AN INFORMATION PRIVACY COMPLIANCE MODEL BASED ON CONFIGURABLE SOFTWARE OBJECTS

Thesis submitted for the Degree of Doctor of Philosophy

in the

Faculty of Engineering, the Built Environment

and Technology

Nelson Mandela University

by

Agbor Takang Kandeh

Promoter: Prof. Reinhardt Botha

Co-Promoter: Prof. Lynn Futcher

April 2022

ABSTRACT

South Africa's Protection of Personal Information Act (POPIA), Act 4 of 2013 requires that organisations enforce information privacy rules in technology systems handling personally identifiable information (PII). This is in line with other national and regional information privacy legislations across the world. However, the absence of a coherent way to implement this legislation, in the form of software objects in technology systems, has created a gap in organisations around the world. To bridge this gap, this thesis proposes a compliance model based on a conceptual framework, a design framework, and a software-based prototype. The objective of this model is to test how best to enforce information privacy regulations in technology systems handling personally identifiable information. The proposed conceptual framework views information privacy compliance as a context-driven reality enforced by configurable software objects. To refine the conceptual framework, a design framework and a software-based prototype was developed using the design science research methodology as the theoretical construct and the UML ontology language and object-oriented programming paradigms as the underpinning practical construct. This prototype will assist organisational stakeholders in understanding and visualising the theoretical and practical constructs of handling personally identifiable information as software objects in technology systems. The design and implementation of this prototype resulted in some practical and theoretical recommendations. These include the adoption of a decision model notation (DMN) as a formal standard to manage privacy rules and the creation of a context-aware privacy compliance zone (CAP). However, the main contribution of this thesis is a reusable conceptual and contextual design framework and a prototype through which POPIA rules, or those of any similar information privacy law, such as the European General Data Protection Regulation (GDPR), can be encapsulated into software objects used in technology systems to ease compliance with information privacy regulations.

ACKNOWLEDGEMENTS

I would like to acknowledge the following people for the support and encouragement I received from them throughout the duration of this study. I hereby express my humble appreciation and gratitude to you all, for the different roles that you played from the inception of this study to its conclusion.

Professor Reinhardt Botha: Main research sponsor, for shaping the research subject, guiding me on the breadth and depth of the research exploration and moderating the research niche in order to meet the research objectives and expectations. Professor Lynn Futcher: Co-Sponsor, for the detailed review of the research content, structure, flow of the research write-up and subject matter.

I would equally wish to express heartfelt appreciation to Mrs Felicita Eyong Tiku and sons for family support and encouragement; to Mr Tengwan Ambe for inspiration and guidance; and to Mr Jonathan Dorothy for academic support and processing of all the requisite documentation. To all whom I might not have mentioned by name, I appreciate your invaluable support and motivation.

Finally, I thank God almighty for granting me the strength and resolve to stay in the programme and to sail right through to the end.

'Not everyone who chased the zebra caught it; but he who caught it, chased it.'

DECLARATION

NAME: **Agbor T. Kandeh**

STUDENT NUMBER: **215388968**

QUALIFICATION: **PhD Information Technology**

TITLE OF PROJECT: An Information Privacy Compliance Model based on Configurable Software Objects

DECLARATION:

In accordance with Rule G4.6.3, I hereby declare that the above-mentioned treatise/ thesis is my own work and that it has not previously been submitted for assessment to another university or for another qualification.

SIGNATURE: ________ _______________

DATE: ________22/03/2022______________________________

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

Introduction

The objective of this chapter is to introduce the research topic entitled ‘An Information Privacy Compliance Model based on Configurable Software Objects’, and to elaborate on the rationale for the study in the context of academic research and business organisations.

This chapter is broken down into eleven (11) sections covering the general background of this study, the state of information privacy research, the background to the research problem, information privacy compliance, the research objectives, the research aim and significance, and finally the chapter layout. The chapter layout gives a brief description of each chapter. This chapter also contains the Thesis Chapter Flow, as illustrated in Figure 1.1, showing the different sections and chapters of this study and how they relate to one another.

1.1 Information Privacy and Compliance for Modern Organisations

The long road towards information privacy and compliance for modern organisations presents many challenges and obstacles. First, these organisations operate in a dynamic environment in which information privacy requirements are constantly changing. For instance, most organisations store, process, and transmit confidential customer data such as credit card information, residential address, social security numbers, and contact details. The storage occurs by means of computerised systems which, if compromised, can have serious repercussions for their customers’ privacy.

Second, these organisations and their employees interact with external information systems from service providers, auditors, hosting providers, and cloud service providers, to mention just a few. All of these boundaries are potential loopholes for exploitation. According to Ernst and Young (2013), organisations are struggling to understand and to enforce information privacy within these boundaries.

To corroborate these assertions, reference is made to some recent incidents involving prominent organisations such as Facebook, Google, and Uber. According to Cimpanu (2018), Google discovered a bug that allowed third-party developers to access the personal data of millions of Google Plus users in its Google Plus API. Instead of informing its users so that they could take steps to protect themselves, Google kept it secret for over a year (Bradley, 2019). According to OECD (2021) ‘personal data’ means any information relating to an identified or identifiable individual (data subject).

In a similar incident involving another prominent global organisation, Facebook, five hundred and forty (540) million Facebook user records were exposed by app developers on an insecure Amazon Web Services (AWS) server in 2019. The peculiarity of this incident is that the majority of the records came from a Mexican media company (Cultura Volectiva), which had a 146GB dataset containing more than 540 million records. The Cultura Volectiva dataset contained information such as account names, IDs and Facebook activity (Computing.co.uk, 2019).

Based on these incidents, it is imperative to enquire whether the impacted data subjects were even aware of a data breach involving their personal information (data) or whether they were compensated in one form or another. The significance of this question can be more appreciated upon considering a MacAfee survey, which revealed that more than 40 per cent of people worldwide feel they lack control over their personal data. The same survey showed that one-third of parents worldwide find it difficult to explain online security risks to their children (McAfee 2018). Most importantly, it should be asked what preventative mechanisms have been put in place by governments to mitigate the potential misuse or unauthorised disclosure of customer data.

These concerns are exacerbated in countries where data protection legislation is still in its infancy. For example, according to Giles (2020), South Africa’s Protection of Personal Information Act 4 of 2013 (also known as POPIA), was signed into law in November 2016 and five (5) years after, the information regulator is still struggling to build the requisite capacity to enforce it. In fact, the implementation compliance date was set for 1 July 2021 and it is still to be seen how effectively organisations will comply with POPIA and how the regulator will enforce compliance.

Hence Shapiro and Baker (2001) argue that it is getting easier to collect personal data from data subjects globally, not purely because of the advances in technology but mostly owing to government inactivity and commercial interests. They conclude that, to date, commercial interests have dictated industry practices. The same interests largely dictate self-regulation outcomes in organisations such as Facebook, Google, and Amazon. Mindful of these arguments and conclusions, it is imperative to protect customer data collected from data subjects globally from the onslaught of big business marshalled by these organisations.

Customer data in this study refers to all personal information that is used to identify a person. According to Matuszewska, Sweeney and Lubowicka (2015), the best-known types of personal data include: email address, home address, phone number, credit card information, birth date, and social security number/ national identification number. They add, however, that personally identifiable data is not limited to the examples stated above. It also includes customer biometric information such as fingerprints, facial structures, iris and voice, medical records, criminal records, and credit records and, until recently, social media behaviour, preferences, and footprint. Traditionally, such information is kept by government (State) where the data subject is a citizen, or by a trusted third party with whom the data owner has explicitly subscribed. It is used only for the purpose for which it was originally collected. However, with the advent of globalisation, information and persona data is seen as a commodity and the fuel for electronic transactions by both public and private organisations globally. This perception thus creates a fertile ground for abuse and other unethical dealings.

In principle, an individual’s right to privacy entails the freedom to disclose (or not to disclose) personal data without incurring the risk of unwanted social control by others (Agre, 1997, p.7). This study will examine how safe it is, in the advent of globalisation, to trust organisations to uphold this principle with personal data, considering that recent events have shown a different trend. An example is the Cambridge Analytica data breach in which millions of Facebook users' personally identifiable information was compromised (Mitra, 2020). Another example is the Uber 2016 incident in which two hackers were able to access data from ride-sharing company, Uber, by breaking into one of the company's third-party cloud services (Khosrowshahi, 2017). As a result, the personal information of fifty-seven (57) million Uber users (including names, email addresses, and mobile phone numbers) was breached, along with the names and driver's licence numbers of over half a million Uber drivers. Instead of disclosing the privacy breach, Uber paid the hacker $100,000 to delete the data (Khosrowshahi, 2017).

Considering the public danger represented by these breaches, this study explores how best to enforce information privacy controls and compliance in the technology systems used in handling personally identifiable information. Information in the wrong hands can be used for all sorts of malicious activities. An example is the Cambridge Analytica breach, where leaked data was used to influence elections around the world. The outcome included Facebook CEO, Mark Zuckerberg, being forced to attend a United States Congressional hearing. Following this hearing, he suffered a staggering 20 per cent loss in his stock. As a result, the company lost $120 billion in value (Mitra, 2019). To crown it all, Facebook was issued an undisclosed fine by the UK Information Commissioners' office for malicious data exposure.

To address some of these privacy challenges, and to force organisations to comply with information privacy best practices globally, many governments and regional bodies have crafted laws and regulations aimed at enforcing fair information privacy practices within organisations. These laws include India's Personal Data Protection Bill (PDPB), the European General Data Protection Regulation (GDPR), the California Consumer Privacy Act of 2018 (CCPA), and the South African Protection of Personal Information Act 4 of 2013 (POPIA), to mention a few. The commonality in all this legislation is the requirement that organisations enforce information privacy rules in technology systems handling personally identifiable information (PII). To comply with these privacy laws and regulations, organisations are required to translate these laws into practical controls that can be enforced in the information systems holding their customer and employee data. These challenges are succinctly summarised by Westin (1967) in the following question: '*What can be done to protect privacy in an age when so many forces of science, technology, environment, and society press against it from all sides?*'(Westin, 1976).

This statement is particularly true in the current context of the fourth industrial revolution or Industry 4.0. According to Schulze (2019), the fourth industrial revolution is a characterisation of the current developing environment in the world in which disruptive technologies and innovations, such as the internet of things (IoT), robotics, virtual reality (VR), and artificial intelligence (AI) are changing the way modern people live and work (Wigmore, 2020). In the context of this study, the disruptive 4IR technologies will be examined especially as they pertain to privacy enhancing technologies or privacy-invading technologies, as elaborated in the next sections of this study. Breaux (2014) states that privacy is a critical design principle that must be balanced with how we utilise personal data in software systems. Furthermore, privacy is frequently defined in information systems (IS) and in many branches of social science research in phrases such as 'the ability of individuals to control the terms under which their personal information is acquired and used' (Culnan & Bies, 2003, p. 326).

This definition is of particular interest to this study as it gives credence to the fact that individual privacy can be protected through information system control objects. According to the Oracle Corporation (2020), software objects can be defined as conceptual representations of real-world objects with a state and behaviour used to represent business rules. Based on this definition, it can be concluded that information privacy requirements can be tailored into software objects and can be used to enforce compliance with privacy regulations by organisations.

1.2 Background to Research Problem

The rapid adoption of technology by organisations today is opening doors of opportunity but it is also creating a tremendous information privacy risk. To support this point, Bélanger and Crossler (2011) state that advances in information technology have greatly expanded opportunities for technical solutions to address information privacy concerns. They add that it has also set the stage for information systems researchers to take a leading role in the practical implementation of technology solutions to mitigate information privacy concerns.

Privacy regulators are doing everything they can to keep up, but as the technology's evolution accelerates its pace, regulators continue to fall behind. According to Ernst and Young (2013), regulation remains a useful tool to improve privacy protection. However, privacy regulators will have to make a fundamental shift from merely acting as compliance officers to also serving as strategic advisors. They will have to work with organisations to facilitate stronger decision-making when it comes to privacy management. Organisations need to be more accountable. If organisations are unwilling to integrate privacy into IT transformation initiatives, as Privacy by Design suggests, regulators should be looking to mandate it.

One such legislation is the POPIA, enacted into law in November 2013. This act requires that both private and public sector organisations protect personal information collected from data subjects. Recent evidence, however, suggests that various industries have already raised concerns about the lack of clarity in terms of compliance and the status quo of the act. Moreover, the delay in the appointment of an information regulator, who has the power to investigate and impose sanctions, is not helping matters. Furthermore, based on Visser (2021), many companies in South Africa have left their preparation too late (especially the large companies) and have not yet established a complete set of practices and processes to ensure their compliance with POPIA's many requirements. Although companies will have about a year before the many POPIA compliance requirements come into effect, one year is simply not enough to prepare adequately. In fact, according to Deloitte (2021), the effective POPIA compliance go live date was fixed for 1 July 2021, and many companies in South Africa, are still struggling to implement an effective data privacy compliance programme that can withstand legal and regulatory scrutiny. Yet, if these companies find a way to implement POPIA requirements easily into their information systems, it could speed the compliance process and they will be able to comply with POPIA and satisfy both the POPIA regulator and their customers at the same time. However, this challenge does not end here. Rainie and Anderson (2014) argue that while companies operate in an environment where they are pushing existing barriers to what they can know and store about their customers on the one hand, government law makers are changing laws constantly to address potential loopholes, on the other. Mindful of this dynamic environment, experts suggest that technology shields will offer better privacy to people as the cyber community grows and assimilates more personal information.

1.3 Problem Statement

In the context of this study, the problem statement relates to the POPI Act that requires all South African institutions to conduct themselves in a responsible manner when collecting, processing, storing and sharing another entity's personal information. However, **most companies in South Africa are struggling to enforce the POPIA regulations in software systems, especially in systems that were designed without privacy in mind**. Failure to comply with POPIA regulations might result in substantial penalties such as civil claims, prison terms and fines of up to R10 million

In response to this problem, this study proposes a practical model through which companies can implement and enforce the POPIA regulatory policies in their information systems by making use of software objects. At face value, some will argue that this is a legal problem and be treated as such, but from the literature reviewed for this study, it is believed to be a technology problem. Hence a practical technology perspective and model is developed.

To develop such a model, Van Vuuren (2015) suggests that POPIA compliance requires a focused, systematic and formalised approach to the management of information. However, many companies have not taken this aspect seriously in the past, thus giving rise to the following research aim, objectives and questions.

1.4 Aim of Research

The overarching aim of this research is to develop a conceptual design model and a POPIA rule engine prototype to enable the enforcement of information privacy regulations in the form of a context-driven reality in configurable software objects. In this way, similar information privacy laws, such as India’s Personal Data Protection Bill or the European General Data Protection Regulation, can also be encapsulated into business software objects or technology systems to ease the enforcement of information privacy compliance.

1.5 Research Objectives

In a nutshell, the primary research objective of this study is:

To recommend a practical model through which organisations can implement and enforce POPIA regulatory policies in their information systems, using software objects built out of data subject personally identifiable information (PII).

In the same vein, particular attention is given to the data subjects whose privacy is at risk and is here represented by the term ***personally identifiable information (PII).*** The POPIA regulatory policies will be encoded in software objects using privacy engineering rules to test the ease of configuration and compliance.

The secondary objectives include the following:

1. To facilitate the translation of POPIA requirements into machine-interpretable language.
2. To determine the best design pattern to meet the technical and operational requirements for POPIA compliance within organisations.
3. To formulate a model to use for the validation and verification of personally identifiable information against POPIA regulations.

1.6 Research Questions

The primary research question is:

How can the POPIA regulations be implemented as software objects used to enforce information privacy and compliance within organisations?

To answer the primary question, this study examined the following sub-questions:

1. How can the common vocabulary of POPIA regulations be expressed as machine-interpretable instructions?
2. Which software architectural synthesis can best satisfy the organisational goal of POPIA compliance?
3. How should individual privacy concerns be validated against POPIA regulatory controls within an organisation?

It is envisaged that a summation of the answers to all these questions will provide insight into the research problem under consideration for this study. In addition, it will also provide the necessary data to solve the primary research problem outlined in this study.

1.7 Research Paradigms

This study was conducted using the design science research paradigm. Vaishnavi and Kuechler (2005), define design science as a research lens which draws on a set of analytical techniques and perspectives for performing research in Information Systems (IS). These authors further declare that design science revolves around the creation of new knowledge through the design of innovative artefacts and the analysis and evaluation of the performance and use of such artefacts to improve the understanding of information systems. This conclusion is supported by Hevner et al (2004), who describe design science as a research approach best suited to creating and evaluating IT artefacts intended to solve identified organisational problems. Hevner et al (2004) put the definition in other words as a paradigm that extends the boundaries of human and organisational capabilities by creating innovative artefacts.

This methodology suits the context of this study, which proposes an artefact based on the research problem and key variables highlighted in the primary research question. This view is also supported by the work of Wieringa (2013). Wieringa (2013) categorises design science as a solution-oriented research paradigm with an engineering cycle used to investigate and model research problems into practical constructs that represent stakeholders of the problem, the research goals, the phenomenon, the evaluation of the research problem, and practical diagnosis of the problem in context. It thus contrasts with the natural sciences and social sciences that are mainly problem oriented. For example, Wieringa (2013) cites three (3) use cases of design science research methodology as follows:

1. Used for observational studies to understand the requirements of an Agile project;
2. Used for observational studies to unpack the pattern of evolution of groupware software systems;

3. Used in experimental studies to understand how software engineers understand UML ontology language and apply it to solve problems.

Based on the above cited use cases, the third use case fits the context of this study as it involves the use of UML ontology language to represent and solve a complex software engineering problem. A similar study at the University of Nevada, Las Vegas, used design science to develop a conceptual framework to improve knowledge sharing in a virtual community (Koneru, 2018). Koneru (2018) did not only end at the conceptual framework level but went ahead to propose a learning grid as a means to implement the model.

The benefit of such a paradigm, according to Wieringa (2013), is the visibility that results from portraying the interaction between the research artefacts from the study and the operational environment (research context). In addition, this interaction is iterative and provides enough opportunities for continuous observation and evaluation through constant replay and testing.

Furthermore, considering the complexity of this study, design science research methodology is chosen because it incorporates a mixed method design within an evaluation cycle methodology. This enables research to benefit from the strength of a mixed method, as stated by Vaishnavi and Kuechler (2005).

Before examining the scope of this study in more detail, the assumptions underpinning this study are outlined below.

1.8 Assumptions of Study

In setting up this study, based on the academic literature sampled, the following assumptions are made regarding information privacy in general:

1. Information privacy concerns have measurable dimensions which can be quantified (Bélanger & Crossler, 2011)
2. The opt-in or opt-out procedures for information privacy common in most software systems, is too simplistic and is not suited to protect customers against the secondary use of personal information (Biselli & Reuter, 2021).

3. Information privacy vectors are dynamic and require intelligent software objects to protect privacy (Bélanger & Crossler, 2011).

Of all these assumptions, assumption 3 is the most relevant for this study as it positions software objects at the centre of the compliance and most importantly, in enforcing information privacy within information systems, as suggested by Bélanger and Crossler (2011).

1.9 Scope of the Study

This study covers the use of software objects to achieve compliance with information privacy regulations and laws within any organisation operating in the public and private sectors. The research embraces design science methodology to build innovative artefacts. The artefacts are then used to simulate and demonstrate how best to enforce the protection of data subjects' personally identifiable information placed under the protection of an organisation. Several other related disciplines, such as information security, privacy engineering, compliance management, to mention a few, will be explored to add context to the research and to determine the best approach to use to build the prototype.

In building this prototype (artefact), attention is given to the following aspects:

1. The technical architectural design pattern best suited to achieve such an engineering effort.
2. The data subject whose privacy is at risk and which elements of their personal data are at risk. This is linked to the conceptual framework designed for this study, which helps to view how information flows within the context of an organisation.
3. To explore design frameworks to use in order to encode the prescriptions of the information privacy legislation, POPIA in this case, into business rules encoded into software objects residing in information systems.
4. The use of the UML use case ontology to model and represent the structure and inner working of the prototype.

5. The activities undertaken as part of the design science research methodology to simulate, test, and evaluate the artefact or prototype built for this study.

1.10 Chapter Layout

This research project is subdivided into three (3) parts, each consisting of a number of chapters, as outlined below.

Part 1 is termed the **introduction**, and is made up of four (4) chapters, namely:

- **Chapter 1 – Introduction:** It deals with the background of the research topic and elaborates on the rationale for the study in the context of academic research and business organisations.
- **Chapter 2 – Information Privacy:** This chapter introduces and defines the main theme of the research study, information privacy. This chapter also explores key theories about information privacy and related disciplines with the aim of highlighting the background to the research problem and objectives.
- **Chapter 3 – Privacy Engineering:** This chapter focuses on privacy engineering as it seeks to introduce how information privacy principles are handled and enforced in information systems used by organisations to process personally identifiable data belonging to their employees and customers.
- **Chapter 4 – Research Design and Methodology:** This chapter focuses on the research paradigm, design, and methodology. The chapter also paints a road map of how the research design, method, and approach are applied to different stages of the research project from the development of the conceptual framework, the framework for the design, testing and evaluation of the prototype, and the different artefact of this study.

The next section is **Part 2**, and it is termed the **body of the thesis**. It is made up of the following three (3) chapters:

- **Chapter 5 – Conceptual Framework:** This chapter deals with the research paradigm and the development of a conceptual framework for the prototype that was developed to test and evaluate the research questions and objectives of this study.

- **Chapter 6 – Framework Design:** In this section, the principles of design science methodology are explored to extract suitable design principles and a framework to be used to design the prototype envisaged for this study. The conceptual framework developed in Chapter 5 is used as the main input into the framework design. Finally, the prototype design and development process utilised here is recorded as an integral part of the research process leading to the final section of the research.

The final section of this research report is **Part 3**, and it is termed the **conclusion**. It is made up of three (3) chapters, namely:

- **Chapter 7 – Implementation of Prototype:** This chapter deals with the testing and simulation of the prototype in the context of an organisation and in line with the research questions and problem domain.
- **Chapter 8 – Evaluation of Prototype:** Stepping further from the testing and simulation covered in chapter 7, this chapter focuses on the observation and evaluation of the ‘live prototype’ in line with the expectations and conditions defined in the research questions, problem, and research objectives.
- **Chapter 9 – Conclusion and Implications:** This is the final chapter of this study and it focuses solely on the conclusions and implications of this study. The chapter covers the general conclusions, and also specific conclusions based on the research questions, problem, and research objectives. Finally, some significant implications of this study are highlighted and discussed in context. **Figure 1.1** below shows the different chapters of this thesis, their position and how they relate to the other chapters, as explained in **Section 1.12 of Chapter 1.**

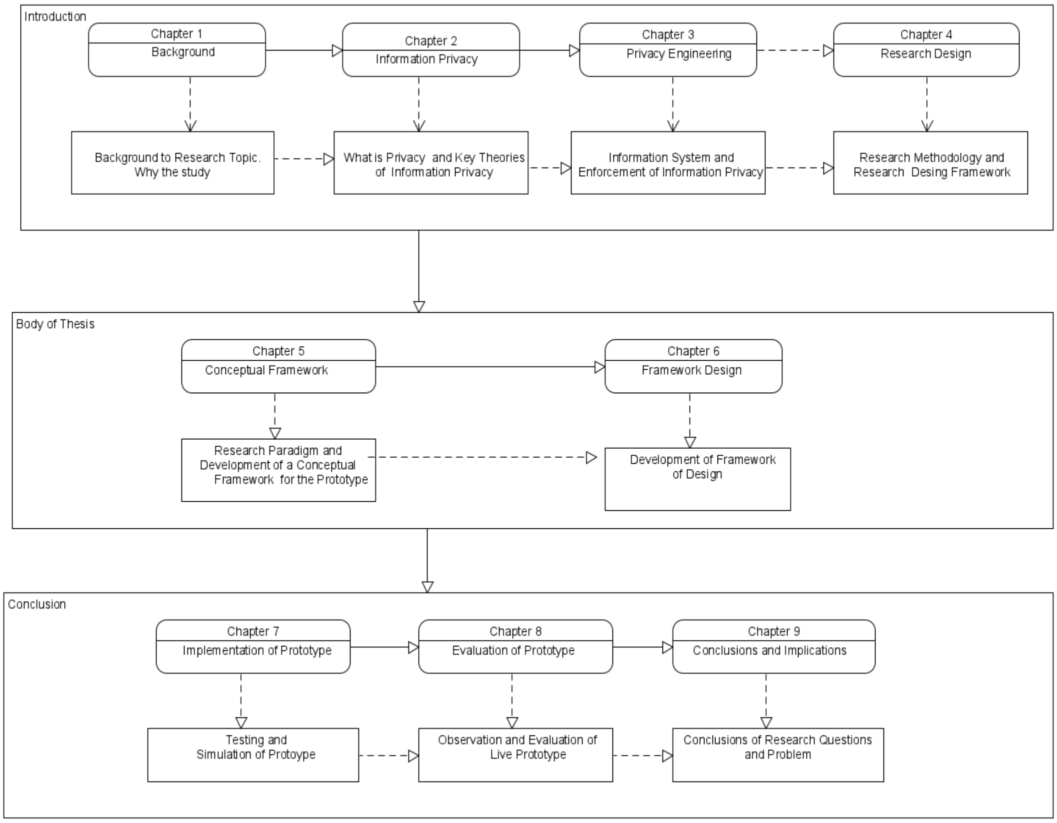


Figure 1.1: Flow of Thesis Chapters

1.11 Conclusion

This chapter introduced and defined the main theme of the research study and the research topic under consideration. It further discussed the background to the research problem and the problem statement, the research significance, the aim, objectives, research question and sub-questions. Then, the research paradigm and methodology were outlined. The next chapter will focus on the literature on information privacy and related disciplines that are relevant to this study.

CHAPTER 2

Information Privacy

2.1 Introduction

Chapter 1 introduced the research topic under consideration and further discussed the background to the research problem, highlighting the research aim and objectives, and the methodology.

Chapter 2 focuses on the literature surrounding the main theme of this research study, information privacy, and how related research literature supports its enforcement using technology controls.

In the first section, entitled Background to Information Privacy, the definition of the concept of information privacy in the context of this study is presented in detail and supported by a historic perspective and timelines of the information privacy discipline.

The next section of this chapter, entitled Related Literature in Information Privacy, explores the state of the current information privacy literature with emphasis on the main contributors to this literature and the key academic concepts and theories put forward by these contributors.

Finally, based on the review of the state of the current literature on information privacy conducted in the previous section, the final section of this chapter identifies the gaps in the current literature and highlights the rationale of this study and the relevance of the research focus area.

2.2 Background to Information Privacy

Information privacy is a central issue in the information age (Wu, et al., 2020, pp. 485–490). According to Serohin (2020), modern society has changed significantly from being power based to being information based, now referred to as the information society; hence, the one with the most information rules the world. To date, we have reached another level of information privacy referred to as the ‘personal information economy’ first referred to by Dennedy et al. (2014). However, according to Biselli and

Reuter (2021), the ever-increasing threat to information privacy has created an environment in which the research and debate about methods to improve the protection of privacy is ongoing at all levels of society.

To contextualise this further, according to Canedo et al. (2020), information privacy is not a new issue. The debates about information privacy go back to the days of Westin in 1967, especially about understanding the practices and guidelines regulating individual and societal privacy concerns. In the light of this, according to Barbosa et al. (2020), privacy concerns exist wherever personally identifiable data is improperly collected, processed, stored, or shared. To expand on the concept of information privacy, the next section examines the definition of information privacy in the context of this study.

2.2.1 Defining Information Privacy in Context

There are many definitions of information privacy in academic literature. Going back in time, Westin (1967), defines information privacy as ‘a person’s desire to freely determine the circumstances and the extent to which they will expose their attitude and behaviour to others’ (Westin, 1967 p78). This definition focuses on the individual and the choices they make and is succinctly summarised by Pavlou (2011, p.1) as the desire to maintain control over one’s personal information.

According to Baselli and Reuter (2021), the current understanding of privacy does not depart far from the 1967 definitions of Westin, although preserving privacy in the digital age is a major challenge as technology is fast evolving and changing the privacy dynamics. To support this view, based on Fenghua et al. (2019), the rapid advances and adoption of information and communication technologies across all industries is generating and accumulating huge amounts of data in the process of serving customers across the globe. Similarly, the multiplicity of digital technologies used by most individuals across the world, and most especially in industrialised nations, is generating tremendous amounts of data for the service providers and provides many benefits to the users of these devices. (Gerber et al., 2018). Consequently, according to Slepchuk and Milne (2020), marketers are using technology to harvest personal data and consumer attention in a very covert manner.

However, according to Pew Research Center (2020), the downside of these benefits is the notion of loss of privacy. Also, according to Pew Research Center (2020), 81 per cent of respondents in a study felt they have little or no control over the data collected, and about 97 per cent of Americans say they have never been asked to approve privacy policies. Similarly, according to Pew Research Center (2020), 81 per cent of Americans think the potential risks of data collection by companies about them outweigh the benefits.

To support this argument, Maple et al. (2021), suggest that the massive accumulation of data by large corporations has instinctively positioned information privacy from a ‘nice to have’ feature to an essential component in the data-driven economies of today. Furthermore, according to Barbosa et al. (2020), in an increasingly connected world, data is collected from diverse sources and the richness of this data has raised several information privacy concerns. Though there are existing rules and guidelines, there is a significant gap in the methodology to integrate these rules into design processes and system controls.

Another threat to information privacy emanating from this massive pool of data collected is the phenomenon of ‘big data analytics’. According to Akter et al. (2016), organisations rely heavily on customer data and advanced technologies such as big data analytics to shape their products and services. Hence, according to Allen (2016), the collection of big data and big data analytics are two technology trends with enormous impact on how business is done globally. Furthermore, according to Allen, big data makes use of a mixture of consensual and non-consensual data gathered from the activities of customers not only to developed customised products or services but also to present supplementary products / service offerings to the customers (Allen, 2016), opening the door for potential information privacy violation by organisations in the context that customers do not always understand the full extent to which their data is collected, analysed and commoditised.

In general, according to Martin and Murphy (2016), the widespread access to customer’s data exposes them to fraud, invasion of privacy, unsolicited marketing and cyber-attacks. Gundu (2019) concludes that big data and big data analytics come with critical security and privacy risks. According to Biselli and Reuter (2021), information privacy in general refers to the prevention of exposure of sensitive information about groups or individuals. This includes, among other things, the nondisclosure of behaviour, communications, and descriptive personal data. Furthermore, with advanced technologies such as Artificial Intelligence (AI), quasi identifiers are used nowadays to uniquely identify individuals without their knowledge, by combining data emanating from different sources and criteria (Vimercati and Foresti, 2011).

Hence, since the concept of privacy has been influenced mostly by technology, a technology-focused definition is provided by Dennedy et al. (2014). According to Dennedy et al. (2014), information privacy describes the individual’s right and ability to define and live life in a self-determined manner by fully controlling the data they generate from the process. To support this definition, according to Zimmer (2014), Mark Zuckerberg of Facebook, building onto of the 1960 techno-activist slogan ‘information wants to be free’, characterises information privacy as the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others. Zuckerberg argues that, in Facebook, users must have control over who they shared their information with. However, the default settings of Facebook lean toward making information public. This definition touches the core of this study by asserting that the data subject should always be in control of their personal information and data, a statement which is not always true, as highlighted by Martin & Murphy (2017) and by Gundu (2019).

As a result of the threat against information privacy, national and regional governments across the word have passed legislations to protect the privacy of their citizens. Some of these legislations are regional, such as the General Data Protection Regulation (GDPR) covering the European Union (EU), while others are country specific such as the Protection of Personal Information Act (POPIA) of South Africa and India’s Personal Data Protection Bill (PDPB), and finally some are in-country state specific, such as the California Consumer Privacy Act.

2.2.2 Historic Timeline and Evolution of Information Privacy

Going back in history, privacy is not a new issue. To give a perspective, as far back as 1967, Westin defined privacy as the right of an individual to determine how and to what extent their personal information is shared with others (Westin, 1967). Even older than this definition of Westin’s, is that of Warren and Brandeis (1929), which characterises privacy as the right to be left alone and recommended that it is the individual's responsibility to main that privacy.

In contrast, more recently, according to Mason (2005), and Smith et al. (2011), individuals were seen to be trading their privacy for goods and services within the e-commerce and online banking space, creating a new perspective on privacy which is quite different and evolved from those posited by Warren and Brandeis (1890) and Westin (1970) in the past. Hence, according to Conger, Pratt and Loch (2013), privacy as a concept is not simple and has been the subject of rigorous debates in the different spheres of society and in different regions of the world over a very long time. Nevertheless, the history of privacy is not the central focus of this study.

In the context of this study, information technology and, more precisely, emerging technologies is used as the lens through which the evolution of the concept of privacy is reviewed. This focus is primarily due to the fact that technology has been the main and consistent catalyst changing the trajectory of the information privacy evolution. For instance, according to Fenghua et al, (2019), advances in information and communication technology have exacerbated the information privacy debate as its adoption has exponentially increased the amount of personal data being collected, processed, and stored by organisations of all types. Furthermore, it is believed that information technology is uniquely positioned to shape the information privacy debate and to provide innovative solutions to privacy challenges. (Fenghua et al., 2019). For instance, according to Conga et al. (2020), the characteristics of the emerging technologies that pose threats to privacy, relate to their ubiquity, invisibility, invasiveness, collectability of previously uncollectible information, programmability, and wireless network accessibility.

To illustrate this view, **Table 2.1** shows the evolution of the information privacy concept alongside the evolution of the underlying technologies.

Table 2.1: Evolution of Information Privacy

(Source: Adapted from Margulis, 2003)

Period	**Characteristics**
Privacy baseline 1945–1960	Limited information technology developments,

Period	**Characteristics**
	high public trust in government and business sector, and general comfort with the information collection.
First era of contemporary privacy development 1961–1979	Rise of information privacy as an explicit social, political, and legal issue. Early recognition of potential dark sides of the new technologies (Brenton, 1964 p112), formulation of the Fair Information Practices (FIP) framework and establishment of government regulatory mechanisms, such as the Privacy Act of 1974.
Second era of privacy development 1980–1989	Rise of computer and network systems, database capabilities, federal legislation designed to channel the new technologies into FIP, including the Privacy Protection Act of 1984. European nations move to national data protection laws for both the private and public sectors.
Third era of privacy development 1990–present	The rise of the internet, Web 2.0 and the terrorist attack of 9/11/2001 dramatically changed the landscape of information exchange. Reported privacy concerns rose to new highs.
Fourth era of information privacy compliance laws, e.g. The POPI Act	Internet of things (IoT), bring your own device (BYOD), big data, cloud computing, social media, e-government, etc.

To explain information privacy in more detail and in context, the next section of this study examines related literature in information privacy research and its relevance to this study.

2.3 Related Literature in Privacy Research

The related literature is reviewed and discussed in the following sections of this study. Similarly, to situate the correct trends properly in the information privacy debate, emphasis is placed on recent literature, not more than five years old.

2.3.1 Literature Review Process

To review the current literature research on information privacy, a literature search was conducted of the major journals over the last ten years (post-2011) using the keywords ‘Information Privacy’ in the search for publications and journal articles dealing with this topic in the electronic database of major academic journals, such as Science Direct, EBSCOhost, Scopus, MIS Quarterly, Google Scholar, SpringerLink, and ProQuest under following sources types: scholarly journals, books, dissertations and theses, newspapers, trade journals and conference papers / proceedings, to mention a few, as illustrated by **Table 2.2.**

Table 2.2 Electronic Database Sources Consulted

No.	Electronic Database	URL Link
1.	Networked Digital Library of Theses and Dissertations (NDLTD)	http://www.ndltd.org
2.	Newspaper Source Premier (EBSCOhost)	https://www.ebsco.com
3.	OECD iLibrary	https://www.oecd-ilibrary.org
4.	OSIRIS	https://osiris.bvdinfo.com
5.	Oxford English Dictionary	https://www.oxfordlearnersdictionaries.com
6.	Passport	https://www.library.hbs.edu
7.	SA ePublications (Sabinet)	https://www.sabinet.co.za
8.	SA Media (Sabinet)	https://www.sabinet.co.za
9.	Sabinet Government Gazettes	https://www.sabinet.co.za
10.	SAGE Knowledge	https://sk.sagepub.com

11.	SAGE Research Methods	https://sk.sagepub.com
12.	Science Direct	https://www.elsevier.com/
13.	Scopus	https://www.scopus.com
14.	SPORTDiscus	https://www.ebsco.com
15.	SpringerLink	https://link.springer.com
16.	ProQuest	https://www.proquest.com
17.	Google Scholar	https://scholar.google.com
18.	Science Direct	https://www.sciencedirect.com
19.	MIS Quarterly	https://www.misq.org

This search was inspired by the 2011 search conducted by Bélanger and Crossler (2011), cited in over 853 different papers and publications and published in the MIS Quarterly journal article entitled 'Privacy in the Digital Age: A Review of Information Privacy Research in Information Systems'. Based on the article published in the MIS Quarterly, 35(4), 1017-1041, authored by Bélanger and Crossler (2011), a review of literature was conducted to analyse the different theory types on the topic of information privacy, adapted from the contributions to theory definitions of Gregor (2006), and illustrated in Table 2.3.

Table 2.3 Privacy Theory Categories and definitions (Adapted from Gregor, 2006)

Theory Category	**Definition**
Analysing	Describe the state of information privacy or the need for information privacy research
Explaining	Explain what is occurring but do not provide testable predictions
Predicting	Provide testable predictions without well-developed causal relationships
Explaining and Predicting	Explain what is occurring and provide testable predictions with causal explanations
Design and Action	Specify a design tool for providing information privacy or a framework to evaluate such tool

These theory types were matched against the following five (5) topic areas of information privacy research, namely information privacy concern, information privacy and e-business impacts, information privacy attitudes, information privacy practices, and finally information privacy and technologies. The statistics elicited a significantly low number of publications in the design and action theory type of information privacy and technologies topic areas (Bélanger & Crossler, 2011).

Based on this search outcome, the design and action theory type was least discussed in the available academic literature reviewed, raising the question of whether enough research has been done in this area. Hence, this study will be focused on this area.

2.3.2 Main Contributors to Information Privacy Research

Following the same approach by Bélanger and Crossler (2011), this section highlights the main contributors to the information privacy research from 2012 to date.

These contributors are classified using the same privacy categories and definitions adapted from Gregor (2006) and presented by Bélanger and Crossler (2011), as illustrated in **Table 2.3.**

In line with the topic of this study and focusing on the research domain, emphasis is placed on contributors in the theory category of design and action under the topic area of information privacy and technology.

To support this approach, Bélanger and Crossler (2011) recommend that future research in information privacy should take the form of practical frameworks to enforce privacy protection, such as the creation of technology artefacts for the protection of information privacy, which will also improve existing theories on information privacy.

Table 2.4 highlights the main contributors to the information privacy debate under the theory category of design and action in the information privacy and technology topic area. Contributions into the other theory categories are also highlighted in **Table 2.4**. Refer to **Appendix A** for a more detailed table highlighting the contributors and the highlights of their contributions.

Table 2.4 Contributors to Privacy Debate According to Theory Category

	Names of Contributors	**Theory Category**
1.	Bellman et al. (2004)	Analysing
2.	Slane (2018)	
3.	Lomas (2019); Conger (2020); Holmes (2021)	Explaining
4.	Slepchuk & Milne (2020)	
5.	Biselli & Reuter (2021)	Predicting
6.	Dias Canedo et al. (2020)	Design and action
7.	Papacharissi & Gibson (2019)	
8.	Cavoukian et al. (2010)	
9.	Microsoft (2020)	
10.	Liu et al. (2011); Wu (2019)	
11.	Kalloniatis et al. (2013)	
12.	National Institute of Standards. NIST (2020)	
13.	Fennessy (2019)	
14.	Martin & Del Alamo (2017)	
15.	Wu et al. (2020)	Explaining and predicting
16.	Bélanger & Crossler (2011)	
17.	Wu et al. (2020)	
18.	Shaar (2010).	
19.	Gerber et al. (2018)	
20.	Akter et al. (2016)	
21.	Allen (2016)	
22.	Teravainen (2020)	Explaining and predicting
23.	Cole (2015)	Explaining and predicting
24.	Fenghua et al. (2019)	Analysing and explaining and predicting
25.	Perera et al. (2020)	
26.	ISO/IEC (2014)	Privacy standard

In view of these contributors and their numerous contributions to the information privacy debate under the design and action theory categories in the information privacy topic area, the following key concepts stand out from the literature guided by these contributors.

2.3.3 Key Concepts in Information Privacy

In today's digital information environment, information privacy concepts and principles are rapidly evolving and debated in the different spheres of society (Bu et al., 2020). According to Slepchuk and Milne (2020), a synthesis of all these debates and research point towards one ultimate goal, which is an improved information privacy practice. From a review of current and past literature on information privacy, we note the following twelve (12) information privacy concepts that are shaping this debate and which this study examines in depth and in context, namely: Privacy by Design (PbD), privacy engineering (PE), privacy paradox, fair information practices (FIP), privacy regulation, privacy standard, data protection, information security, cloud computing, social networks, and internet of things (IoT).

To contextualise these concepts, **Figure 2.1** provides an overview of how these concepts intersect with one another and contribute towards the ultimate goal of improving information privacy.

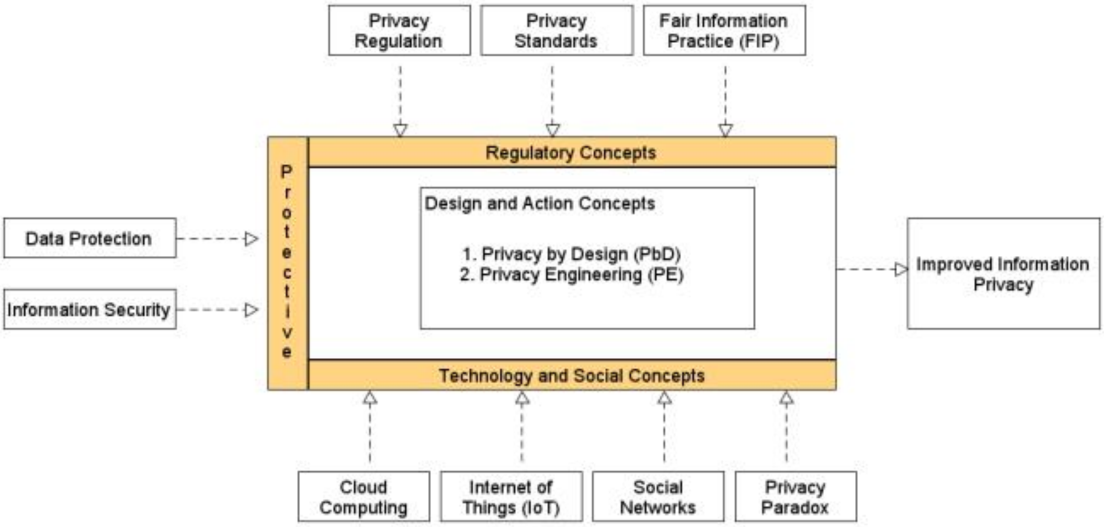

Privacy Regulation
Privacy Standards
Fair Information Practice (FIP)
Regulatory Concepts
Protective
Data Protection
Information Security
Design and Action Concepts
1. Privacy by Design (PbD)
2. Privacy Engineering (PE)
Improved Information Privacy
Technology and Social Concepts
Cloud Computing
Internet of Things (IoT)
Social Networks
Privacy Paradox

Figure 2.1: Information Privacy Concepts

At the centre of **Figure 2.1** are the concepts of Privacy by Design and privacy engineering. These two concepts in the context of this study can be referred to collectively as the design and action concepts. These concepts are central to this study as they align with the goal of this study, which is to design software-based objects to improve privacy controls within technology systems.

To start, Privacy by Design (PbD), was introduced when it was felt that the legal and regulatory frameworks established to protection information privacy was not enough to ensure the protection of personal data in the rapidly growing information industry (Shaar, 2010). According to Cavoukian (2009), (see also Hustinx, 2010), the concept of PbD emphasises proactive protection and claims that privacy protection should be considered throughout the products' entire lifetime, from initial conception to the end of the service life (Cavoukian, 2009; Hustinx, 2010). PbD is a new privacy protective paradigm, which could provide wider privacy information protection (Hustinx, 2010). Cavoukian et al. (2010), propose seven (7) fundamental principles of PbD, which this study will not be focusing on. However, this study rests on the PbD principles that information privacy protection should be preventative rather than remedial and that privacy should be embedded in the design of privacy systems and tools and should be an integral part of the life cycle of any products.

Proceeding to the other concept of privacy engineering (PE), according to NIST (2020), privacy engineering is a new and developing discipline with the primary purpose to build trusts in information systems designed to handle personal information and, as a consequence, to support the growth of the digital economy and improve individual quality of life.

Given the recent concerns on how information systems and technology can affect the privacy of individual, their social levels and identities, privacy engineering is emerging alongside similar disciplines such as information security, cloud computing and big data, to build privacy protection into information systems at design stage as well as to offer privacy protection that can scale. (NIST, 2020).

In terms of definition, according to NIST (2020), privacy engineering is an emerging field of study, and its exact meaning is still evolving. Privacy engineering brings tools, techniques, metrics, and taxonomy to implement 'Privacy by Design'. The most widely accepted definition of privacy engineering is the definition from NIST (2020), as 'a specialty discipline of systems engineering focused on achieving freedom from conditions that can create problems for individuals with unacceptable consequences that arise from the system as it processes PII.' (NIST, 2020 p45). This definition takes into consideration three (3) important aspects of this study. First, the individual whose information is at risk, here referred to as the data subject; second, consequences for non-compliance with the provisions of information privacy which, in the context of this study and according to some privacy legislations like POPIA, can come up to R10 million in fines and penalties and even up to three (3) years' prison term; and third, the information system or systems responsible for the handling and processing of this information.

Based on Martin and Del Alamo (2017), privacy engineering is multidisciplinary, and thus subject to multiple reference frameworks and paradigms, some being social, legal, risk, or technical. Furthermore, previous research has focused more on using technical means to solve information privacy issues. Not an equal amount of effort has been placed on investigating a systematic way of generalising and standardising engineering solutions to information privacy. In addition, according to Martin and Del Alamo (2017), even if a given privacy engineering framework is crafted, the diversity of information systems platforms such as APIs, microservices, web services and portals, coupled with a myriad of software development methodologies such as Agile, Waterfall, and Scrum makes it difficult to formulate adequately a one-size-fits-all approach. To expand on this point further, drawing on the definition shared by NIST, the essence of privacy engineering is developing a trustworthy information system by applying measurement science and systems engineering principles to the creation of frameworks, risk models, guidance, tools, and standards that protect privacy and, by extension, civil liberties. (NIST, 2021).

This definition connects the practical aspect of system development and design with the theories and regulations guiding the practice of information privacy. In a similar manner, Fennessy (2019) of the International Institute of Privacy Professionals (IAPP) characterises privacy engineering as the technical side of privacy by which privacy considerations are integrated into privacy design. In the industry today, according to Fennessy (2019), privacy engineers work as part of technical design teams as well as part of the security team. However, there is a different school of thought that privacy engineers are more part of the process design team. In this case, privacy engineers are seen as belonging more to the product / legal and compliance teams. However, according to Fennessy (2019), the privacy engineering programme at Carnegie Mellon University expresses the need for privacy engineering professionals whose skills and know-how can span both the technical and theoretical aspect of information privacy, meaning that these professionals will be able to understand information privacy theories and principles and be able to integrate this knowledge in the software design and development process and further extend it into the information security space.

The cornerstone of this study is to see how to employ these design and action concepts to build software-based objects to enforce information privacy in technology systems as guided by the gaps in knowledge identified in this study. Departing from these design and action concepts, the next section looks at the social and technology concepts influencing the information privacy debate. One of the concepts in this area is the ‘privacy paradox’, described simply by Gerber et al. (2018) as a dichotomy between privacy attitude and privacy behaviour of data subjects, by which they advocate the importance of their privacy publicly but willingly give it away privately.

Information privacy researchers have made several attempts to explain the privacy paradox during the last ten years (Kokolakis, 2017). According to Gerber et al. (2018), several theoretical explanations have been put forward leaning on empirical results from various studies. However, no satisfactory explanation has emerged as a comprehensive rationale for the actions of the users. Consequently, the privacy paradox still remains a complex phenomenon that cannot be entirely explained.

To illustrate the complexity of this this phenomenon, according to Wu et al. (2020), privacy researchers have long observed a ‘privacy paradox’ phenomenon (that is, people claim to care about privacy but behave as if they do not care), but few have examined systematically in which contexts this attitude–behaviour dichotomy is likely to manifest, or how to resolve the dichotomy through technology design. This study does not delve into the theories and empirical evidence justifying the different explanations of the privacy paradox but will consider the privacy paradox as one of the inputs in the technology and social concepts influencing the information privacy concept in general, as illustrated in **Figure 2.1.** Alongside the privacy paradox, social networks (SN) is another important technology and social concept influencing the information privacy concept in general, as illustrated in **Figure 2.1.** According to Wu et al. (2020), from a social network perspective, information privacy can be understood as a process of managing boundaries across different social contexts. The boundaries may shift, collapse, or re-emerge as social circumstances change. For example, on Facebook, users navigate a variety of audiences and social contexts, with different boundaries for their disclosures.

In doing so, social network users must negotiate equally the content they share and who the perceived audience is, while being mindful of the fact that there might be negative repercussions if the wrong content is published, for instance. According to BBC.com (2021), Twitter permanently suspended the account of the former US President Donald Trump for allegedly using his Twitter account to incite violent attacks on the US Capitol on 6 January 2021. This delicate balance is an ongoing activity for all users of social networks. To illustrate, from an information misuse and leakage perspective, according to Conger (2020), social media giant Twitter was investigated and fined $150 million by the United States Federal Trade Commission (FTC) for misusing people's personal information to serve advertisements. Similarly, according to Holmes (2021), over 500 million personal records of Facebook users have been leaked online in a low-level hacker's forum. This leaked data includes phone numbers and other personal data that cyber criminals could use to impersonate and scam these users online. All this is happening on the back of the Cambridge Analytica data breach in 2016 where the data of 80 million Facebook users was leaked and used to target them with political advertisements, by which they influenced the results of the presidential elections in the United States of America (Lomas, 2019).

Hence, in view of the immensity of these fines and the extent of leakages involving hundreds of millions of users, the social network concept has a significant influence in the general debate about the information privacy concept, and its context is very relevant to the process undertaken in this study to build privacy-sensitive software objects to improve the protection of information privacy.

In addition to the privacy paradox and social network, there are two more important information technology concepts, namely internet of things (IoT) and cloud computing (Cloud) that are greatly influencing the information privacy concept and debate. First, according to Perera et al. (2020), the internet of things (IoT) is an interconnected collection of physical objects or 'things' that have computing, sensing and actuation capabilities, together with the ability to communicate with each other and other systems to collect and exchange data. The fear of IoT is not in the connectedness of devices but in the fact that IoT applications can generate large amounts of data, which can be used to derive sensitive information about data subjects. To support this view, according to Singhai and Sushil (2021), the purpose of IoT is to collect information from users through a mainly wireless network of connected devices. As of 2020, there are over 50 billion interconnected devices worldwide. (Singhai & Sushil, 2021).

To further illustrate the importance and power of this concept to change the information privacy debate, Gartner, a technology consultancy company, predicted in 2017 that, by 2022, there would be 8.4 trillion linked things operating ubiquitously. (Gartner, 2021). The idea of these ‘things’ operating ubiquitously in the background provides a very fertile ground for the infringement of privacy of data subjects; hence, it is absolutely important to consider the influence of IoT on the information privacy debate. Similarly, the concept of cloud computing is also proving to have significant influence in the information privacy debate. According to Kalloniatis et al. (2013), cloud computing (simply referred to as Cloud) is a new generation of technology that has invaded our lives positively providing a number of capabilities that have made our digital behaviour much easier than before.

In fact, various well-known services such as email, messaging, databases, storage, networks, applications, and content management that were traditionally hosted in proprietary environments can now be consumed as cloud services over the internet. Furthermore, most personal data today resides in the cloud of major hosting companies such as Microsoft Azure, Amazon Cloud Service, and Google Cloud Service (Microsoft, 2019).

Like IoT, cloud computing is a very broad and complex concept that this study will not delve into but will review its impact in shaping the discussion on information privacy.

Transitioning to the regulatory concepts and how they impact the information privacy concept, we note the following concepts: fair information practices, privacy regulations and privacy standards. In terms of fair information practices (FIPs), they were introduced by the US Department of Health, Education and Welfare (HEW) in 1973 (Bellman et al., 2004). These FIPs principles, according to Dixon (2007), are a set of best practices that describe how in a data-driven global technology environment, organisations should handle, store, and manage information with fairness and with consideration for privacy. FIPs principles were further transcribed, in 1980, by the Organisation of Economic and Cooperation and Development (OECD) into eight principles commonly known as the OEDC Guideline of FIPs. Based on Dixon (2007), OEDC is known to make generally acceptable standards and best practices.

Furthermore, according to Dixon (2007), if these FIPs principles are consistently applied by firms it will go a long way to complying with some of the toughest global information privacy legislations, such as the GDPR of the EU, and the California Consumer Privacy Act in California, USA, host to the Silicon Valley.

According to Slane (2018), The EU, Canada, and the US have in recent times grappled with the issue of what fair information practices are when collecting processing, transmitting, and storing data. Especially in the era where technology has advanced to the point where we have 6 billion devices globally connected to one another. Hence, in the context of this study, FIPs is contributing to the information privacy debate and, if properly implemented, will help in improving the protection of information privacy. Building on FIPS from a regulatory perspective, this study examines two equally important concepts, namely privacy regulation and privacy standards, and how they impact the information privacy concept and debate.

First, in terms of privacy regulation, according to Akter et al. (2016), businesses, assisted by advanced information and communication technologies, rely heavily on customer data and advanced technologies such as big data analytics to shape their products and services (Allen, 2016). Furthermore, according to Allen (2016), big data analytic makes use of a mixture of consensual and non-consensual data gathered from the activities of customers not only to develop customised products or services but also to present supplementary products / services offerings to the customers, thereby opening the door for potential information privacy violation.

To support this view, Martin and Murphy (2017) posit that the widespread access to customers' data exposes them to fraud, invasion of privacy, unsolicited marketing and cyber-attacks. In addition, according to Nakagaki (2018), this dependency on consumer data to spur innovation by organisations comes at a cost to the customer's information privacy. For instance, the Cambridge Analytica incident (Rosenburg, 2018) involved the data of over 50 million customers from a born-global firm and was illegally exploited to influence their voting behaviour in the American presidential election. Nakagaki (2018) further suggests that companies should embed information privacy as part of their corporate strategy.

Hence, national and regional governments across the word have passed legislations to protect the privacy of their citizens. Some of these legislations are regional, such as the General Data Protection Regulation (GDPR) covering the European Union (EU), while others are country specific such as the Protection of Personal Information Act (POPIA) of South Africa, and India's Personal Data Protection Bill (PDPB) and finally, some are in-country state specific, such as the California Consumer Privacy Act.

In term of information privacy standards, there is no universally accepted privacy standard. However, there have been different common bodies of knowledge proposed, such as InfoPrivacy, championed by Lavranou and Tsohou (2019), although it has received very low validation by information privacy experts. Similarly, in January 2020, the National Institute of Standards (NIST) released its information privacy framework, which it describes as a tool to help optimise the beneficial use of data while protecting individual privacy (NIST 2020). NIST emphasises that the use of this tool is voluntary and not a law or a standard on its own. However, according to NIST, the proper use of this tool can contribute towards complying with different information privacy laws, such the GDPR and the California Consumer Privacy Act.

From the International Standards Organisation (ISO/IEC) perspective, the ISO standard ISO/IEC 27701 provides the standard to management of information privacy risk surrounding personally identifiable information (PII) within an organisation. This standard is an extension of the ISO/IEC 27001 standard dealing with information security programmes and controls within an organisation. Again, this standard is not a law or a formal regulation but will help organisations comply with regulations such as the GDPR.

In the context of this study, these standards are helpful to organisations to improve the protection of information privacy through practice. In the context of an organisation, to protect technology systems in practice, reference is made to two important information privacy concepts, namely information security and data protection, as illustrated in **Figure 2.1.**

These two concepts will be discussed together since information security is closely related to data privacy within any organisation. According to Singhai and Sushil (2021), the National Institute of Standards (NIST) has set the high-level goal of information security to be data / system integrity, availability and confidentiality. These goals form the backbone of every information security programme and is often referred to as the CIA of security.

To accomplish this goal, mechanisms such as encryption, access control, and authentication are used. These mechanisms, according to Singhai and Sushil (2021), have a direct bearing on data privacy as they are the same mechanisms that will be used to enforce data privacy. According to Cole (2015), information has become one of the biggest business assets in recent times and organisations across the world are scrambling to build large pools of information to gain a competitive advantage over their competitors (Cole, 2015).

In view of this, the threat to data privacy and information security comes in many forms and media such as computer viruses and worms (referred to as malware), social engineering, phishing, and identity theft, to mention just a few.

This increasing value of information within organisations has changed the approach of organisations towards protecting the data in their custody. Hence, the increased awareness and drive by organisations towards a culture of information security. In fact, according to Cole (2015), information security processes and procedures have become a big part of any successful business organisation and the role of information security professionals is becoming sought after in the job market and within organisations of all natures.

In terms of a formal definition, according to Teravainen (2020), information security is a set of strategies for managing the processes, tools and policies necessary to prevent, detect, document, and counter threats to digital and non-digital information. In summary, information security is about building a set of business processes, practices and policies to protect the information assets of an organisation from unauthorised access, modification or disclosure, whether this information is in transit, being processed, or at rest (storage).

In the context of this study, information security is weighted at the same level as information privacy. Later on in this study, the sister relationship between information privacy and information security will be examined, compared, and contrasted in the context of protecting personally identifiable information using configurable software objects.

After a detailed review of the different information privacy concepts and how they influence the objective of improving information privacy, the next section will look at the different theories guiding the information privacy debate, as seen through the lens of this study.

2.3.4 Key Theories in Information Privacy

To begin, Westin's 1967 theory of privacy addresses how people protect themselves by temporarily limiting the access of others to themselves (Westin, 1967). Through the lens of Westin, privacy is the claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others.

This view of privacy might have worked well in the early days of technology where privacy could only be defined in four states, namely personal anatomy, emotional release, self-evaluation, and limited and protected communication. Nowadays, with a plethora of advanced information and communication technology tools and concepts such as service-oriented architecture, social media, cloud computing, mobile computing, etc., coupled with numerous national and regional legislations regulating privacy, the dimensions of privacy have changed.

Moreover, this rather old-fashioned understanding of privacy is contrasted by the privacy calculus theory, which proposes that an individual's intention to disclose personal information is based on the trade-off between the risk and benefits envisaged. This theory is found in various works such as those by Klopfer and Rubenstein (1977); Laufer and Wolfe (1977); Posner (1981); and Stone and Stone (1990). They view the concept of privacy as not absolute but rather, as subject to interpretation in 'economic terms' (Klopfer & Rubenstein, 1977, p. 64). That said, in most instances, the privacy calculus is often used in conjunction with the risk calculus, the trade-off between privacy risks and the efficacy of coping mechanisms. These two trade-offs are together called the dual-calculus model.

A decision table based on the dual-calculus model is provided to predict an individual's intention to disclose personal information online. In a similar manner, the communication privacy management (CPM) theory suggests that 'individuals maintain and coordinate privacy boundaries with various communication partners depending on the perceived benefits and costs of information disclosure'. It was first developed by Petronio in 1991. This theory operates in the same frame as the dual-calculus model and is relevant in the behaviours of data subjects in recent times. Furthermore, according to Petronio (2010), 'after individuals disclose their personal information, the information moves to a collective domain where collectives (i.e., data subjects and data recipients) manage mutually held privacy boundaries'. This collective domain is nothing less than information systems and the information permeates across the boundaries of these systems.

Another interesting theory and perspective to privacy is the 'privacy regulation theory' formulated by Altman in 1975. The key aim of his theory is to explain why people sometimes prefer staying alone but at other times like to get involved in social interactions. According to Altman (1975), privacy is not static but 'a selective control of access to the self or to one's group'. Although Altman proposed the privacy regulation theory in 1975, well before the cyber age, recent studies have applied the theory to suggest new ways of thinking about privacy in socio-technical environments. Some of them state that, with information technology privacy extended from physical space to virtual space.

Effectively a new virtual context was created, which required a new balance as the physical boundary was stretched to a virtual one. This theory supports the context of this research project as it outlines two key elements of the research, namely selective control of access and the use of social technical systems. Altman (1975) believes that the goal of privacy regulation is to achieve the optimum level of privacy, which he describes as the state in which we can restrict access to ourselves as much as we want and at the same time enjoy the desired social contact that we want. However, owing to the dialectic nature of this theory, there are two extremes which he mentioned. They are:

1. Loneliness / isolation: This will occur when the actual privacy accorded to oneself is greater than the desired privacy.
2. Crowdedness or annoyance: This will occur when the level of privacy accorded to oneself is lower than the desired privacy.

Figure 2.2 is a diagrammatic representation of Altman's Privacy Regulation Theory (1975).

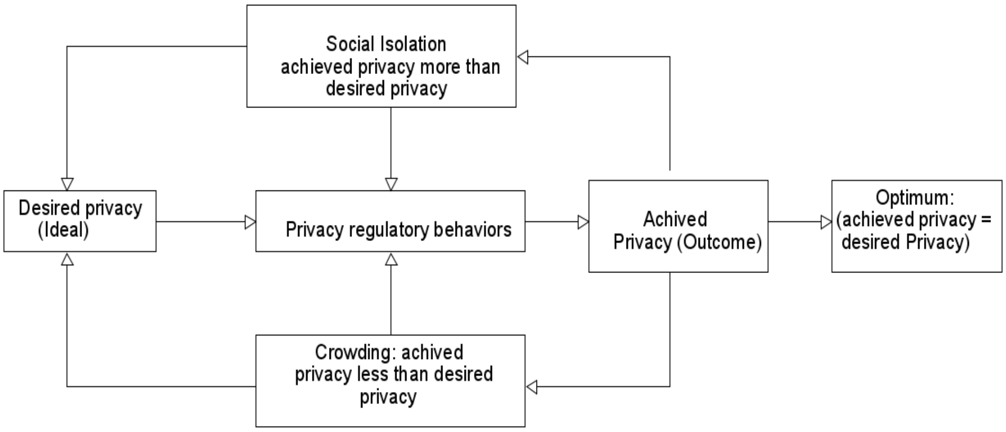

Social Isolation
achieved privacy more than desired privacy
Desired privacy (Ideal)
Privacy regulatory behaviors
Achived Privacy (Outcome)
Optimum: (achieved privacy = desired Privacy)
Crowding: achived privacy less than desired privacy

Figure 2.2: Altman's Privacy Regulation Theory

(Source: Adapted from Elprama et al., 2011)

According to Elprama et al. (2011), as illustrated in **Figure 2.2**, privacy is seen as a balance between social isolation and desired exposure. In order to regulate our privacy, we need to combine several mechanisms such as behavioural, environmental, and territorial realities. However, in the current technology context, there is one limitation to this theory which this study might contribute to address, thereby extending the theory. The limitation is that this theory was developed in 1975 and at that time, the cyberspace or the cloud in which today's data privacy needs to be enforced was not yet fully developed. As a result, the aspect of data handling when it is out of the control of the data subject was not adequately addressed.

Extending the privacy regulation theory to cover the regulation of data residing in cyberspace or the cloud and managed by third parties or proxies is thus necessary. Furthermore, according to Deloitte (2016), contracts with cloud service providers should define data protection standards and establish service level agreements (SLAs) that outline security and privacy measures. This leads to the main rationale of this study, which is to determine how best to enforce information privacy rules and compliance in technology systems responsible for the handling of personal information in organisations.

Another significant theory that this study makes reference to is Moor’s 2004 comprehensive theory of privacy, which characterises privacy as a situation in which an individual is protected from intrusion, interference, and information access by others. Moor’s theory is used in this study because it has incorporated all of the key elements of the other classic theories of privacy, namely non-intrusion, non-interference, and control/restricted access to personal information. (Moore, 2008).

Furthermore, the research instruments for building the software-based information privacy objects employ the elements of Moor’s theory to determine neither the traditional issues of privacy violation nor loss of privacy but specifically how the different information privacy regulations impact the operations of organisations.

2.3.5 Limitations of Key Theories in Information Privacy

The theories and ideas propounded by Westin and Altman have stood the test of time, considering that they still figure prominently in recent information privacy reviews such as that by Margulis (2011) in the Journal of Social Issues. More recently, they figured in Martin et al. (2015), a review presented at the Australian Conference on Information Systems in 2015. Based on their review of Westin and Altman, Martin et al. (2015) conclude that both authors focus on traditional privacy, which is based on the concept of personal information protection (PIP).

According to Smith et al. (2011), PIP was primarily the responsibility of the individual, based on the view that privacy is an inherent right of the individual to be left alone. Smith et al. (2011) thus consider that only the individual will determine whether they want to maintain that right or not. This view draws heavily on Westin’s 1967 definition of privacy as the right to define for oneself when, how, and what extent of information should be released or shared.

However, according to Martin et al. (2015), Westin does not address how information is managed once it leaves the data owner and is in the custody of the organisation or entity responsible for its management. This is especially true in the context where this data is exchanged between entities collecting the data and entities disclosing the data to third parties.

This limitation is made more visible with the advent of the internet and the fourth industrial revolution, creating an explosive growth of new technologies and contexts of data exchange and management such as mobile / wireless technologies, cloud computing, internet of things (IoT), electronic commerce, social media and e-learning. With reference to all of these new and modern trends and contexts of data exchange and management, privacy is seen as a context-driven phenomenon.

To support this assertion, different privacy laws such as the General Data Protection Regulation (GDPR) for the European Union, Fair Information Practice Principles (FIPP) and Health Insurance Portability and Accountability Act (HIPAA) for the United States Federal Government, and the Protection of Personal Information Act (POPIA) for South Africa, are all flagship laws passed to provide geographic region-specific requirements in terms of how privacy must be handled by entities operating within each specific region.

2.4 Gaps in Current Literature

Based on the review of current literature and the examination of the different concepts influencing the information privacy debate and discussion within the privacy research community, as highlighted in **Section 2.3,** the following four points have been identified as gaps in the current research in the context of this study:

1. Previous research has focused on theoretical frameworks;
2. Lack of a practical design framework to enforce privacy as technology objects;
3. Lack of standardisation in methods of enforcing information privacy controls; and
4. Lack of a mechanism to enforce information privacy using a multi-contextual software object.

To justify these gaps the following points are highlighted from existing literature review.

1. According to Dias Canedo et al. (2020), information privacy violation can be prevented if privacy requirements are properly elicited in the early stages of a system development process that exists both at the functional and non-functional requirements gathering phases of the Software Development Lifecycle (SDLC). Regrettably, many current systems and platforms still fail to protect user privacy

because privacy is an afterthought of system design (Papacharissi & Gibson, 2011).

2. There is a lack of a practical design framework for implementing information privacy requirements into technology systems responsible for storing, processing, and transmitting personal information or personal data. This view is supported by Wu et al. (2019), when they posit that there is a still research gap in proposing practical and innovative privacy enhancing solutions, despite the enormous resources that have been invested in multidisciplinary privacy research. For instance, since the research conducted by Bélanger and Crossler (2011), highlighting the opt-in and opt-out mechanism as a means of enforcing privacy, and concluding that it is inadequate; to date, social network sites are still using the opt-in and opt-out mechanism for privacy protection (Liu et al., 2011; Wu, 2019). Fewer studies to date have proposed an empirically tested alternative mechanism for information privacy enforcement.
3. Lack of standardisation in methods of enforcing information privacy controls. This view is supported by Fenghua et al. (2019), when they conclude that most information privacy schemes are focused on relatively isolated software application scenarios and technical points and mostly proposed solutions to specific problems within an application scenario. For instance, according to Fengua et al. (2019), the problem of privacy in large data environments such as the cloud, social networks, and cyberspace still remains a problem. For this reason, this study proposes how information privacy can be protected using multi-contextual software-based objects.
4. The use of context in information system research and design is key (Wu et al, 2020). According to Dias Canedo et al. (2020), protecting information privacy in software systems is a complicated issue that encompasses several aspects such as privacy regulation, international standards, organisational controls and methodological support, and most importantly, the software developers' perceptions and technologies. In fact, according to Biselli and Reuter (2021), the current insufficiencies and possibilities for improving information privacy is being studied intensely. Hence, there is a requirement in the body of knowledge to improve on the level of technology used to protect information privacy.

5. Wu et al. (2020) write that the popular Westin 1967 conceptualisation of privacy does not align with the reality of the digital world of today. They suggest that a growing number of privacy scholars are advocating for a more contextual approach to information privacy with emphasis on the conditions and context guiding the individual's disclosure of privacy.

Based on the gaps identified through the review of past and current literature on information privacy highlighted here, this study will dwell on these gaps to contribute positively to the body of knowledge on information privacy in line with the objectives of this study as outlined in **Section 1.5 of Chapter 1**.

Other contributions are elaborated at the end of the study once the prototype is developed, implemented and evaluated in context and in practice as part of this research study.

2.5 Conclusion

This chapter examined the literature underlying the main theme of this study, information privacy as it is understood by modern organisations in terms of compliance and technology. The chapter also elaborated on the key theories of information privacy, how it has evolved over time, and the state of information privacy research at the time of this research through a comprehensive literature review.

Furthermore, to understand the role of information privacy fully in the right context within organisations, this chapter explored related concepts such as information security and privacy engineering, data protection, IoT, PbD, privacy standards, and privacy regulations to find the areas of intersection and specialisation.

In conclusion, this chapter reviewed and presented relevant literature on the information privacy subject matter leading to Chapter 3 entitled Privacy Engineering. This chapter focuses on how information privacy and related concepts can be engineered into information systems handling personally identifiable information.

CHAPTER 3

Privacy Engineering

3.1 Introduction

In Chapter 2, information privacy concepts and theories were introduced and discussed in line with past and existing literature on information privacy. Building on this, Chapter 3 focuses on privacy engineering (PE). This chapter examines how information privacy concepts and theories can be engineered as technology system controls to protect information privacy within organisations.

The chapter is divided into nine (9) sections. Section 3.1 is entitled Background to Privacy Engineering and covers the introduction and definition of privacy engineering. Concluding this section, related literature is examined on privacy engineering.

Section 3.2 is entitled Definition of Privacy Engineering and examines two important aspects of privacy engineering, namely privacy engineering-related disciplines (for example, requirements engineering, system engineering, data engineering) and privacy engineering key concepts including privacy engineering frameworks and tools, meta-models, and Privacy by Design.

Section 3.4 outlines the privacy engineering process and adapts this process to the current research context, which is aimed at building multi-contextual software-based objects to protect information privacy using the design science paradigms.

3.2 Definition of Privacy Engineering

According to NIST (2020), privacy engineering (PE) is a new and developing discipline primarily for the purpose of building trust in information systems designed to handle personal information (NIST, 2020). Given recent concerns regarding how information systems and technology can affect the privacy of individuals, their social levels and identities, privacy engineering is emerging alongside similar disciplines such as information security, cloud computing, and big data (NIST, 2020). The aim of privacy engineering is to build privacy protection in information systems at design stage as well as to offer scaled privacy protection. NIST (2020) describes privacy engineering as an emerging field of study, and its exact meaning is still evolving. According to NIST, privacy engineering brings tools, techniques, metrics, and taxonomy to implement Privacy by Design (PbD).

The most widely accepted definition of privacy engineering is the definition proposed by NIST (2020), as ‘a specialty discipline of systems engineering focused on achieving freedom from conditions that can create problems for individuals with unacceptable consequences that arise from the system as it processes PII’ (NIST, 2020 p59). This definition takes into consideration three (3) important aspects of this study. First, it concerns the individual whose information is at risk, herein referred to as the data subject. Second, it addresses the consequences of non-compliance with information privacy provisions. In the context of this study, and according to POPIA, these consequences include fines and penalties of up to R10 million and even a three-year prison term. Third, it considers the information system or systems responsible for the handling and processing of PII.

Considering the importance of the system component, as highlighted by the NIST definition, this study delves more into the PbD concept in **Section 3.5,** which is essentially to design and incorporate privacy techniques during the inception of new systems. According to NIST (2020), embedding privacy as such is the goal of modern information system owners as they are required to comply with the information privacy or data protection laws of their respective countries and regions.

A more contextual description of privacy engineering is provided by Martin and Del Alamo (2017). In their description, privacy engineering is seen as a multidisciplinary concept, and is thus subject to multiple reference frameworks and paradigms, including social, legal, regulatory, risk or technical, and even industry-driven standards. The relevance of this definition is highlighted in the following sections, where the disciplines related to privacy engineering are elaborated upon.

A more succinct definition of privacy engineering is that provided by Watson (2019), as ‘the systematic application of engineering concepts for protecting sensitive information’.

The following two sections of this study delve into some of the related literature on privacy engineering (**Section 3.3**) and the related disciplines of privacy engineering (**Section 3.4**).

3.3 Related Literature on Privacy Engineering

According to Hoel et al. (2020), privacy engineering is described as the deliberate approach of interjecting data protection requirements into complex systems based on regulatory and ethical corporate strategies.

The highlight of this description is the concept of deliberate interjection of privacy rules into information systems, resulting in privacy no longer being considered as an afterthought. In line with his definition, according to MITRE (2019), privacy engineering focuses on 'methods and standards, technical elements of information infrastructure and individuals and collectors of personal data with the goal to integrate privacy into existing systems and engineering processes'. Furthermore, Watson (2021) defines privacy engineering as the systematic application of engineering concepts for protecting sensitive information.

First, this definition highlights the fact that privacy engineering is not an island on its own, but that it exists alongside and within other engineering disciplines. **Section 3.4** expands on these related disciplines and their relationship with information privacy engineering.

Second, the protection of sensitive information is highlighted. To understand how this sensitive information is protected, the information privacy concepts need to be understood. Some of these concepts include Privacy by Design, privacy patterns, privacy frameworks and tools, to mention a few.

Finally, to understand what sensitive information is, the information privacy laws, regulations and standards need to be considered, as in **Section 1.1**. Therein, the definition of the data elements of sensitive information are elaborated upon in the right context.

3.4 Privacy Engineering-Related Disciplines

Privacy engineering does not exist on its own. As a new and developing discipline, privacy engineering is related to several other disciplines, as highlighted by Martin and Del Alamo (2017) in **Section 2.3.**

To expand on the related disciplines, the definitions of privacy and privacy engineering provided by MITRE are called upon as well as the description of privacy engineering provided by Martin and Del Alamo (2017) in **Section 3.2.**

MITRE (2021) defines privacy as the ability of an individual to control the collection, use and dissemination of his or her personally identifiable information (PII). PII can be defined as any information which can be used to trace an individual's identity. (MITRE, 2021). Furthermore. MITRE (2021) defines privacy engineering as a systematic, risk-driven process that operationalises the Privacy by Design (PbD) philosophical framework within IT systems (MITRE, 2021).

Mindful of these definitions, all of the disciplines that are involved in the scope of collection, use and dissemination of information are related to privacy engineering. More specific to the disciplines that directly help to formulate and operationalise technology systems, **Figure 3.1** shows some of these disciplines and how they are related to privacy engineering.

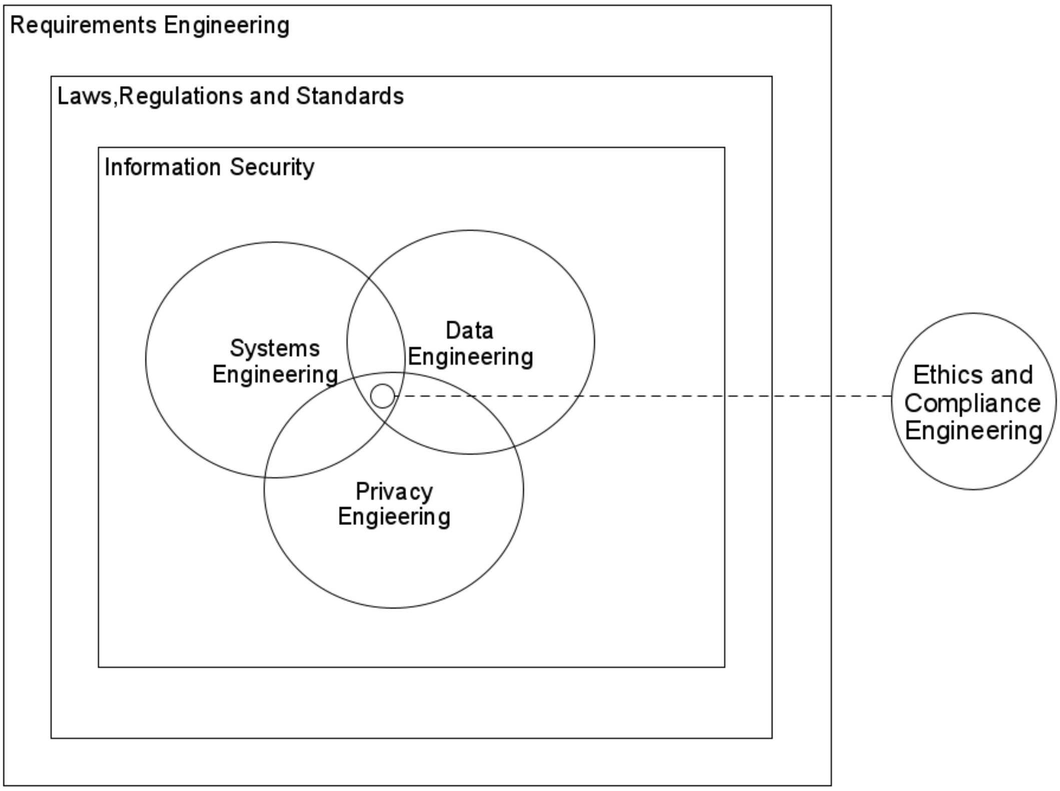


Figure 3.1: Privacy Engineering in Context

Figure 3.1 highlights six (6) related disciplines linked to the privacy engineering concept, inspired by the definitions of privacy engineering and information privacy by MITRE (2021) in **Section 3.4**. **Figure 3.1** takes an outside-in approach, starting with the premise that all privacy engineering activity is based on a set of requirements which informs the applicable laws, regulations and standards applicable to the privacy engineering context. For example, if the organisation is operating in South Africa, then the requirements will inform the need to comply with the Protection of Personal Information Act (POPIA).

Moving inward in **Figure 3.1**, the section on applicable laws, regulations and standards defines the information security context to which the engineering of privacy takes place. For instance, if the organisation is operating in the European Union (EU), then GDPR regulations are enforced and the GDPR mandates the ISO/IEC 27001 security framework to be complied with.

Moving further in, based on **Figure 3.1**, to implement privacy engineering at a technical level, the systems engineering, and data engineering disciplines are required.

At the core of all these related disciplines lie ethics and compliance engineering. This central position occupied by the ethics and compliance engineering discipline, is in line with the assertion that the outcome of privacy engineering is compliance and the ethical handling of personally identifiable information.

The following sections examine these related disciplines and highlight their relationship with the privacy engineering disciplines in context.

3.4.1 Requirements Engineering

According to Nuseibeh and Easterbrook (2000), requirements engineering involves defining, documenting, and maintaining requirements in the design process and plays a key role in systems engineering and, by induction, privacy engineering. Nuseibeh and Easterbrook (2000) further posit that the primary measure of success of any technology system is the degree to which it meets the purpose for which it was intended, and requirements engineering is the path to establishing that purpose.

A formal definition of requirements engineering is provided by Zave (1997). He defines requirements engineering as:

> *'The branch of software engineering concerned with the real-world goals for, functions of, and constraints on software systems. It is also concerned with the relationship of these factors to precise specifications of software behaviour, and to their evolution over time and across software families'.* (Zave, 1997 p106).

This definition highlights a number of important points. First, that requirements are 'real-world goals' that software engineers hope to achieve in a software system. Second, the relationship between the different components of a software system is highlighted. Third, the precise specification is also mentioned. This forms the basis for validating and verifying requirements and, in the case of privacy engineering, the precise units of measurements of the privacy metrics. According to Nuseibeh and Easterbrook (2000), requirements engineering came to prominence in the 1990s with the International Requirements Engineering Conference, which was granted an 'A' rating from both Australian and Brazilian rankings of Information and Communications Technology (ICT) conferences.

Based on Royce (1970, pp.1–9) and Somerville (2009), referencing the waterfall software development methodology, requirements engineering constitutes the first phase. However, in latter systems engineering methodologies such as Agile and Rational development methodologies, requirements engineering continues throughout the lifecycle of a product development process. According to Sommerville (2009), some of the activities involved in the requirements engineering process include:

1. **Requirements elicitation:** This is the actual first step in the requirements engineering process. In this step, the requirements are extracted from the stakeholders of the intended system, interpreted, analysed, modelled, and validated to ensure that they are complete.
2. **Requirements analysis:** This is where requirements are analysed, both old and new, and conflicts with stakeholders are handled. To perform this analysis, text and graphic analysis tools are used, for instance UML use cases.
3. **Systems modelling:** Modelling involves creating a design blueprint of the product visually using a modelling tool, such as Lifecycle Modelling Language or UML, to enable the stakeholders to approve the design before the actual development or fabrication starts.
4. **Requirements specification**: In terms of specification, the requirements are documented into a formal artefact called requirements specification (RS).
5. **Requirements management:** This involves managing all the activities related to the requirement engineering lifecycle, from inception right through to completion- and post-completion activities, such as changes and extensions.

Though these activities are presented here in a chronological order, in practice there is considerable interleaving of these activities. To carry out these activities, the different stakeholders of the system or product need to be engaged through formal and informal processes to establish their needs and expectations of the system. Critics of requirements engineering have advanced arguments that it reduces design performance and sometimes the entire requirements engineering exercise results in a situation where requirements do not exist. In other instances, design decisions are misconstrued for system / product requirements.

In the context of this study, the requirements engineering activities were encoded into software-based objects and implemented into information systems as business rules using the business rules approach. According to Gougeon (2003), business rules have been defined and redefined over the years. In the context of this study, business rules and the business rules approach are used to implement the POPIA regulatory requirements as software objects in information systems.

To date, there is no generally accepted formal definition of a business rule. Loosely speaking, a business rule is a natural statement that describes a constraint related to a business process or activity. For example, a business rule can be a statement like:

a) In the case of a business organisation: The selling price of all shoes below size 7 must be discounted by 20 per cent for the next two months.

b) In the case of a country or national government: All national borders of South Africa should be closed for all non-business travellers until further notice.

Mindful of these examples, it can be concluded that business rules are at the heart of any business and they represent the core business policies of any organisation (Valatkaite & Vasilecas, 2004). From among the many and diverse definitions of business rules proposed, this study will retain the following two, which suit the context of the research undertaken.

1. A communication tool that expresses the rules and policies of an organisation as they relate to data (Sandifer & von Halle, 1991).

2. A constraint placed upon a business organisation (Moriarty, 1993).

The highlights of these two (2) definitions are that they both bring to life two major concepts on which this study depends heavily. The first concept is the tool which, in the context of this study, is the artefact used to handle personally identifiable information within an information system. The second concept is the organisation.

This study focuses on helping organisations conform to information privacy laws using configurable software objects. The business rules approach to information systems development arose from the growing need for businesses to manage their knowledge explicitly and to map it effectively into the information systems used in daily business operations (Valatkaite & Vasilecas, 2004). According to von Halle (2002), the business rules approach is the most efficient approach for building rules in information systems in terms of roll-out efficiency. This approach enables easy, changeable and faster rule building than previous approaches. Von Halle (2002) adds that a system built using the business rules approach has many advantages compared to a system built using the traditional approach of hard coding business rules in the system logic.

Until recently, business rules were formalised as system requirements and hard coded into software systems logic. This view is supported by Simsion (1993) through his declaration that, in traditional information system development methodologies, rules were not treated as modelling formalisation at the early stage of the development of a system. Rather, they were hidden in database constructs and procedural code.

This approach worked well when systems were still small. As systems grew in functionality and complexity, however, it became a nightmare to locate these rules within the information systems and to apply necessary updates or modifications. The essence of business rules is captured by Moriarty (1993), who suggests that system analysts are still striving for a paradigm that can bridge the communication gap between businesspeople and information systems professionals. It is this gap that the business rules approach seeks to bridge. However, the most significant benefits of the business rules approach can be narrowed down to these two points:

1. The business rules approach helps to design systems in a way that they can easily accommodate business rule changes with minimal system disruptions. This is particularly important for modern and dynamic business organisations where marketing time is very short and change is constant.

2. The business rules approach introduces tracking, which is vital for large systems such as enterprise resource planning systems. This is because such systems host thousands of business rules and also require that these rules be modified frequently to align with legislation and other business imperatives and directives. A case in point is the Covid-19 pandemic, in response to which businesses were required to suspend taxes and implement rate changes on certain products and commodities. Said changes include the deferment of excise taxes on alcoholic beverages and tobacco products

(SARS, 2020). This measure was a direct response to the restrictions placed on the sale of alcoholic beverages and tobacco products to ease pressure on South African hospital emergency sections. According to Theodoulidis and Youdeowei (2000), when using a business rules approach within an information system, emphasis should be placed on the analytic methods and the system architecture that is relevant to support the business rules environment.

3.4.2 Laws, Regulations and Standards

To understand the role of information privacy laws, regulations and standards in general, Habermas (2010, p. 473), frames information privacy as a fundamental aspect of human dignity that needs to be protected in the context of new invasive technologies applied to customer data, especially in countries where information privacy laws are still in their infancy or are not in sync with those in the developed markets.

This view is supported by the newly enacted global data protection regulation (GDPR), which came into force on 25 May 2018 for the European Union (EU) and positions data protection as a fundamental human right in **Chapter 1, Article 1** of the legislations. According to Campbell et al. (2015), information privacy laws impose certain constraints on organisations regarding how to handle the information of their customers and employees, and in context, they will have to comply with the applicable information privacy legislation pertaining to the geography in which they are operating.

Based on studies conducted by the United Nations Conference on Trade and Development (UNCTAD, 2020), only 66 per cent of countries in the world, mostly in developed economies, have any sort of information privacy legislation, while the remaining 34 per cent do not have any form of information privacy legislation

Furthermore, for those who have some form of legislation, these legislations vary from country to country and are applicable only to specific regions that the specific law covers. For instance, some of these legislations are regional, such as the General Data Protection Regulation (GDPR) covering the European Union (EU), while others are country specific such as the Protection of Personal Information Act (POPIA) of South Africa and India’s Personal Data Protection Bill (PDPB), and finally some are in-country state specific, such as the California Consumer Privacy Act.

As highlighted in **Section 1.1**, disruptive 4IR technologies are eroding the information privacy of data subjects across the globe. To support this assertion, according to Akter et al. (2016), companies are relying heavily on customer data and advanced technologies such as big data analytics to shape their products and services. Big data analytics makes use of a mixture of consensual and non-consensual data gathered from the activities of customers not only to develop customised products or services, but also to present supplementary products and services to customers. This opens the door for potential information privacy violation by organisations both public and private since customers do not always understand the full extent to which their data is collected, analysed, and commoditised by these organisations.

In general, according to Martin and Murphy (2017), the widespread access to customers’ data exposes them to fraud, invasion of privacy, unsolicited marketing and cyber-attacks. In addition, according to Nakagaki (2018), this dependency on consumer data to spur innovation by organisations comes at the cost of the customers’ information privacy, for instance, the Cambridge Analytica incident (Rosenburg, 2018) where the data of over 50 million customers on Facebook was illegally exploited to influence their voting behaviour in the American presidential elections of 2016. Nakagaki (2018) further suggests that companies should embed information privacy as part of their corporate strategy.

Moving away from information privacy laws, industry regulatory bodies, such ISO/IEC and NIST, have also recommended best practice regulations and standards to guide the protection of information privacy. Some examples of these best practice regulations are:

- ISO/IEC 27701 presents ISO/IEC’s Privacy Information Management System (PIMS), which outlines a framework for personally identifiable information (PII) controllers and PII processors to manage data privacy.
- ISO/IEC 27001 presents ISO/IEC’s Information Security Management System (ISMS) as a framework for ensuring the confidentiality, integrity, and availability (CIA) of information, as well as compliance. It should be noted that there is significant overlap between the ISO/IEC27001 and the ISO/ IEC27701.

- The NIST Privacy Framework is a tool for improving information privacy through enterprise risk management.
- The NIST cybersecurity framework is a tool to prioritise cybersecurity risk for enterprise risk management. It is noted that the NIST Privacy Framework and the cybersecurity framework are designed to complement each other.

|In the context of this study, to conduct privacy engineering fully of a technology system, these laws, regulations and best practice standards should be examined and incorporated into the requirements of the technology product in line with its scope and relevance. These arguments are summarised explicitly by the international association of privacy practitioners (IAPP) when they conclude that privacy laws mandate privacy engineering in practice to take effect and it demands that organisations implement appropriate organisational and technical measures to enable data protection principles and safeguards (IAPP, 2021).

3.4.3 Information Security

According to Cole (2015), information has become one of the biggest business assets in recent times. In fact, the value of an organisation lies in the information it holds. Organisations across the world have taken cognisance of this and are scrambling to build large pools of information, in order to get an edge over their competitors.

This increasing value of information within organisations has changed the way that organisations approach the protection of the data in their custody. The security of such information is critical for business operations and for maintaining credibility in the face of their clients. Hence, the increased awareness and drive by organisations towards a culture of information security.

Cole (2015) further reveals that information security processes and procedures have become a big part of any successful organisation, and information security professionals are becoming highly sought after in the job market. The demand for these professionals is increasing in all types of organisations. Teravainen (2020) defines information security as ‘a set of strategies for managing the processes, tools and policies necessary to prevent, detect, document and counter threats to digital and non-digital information’ (Teravainen, 2020). In other words, information security is about building a set of business processes, practices, and policies to protect the information asset of an organisation from unauthorised access, modification or disclosure, whether this information is in transit, being processed, or at rest (storage).

Nowadays, many organisations employ a dedicated security team to implement the information security programme of the organisation, under the leadership of a Chief Information Security Officer (CISO). According to TechTarget (2020), the information security programme is built around the CIA triad. CIA is the acronym for confidentiality, integrity, and availability. To unpack these important concepts, confidentiality ensures that sensitive information is only disclosed to authorised parties, while integrity takes care of the unauthorised modification of information, and availability refers to the uptime and guarantees that the data and systems can be accessed by authorised parties at all times. As expanded upon in this study in **Chapter 7**, one of the key techniques of enforcing information security is by means of encryption. The threat to sensitive data and privacy comes in many forms and media such as computer viruses and worms, referred to as malware. Other threats include social engineering, phishing and identity theft, to mention a few. It is common knowledge within the security community that absolute security does not exist. Hence, the ultimate goal of any security programme in any organisation is to maintain the confidentiality, integrity, and availability of its information systems and the underlying business data to an acceptable degree to enable and ensure the comfort of stakeholders.

In the context of this study, as illustrated in **Figure 3.1**, information security is a broader discipline and privacy engineering is an integral part of it as it works towards enforcing information security within an organisation. More precisely, information security is viewed in the context of protecting personally identifiable information using configurable software objects.

3.4.4 Systems Engineering

According to the systems engineering body of knowledge stipulated in the ISO/IEC notes, systems engineering is defined as an interdisciplinary approach (technical and managerial) to transform a set of customers’ needs, expectations and constraints into solutions (ISO/IEC/IEEE, 2010). Similarly, according to the

(ISO/IEC/IEEE, 2010), software engineering is a subset of systems engineering and, in the context of this study, the focus is on system engineering as the overarching discipline.

The National Aeronautics and Space Administration (NASA) (2020), provides a more contextual definition of systems engineering as ‘a methodical, multidisciplinary approach to design systems and manage them to their retirement’. This definition emphasises the aspect of systems and further describes a system as a combination of elements that function together to produce the capability required to meet a need. These elements could include hardware, software, people, and procedures, to mention these few.

Privacy engineering is intimately linked to systems engineering and cannot be discussed in isolation without first considering the concept of information systems as per the definition of MITRE in **Section 3.4.** Information systems involves a variety of information technologies. According to Boell and Cecez-Kecmanovic (2015), an information system involves computers, databases, communication systems, the internet, mobile devices, and much more, that serve to perform specific tasks, interact with and inform various actors in different organisational or social contexts. Furthermore, Boell and Cecez-Kecmanovic (2015), in their literature review on information system definitions, identify four distinct conceptualisations of information systems, namely a technology view, a social view, a socio-technical view and a process view.

Another definition views information systems as a tool in the world to be used by humans to support their day-to-day activities. It holds that ‘information systems are primarily intended to model the states and behaviour of some existing or conceived real-world system’ (Wand & Weber, 1990). Adapting these definitions to the context of this study, an information system can be viewed simply as a set of tools used by organisations to store, process, transmit, and transform data, and more specifically, personally identifiable information. According to Stair (2009), information systems have four major parts, namely input, processing, output and feedback. **Figure 3.2** illustrates the relationship between these major parts, with data being the major input and information being the major output. Between data and information, there is a process to transform or give meaning to the data championed by the information system. At the top of the information system is a feedback mechanism, which passes messages back and forth between the data and information states. For instance, if the data was successfully processed into information, the system will notify all the stakeholders that the operation was successful. This process holds true for any type of information system.

The above description is similar to the two-part definition of information systems cited by Alter (2008), where information systems are characterised as a work system, in which:

1. Processes and activities are devoted to processing information, i.e., capturing, transmitting, storing, retrieving, manipulating, and displaying information
2. Human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce specific products and/or service (Alter, 2008. P100).

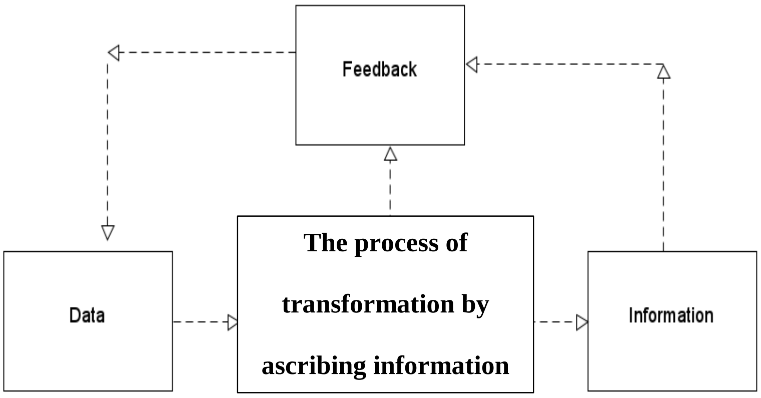


Figure 3.2: The Major Parts of any Information System

The four major parts highlighted in **Figure 3.2**, are deemed to represent the context of this study adequately and will help in building the use case scenarios for the POPIA rule engine prototype designed to test the assertions of this study, as highlighted in **Chapter 7**.

3.4.5 Data Engineering

According to Black and Steel (2017), as engineers design and build things, data engineering involves the building of pipelines that transform and transport data into formats that are usable to data consumers and practitioners.

From a historic perspective, data engineering is an evolution of the term information engineering that was coined in the 1980s to describe database design and data analysis, as required by software engineering. With the evolution of the internet and data-driven technology in the 1990s and 2000s, large technology companies, such as Facebook and Google introduced the |term ‘data engineering’ to describe the role that moved away from the traditional extraction, transformation, and loading (ETL) developer roles to roles designed to handle large volumes and high velocity of data, quickly and correctly. According to Taylor (2015), today data engineering has evolved to an aspect of software engineering that focuses on data, data infrastructure, data warehousing, data mining, modelling, crunching, and metadata management.

From this definition, according to Vikram (2021), data engineering is a technical role responsible for the architecting, building, and maintaining of a data system. In the context of this study, the data engineering principle is employed in privacy engineering to understand the data and the data infrastructure involved in privacy protection as mandated by either the privacy regulation, privacy standard, or privacy law that is applicable in a particular instance. Hence, the next section examines ethics and compliance engineering as they define the scope and instances of regulations and best practices that are applicable during privacy engineering.

3.4.6 Ethics and Compliance Engineering

Ethics and compliance engineering span across many professional disciplines and fields of study, such as law, medicine, engineering, military, communications, and aviation, to mention a few. In the context of this study, ethics and compliance engineering is examined in two particular domains, namely legal and technology to align with the research focus of this study.

Compliance engineering is described as ‘a combination of general engineering principles and human risk factors with an interpretation of the legal requirements and regulations’ and defined broadly as ‘designing and developing products to meet the applicable market compliance requirements’ (Bayswater, 2016).

Ethics, on the other hand, according to Cavalier (2014), is defined as ‘a branch of philosophy that defines and recommends concepts of right and wrong in terms of human behaviour’ (Cavalier, 2014). This study does not delve into the broad philosophical debates around the subject of ethics; rather, it highlights two important ethical principles, which are, ‘right’ and ‘wrong’ in terms of human behaviour.

To adapt these ethical principles into context, this study takes the position that the essence of privacy engineering is to promote the right ethical behaviour amongst humans operating in organisations. To support this argument, according to Mai (2016), people reveal personal information consciously or unconsciously, willingly or unwillingly as they perform their daily activities such as shopping, communicating with family members, or even reading the news online. It is therefore ethically right to protect the privacy of the users of these systems by using privacy engineering tools and techniques.

From a technological perspective, according to Wagner and Eckhoff (2018), recent innovative technologies such as artificial intelligence and robotics are raising ethical issues around regulation, governance, and humanity. These technologies fall under the banner of the fourth industrial revolution (4IR). It is recorded that the fourth industrial revolution or industry 4.0 is the successor of the first, second, and third industrial revolutions. The first industrial revolution started in Britain around 1760 with major inventions such as the steam engine, which powered manufacturing and factories. It led to the second industrial revolution, which was characterised by mass production in industries like steel, oil and electricity, and notably the invention of the telephone, light bulb and the internal combustion engine for automobiles (Schulze, 2019). Following the second industrial revolution, is the third, which started in the 1960’s. It is popularly known as the digital revolution. Most economies, especially in developing countries like South Africa, India, Brazil, and Malaysia, are still locked in this digital revolution while developed economies are making inroads into the fourth industrial revolution. The fourth industrial revolution is a characterisation of the current developing environment in the world in which disruptive technologies and innovations, such as internet of things (IoT), robotics, virtual reality (VR) and artificial intelligence (AI) are changing the way modern people live and work (Wigmore, 2020).

To regulate such incursions, professional bodies such as the European Group on Ethics in Science and New Technologies (EGE) have been set up and tasked with providing guidelines on ethical principles and democratic prerequisites for such technologies. In the context of this study, the focus is on two major categories of technologies that are shaping the ethics debate, namely privacy enhancing technologies and privacy-invading technologies, each sitting on opposite sides of the ethics fence. First, privacy enhancing technologies (PETs) come before the backdrop of massive risk to information privacy caused by the heavy reliance on data by organisations to expand their business operations and create new innovative products and services enabled by 4IR technology trends.

This view is supported by Noble of the Privacy Enhancing Technology Work Group, who declares that the scale and rate at which data is collected, used and analysed is rapidly offering significant new and developing benefits to society and the economies of most countries and organisations. However, there needs to be a balance between exploiting the data and the risk it poses to the data subject whose personal data is being used by the organisation (Noble, 2019). Failure to mitigate adequately the risk to data subjects may result in reputational damage for the organisation, and in some cases, civil lawsuits. This might also result in organisations not being able to use technologies to derive benefits from the data in their custody. However, certain risk can potentially be mitigated and managed with a set of emerging technologies and approaches collectively referred to as privacy enhancing technologies (PETs).

The focus of PETs is different from that of information security in the sense that, while information security is focused on preventing unauthorised access to data, PETs are focused more on deriving useful analysis and results from data without enabling the requestor to access or deduce the data, in order to protect the data owners and data custodians. According to Noble (2019), this set of technologies will promote greater use and access to data in a trustworthy and privacy preserving manner.

Going forward, PET and its discipline is still new and evolving. This study does not delve into the details of privacy enhancing technology but recognises the fact that it impacts the way privacy legislations are rolled out and the way technology is used to process and handle data in the long run.

In contrast to privacy enhancing technologies are privacy-invading technologies (PITs). According to Klitou (2014), there is a rapid rise in the development and use of PITs globally, which is posing a serious challenge to the enforcement of information privacy. Also, according to Klitou (2014), PITs are developed owing to the need to enhance public and personal security but recently they have been the subject of serious violation to privacy. Examples of PITs include:

a) **Body cameras** worn mostly by law enforcement officers operating as first responders, or on the frontlines of crime-prevention operations, in some countries such as the United States of America (USA).

b) **Public space CCTV microphones, loudspeakers and cameras** are used in many metropolitan cities such as London, New York, Seoul, and Moscow, as well as in streets and on street corners of some other cities, and school campuses.

c) **Human-implantable microchips (RFI/GPS),** to track high-profile prisoners and employees of some high-profile security establishments.

Ultimately, although these PITs provide efficient security protection for the state and for some of the users, it also comes with a high cost to the privacy of the latter. This study will not delve into the details of the effect of these PITs on information privacy nor into the ethical issues linked to PITs. Instead, it will frame PITs as another breed of technologies that poses a challenge to the enforcement of information privacy. In terms of compliance and compliance engineering, compliance on its own is a very broad subject, touching on many disciplines and areas of study that fall out of the scope of this study.

According to Le Grand (2020), compliance can be defined broadly as the act of following rules and, according to Dias Canedo et al. (2020), these rules are often external rules or requirements, for instance, compliance to POPIA, which is a requirement imposed by the government of South Africa on organisations, both public and private. Compliance also involves following the organisation's internal rules, policies, and procedures, and acting in accordance with ethical practices.

However, in the context of this study, the issue is not compliance on its own, but rather the management of compliance. According to Le Grand (2020), compliance management refers to how organisations ensure that they are operating in accordance with the rules, laws and regulations and other requirements to which the organisation is subject. According to GAN Integrity (2021), for companies to achieve compliance, they are required to build a compliance culture. This culture is defined as when compliance is a central and unalterable part of the corporate culture of the organisation. However, this does not happen overnight. It requires the organisation to impose strict adherence to compliance and ethics at every level of the organisational structure. This should include top management, middle management, junior management, and even the lower-level staff of the organisation, such as drivers and cleaners (GAN Integrity, 2021).

To support this view, Le Grand (2020), argues that compliance management involves oversight, assessments, reporting and remediation on actions requiring compliance. Some international industry regulations like Sarbanes-Oxley (known as Sox) have been enacted to enforce compliance and compliance management in financial institutions globally as part of their broader risk management framework. To build a culture of compliance, GAN Integrity (2020) suggests that organisations incorporate compliance in their vision, mission and value statements since these constitute the pillars of the organisation to be respected by every employee. They also suggest that management should communicate constantly the values of compliance and ethics to the rest of the organisation.

As an example, according to Le Grand (2020), the list of compliance requirements is growing steadily for most organisations and is seen to involve the following areas of their business and industries.

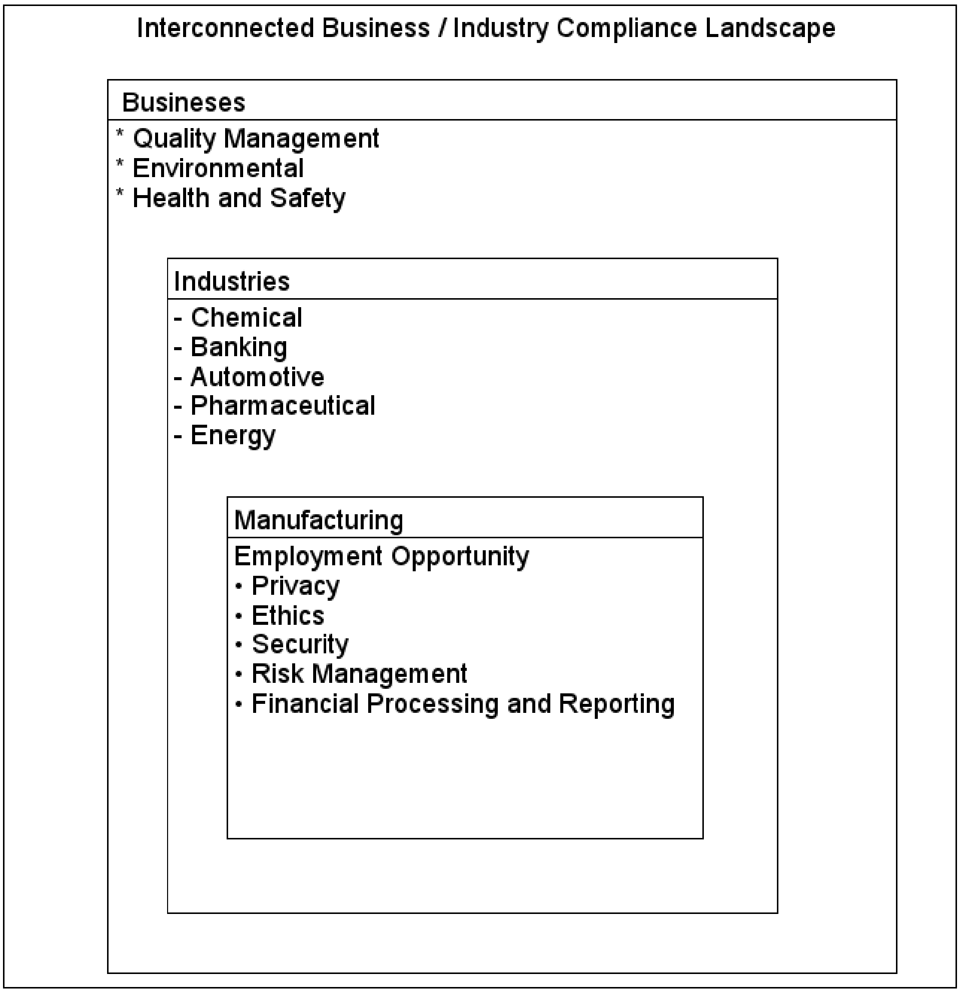


Figure 3.3: Business –Industry Compliance Landscape

In **Figure 3.3,** a logical linkage is shown in terms of how the different business areas and industries are mapped into interconnected business areas and industry compliance landscape**.**

In terms of compliance engineering, both the PITs and PETs highlighted in this section have to operate within the framework of compliance, which is to comply with the relevant legislation, regulations, and best practices applicable to that industry or geography or domain of operations.

Despite all these areas of compliance within an organisation, this study focuses on compliance as it relates to the area of information privacy. To implement privacy engineering in accordance with the related disciplines highlighted in **Sections 3.4, 3.5,** and **3.6,** respectively, focus is on the actual frameworks and tools used to engineer privacy into technology systems using context-aware configurable software-based objects

3.5 Privacy Engineering Frameworks

According to MITRE (2020), the goal of privacy engineering is to integrate privacy into existing systems and engineering processes. For this to happen, privacy requirements must be defined and implemented into the privacy engineering process.

According to Martin and Del Alamo (2017), engineering privacy in technology systems requires systematic methods to capture and address privacy issues throughout the development process. Similar to any other engineering disciplines, this requires the use of an appropriate privacy engineering framework and tools complimented by a proper understanding of privacy engineering concepts. However, the diversity of privacy and engineering approaches together with the privacy context has given birth to a plethora of privacy engineering frameworks and tools. Nevertheless, the goal of any such framework or tool is to improve the protection of information privacy through the privacy engineering discipline. To narrow down on this abundance of frameworks and tools, this study makes reference only to the frameworks recommended by the related industry associations and certification bodies as best practices or standards to be used in privacy engineering processes, namely the NIST Privacy Framework and the ISO 27701:2019 Framework for Information Privacy.

3.5.1 NIST Privacy Framework

According to NIST (2021), the NIST Privacy Framework is not a regulation or law, but rather a tool that can help organisations to improve their information privacy practice in their products and services, and to comply with privacy laws and regulations that might affect them, such as the GDPR in the European Union (EU), and POPIA in South Africa. According to NIST, the recently released Version 1 of this privacy framework should be used in conjunction with NIST’s Enterprise Risk Management Framework in order to develop risk-based strategies and building blocks to achieve organisational privacy goals.

Another tool mentioned by the NIST Privacy Framework is the NIST Cybersecurity Framework. According to NIST (2020), the two frameworks are designed to be complementary of each other, fitting the narrative that privacy and security are related but distinct concepts in terms of focus. For instance, according to NIST (2020), adopting a good security posture does not mean that the organisation is addressing its privacy compliance needs. In fact, according to Biselli and Reuter (2021), there is no consensus on the exact relationship between privacy and security resulting in many studies being conducted to conceptualise the relationship, both theoretically and practically. In the context of this study, privacy and security are viewed as complimentary tools used to support information privacy.

Moving forward with the industry association recommendations, another important information privacy methodology, tool and technique is the ISO/IEC 27701 standard. According to ISO.org, the ISO/IEC27701:2019 is the de facto standard for implementing a privacy information management system (PIMS) (ISO.org, 2021).

3.5.2 ISO/IEC 27701:2019 Framework

Similar to the NIST Privacy Framework discussed in **Section 3.5.1,** the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) developed the ISO 27701:2019 framework for information privacy. This framework is an extension of the ISO/IEC 27001 framework, which focuses on the information security management system (ISMS) of an organisation. The focus of ISO 27701:2019 information privacy framework is to protect personally identifiable information (PPI) through the use of PII controllers and PII processors. For an organisation to be certified ISO 27701 compliant, they must first have completed the ISO 27001 certification as a prerequisite. In a similar fashion to the NIST framework highlighted in **Section 3.5.1**, organisations undertake the ISO 27701:2019 information privacy certification to comply with information privacy regulations, such as the GDPR. From an engineering perspective, MITRE (2020) proposed the Privacy Engineering Framework (PEF). This framework is based on the Privacy by Design philosophical framework and aligns with the systems engineering lifecycle (SELC) with specific methods that account for information privacy.

According to MITRE (2020), the goal of the Privacy Engineering Framework is to integrate privacy into the systems engineering processes and not to create a separate privacy engineering process. **Figure 3.4** illustrates how the core privacy engineering activities map to the stages of a classic systems engineering life cycle.

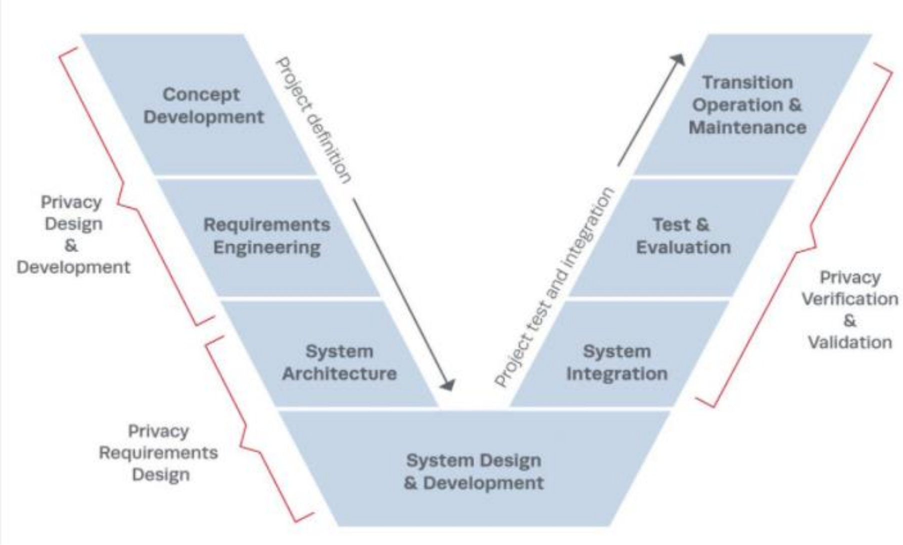

Concept Development
Project definition
Requirements Engineering
System Architecture
System Design & Development
System Integration
Test & Evaluation
Transition Operation & Maintenance
Project test and integration
Privacy Design & Development
Privacy Requirements Design
Privacy Verification & Validation

Figure 3.4: MITRE Privacy Engineering Framework

(Source: Adapted from Mitre.org)

Figure 3.4 depicts two major privacy engineering activities within the project definition phase, namely Privacy Design and Development and Privacy Requirements Design. Essentially, these are the two privacy engineering activities that need to happen before the product is developed.

Once the product is developed, under the project test and integration phases, as depicted in **Figure 3.4,** there is only one privacy engineering activity that needs to take place, which is the privacy verification and validation activity. Next, this study looks at privacy engineering tools

3.6 Privacy Engineering Tools

In terms of tools, this study makes reference to a collection of diverse tools found in the literature of privacy engineering and related disciplines that suit the context of this study, namely prototyping, PbD, privacy patterns, differential privacy, UML ontology, privacy metrics, data modelling and data flow, and the privacy engineering meta-model. These tools are all unique and all play different roles in building technology systems that are engineered with privacy best practices and standards. The next section examines these tools individually and demonstrates how they are used in privacy engineering.

3.6.1 Prototyping

One of the tools used in systems engineering, which is also applicable in privacy engineering, is prototyping. According to Amy et al. (2019), a prototype is an initial model of an object built to test a design. The word prototype comes from the Greek word for ‘primitive forms’. In the context of engineering in general, prototypes are used to perfect items or products before they can be produced on a large scale. Prototypes are often referred to as proof of concept (POC). For example, in the automobile industry, car designers normally build prototypes of new cars to determine whether their ideas can work practically (Amy et al., 2019). This saves time and money linked with producing large quantities of cars that might be dysfunctional and unfit for purpose. However, the same authors note that prototypes are not meant to be the perfect versions of the products or items under development.

This view is based on the consideration that designing and developing a new product can be a very complex process that might require several iterations before the end product is obtained successfully. Also, it is important to note that a prototype might replicate only an aspect of the product. For instance, a prototype might replicate the look and feel of a product without showcasing its full functionalities. In the software engineering world, prototypes are seen as working models of a software product with limited functionality. Similarly, in the context of this study, a prototype will be built to test how information privacy rules can be implemented as configurable software objects in information systems. This prototype, in the context of design science methodology, is referred to as an artefact. In terms of a formal definition, the Oxford Dictionary (2021) defines an artefact as ‘something observed in a scientific investigation or experiment that is not naturally present but occurs as a result of the preparative or investigative procedure’.

In the light of design science research (DSR), the research methodology used in this study, it is suggested that design science in information systems research relies on the creation of artefacts to solve real-life problems (Prat et al., 2014). A different view is expressed by Vaishnavi and Kuechler (2005). They argue that design science research involves the creation of new knowledge through the design of innovative artefacts. These artefacts can be things or processes that have or can have material existence. The analysis and evaluation of the behaviour and performance of these artefacts help researchers to improve their knowledge, which fits well with the prototyping technique of engineering.

To elaborate on prototyping, according to Brown (2020), prototyping as an experimental process involving a prototype. In this process, experimental teams implement conceptual ideas into tangible forms to capture design concepts and to test them on users. As such, it is easy to refine and validate a product before taking it to the end users and the market. According to Brown (2020), critics of the prototyping concept complain that by taking the time to prototype ideas, business objectives are slowed down, resulting in loss of revenue. This stance overlooks the costly risk of sending malfunctioning products into the market.

From a different school of thought, but still in the domain of systems engineering, prototyping is seen as a software development methodology in which a prototype is built, tested and reworked until a final product is released. From a systems engineering perspective, using such a methodology becomes appropriate when the actual requirements of the system are known explicitly. The prototyping process is illustrated in **Figure 3.5**. This illustration is adapted from Guru99 (2020).

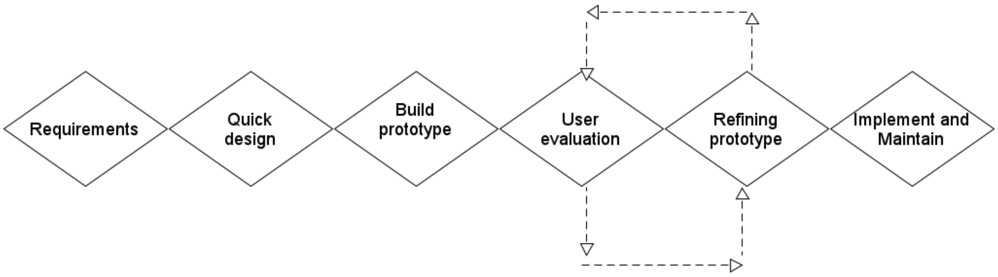


Figure 3.5: Prototyping Process

(Source: Adapted from Guru99.com)

As shown in **Figure 3.5**, the process starts with the project team eliciting some broad requirements and then proceeding to a quick design often called a 'mock-up' design of the intended software. Once the mock-up design is accepted, then a prototype of the software system is built.

Once this prototype is built, the iterative process of testing and refining the prototype begins. This is often done with the full participation of all the stakeholders and end users of the software. This process continues until the software is deemed fit for purpose and is eventually implemented in a live environment for production.

In the context of this study, the principles of prototyping are employed to engineer privacy requirements into the artefacts, as prescribed by DSR. The artefact is used to evaluate and test the research questions and to evaluate the research problem highlighted in **Sections 1.7 and 1.2**, respectively.

3.6.2 Privacy by Design

Organisations still struggle to comply with data protection within their information systems and are now considering PbD as a tool to implement low-level privacy protection. (Bednar et al., 2019). PbD was coined by Cavoukian in 1997 as a methodology to embed privacy into the design of a system and throughout the lifecycle of collecting processing and storing data (Coelho et al., 2021).

Furthermore, according to Spiekerman (2012, p.39), it is the responsibility of systems engineers, software architects, software engineers, system developers, information architects, and product designers to realise privacy protection in information systems According to Alharbi et al. (2012), PbD is defined as a philosophy and approach of embedding privacy into the design specification of various technologies. They base this definition on the trilogy of Privacy by Design principles proposed by Cavoukian (2011) and comprising three elements, namely technology systems, accountable business practices, and physical design and network infrastructure, as illustrated in **Figure 3.6.**

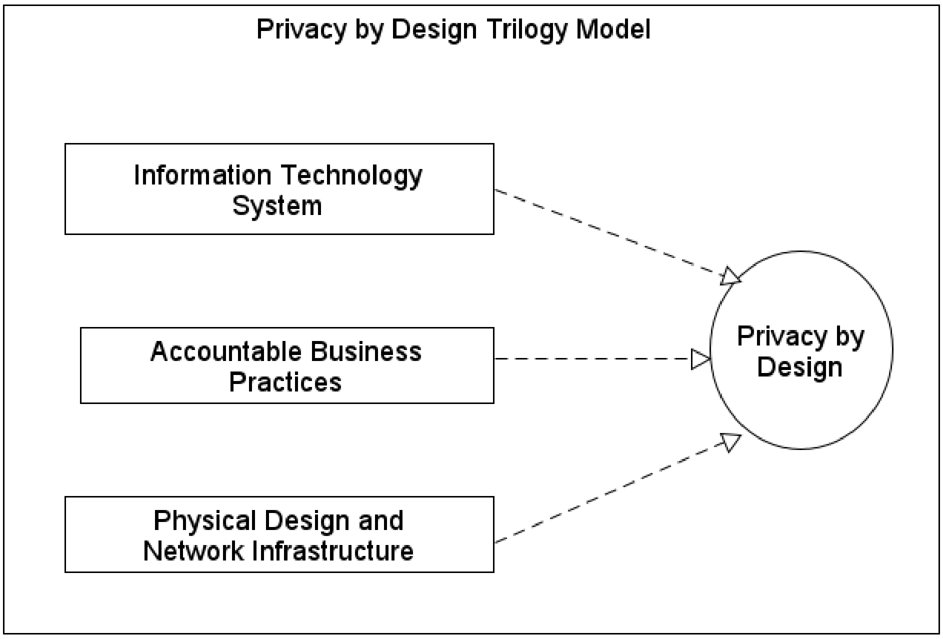

Privacy by Design Trilogy Model
Information Technology System
Accountable Business Practices
Physical Design and Network Infrastructure
Privacy by Design

Figure 3.6: Privacy by Design Trilogy Model

In the context of this study, and based on **Figure 3.6**, emphasis is placed on the first block of the Privacy by Design Trilogy Model, namely the Information Technology System. The rationale for this emphasis is based on the fact that the enforcement of privacy happens within the information technology system components. In view of the importance of the information system component, this study examines how the Privacy by Design concept can be adapted and incorporated into the design of new systems at inception. According to Cavoukian (2011) Privacy by Design minimises the threats to privacy posed by the deployment and use of privacy-invading technologies (PITs). Although Privacy by Design is now increasingly being recognised as a privacy engineering model, it has been critiqued as vague and leaves many questions open about its application during systems engineering. More so, the current definition of PbD does not address the methodological aspect of systems engineering (Cavoukian, 2011).

Hence, the need to investigate how privacy controls can be tailored into information system objects, in practice, and consequently to embed privacy in the design of new systems. Consider that embedding privacy is the goal of modern information system owners in order for them to comply with the applicable information privacy laws, best practices and standards. Furthermore, the goal to embed privacy becomes more important based upon the fact that the basic right to privacy is often ignored by designers and software engineers when designing or building new information systems. In addition, Privacy by Design mandates that privacy be embedded into the design and architecture of information technology systems as well as business practices, as illustrated in **Figure 3.6.** In fact, several Privacy by Design methodologies have been developed that integrate information privacy in the different aspects and stages of the system development life cycle. In the context of this study, heavy reliance is placed on the PbD concept to drive the development of the POPIA artefact to test the assertions of this study.

3.6.3 Privacy Patterns

Another tool used in privacy engineering is privacy patterns. Generally, patterns are reusable solutions to commonly recurring problems (Caiza et al., 2020). By instantiating a pattern, an engineer can arrive at a solution to a problem with minimal effort.

Doty and Gupta (2013) describe patterns from a design perspective as abstract solutions to common problems within a particular context. In the context of privacy engineering, the focus is on the engineering of patterns, which allows systems to verify and comply with privacy laws. According to Buchmann and Anke (2017), privacy patterns help privacy practitioners to realise data minimality and Privacy by Design by implementing privacy oriented processes within technology systems.

Hence, privacy patterns can be viewed as tools for practical Privacy by Design. To support this view, Doty and Anke (2013) posit that Privacy by Design focuses primarily on the step to implement high-level principles into concrete engineering practices. In order to bridge this gap, they propose privacy design patterns as the link. According to Papoutsakis et al. (2021), there have been various efforts to provide architectural and design patterns for the protection of privacy. However, the key challenge has been the lack of established and standardised terminologies to describe privacy principles at the same level as it is done in the software engineering and information security domains (Pfitzmann & Hansen, 2010). For instance, in some literature, anonymity and pseudonymity are grouped together while in others they are separated. (Avizienis et al., 2020). They went on to emphasise that there is a lack of a taxonomy to define the relationship between the privacy properties, concepts, and related patterns. From a literature review perspective, there are conflicting definitions and meanings of the terms used in the information privacy discipline (Pfitzmann & Hansen, 2010).

However, there is a plethora of privacy design patterns available in the literature of privacy engineering. Most of these patterns are context specific and are published in the dedicated website **https://privacypatterns.org/** Privacypattern.org provides a living lab of privacy patterns being updated continually and that are reusable to privacy engineers globally. Every pattern presented here has a title, context, problem context, the solution and examples. **Table 3.1** presents a few of these patterns and their usability in the privacy engineering domain.

Table 3.1: Sample Privacy Patterns (Source: privacypatterns.org, 2020)

Pattern Name	**Problem Context**	**Description of Context**
Protection against tracking	Every single interaction in the web leaves footmarks and clues about yourself. Cookies, for example, enable webservers to gather	This pattern is applicable when personally identifiable information is tracked through software tools,

	information about web users, which therefore affects their privacy and anonymity.	protocols or mechanisms such as cookies and the like.
Location granularity	Many location-based services collect current or ongoing location information from a user in order to provide some contextual service (nearest coffee shop, local weather, etc.)	When a service is collecting location data from or about a user or transmitting location data about a user to a third party.
Minimal-information-asymmetry	Controllers have far more information than the users who utilise their services, which makes the users vulnerable to exploitation	Users frequently interact with controllers whose services (or products) they have not used before and do not understand the details of the privacy policies.

All of these patterns are developed as mini design solutions for common privacy problems and give life to practical implementation of Privacy by Design to solve organisational challenges of protecting information privacy.

In Diamantopoulos et al. (2017), five basic privacy patterns are defined in order to better understand concepts regarding privacy that need to be addressed when designing privacy-aware systems. These are briefly described as follows:

1. **Anonymity** refers to a characteristic that does not allow PII to be identified directly or indirectly;
2. **Pseudonymity** refers to an alias which is used instead of PII;

3. **Unlinkability** is the use of a resource or a service by a user without a third party being able to link the user with the resource or service;
4. **Undetectability** is the inability of a third party to distinguish who the user is; and
5. **Unobservability** refers to the inability of a third party to observe whether a user is using a resource or a service).

These patterns can serve as templates to develop other privacy patterns.

According to Coelho et al (2021), privacy patterns work as a central building block to ensure the translation of privacy into technology systems. This translation ensures that systems conform to privacy laws and regulations which, in turn, fulfils the Privacy by Design principles.

This study will not delve into the different types of patterns; rather, it employs privacy patterns as tools / techniques in the overall effort to engineer privacy into information systems.

3.6.4 Differential Privacy

Another important privacy engineering tool is differential privacy. The relevance of differential privacy is highlighted by Zhu (2018), that big tech companies like Facebook, Google, Apple, and Amazon are continually infiltrating personal and social interactions to collect vast amounts of data (Zhu, 2018).

Proponents of differential privacy claim that it can help protect personal data better than traditional methods of data protection, especially since it is based on the discipline of mathematics. According to Ho Au and Cho (2017, p. 247), differential privacy defines privacy from a new perspective and ensures that data is mined and used without compromising the data subject by introducing trusted data curators that hold private databases.

In this model, the individual whose private data is used is still protected as the data and the data owner are kept separate from each other in trusted curatorship and formal procedures. To illustrate the architecture of this model, **Figure 3.7** shows how the different blocks of a differential privacy construct is structured.

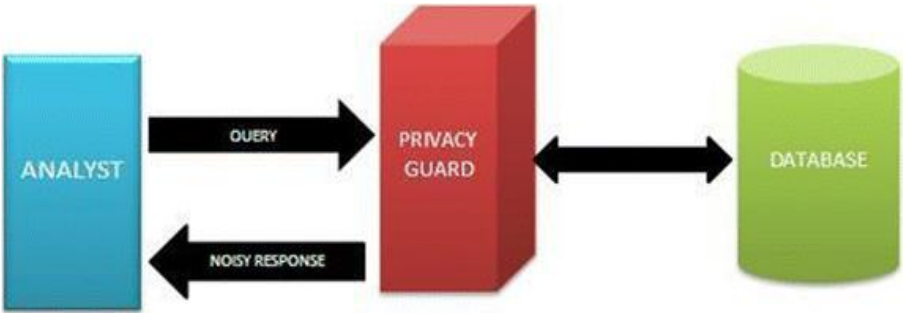


Figure 3.7: Differential Privacy Model

(Source: Adapted from Bigdata.com)
https://journalofbigdata.springeropen.com/about

Notable in this construct, is the role of the privacy guard, also called data curator, that implements formal procedures and mathematical algorithms to separate data subjects from the underlying data, thereby enhancing privacy. This study does not delve into the details and different mathematical algorithms and science driving the differential privacy concept but draws on and adopts the differential privacy concepts as one of the techniques in ensuring the protection of privacy through privacy engineering.

3.6.5 UML Ontology

Ontologies come with a wide variety of meaning, adaptation, and usage, which stretches far and wide in many domains of knowledge. In recent times, the use of ontology in the different fields of studies and research has become very pervasive. According to Manraj and Sivakumar (2010), ontology is a formal representation of a set of concepts within a domain and the relationships between those concepts. It is used to reason about the properties of a domain and may be used to define the domain. Consequently, an ontology language is deemed to be a formal language used to encode or represent a domain of knowledge. This definition focuses more on the adaptation of ontology to resolve real-world problems. Similarly, in both computer science and information science, an ontology is viewed as a data model that represents a set of concepts within a domain and the relationships between those concepts.

These ontologies also form a hierarchical structure, as depicted in **Figure 3.8**. First, we begin with the top-level ontology followed by domain/task ontology. The last level is the application ontology.

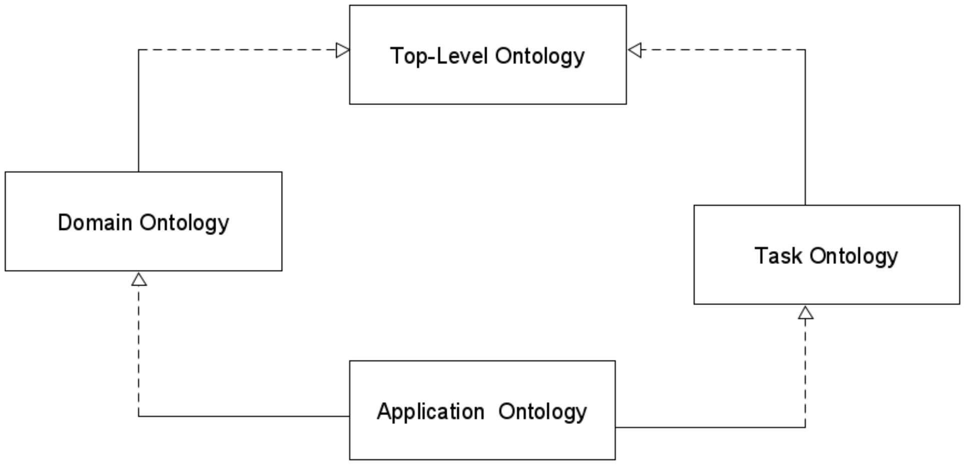


Figure 3.8. Ontology Classification

(Source: Adapted from Slimani, 2011)

Based on **Figure 3.8**, one of the application ontology languages is UML. UML stands for Unified Modelling Language and, according to Padmanabhan (2012), it is a graphical language for visualising, constructing, and documenting the artefacts or building blocks of a software intensive system. The idea of using UML in documenting a software system is justified by the fact that it offers a standard, with agreed-upon notations, to design a system's blueprint.

To support this view point, Bell (2003) declares that the Object Management Group (OMG) released UML with the main purpose of providing the development community with a stable and common design language notation that information technology (IT) professionals have been waiting for.

UML uses techniques from data modelling, business modelling, object modelling, and component modelling throughout the software development life cycle and across different implementation technologies (Padmanabhan, 2012). UML is a programming language agnostic that enables IT professionals to read and communicate system structures and design plans, in a similar way as construction architects can design and share building plans. In simple terms, UML is a software design tool.

Hence, in the context of this study, ontology is used as a tool to model the privacy engineering requirements into a reusable knowledge domain which can be embedded into software objects.

More precisely, the UML use case diagram is used to model the different components of the POPIA artefact in line with DSR methodology with the aim of showing how the artefact is structured both architecturally and functionally. Bell (2003) asserts that the UML use case diagram is primarily used to help development teams to visualise the functional requirements of a system, which includes the relationship between the users (actors) and the system processes or components. By doing this, the use case diagrams communicate the high-level functions of the system and the system's scope.

3.6.6 Privacy Metrics

According to Wagner and Eckhoff (2018), the goal of privacy metrics is to measure the degree of protection experienced by a user of a system and ultimately, the amount of protection offered by the technology system. Clifton (2009) defines privacy metrics as a measure of the susceptibility of data or a dataset to revealing private information. This definition highlights the likelihood of a technology system to reveal PII, intentionally or unintentionally. Consequently, privacy metrics can be used as a tool to measure the level of protection accorded to PII by a technology system. According to INCOSE (2010), engineering privacy, like any other engineering discipline, executes processes to produce their products, which could be requirements, plans, design verification, and validation procedures, hardware, software and integration, to mention a few. As these products become more complex and sophisticated, managers require more advanced techniques, such as measurements, to allow them to monitor and control the use of these complex products.

In the context of privacy engineering, mindful of the fact that privacy metrics contribute to improving privacy in the digital world, there is still a lack of standardisation around privacy metrics. The problem is that the units of privacy measurement keep on changing with advances in technology (Wagner & Eckhoff, 2018). Normally, privacy metrics are determined through questionnaires posed to data subjects to determine their privacy behaviour.

This study does not delve into the validity of privacy metrics and units of measurement used; rather, it explores how privacy units of measurement can be used to validate and verify privacy in technology systems and, in the case of this study, software-based objects built to enforce information privacy. The most common place is in the privacy verification and validation phase of the systems engineering lifecycle (SELC), as illustrated in **Figure 3.4.**

3.6.7 Data Modelling

In the context of this study, data modelling is used as a tool to understand how data is processed and stored in information systems to ensure maximum protection from unauthorised exploitation of disclosure, which could compromise the privacy of data subjects. To understand data modelling fully, it is important to understand the role of data in the context of information systems research. According to Sanders (2016), in a normal conversation, the terms data, information, and knowledge can be used interchangeably. Although the three terminologies are closely linked in a hierarchical manner, they are seen differently in the context of information systems research. **Figure 3.9** shows the hierarchical relationship between these terms, as suggested by the support guide (Cambridge International, 2020).

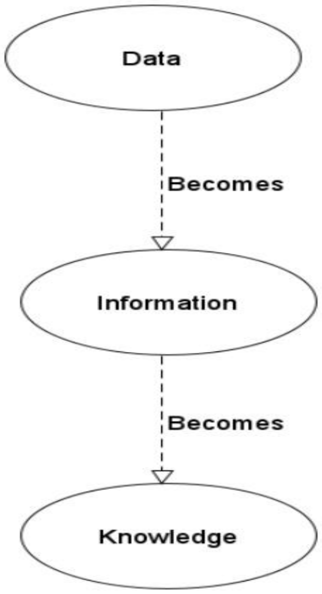
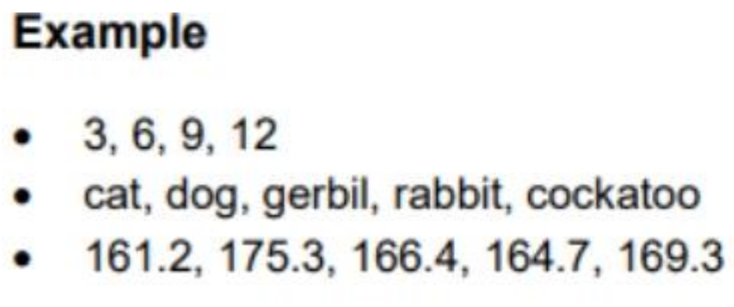


Figure 3.9: Hierarchical Relationship between Data, Information and Knowledge

(Source: Adapted from www.cambridgeinternational.org)

Data is the plural for the word datum. The latter is seldom used in information system research. In terms of definition, data refers to a representation of facts, concepts or instructions in a formalised manner that is suitable for communication, interpretation, or processing by humans or by automatic means (Hicks, 1993, p. 668, as cited in Checkland & Holwell, 1998). This definition highlights the main aspects of data that are linked directly to personally identifiable information handling used extensively in this study. That is, data suitable for communications, which technically means transmission, processing, and interpretation, leaning on the side of data classification and sharing. Another important aspect highlighted by this definition is the representation of facts. This means that for data to be meaningful, it must represent facts resulting from the recording or observation of an event or object. Another noteworthy definition of data is provided by Martin and Powell (1992, p.10). Here, data is defined as the raw material of organisational life; it consists of disconnected numbers, words, symbols, and syllables relating to the events and processes of the business. An example of this representation of data is shown in **Figure 3.10.**



Example

- 3, 6, 9, 12
- cat, dog, gerbil, rabbit, cockatoo
- 161.2, 175.3, 166.4, 164.7, 169.3

Figure 3.10 Representation of Data

This definition highlights the wide and diverse forms in which data is represented in information systems, as well as in data repositories. From this definition of data, and as illustrated in **Figure 3.10**, data is in its pure form and has no meaning. According to the guide of Cambridge International (2021), such data is often referred to as raw data and has to be given context before it starts to take form and get meaning. Once data takes form and meaning, it now qualifies to be called information. The process of interpreting and ascribing meaning to data mostly happens in information systems. This point is succinctly stated by Sanders (2016), when he suggests that 'the so-called information overload is in fact a data overload'. Hence, from an information system perspective, data is seen as input or raw unprocessed collected facts, which will then be processed by the information system (given context and meaning) resulting in an output which is referred to as information. This whole process is illustrated in **Figure 3.11.**

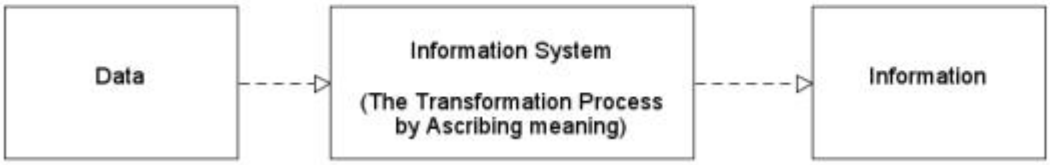


Figure 3.11: Data Transformation Process

This study does not delve into the detailed arguments and counter arguments on the relationship between data, information, and knowledge. These details are considered to be out of the scope of this study. It therefore suffices to say that this study focuses mainly on data and will use the term interchangeably with information, as grounded in the concepts of information privacy and personally identifiable information (PII). Having a good understanding of the concept of data and information within information systems, the concepts of data modelling can be examined in context. According to Taylor (2021), data modelling is the process of creating a model for data to be stored in a database. The idea of data modelling is to create a visual representation of the data to be stored in a database.

According to Taylor (2021), the real advantage of data modelling ensures consistency in naming conventions, default value, semantics, and security of data while ensuring the quality of data.

There are two types of data modelling techniques, namely entity-relationship (E-R) model and UML (unified modelling language):

- **Entity-relationship (E-R) model**: An entity-relationship model is a high-level data model based on entities and relationships among entities. An entity is a real-world object about which data is collected (Hadzilacos & Tryfona, 1997).
- **UML (unified modelling language):** This has emerged as a standardised notation for describing object-oriented models. To use UML effectively, it needs to be applied in conjunction with object-oriented analysis and design method (Goma, 2006)

In the context of this study, the different data modelling techniques are adapted as follows;

- **The E-R model**: The E-R model is used in the context of the technology system to model the different data entities and to show the relationships and flow of data within the technology system. Furthermore, since the context is to protect information privacy, the E-R model is also used to view the data elements that are within the scope of protection in line with the applicable regulations and privacy laws.
- **The unified modelling language (UML):** UML is used to model the system components during the privacy engineering process. According to Taylor (2021), UML models what the system contains at different levels of abstraction, such as conceptual, logical, and physical levels to cater for the understanding of the different stakeholders. For instance, the physical model describes how

the system will be implemented, and typically, it is created for software engineers while the conceptual model is created primarily for business stakeholders with the purpose of organising, scoping and defining business rules and concepts.

These modelling and flow techniques are used in this study to develop the privacy engineering meta-model as the foundational framework upon which to build the prototype used in this study.

3.6.8 Privacy Engineering Meta-Model

One school of thought proposes that to encourage acceptability within the privacy engineering community, researchers should not focus on building an entirely different framework or methodology for privacy engineering (Martin & Del Alamo, 2017). Instead, all efforts surrounding privacy engineering should be aligned with the more general efforts on software and systems engineering, such as the ISO/IEC24744 (2021) standardised software engineering meta-model for development methodologies (SEMDM). The strength of SEMDM is that it proposes three layers of abstraction through which new methodologies can be developed and instantiated. This instantiation creates elements which method engineers can use to enact methodologies.

Through these methodologies, system developers and designers can construct products or deliver services in the context of their endeavour. In the case of this study, this amounts to enforcing POPIA rules as configurable software objects in information systems responsible for the handling of data subjects' personally identifiable information. According to Martin and Del Alamo (2017), the SEMDM describes a set of concepts which can be part of any methodology and represents it in three dimensions (processes, producers (human and non-human resources), and products).

In summary, SEMDM provides a comprehensive meta-model for software and systems engineering. Hence, this study addresses a subset of that problem by creating a model for enforcing privacy through user-configurable software objects. Another approach to privacy engineering that is prevalent, especially in the financial service and banking industry, is encryption and hashing. All of them are part of the science of cryptography, which serves to hide data and protect it from unauthorised access.

However, these sciences do not exist on their own and are normally implemented within a certain legal framework. This leads to the next section entitled Legislative Framework for Privacy Engineering.

3.7 Legislative Framework for Privacy Engineering

Before examining the privacy engineering processes in Section 3.8, reference is made to the legislative context in which the privacy engineering process is undertaken. For this study, the privacy engineering process was implemented in line with the legislative framework applicable in South Africa (POPIA).

POPIA was signed into law in South Africa in November 2013 and requires that all organisations protect the personal information collected from their customers and employees, herein referred to as data subjects. However, recent evidence suggests that various organisations have raised ongoing concerns about the lack of clarity regarding compliance with the Act and its status quo. Recent information, according to Michalson (2021), posits that the POPIA commencement date and effective date, which were set for 1 July 2020 and 1 July 2021, respectively, have both lapsed. The implication of this is that POPIA is now fully effective and operational in South Africa, eight (8) years after it was signed into law. This means that responsible parties (known as data controllers) have to ensure that all data processing conforms to the specifications of POPIA. However, according to Michalson (2019), POPIA does not exist on its own; it has to co-exist alongside other laws, and where there is a conflict between POPIA and another law, POPIA will prevail. To compound this debate, Michalson (2019), argues that if another law is more protective of the personally identifiable information, then said law will take precedence over the POPIA, in case of a conflict. In the context of South Africa, the following are some of the laws and legislative texts relating to the protection of personal information:

1. **The Consumer Protection Act (CPA):** This act aims to promote fairness and good business practices between suppliers and consumers of goods and services. (Government Gazette, 2009).
2. **The National Credit Act (NCA):** NCA promotes responsible access to the consumer credit marketplace (NCR.org.za, 2019).

3. **The Regulation of Interception of Communications Act (RICA):** 'RICA is the piece of legislation in South Africa that governs the interception or monitoring of communications' (Michalson, 2019).
4. **The Promotion of Access to Information Act (PAIA):** 'It gives effect to any information that is held by another person and that is required for the exercise or protection of another person's rights' (Justice.gov.za, 2000).
5. **The Cybersecurity Act**: This act seeks to criminalise cybersecurity offences and to protect the masses from cyber-related attacks, among other issues (ITWeb, 2019).

A visual representation of these laws is shown in **Figure 3.12**

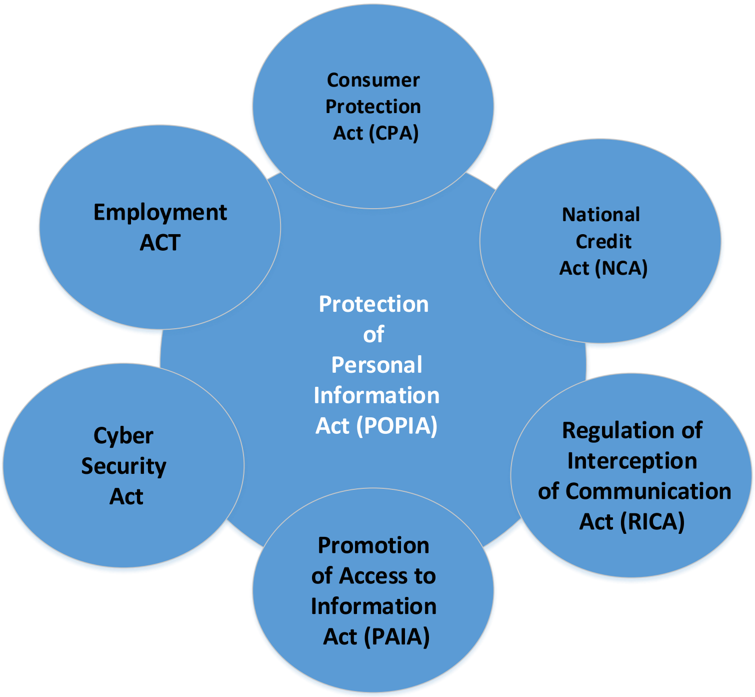

Consumer Protection Act (CPA)
Employment ACT
National Credit Act (NCA)
Protection of Personal Information Act (POPIA)
Cyber Security Act
Regulation of Interception of Communication Act (RICA)
Promotion of Access to Information Act (PAIA)

Figure 3.12: POPIA and Other Privacy Laws in South Africa

According to Michalson (2019), analysing all these laws and noting their points of intersection and overlap is a subject for a detailed legal review. Mindful of this, there is a gap in academic literature covering these overlaps and intersections, leading Michalson (2019) to conclude that, according to jurisprudence, if another law gives the data subject greater protection, the other law will prevail. This implies that companies may end up having to comply with another law instead of with POPIA. For instance, in the healthcare sector, POPIA does not require the healthcare professionals to get consent from the patient, but the National Health Act does require consent. Therefore, the National Health Act will apply and not POPIA (Michalson, 2019).

In the light of all these laws and emerging technologies broadening the scope and context of information harvesting, storage, processing, and disclosure, a key question comes to mind: What is the best way to enforce the POPIA privacy rules and guidelines into technology systems responsible for the handling of these data? To answer this question, a privacy engineering process is deemed necessary. **Section 3.8** expands on this process.

3.8 Privacy Engineering Process

The privacy engineering process covers the steps required to implement Privacy by Design (MITRE, 2019). A plethora of privacy engineering processes exist in academic literature and in practice. This study does not delve into all the different and distinct privacy engineering processes; rather, it examines two distinct privacy engineering processes based on two distinct approaches. These two approaches are selected based on the two pillars of the ISO/IEC Privacy Engineering Framework, namely a risk-based approach and an engineering-based approach.

First, in terms of the risk-based approach, MITRE (2019) proposes the following privacy engineering process. This process is risk driven and tailored to operationalise the Privacy by Design philosophical framework within a technology system.

The process is made up of three distinct steps, as outlined below:

- **Step 1: Segment Privacy by Design** into distinctive privacy activities that are aligned to the systems engineering lifecycle of the technology system. An example of an activity in this step is mapping the Privacy by Design principle into the chosen systems engineering lifecycle steps of the organisation.

- **Step 2: Identify, define and implement privacy requirements** to address privacy risk within the systems development lifecycle, such as compliance risk, ethical risk through system functionalities, and technical control. An example of an activity in this step is sourcing baseline and custom privacy system requirements.
- **Step 3: Continuous alignment of technology systems with the broader privacy programme of the organisation.** An example of an activity in this step will be privacy testing and review (MITRE, 2020).

In comparison, the engineering approach, according to Mania (2021), proposes a four-step privacy engineering process, which is based primarily on the tenet of solving the privacy engineering process in modern systems that are running fourth industrial revolution technologies such as artificial intelligence, big data, IoT and virtual reality. The steps put forward are as follows:

- **Step 1:** Make privacy a key system requirement. That is, making privacy a non-functional requirement when the system is initially scoped. In this way it will take into consideration regulatory requirements and best practices from the outset and avoid the pain of having to force privacy policies to work with a technology system that is non-compliant at a later stage.
- **Step 2:** Understand the business domain. In this step, the engineering team must understand the business domain to understand the kind of data to be handled and to design the best type of controls. For instance, engineers will know the data flow and determine which data to encrypt and which to expose.
- **Step 3:** Apply industry standards to the handling and processing of personal data. This step is important as a system is only trusted when it is compliant with the requisite and applicable industry standards and best practices. For instance, if it is a card-processing system carrying card data, then the data security standard (PCI-DSS) controls of the payment card industry will be evoked, and the system must be compliant with the standard.

- **Step 4:** Add privacy-specific test cases in a quality assurance process. In this step, privacy-specific testing and validation is conducted to ensure that the system meets the baseline, best practices and scope privacy requirements specified in **Step 1.**

In this context, any of these approaches is fit for purpose to engineer information privacy into technology systems.

3.9 Conclusion

As elaborated on in the different sections of this chapter, privacy engineering is a very important and integral component of enforcing information privacy within information systems.

In terms of benefits, privacy engineering benefits both the data subject whose privacy is at risk as well as the information system owners, as mandated by the different information privacy laws and industry best practices.

In conclusion, despite the complexity of the applicable laws and regulations, and the diversity of engineering approaches, the evolution of technology and the concept of big data / cloud computing have created a plethora of approaches to privacy engineering

This multiplicity of approaches requires careful planning to engineer privacy regulations efficiently into information systems. Therefore, engineering a privacy-friendly system is still a challenge for software engineers.

However, in the context of this study, the next chapter will develop a conceptual framework as a guide in the process of integrating privacy in technology systems as a contextual software-based object.

CHAPTER 4

Research Design and Methodology

4.1 Introduction

This chapter presents the research design and methodology used in this study. The chapter is structured into six sections covering the research design and methodology, the research participants, the research instruments and procedure, and the research analysis approach.

The final section of this chapter highlights the limitations of the research study and the ethical considerations. This chapter provides the road map used in structuring the body of this thesis, starting with the conceptual framework in Chapter 5, followed by the research design in Chapter 6 and ending with the implementation of the prototype in Chapter 7.

4.2 Research Method

This study was conducted using the design science research methodology. (DSR). It is important to highlight that the main contribution of this study, in line with the primary research objective, takes the form of an artefact: a practical model for POPIA compliance.

This artefact was created using the three-phase design science approach proposed by Peffers et al. (2008) and depicted in **Table 4.1**. Furthermore, to accomplish these phases, several methods or steps were used, as shown in the methods column of **Table 4.1**. Additionally, a literature review was conducted in Chapter 2 and Chapter 3 to seek a critical appraisal on the following aspects of this study.

1. To illustrate the extent to which the POPI Act is influencing organisations. Data sources (government gazettes, legal expert reviews and various online websites and accredited journals) have presented

write-ups, reviews, and opinions on the importance of compliance with the Act.

2. To translate the common vocabulary of the POPIA requirements into machine-interpretable instructions, data sources on software ontology creation literature, modelling techniques, metrics identification models and other documents about machine language specification were consulted for this study.
3. To determine the best architectural design patterns that can be used for POPIA compliance, data sources were consulted on design science theories, modelling techniques and other source of relevant literature, and the evaluation of use cases for this study.
4. To ascertain how individual privacy concerns are measured, quantified and validated, data sources were consulted such as journal papers, conference papers, online articles and presentations.
5. According to Creswell (2009), a case study involves an up-close, in-depth, and detailed examination of a subject (the case), as well as its related contextual conditions. In this vein, several cases studies were performed to test the use cases of the POPIA compliance model from a technical and operational perspective.
6. To validate the literature, four important knowledge questions were asked. These questions are based on the design science literature proposed by Wieringa (2013), and include:
    a. Effect questions: Does (artefact x context) produce effects?
    b. Trade-off questions: Does (alternative artefact x context) produce effects?
    c. Sensitivity questions: Does (artefact x alternative context) produce effects?
    d. Requirement satisfaction questions: Do (effects satisfy requirements?)

Finally, to conduct the research an adaptation of design science, based on Peffers et al. (2008) was employed, as illustrated in **Table 4.1.**

Table 4.1: Adaptation of Design Science Research
(Source: Peffers et al., 2008)

No.	Phases	Methods
1	Problem identification and motivation	• Identify problem • Literature research • Expert interviews • Pre-evaluate relevance
2	Solution design (design artefact)	• Use case analysis • Modelling techniques • Simulation and prototyping • Develop a system architecture • Analyse and design the system • Case study • Build the system
3	Evaluation and demonstration	• Refine design model • Iterative build and evaluation • Summarise results /Argumentation

To summarise, the mapping between the research questions, the research objectives, and methods is shown in **Table 4.2**

Table 4.2: Mapping between research questions, research objectives, and methods

Research questions	Research objectives	Research methods
1. How can the common vocabulary of POPIA requirements be expressed as machine-interpretable instructions?	To facilitate the translation of POPIA requirements into machine-interpretable language.	• Literature review • Ontology creation/ Language specification • Modelling techniques
2. Which software architectural synthesis can best satisfy the organisational goal of POPIA compliance?	To determine the best design pattern to meet the technical and operational requirements for POPIA compliance within organisations.	• Literature review (Design science theories) • Use case analysis • Static and dynamic modelling techniques
3. How should individual privacy concerns be validated against POPIA regulatory controls within an organisation?	To formulate a model to use for the validation of individual privacy concerns against POPIA regulatory controls.	• Literature review • Modelling techniques • Iterative build and evaluation of artefacts • Argumentation

Based on the mapping in Table 4.2, the researcher is guided in how to conduct the research study without moving away from the objectives of this study, always keeping in mind the research question that this study is designed to answer. Similarly, Table 4.2 guides the researcher in terms of the exact steps of research method to answer the specific research question and meet the corresponding research objective. Based on Table 4.2, the next section focuses on the research design.

4.3 Research Design

The research design of this study was based on the research design framework developed in this chapter. The framework is based on the design science research paradigm chosen for this study. The choice of design science is motivated by the fact that it seeks to enhance technology and science knowledge via the creation of innovative artefacts to solve specific research problems (Bocke et al., 2020). In the case of this study, the main innovative artefact is the POPIA prototype. The next section examines the approach to develop the research design framework for this study, which is also based on two different conceptual frameworks for understanding, executing, and evaluating of design science research.

4.3.1 Framework Objectives

The objectives of developing the research design framework are captured in the following points:

- First, the research design framework developed here helps in giving form and structure to the body of this research study by elaborating on the different stages of the thesis design and how they are logically linked and connected.

- Second, this research design framework shows the focus of every stage of the thesis design. For instance, the focus of Stage 1 is on the analysis of the problem domain; Stage 2 is linked to the development of the design framework; and Stage 3 focuses on the test of the prototype.

- Finally, the research design framework developed here logically links the output of the research process with the specific chapter in the research study. For instance, the POPIA conceptual framework is linked to Chapter 5 of this study; the design framework is linked to Chapter 6 of this study, and finally, the research findings are linked to Chapter 7 of this study.

In light of the objectives of the research design framework listed above, the next section focuses on the approach used to develop the research design framework.

4.3.2 Approach Used to Develop the Framework

This study is underpinned primarily by the design science research methodology. According to Hevner et al. (2004), two key paradigms characterise research in the information systems discipline, namely behavioural science and design science. On the one hand, the behavioural science paradigm focuses on the development of theories about human and organisational behaviour with regard to information systems; on the other hand, the design science paradigm seeks to grow both human and organisational reach by creating innovative artefacts. Hence, to develop the research design framework for this study, the conceptual framework for conducting design science research by Wieringa (2013) is utilised. Wieringa (2013) defines design science research as the ‘design and investigation of an artefact in context’ (Wieringa, 2013), for example, the design and investigation of an agent-based route planning algorithm. This author adds that design science is solution oriented, which contrasts with natural and social sciences, which are problem oriented. Wieringa’s (2013) proposed approach to solving a research problem using design science is shown in **Figure 4.1**.

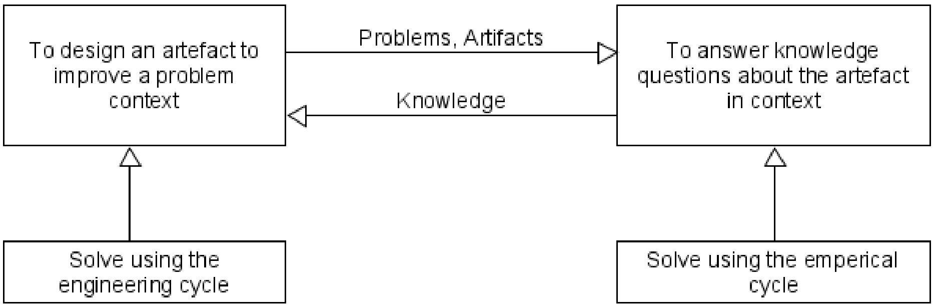

To design an artefact to improve a problem context
Problems, Artifacts
Knowledge
To answer knowledge questions about the artefact in context
Solve using the engineering cycle
Solve using the emperical cycle

Figure 4.1: Design Science Approach to Solve Problems

(Source: Adapted from Wieringa, 2003)

The Wieringa (2003) approach is much simpler and more straightforward to implement; however, Vaishnavi and Kuechler (2005) provide a more comprehensive conceptual framework (design science process), in which the entire design science development process is outlined, as illustrated in **Figure 4.2**, to help solve sophisticated design science research problems.

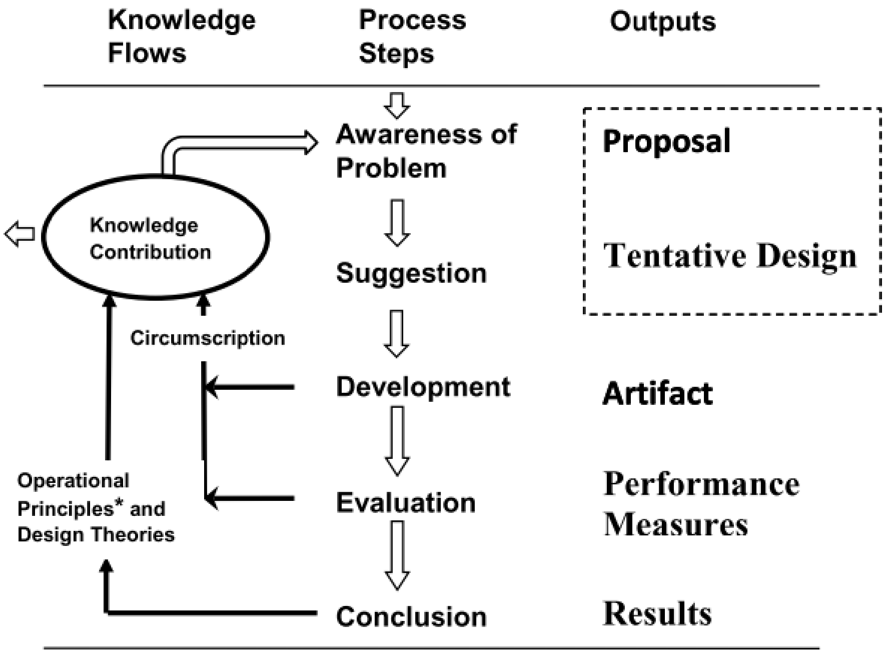


Figure 4.2: Comprehensive Design Science Research Process Model

(Source: Adapted from Dasgupta, 1996; Purao, 2002)

Based on the Vaishnavi and Kuechler (2005) conceptual framework to conduct design, as illustrated in **Figure 4.2**, the following steps are highlighted and adapted to formulate the research design framework for this study.

1. **Awareness of Problem:** According to Vaishnavi and Kuechler (2005), awareness of an interesting problem may come from several sources, such as new trends in the industry, innovation, natural occurrence or problems encountered in the different spheres of society. In the context of this study, the problem is how to best enforce information privacy rules in the technology systems used in handling personally identifiable information. This problem is becoming more visible, owing to the rapid innovation and adoption of advanced technologies to process information both on the cloud and in the public internet. For instance, according to Wagner and Eckhoff (2018), recent innovative technologies, such as artificial intelligence and robotics, are raising ethical issues about regulation, governance, and humanity

2. **Suggestion**: Once a reasonable problem has been identified, the next phase in the design science process is to construct a tentative design, also called a suggestion. This design is most often, as shown in **Figure 4.2**, combined with the proposal, which is a write-up by the researcher of the problem identification for suitability and further development potential.

3. **Development:** In this step, the tentative design or artefact(s) is/are further designed and developed. In the context of this study, the conceptual framework is first developed**,** showing the different contexts, their distinctive input and output elements and assumptions, in Chapter 5. Following the conceptual framework is a quick transition to the design phases, starting with the development of the design framework in Chapter 6**,** followed by implementation (test) of the actual prototype in Chapter7.

4. Evaluation: In this phase of the design science research process, in the context of this study, the POPIA prototype is evaluated in context. Next, observations are made based on the implementation (test) findings. The activities of this phase are covered in **Chapter 8** of this study. The findings and observations recorded from this phase are further analysed based on the research objectives initially defined for this study in

Section 1.5 of Chapter 1, and highlighted as follows:

- To facilitate the translation of POPIA requirements into machine-interpretable language.
- To determine the best design pattern to meet the technical and operational requirements for POPIA compliance within organisations.
- To formulate a model to use for the validation and verification of personally identifiable information against the background of POPIA regulations.

It is important to highlight the objectives of this study at this juncture because the outcome of the research process can only be measured against the set research objectives

5. Conclusion: In concluding on the design science research process in general, and according to **Figure 4.2,** the design science research methodology requires that every conclusion should be set against the design framework and theories guiding the study.

Hence, to create the final research design framework for this study, elements of the Wieringa (2003) – Design Science Approach to Solve Problems, and the Vaishnavi and Kuechler (2005) – Comprehensive Design Science Research Process Model, are adapted and combined to create a process that is both simple and comprehensive. The final proposed research design framework for this study presented in the next section, has elements of both processes, illustrated in **Figure 4.1 and 4.2**, respectively.

4.3.3 The Research Design Framework

To recap, according to Cresswell (2009), a framework is a support structure or frame that holds parts together. In other words, according to Collins and Stockton (2018), a framework can be seen as a particular set of rules used to deal with a problem. Following this definition, the research design framework for this study is crafted, first to structure the research problem and second, to understanding the problem in a new way. As specified in **Section 4.3.1**, elements of the Wieringa (2003), and Vaishnavi and Kuechler (2005) models, are condensed to produce the research design framework for this study, as illustrated in **Figure 4.3**.

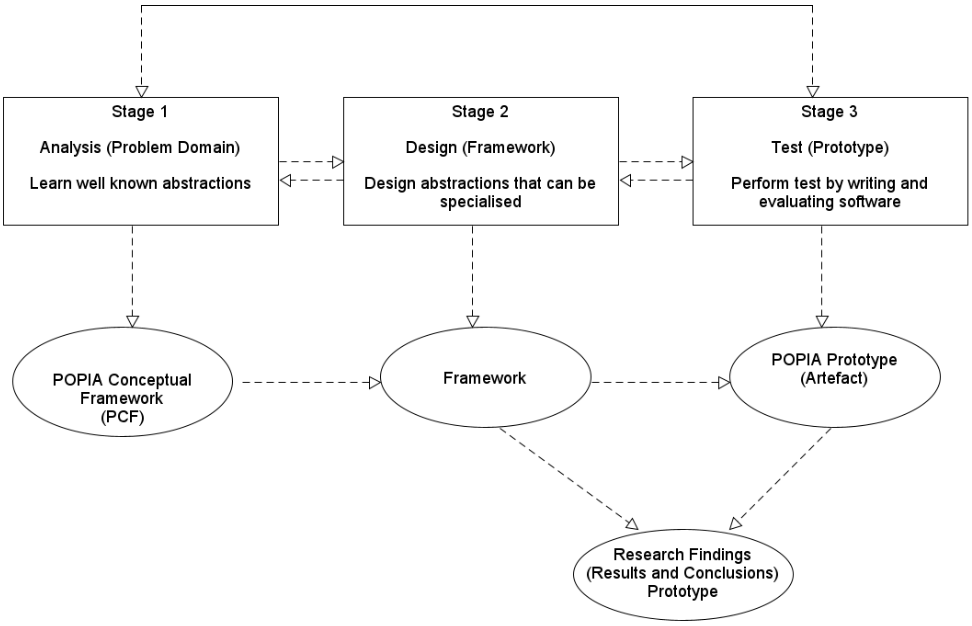

Stage 1
Analysis (Problem Domain)
Learn well known abstractions
Stage 2
Design (Framework)
Design abstractions that can be specialised
Stage 3
Test (Prototype)
Perform test by writing and evaluating software
POPIA Conceptual Framework (PCF)
Framework
POPIA Prototype (Artefact)
Research Findings (Results and Conclusions) Prototype

Figure 4.3 Research Design Framework

First, this research design framework comprises three stages, namely analysis, design, and test, respectively. The stages are all connected to one another and each can serve as both an input and output element into the preceding or following stage, and the entire process is iterative.

According to the research design framework depicted in **Figure 4.3**, the first stage (**Stage 1**) is where the problem and the problem domain are analysed. Based on this analysis, abstractions are inferred and examples of programmes to be built using these abstractions are decided upon and supported by appropriate justifications. The main output of this stage is the conceptual framework and the different contexts developed in Chapter 5 and notably, the organisational context, which is seen as the main stage for information privacy compliance activities within the organisation**.** Considering that compliance with information privacy is an organisation-wide problem, proper organisational scoping and governance activities need to happen to enforce information privacy**.**

The second stage (**Stage 2**) makes reference to the design abstractions. These design abstractions are developed using the design science research principles and object-oriented software development methodologies, and Rauch’s (2012) three-level model, discussed in **Section 4.4** of this study. The different contexts of the conceptual framework developed in Chapter 5 serve as input to this stage. The main output is first the design framework, from which the POPIA prototype (artefact) and its components can be designed. These designs are used directly as models to explain the research problem based on the design theory adopted for the particular study which, in this case, is the design science research methodology (DSR). Once the designs of the abstractions are completed, the next stage is the testing phase.

The testing phase constitutes the last stage in the proposed framework for the design of the POPIA prototype. Based on the proposed framework, the most effective way to test the proposed framework is to apply the framework in solving a particular problem, as demonstrated in Chapter 6 of this study. By solving the problem, the different components of the framework are called upon and eventually the actual testing is performed by writing software programs. These software programs, in the context of this study, are represented by business rules encoded in software-based objects using methods of business rules engine execution. In the next section, in which the POPIA prototype is developed, the details of the business rules execution methods are elaborated upon in more detail and in context.

Based on this proposed framework, the practical implementation and testing of the prototype and by implication, the framework, is discussed in Chapter 7, resulting in research observations and findings, discussed in Chapter 8**.** Once a problem has been subjected to these three blocks and is eventually solved, then the research design framework can be confirmed to be working and capable of solving problems. **Figure 4.4** illustrates the link between the proposed research design framework used in the implementation and execution of the different sections of this research study. Based on **Figure 4.4**, the three phases of the proposed research design framework are interactive and linked, with each phase serving as input and output to the next phase. As illustrated in F**igure 4.4**, the three main outputs of the research design framework are highlighted as follows:

1.) **The POPIA conceptual framework:** This output occurs as a result of the analysis of all the abstractions of the research problem, as demonstrated in Chapter 5**.**
2.) **The design framework:** This output exists as a result of the design of the different abstractions of this study, as represented by the different contexts of the conceptual framework which serve as input, as discussed in Chapter 6**.** The design framework produced here is used as a template to develop the POPIA prototype implemented in Chapter 7. To generate this design framework, the Rauch (2012) three-level model is used. **Section 4.4** throws more light on the Rauch (2012) three-level model for the development of a design framework.
3.) **Research findings:** This output occurs a result of the testing and simulation of the prototype, which is explained in Chapter 7**.** The main input into this process is the prototype itself or artefact, implemented in Chapter 7.

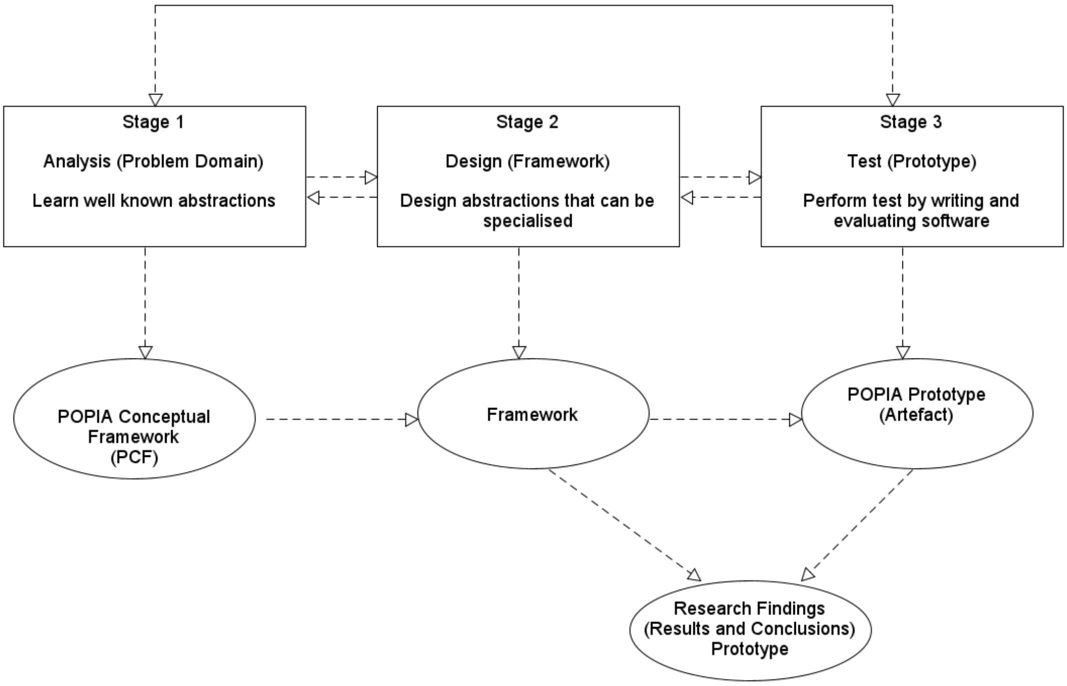

Stage 1
Analysis (Problem Domain)
Learn well known abstractions
Stage 2
Design (Framework)
Design abstractions that can be specialised
Stage 3
Test (Prototype)
Perform test by writing and evaluating software
POPIA Conceptual Framework (PCF)
Framework
POPIA Prototype (Artefact)
Research Findings (Results and Conclusions) Prototype

Figure 4.4. Research Design Framework

4.4 Rauch's Three-Level Model to Create Frameworks

To develop the design framework for this study, represented as **Stage 2** of the research design framework, illustrated in **Figure 4.4,** a three-level approach is used based on the Rauch (2012) model. **Figure 4.5** illustrates the Rauch (2012) model.

The rationale for using the Rauch (2012) three-level model is to have a common ordering scheme and language that serves as a guide to ease complexity linked with the design of the framework for this study. The three (3) levels of the Rauch (2012) model are: the design level, the planning level, and the execution level. This model was applied to the design of the framework of this study in Chapter 6.

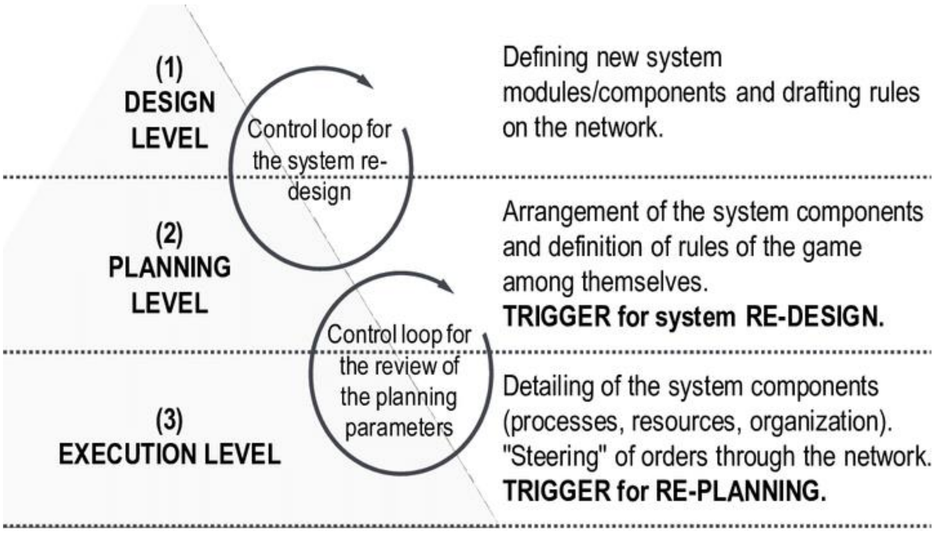

(1) DESIGN LEVEL
Control loop for the system re-design
Defining new system modules/components and drafting rules on the network.
(2) PLANNING LEVEL
Arrangement of the system components and definition of rules of the game among themselves.
TRIGGER for system RE-DESIGN.
Control loop for the review of the planning parameters
(3) EXECUTION LEVEL
Detailing of the system components (processes, resources, organization).
"Steering" of orders through the network.
TRIGGER for RE-PLANNING.

Figure 4.5. Rauch's Three-Level Model to Create a Framework

(Source: Adapted from Rauch, 2012)

In terms of the details of the Rauch (2012), three-level model, the following is noted.

- **Design level:** At design level, the components and modules of the framework are identified and highlighted.
- **Planning level:** At the planning level, the arrangement of the components and

definition of their rules and roles within the framework are discussed in context.

- **Execution level:** At the execution level, inner details of the framework components, such as processes, resources, and organisation are discussed in context.

What is noteworthy is that the three levels of the Rauch (2012) model are interactive in nature and the process to create these levels might loop through several iterations between the levels to arrive at the final outcome. In terms of utilisation, Rauch used this model in the design of a technology framework used in the development of technology systems for the food production chain with much success (Rauch, 2012). Similarly, in this study, this model was used to develop the framework used in building the prototype for POPIA compliance.

4.5 Conclusion

In this chapter, the design elements of two (2) different and distinct conceptual frameworks used for understanding, executing and evaluating design science research were adapted and combined to create a research design framework for this study. The conceptual frameworks used were the Wieringa (2003) – Design Science Approach to Solve Problems, and the Vaishnavi and Kuechler (2005) – Comprehensive Design Science Research Process Model.

The rationale for combining these two conceptual frameworks and creating the unique research design framework for this study was first, to create a design science research process that is unique, simple, and comprehensive enough to cater for the specificity of this study and second, to create a framework that is used as input into the following chapters of this study to give form and structure to the research process and the output of this study.

Additionally, the Rauch three-level model to build frameworks was also introduced in this chapter. This model was used in building the framework used in Chapter 7 to implement the prototype.

In conclusion, this chapter achieved the goal of creating a comprehensive Design Science Research Process Model used in this study to analyse the research problem, build and evaluate the research artefact in context.

CHAPTER 5

Conceptual Framework

5.1 Introduction

In Chapter 4, the research design framework for this study was elaborated upon where the structure, form, and output of the following chapter were outlined. This chapter draws on the research design framework, focusing on the first stage of the research design framework, which is dedicated to the analysis of the problem domain and having the POPIA conceptual framework as its major output. **Figure 4.1,** illustrates the sections of the conceptual framework that on which this chapter focuses. The chapter is divided into four (4) sections**.**

Section 5.2 discusses the background to the conceptual framework; here a top-down approach is used to introduce all of the contexts linked to the conceptual framework. Also in this section, the objectives, the research setting, and the compliance domain of the conceptual framework are discussed

In **Section 5.3**, the conceptual framework and its different components and contexts are developed following the Regoniel (2015) four (4) broad steps approach. Using this approach, the important variables and concepts of this study emanating from the literature review conducted in Chapters 2 and 3, respectively are highlighted and discussed.

Finally, in this section, six (6) contexts are identified and modelled through which information is handled and through which information flows in the real world**.** **Section 5.4** is the final section of the chapter, which summarises the chapter and highlights the main concepts to serve as input for the design framework in Chapter 6.

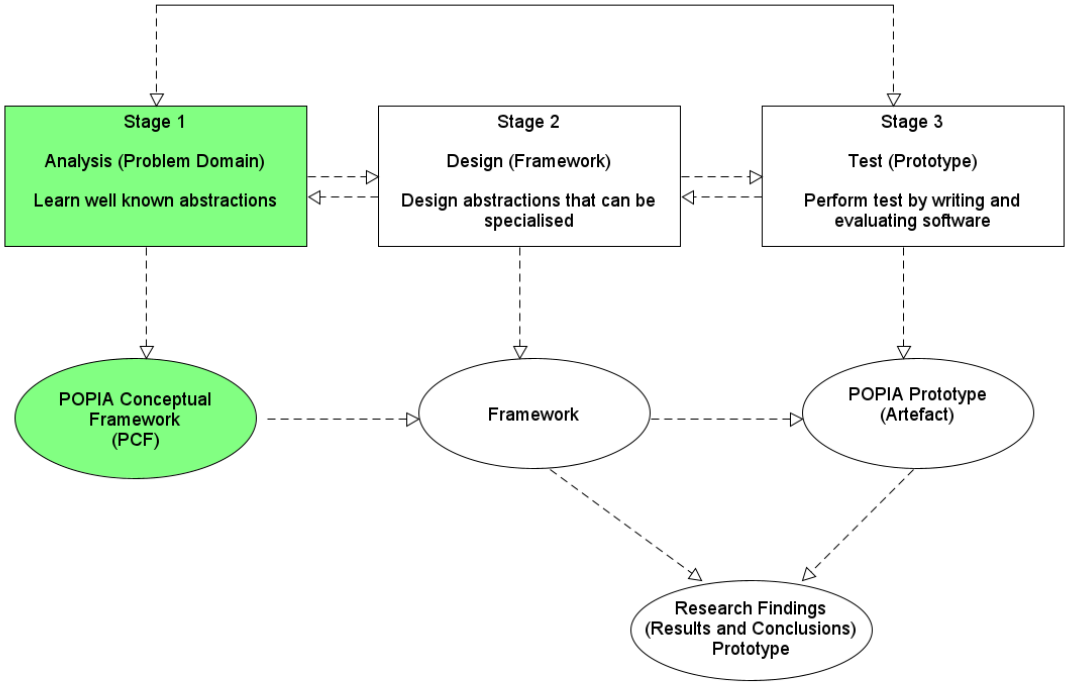

Stage 1
Analysis (Problem Domain)
Learn well known abstractions
Stage 2
Design (Framework)
Design abstractions that can be specialised
Stage 3
Test (Prototype)
Perform test by writing and evaluating software
POPIA Conceptual Framework (PCF)
Framework
POPIA Prototype (Artefact)
Research Findings (Results and Conclusions) Prototype

Figure 5.1. Research Design Framework: Stage 1

5.2 Background to the Conceptual Framework

In general, according to Regionel (2015), all conceptual frameworks are based on the identification of key concepts and the relationship among these concepts. Furthermore, according to Regionel, a conceptual framework lies within a much broader framework called the theoretical framework. In turn, the theoretical framework draws from time-tested theories developed by researchers to explain their findings. In addition, the theoretical framework also provides a focal point to investigate the unknown in a specific area of study. (Regionel 2015). In the context of this study, the same principle will apply. This means that the main concepts of this study are captured and the relationship between the concepts are highlighted with the objective of showing a representation of this study based on the observation of the researcher in order to develop a conceptual framework. In fact, according to Regionel (2015), a conceptual framework is described as a researcher’s understanding of how the particular variables in his/or her study connect with each other. This very basic description of the conceptual framework provides a powerful mechanism which allows the researcher to identify the required variables in the research investigation and to build a map linking these variables together based on the peculiarities of the research. In fact, Regionel proposes the following four (4) broad steps to develop a conceptual framework:

1. Choose your topic;
2. Do a literature review;
3. Isolate the important variables; and
4. Generate the conceptual framework.

However, there is no undisputed way of building a conceptual framework. Ravitch and Riggan (2016) also reflected on how best to construct a conceptual framework and settled on two approaches, namely structured reflexivity and dialogic engagement. First, structured reflexivity is centred around reflective writings, which dwell on questions about the research such as the research design and questions/topic, the research method and process; and second, dialogic engagement focuses on structured conversations with individuals and groups to enable the researcher (s) to develop opinions about the research process, methods, data, and scope. To develop the conceptual framework for this study, structured reflexivity is consulted as it is more design focused and iterative, unlike dialogic engagement which involves individuals and group discussions to formulate the conceptual framework. However, the Regionel four (4) broad steps approach is used primarily to guide the process to create the conceptual framework.

In terms of the envisaged structure of the conceptual framework, according to Jabareen (2009), a conceptual framework is defined as a network or a ‘plane’ of linked concepts. This definition sets the tone for leveraging a conceptual framework as a mechanism to analyse related concepts. In line with this definition, the conceptual model for this study views information privacy compliance as a context-driven reality operating within a privacy context which, in this case, is the POPIA compliance domain**.**

5.2.1 POPIA Compliance Domain

In general, the essence of the POPIA compliance domain is to understand the guidelines and boundaries required for the lawful handling of personal data according to the POPIA legislation. In this study, the POPIA compliance domain helps in understanding and ring-fencing how best to protect the privacy of information flowing within and out of the organisational information systems. A good understanding of the boundaries of POPIA helps in the formulation of the POPIA rule engine conceptual framework for this study. The POPIA compliance domain, according to Naveg (2016), highlights eight principles that are accepted and agreed upon in South Africa as the standard for the implementation of POPIA. These principles are listed and briefly described as follows:

- **Accountability**: Every organisation must have an accountable officer who is appointed to be in charge and responsible for POPIA compliance.
- **Processing limitation**: This condition deals with the legality of processing personally identifiable information, which includes recording the consent, justifications, and objections of the data subject.
- **Purpose specification:** This covers the purpose for which the information is being collected and must be disclosed upfront and made clear to the data subject.
- **Further processing limitation:** This principle takes effect when further processing is done on the data and such processing must be compatible with the purpose for which the information was initially collected.
- **Information quality:** This involves making sure that the personal information collected is complete, up-to-date, accurate, and not misleading.
- **Openness**: This deals with the requirement of transparency to both the data subject and the information regulator in the process of handling personal information.
- **Security safeguards:** In this condition, the safeguarding of the personally identifiable information should be guaranteed through administrative and technical means.
- **Data subject participation:** The data subject can exercise their right, at any

point in time, to enquire whether any of their information is being held by the data custodian.

Figure 5.2 helps to simplify and contextualise the POPIA compliance domain and the eight elements required to process personally identifiable information legally as part of the POPIA compliance domain as well as the different actions that can be undertaken on the POPIA data.

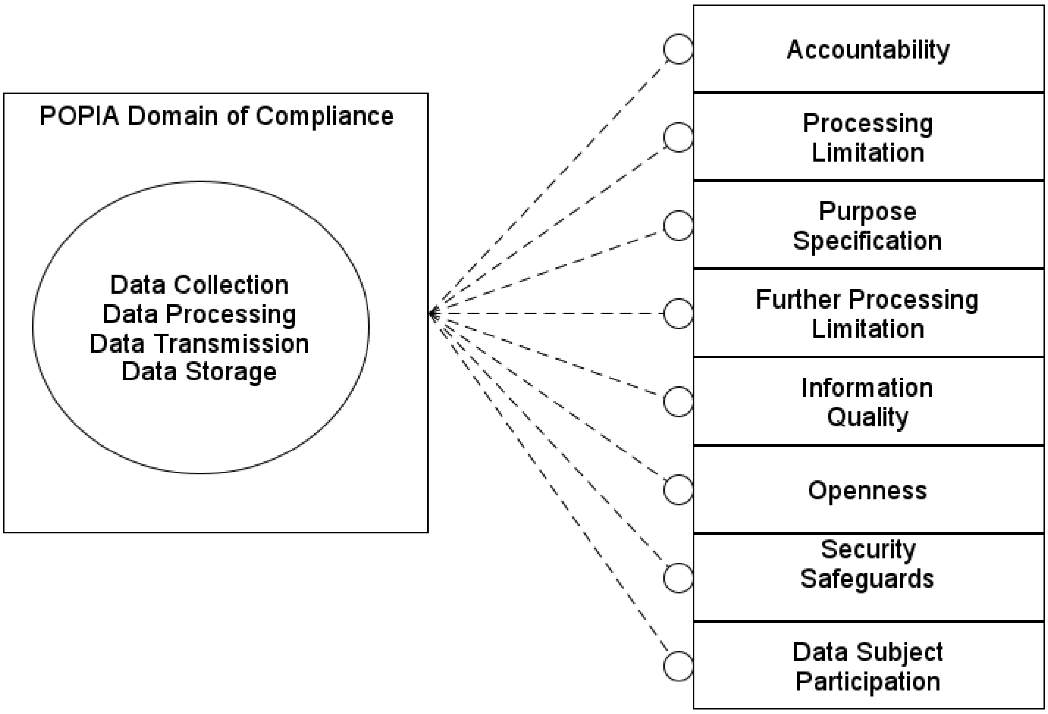


Figure 5.2: The POPIA Compliance Domain

A good understanding is required of the POPIA compliance domain and its guidelines and boundaries for the lawful handling of personal data. The next section showcases the objectives of building the conceptual framework in the context of this study.

5.2.2 Conceptual Framework Objectives

Before delving into the formulation and design of the conceptual framework in **Section 5.3**, this section highlights the conceptual framework objectives.

First, the primary objective to developing this conceptual framework called the *POPIA Rule Engine Conceptual Framework* or simply, *the POPIA Conceptual Framework (PCF),* is tied to the common rationale and essence of any conceptual framework from an academic perspective, which is to understand some of the key variables and theoretical concepts that underpin this study such as the POPIA, the POPIA compliance domain and the research paradigm which, in this case, is the design science research methodology.

Second, in terms of this study, the need to develop a conceptual framework is born out of the research objectives of this study, as highlighted in **Section 1.5** of Chapter 1 and outlined in **Objectives 1, 2, and 3**, respectively. More so, this conceptual framework serves to illustrate the context of this research and to showcase the different variables, particularly, how they relate to one another in the context of enforcing information privacy within an organisation by means of user-configurable software objects in line with **Objective 2** of **Section 1.5** of Chapter 1

Third, the conceptual framework developed in this chapter will serve as input into the design framework to be developed in Chapter 5.

Finally, to expand on the understanding of the role and context of this conceptual framework as it pertains to this study, the next section deals with the research settings on which this study hinges.

5.2.3 Conceptual Framework Research Setting

In general, this study is underpinned by the design science research methodology (DSR). DSR, as explained in **Section 1.8**, is a solution-oriented research paradigm with an engineering cycle used to investigate and model research problems into practical constructs that represent stakeholders of the problem, the research goals, and the phenomenon, the evaluation of the research problem and practical diagnosis of the problem in context. Hence, to investigate the research problem of this study, which is how POPIA can be used effectively to enforce information privacy with an organisation using software-based objects, the POPIA conceptual framework was developed. This PCF was constructed using the prescriptions of the DSR, which mandate the creation of an artefact which becomes the main research tool and domain of enquiry of the research. Therefore, DSR method is used to create this PCF, which connects all the different aspects of this research as well as serving as a guide to the researcher in the process of conducting the research and building the prototype.

In the particular setting of this study, this conceptual framework serves as input into an information systems artefact or prototype that is implemented in Chapter 7 of this study. In addition, the conceptual framework also provides the research context, guide and scope / boundaries of the prototype created for this study.

In Chapters 7 and 8**,** this prototype is observed in context to see how it reacts to the research problem, which is to help organisations operating in South Africa and struggling to enforce information privacy compliance in their technology systems. This prototype assists the research by helping to answer the research questions of how POPIA can be implemented as software-based objects and can be used to enforce information privacy compliance.

In summary, the PCF developed for this study is intended to achieve two major outcomes, namely:

1. To provide a guide to investigate the research topic in the right research context
2. To make the case of how the research design, in this case DSR, helps in answering the research questions.

The next section focuses on the development of the POPIA conceptual framework of this study using the Regionel process as a guide, as elaborated upon in **Section 5.2**.

5.3 POPIA Conceptual Framework

In this section, a top-down approach is used, based on the Regionel (2015), proposed four (4) broad steps to develop a conceptual framework to introduce and develop the different components of the conceptual framework. To recap, these steps are: choose your topic, do a literature review, isolate the important variables, and generate the conceptual framework.

Step 1: Choose your topic

The topic of this study is: ***An information privacy compliance model based on configurable software objects***. This topic covers the main concepts and key themes of this study and was chosen at the onset of the study after consultation with the research sponsors and approval was obtained through a successful research proposal process administered by the university. The next step in the regional process is to do a literature review.

Step 2: Literature review

A concise and comprehensive literature review was conducted in **Chapters 2 and 3** of this study. In this literature review, the main themes of this study, information privacy and privacy engineering as it is appreciated by modern organisations in terms of compliance and technology was discussed in detail and in context. In addition, the key theories of information privacy, how it has evolved over time and the state of information privacy research at the time of this research were also examined in detail. Furthermore, to understand the role of information privacy fully in the right context within organisations, the literature review explored related concepts, such as information security, systems engineering, data protection, IoT, PbD, privacy standards, and privacy regulations to find the areas of intersection and specialisation leading to the next step in the Regionel process, which is to isolate the important variables.

Step 3: The main variables

Based on the extensive and comprehensive literature review covered in Chapters 2 and 3 of this study, a number of important variables were isolated for this study. These variables serve different purposes. Some are input variables; others are output variables, while some serve as both input and output feeding into the next steps and iterations of the research process. **Table 5.1** lists these variables, their direction (input, output, or input / output) and gives a brief description of their role in the context of this study.

Table 5.1 Important Variables and their Description

	Variable name	**Direction**	**Description**
1	Design science research method	Input	The main research methodology used in this study to create and evaluate the research artefact or prototype.
2	Business rules approach	Input	The frameworks used to encode the prescriptions of POPIA as business rules into software-based objects.
3	POPIA	Input	The information privacy legislation of South Africa that requires all organisations operating in South Africa to comply. The outcome is a POPIA compliance domain.
4	Software development methodology	Input	The software development approach used to build the software-based objects applied to implement the business rules within the prototype.
5	UML use case ontology	Input	The language used to model and represent the structure and inner working of the artefact or prototype.

	Variable name	**Direction**	**Description**
6	POPIA conceptual framework	Input/Output	A framework that captures the main concepts and variables of this study and shows the relationship between them. It serves as input into the design framework of this study.
7	POPIA design framework	Input/Output	A framework that shows the steps and approach of translating the conceptual framework into a prototype design for this study. It serves as input into the design of the prototype.
8	Business rule engine	Input/Output	The business rules engine is used to translate POPIA into reusable business rules. This serves as an input into the software-based objects created for this study.
9	Software objects	Output	These are the information systems components created to enforce business rules within the information systems of the organisations.

	Variable name	Direction	Description
10	Information systems	Output	These are the organisational technologies and tools use to collect, process and store data within an organisation and must comply with the prescriptions of POPIA.

In view of these important variables identified for this study and highlighted in **Table 5.1,** the next section focuses on generating the POPIA conceptual framework for this study.

Step 4: Generating the conceptual framework

To generate this framework, reference is made to the research setting discussed in **Section 4.2.3**. Generally, a research setting provides a lens through which the research approach, arguments, findings and conclusions can be viewed. In light of the different variables and concepts identified, this research phenomenon is seen and modelled as a context-aware phenomenon. In this model, the variables and concepts of this study are broken down and examined in six (6) different contexts in which information exists and through which it flows in the real world. Some of these contexts are sub-contexts of the main contexts identified. In addition, each of these contexts has its own distinctive input and output elements or criteria which feed into the next context in this framework.

In terms of this model, the top two main contexts are the research context and the organisational context, as illustrated in **Figure 5.3**.

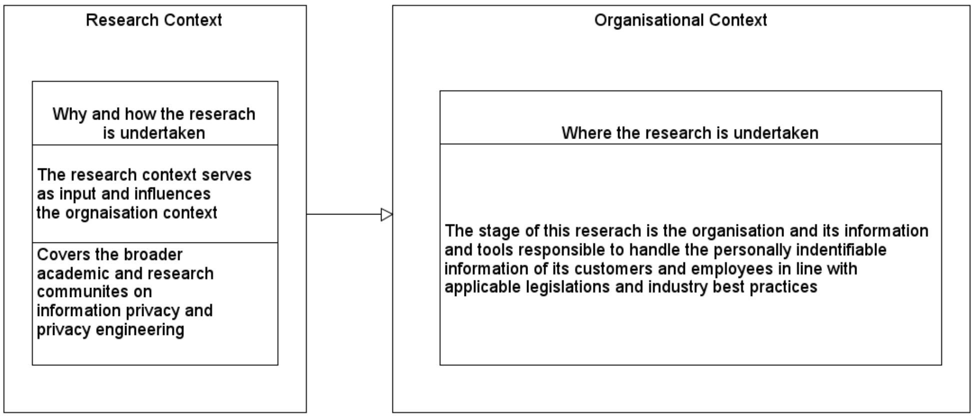

Research Context
Why and how the reserach is undertaken
The research context serves as input and influences the orgnaisation context
Covers the broader academic and research communites on information privacy and privacy engineering
Organisational Context
Where the research is undertaken
The stage of this reserach is the organisation and its information and tools responsible to handle the personally indentifiable information of its customers and employees in line with applicable legislations and industry best practices

Figure 5.3: Structure of Top Two Main Contexts

Based on **Figure 5.3**, the research context focuses on why and how the research is undertaken while the organisational context covers where the research is undertaken. The other contexts discussed are all subsets of the organisational context, notably the information system context which, in turn, also has its own sub-contexts, such as validation context and the technology context which serve as direct input into the information system. **Figure 5.4** illustrates the hierarchical structure of all these contexts.

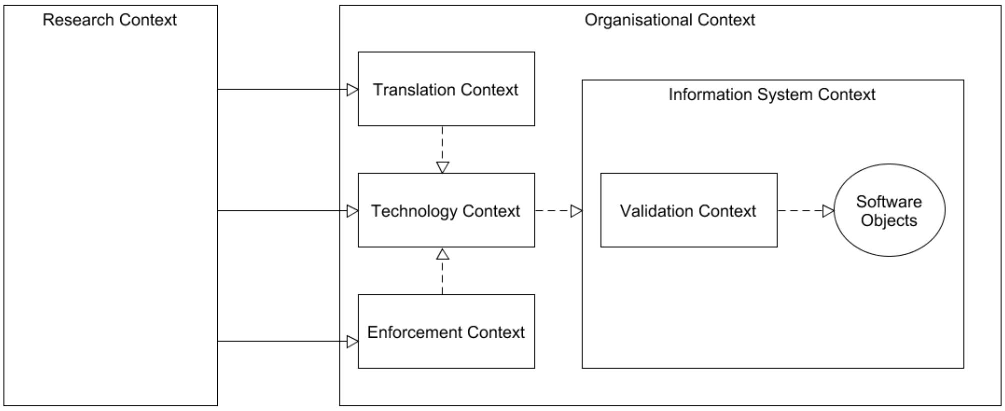


Figure 5.4: Hierarchical Structure of Contexts

5.3.1 The Research Context

This context lies within the academic and research community worldwide. It leans on the different theories covering information privacy compliance. As such, it leads to the definition of privacy dimensions; in addition, to how they can be quantified and measured and finally, to the ethics of what is allowable within a privacy-sensitive domain of business and human rights. This context can be seen as the strategic enabler shaping the direction in which information privacy is moving and, in the context of this study, influencing the organisations handling personally identifiable information. The research context is mostly argued and counter-argued by academics, policy-makers, human rights organisations, ethics organisations and the civil society in general. In this study the research context focuses on the use of software-based objects to enforce information privacy compliance using POPIA as the main regulation to be enforced since the research is taking place in South Africa and POPIA is the de facto information privacy regulation for all organisations operating in South Africa.

To conceptualise the research context fully, the design science research methodology is employed to develop a POPIA prototype or artefact, which is made up of software objects. These objects encode POPIA regulations as business rules within information systems used by these organisations to handle personally identifiable information. The design science research approach is used because it is solution oriented and mandates the creation of an artefact to simulate and evaluate the research problem in context to understand and answer the research question. To design and build this artefact practically, as shown in Chapters 6 and 7, the architectural synthesis employed uses agile software development methodology and UML ontology to model the different software components or objects of the POPIA prototype.

This research context leads and serves as input into the next context, which is the organisation context, as illustrated in **Figure 5.4**. In terms of the organisational context, it is the main context of this study and the serves as the stage of information privacy compliance.

5.3.2 The Organisational Context

According to POPIA, all organisations handling personally identifiable information in South Africa are mandated by law to comply with POPIA. As a result of this, the organisation becomes the main stage through which POPIA privacy compliance operations take place. In fact, the organisation is the primary setting for this research owing to the following reasons:

1. The organisation is the main entity that is responsible and accountable for the handling of personally identifiable information belonging to the data subjects under their custody.
2. The information system that stores, processes, and transmits this data is owned and operated by the organisation.
3. The human resources that are responsible for handling this information are either employees or service providers to the organisation.

Mindful of these reasons, and coupled with the extent, size, and nature of data that is collected, processed and used by these organisations, it is important to understand the different components of POPIA compliance within the organisational context. This challenge is further compounded by the diversity of information systems tools and technologies used to handle data within these organisations. Hence, the organisational context represents all the elements of the information custodianship domain within public or private organisations compelled by law to comply with conditions for the lawful processing of personal information, as determined by the de facto information privacy legislation (POPIA).

This context contains all the other contexts, except the research context that lies outside organisational context and serves as direct input into the organisational contexts and its sub-contexts, as highlighted in **Figure 5.3**. In essence, all POPIA privacy compliance happens within the organisational context. The organisational context is made up of the following sub-contexts: information systems context, translation context, technology context, and enforcement context, as illustrated in **Figure 5.5**.

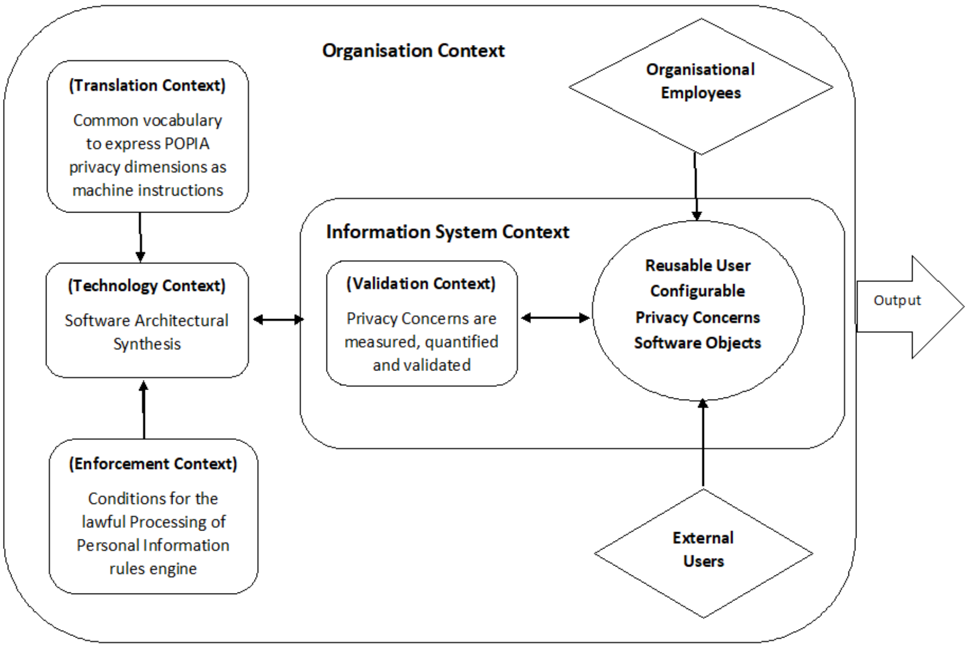

Organisation Context
(Translation Context)
Common vocabulary to express POPIA privacy dimensions as machine instructions
Organisational Employees
Information System Context
(Technology Context)
Software Architectural Synthesis
(Validation Context)
Privacy Concerns are measured, quantified and validated
Reusable User Configurable Privacy Concerns Software Objects
Output
(Enforcement Context)
Conditions for the lawful Processing of Personal Information rules engine
External Users

Figure 5.5: Organisational Context

5.3.3 The Information System Context

This context covers mainly the validation context, the reusable user-configurable privacy concern software objects, the employees of the organisation who are responsible for collection, transmission, processing, and storage of the data subjects' personally identifiable information as well as the users whose personally identifiable information is being kept in the information system of the organisation. This is the main context through which information privacy and compliance is either enforced and complied with or violated. In this same context, data subject privacy concerns are validated against POPIA regulatory rule objects to ensure that the data subject's privacy is respected. This is the context that will be used in this study to create the POPIA reusable prototype. **Figure 5.6** shows the elements of the information system context and how they are linked to each other.

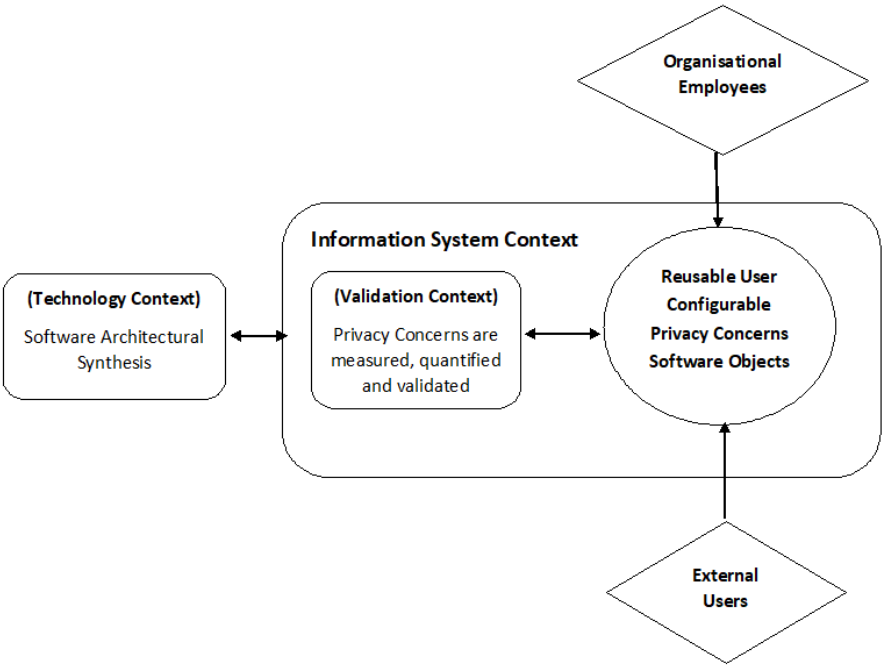

Organisational Employees
Information System Context
(Technology Context)
Software Architectural Synthesis
(Validation Context)
Privacy Concerns are measured, quantified and validated
Reusable User Configurable Privacy Concerns Software Objects
External Users

Figure 4.5: Information System Context

In terms of input, the technology context, which determines the main software architectural synthesis, is a direct input into the information system context. The architectural synthesis in this case refers to the software design pattern and software development methodology used to build the POPIA software-based objects.

With reference to **Figure 5.6**, emphasis is equally placed on the reusable user-configurable privacy concern software object. This object is part of the information system context of the broader POPIA rules engine conceptual framework. It is expected that the user's privacy preferences are captured and configured in this object. Before this happens, the privacy preferences are measured, quantified and validated in the **validation context,** which is also part of the information system context. The verification of the metadata of the data subjects is validated in this context to determine which rules to apply when dealing with the particular data subject. Based on **Figure 5.6**, the data elements used to drive these contexts are obtained from two main sources, namely the external users who are using this system and the employees of the organisation who are also using the system. In this study they are collectively referred to as the data subjects. Based on **Figure 5.6**, all the manipulation of data takes place within the information system context and precisely within the reusable user-configurable privacy concern software objects. Furthermore, the output of this context is the software-based objects which can be shared with other organisations operating in the same domain to enforce information privacy. Serving as the main input to the information system context and its sub-contexts is the technology context.

5.3.4 The Technology Context

The technology context lies within the organisational context and serves as the main input into the information system context. This context focuses on the choice of appropriate technology, software design pattern and software development methodologies to be used for the technical and operational requirements of enabling POPIA privacy compliance within the organisational context. In the context of this study, in Chapter 7, tools like UML use case analysis, static and dynamic modelling techniques, and case study evaluation are used to determine the appropriate software architectural design synthesis to best achieve POPIA compliance for the organisation when the prototype is being implemented. The next sub-context within the organisation context is the translation context.

5.3.5 The Translation Context

This context lies within the organisational context and represents the technical and analytic activities involved in creating a common vocabulary to express the POPIA privacy dimensions and measurements as machine-readable instructions for consumption by a software model. This context uses an ontology creation language such as UML and its modelling techniques to represent the information privacy metrics. The goal of this context in this research is to facilitate the translation of POPIA requirements into machine-interpretable language and the focus is on data management and data classification. To achieve this, this context identifies sensitive and non-sensitive data within the organisation and applies different sets of rules for their utilisation. This leads to the enforcement context.

5.3.6 The Enforcement Context

The enforcement context also lies within the organisation context and it is primarily driven by the regulatory landscape which defines the conditions for the lawful processing of personal information. The business rules engine structured after POPIA defines the conditions for the lawful processing of personally identifiable information. The key input is the type of industry and type of information being handled.

Based on all the six (6) contexts described in the preceding sections, **Figure 5.7** shows the relationship and data flow between all these contexts. It is presented here as the ***POPIA conceptual framework (PCF)*** for this study.

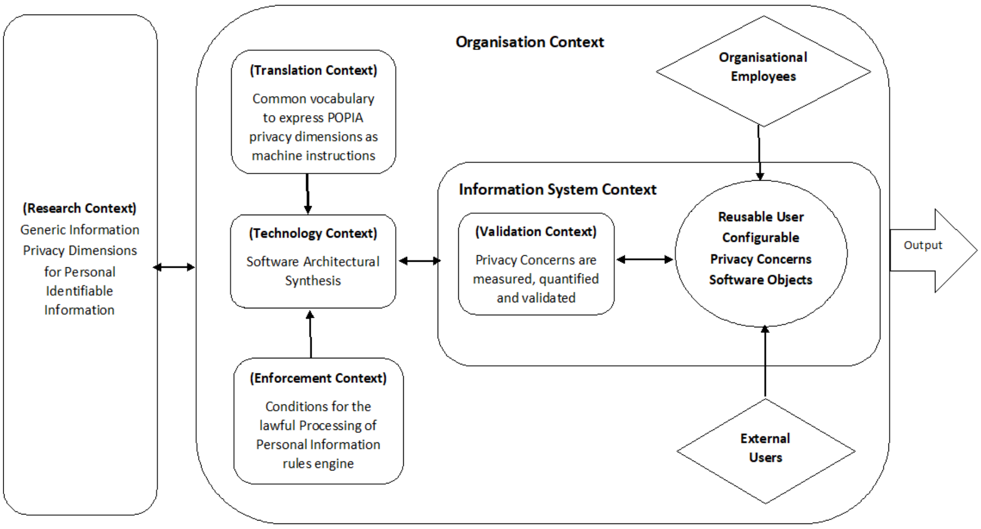

(Research Context)
Generic Information Privacy Dimensions for Personal Identifiable Information
Organisation Context
(Translation Context)
Common vocabulary to express POPIA privacy dimensions as machine instructions
Organisational Employees
(Technology Context)
Software Architectural Synthesis
Information System Context
(Validation Context)
Privacy Concerns are measured, quantified and validated
Reusable User Configurable Privacy Concerns Software Objects
Output
(Enforcement Context)
Conditions for the lawful Processing of Personal Information rules engine
External Users

Figure 5.7: POPIA Conceptual Framework

Mindful of all the contexts constituting the POPIA conceptual framework of this study, going forward this study focuses only on the organisational context and its sub-contexts, namely:

1. The translation context
2. The technology context
3. The enforcement context
4. The information system context, which is made up of the validation context, the reusable user-configurable privacy

concerns software objects, the organisational employees, the external users and the output.

The rationale for this choice is based on the arguments in **Section 4.3.1** which justify that the organisation is the main stage for all POPIA compliance activities in the context of this study.

5.4 Conclusion

In this chapter, the goal of developing a conceptual framework entitled the POPIA conceptual framework (PCF) was achieved. This framework serves to understand how the different variables and concepts of this study relate to one another within the information privacy and POPIA compliance domain.

From the proposed conceptual framework model, one main context (the organisational context) and its sub-contexts, highlighted in **Sections 5.3.2, 5.3.3, 5.3.4, 5.3.5,** and **5.3.6**, respectively aim to provide a guide for the development of the design framework in Chapter 6 and eventually, for the implementation of the prototype in Chapter 7, and notably, these contexts, were identified as most valuable for this study.

CHAPTER 6

Design of the Framework

6.1 Introduction

In Chapter 5, a conceptual framework was developed for this study. In this chapter, a framework is designed based on the elements of the conceptual framework developed. This framework is the second stage (**Stage 2**) of the overarching research design framework developed in Chapter 4 and highlighted in **Figure 6.1**. The framework is used as a generic model for the implementation of the prototype in Chapter 7.

To develop the framework, a three-level model is used based on the Rauch (2012) model. Hence, this chapter is divided into three main sections.

The first section focuses on the design level activities; here, the components and modules of the framework are identified and highlighted. The next section focuses on the planning level activities. In this section, the arrangement of the components and definition of their rules and roles within the framework are discussed.

The last section of this chapter focuses on the execution level activities. Here, detailing of the design framework components are discussed in terms of processes, resources and organisation. The chapter concludes with summaries of the key points discussed and introduces the next chapter **(Chapter 7),** which focuses on the implementation of the prototype.

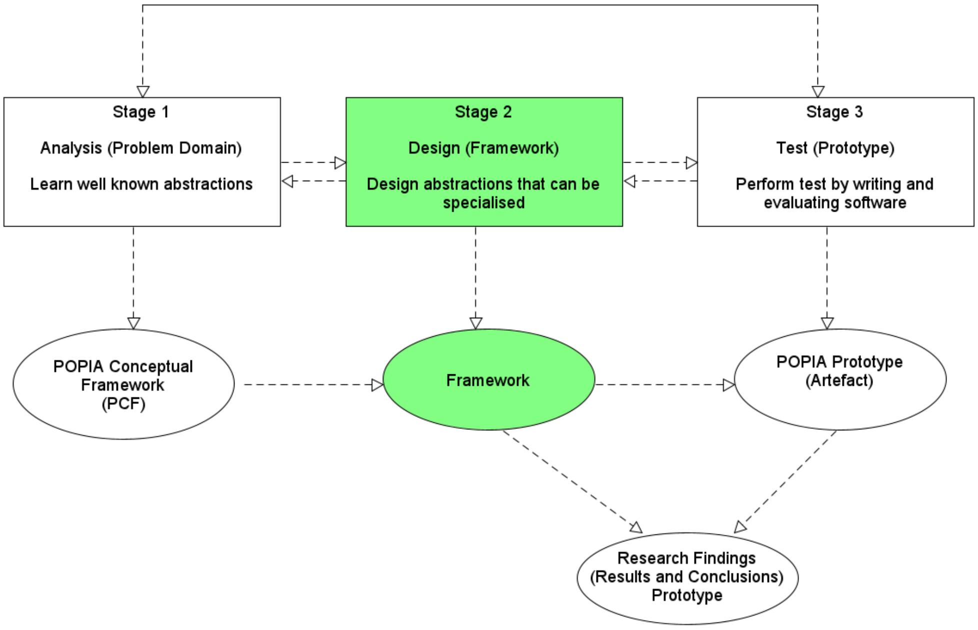

Stage 1
Analysis (Problem Domain)
Learn well known abstractions
Stage 2
Design (Framework)
Design abstractions that can be specialised
Stage 3
Test (Prototype)
Perform test by writing and evaluating software
POPIA Conceptual Framework (PCF)
Framework
POPIA Prototype (Artefact)
Research Findings (Results and Conclusions) Prototype

Figure 6.1. Research Design Framework: Stage 2

To recap, based on the conceptual framework developed in Chapter 5, the following contexts were highlighted, namely the research context, organisational context, information system context, technology context, translation context, and enforcement context. Within and between these contexts, information flows and data should be handled lawfully according to the dictates of POPIA. It is these same contexts that forms the basis for the development of the different components of the framework using the Rauch (2012) three-level model that is covered in the next section.

6.2 Design of the Different Components of the Framework

In this section, the Rauch (2012) three-level model was used to design the different components of the framework envisaged for this study. The next section starts with the design of the organisational context.

6.2.1 Design of the Organisational Context

Phase 1: Design Level: Organisational Context

In general, word organisation is used extensively in our daily life. According to Diksha (2020), organisation, comes from word 'organising', which is the function of gathering resources, and establishing the orderly use of such resources in a structured way to achieve a plan. Going from this, the term organisation has been defined in a number of ways by different practitioners, such as sociologist, anthropologist, scientist, psychologist, management theorist, to mention just a few. However, according to Diksha (2020), one such definition provided by Barnyard many years ago still remains very popular and is widely used within the research community. According to him an organisation is a system of consciously coordinated activities or efforts of two or more persons. In the context of this study, the definition of organisation provided by Diksha (2020) suffices, as it covers two (2) main points relevant to this study, that is, people and activities. Adapting this definition to the Rauch (2012) three-level model, the main components and modules of an organisation are identified as the people and the activities.

Phase 2: Planning Level: Organisational Context

At the planning level, based on the Rauch (2012) model, the activities are synonymous with the 'consciously coordinated activities' mentioned by Diksha (2020) and the people are referred to here as 'two or more persons'. Further adapting this definition to the study, the persons mentioned in this definition involve all the stakeholders who are involved in handling personally identifiable information within the organisation. The consciously coordinated activities mentioned in this definition involve all the data manipulation activities that the data is subject to, such as storage, transmission, processing, and deletion, archiving, and sharing. Hence, from this definition, to design this organisation becomes a simple process of identifying the activities of this organisation and the stakeholders. In this case, the activity is to comply with POPIA regulations, and the stakeholders are the employees of the organisation in addition to the users of the information systems of the organisation. In a nutshell, the organisation represents the stage in which all information privacy compliance activities is undertaken.

Phase 2: Execution Level: Organisational Context

In terms of the building blocks of this organisation, the two main components involved are the people (stakeholders), and the activities (data manipulating activities). Also important to note is that the organisation is being influenced actively by the research community and industry practices surrounding it. Hence, the people in this instance are the stakeholders of the POPIA system and the activities are the use cases that these stakeholders interface with as well as the research community and industry practices that inform, guide, and regulate these use cases.

Figure 6.2 illustrates the overarching architecture of the organisation based on the Rauch (2012) three-level model.

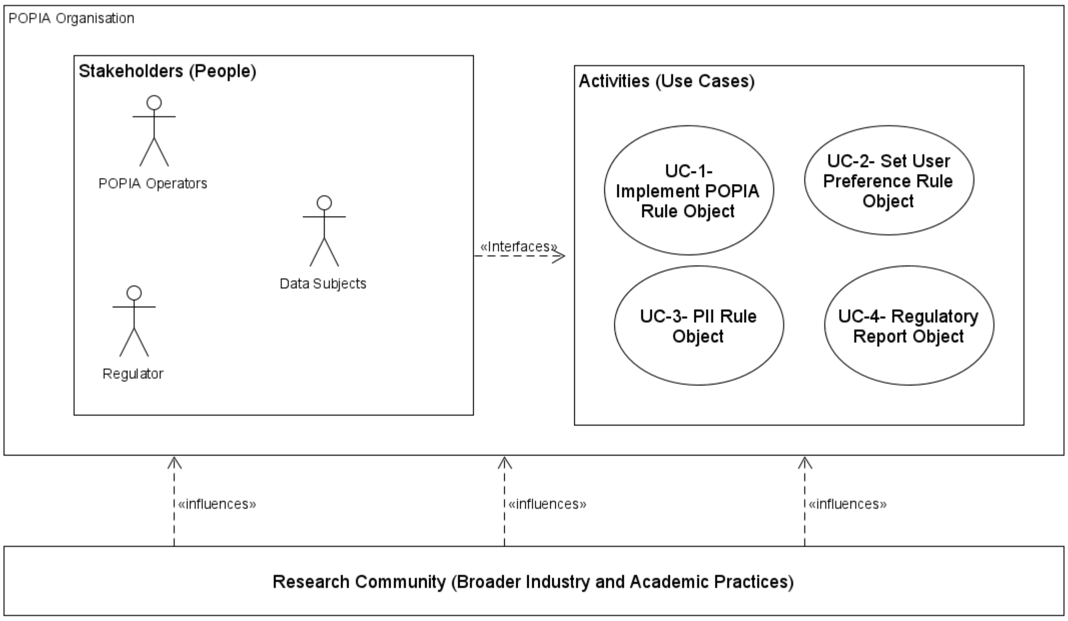


Figure 6.2. Organisation Design: Execution Level

6.2.2 Design of the Translation Context

Phase 1: Design Level: Translation Context

The goal of the translation context is to facilitate the translation of POPIA requirements into machine-interpretable language and the focus is on data management and how to apply different set of rules for data utilisation. To design this translation context, the full universe of data elements defined by POPIA are identified and incorporated into a data dictionary that is used for validation and inference. In this study, as defined by POPIA, the universe of data elements is referred to collectively as personally identifiable information (PII). According to The Protection of Personal Information Act (2013), PII is defined as ‘information relating to an identifiable, living, natural person, and where applicable, an identifiable, juristic person’. Simply put, it is any data that can be used to identify a natural or juristic person. According to Stringer (2011), virtually every organisation acquires, stores, and processes personally identifiable information (PII). Depending on the type of organisation, this might include information on students, patients, residents, debtors, prisoners, to mention just a few, and it is the most central concept in information privacy regulation (Swartz & Solove, 2011). Based on the POPIA legislation, this information about a person includes, but is not limited to:

- Race
- Gender
- Sex
- Pregnancy
- Marital status
- National / ethnic / social origin
- Colour
- Sexual orientation
- Age
- Physical or mental health
- Disability
- Religion / beliefs / culture
- Language
- Educational / medical / financial / criminal or employment history
- ID number
- Email address
- Physical address
- Telephone number
- Location
- Biometric information

- Personal opinions, views or preferences (The Protection of Personal Information Act, 2013)

Of the 21 items listed here, it is important to highlight that they must be used in a particular sequence to become a risk to the individual. Conversely, if they are used in another sequence the risk to the data subject is significantly lower. For example: John Peter, a white male living in Pretoria is an indicative statement but does not pose a direct risk to the individual. However, if they use his ID number or Passport Number and home address, then it points directly to the specific individual. Based on this sequencing of usage, **Figure 6.3** shows a hierarchical structure that is developed to represent the POPIA personally identifiable information (PPI) visually.

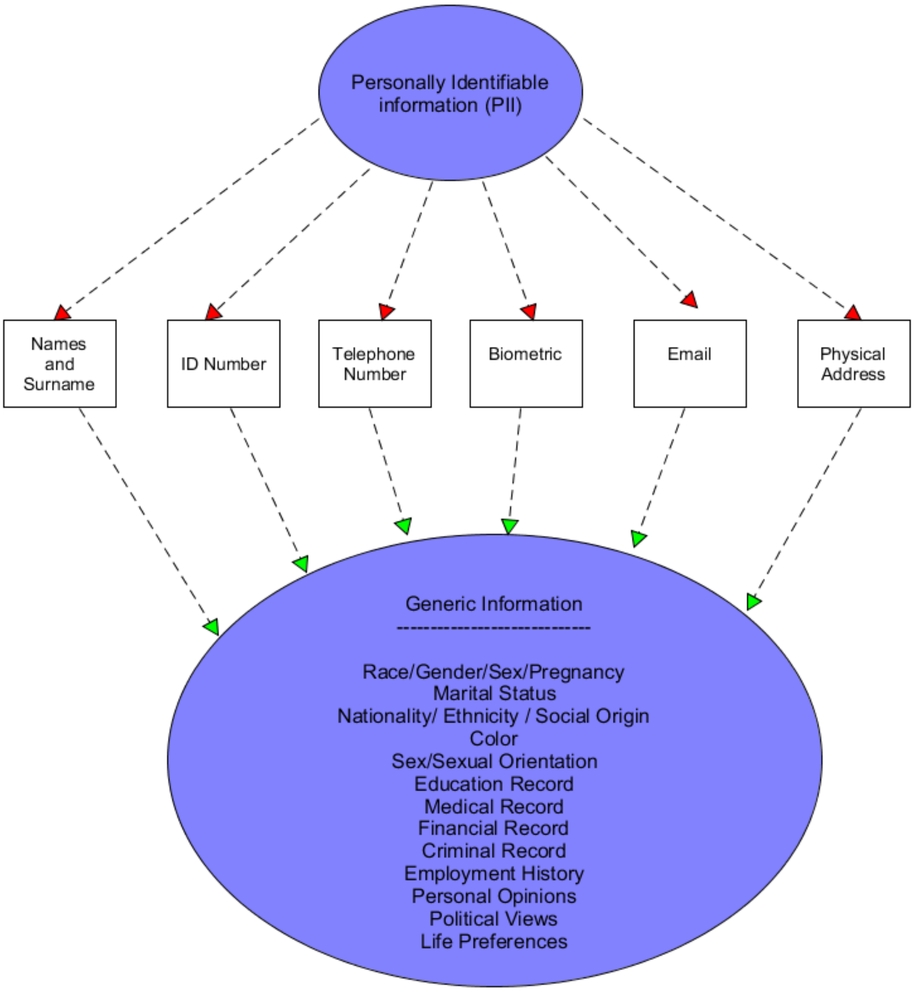


Figure 6.3. Hierarchical Structure of POPIA PII

This hierarchical structure highlights two levels of personally identifiable information. The top level is made up of six (6) elements, namely name and surname, ID number, telephone number, biometric, email, and physical address while the bottom level is made up of 13 elements, which are considered as generic information elements and can only be a threat to the data subject if they are used in combination with any one of the top level elements. In the translation context design, these elements are segmented as top- and bottom level elements and are encoded into a data dictionary of metadata, which constitutes the main design artefact for this translation context. This dictionary of metadata constitutes the common vocabulary and reference point for machine instructions. Adapting this to the Rauch (2012) model, the main components and modules of this context are: a) data dictionary of metadata represented by object-oriented class properties (fields); b) a common vocabulary for expressing the rules on the data represented by object-oriented class methods (functions).

Phase 2: Planning Level Activities: Translation Context

In terms of the planning activities, the translation context, which represents the common vocabulary to express POPIA rules, takes the form of an object-oriented class object, and the UML modelling language is used at the execution level to model the class object. This is premised on the fact that, according to Kindler and Krivy (2011), software class objects can contain both data and code. Data is represented by the fields of the class also known as attributes, and code in the form of procedures, also known as methods. Consequently, in designing the translation context of this study there will be a translation class called **<clsTranslate>**, which is a super-class made up of sub-classes, namely <**testforGenericData>** and <**testforTopLevelData>.** Their fields represent the top-level PII data and the generic level PII data elements, as illustrated in **Figure 6.3**.

Phase 2: Execution Level Activities: Translation Context

To execute the translation context, UML modelling language is used to design the super-class **<clsTranslate>** and its underlying sub-classes <**testforGenericData()>** and <**testforTopLevelData>** respectively. **Figure 6.4** illustrates the relationship between the super-class and the base class.

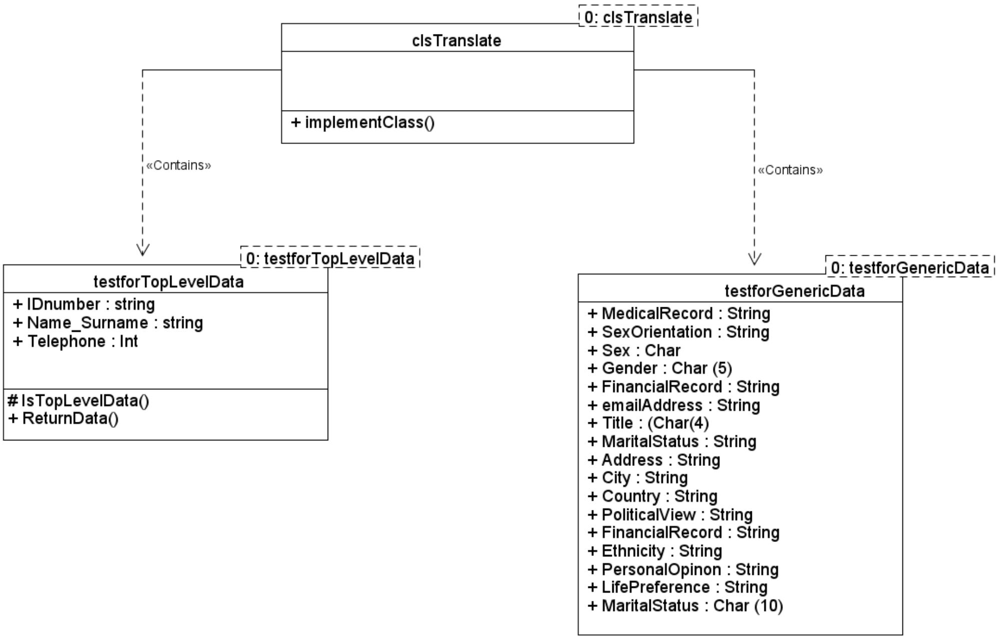


Figure 6.4. Translation Class Structure

Based on **Figure 6.4**, both classes have properties that are linked to the data elements and methods that are used to test the data and to return the data to the system for onward processing. Tables are a useful mechanism for arranging data. Typically, they group elements of the same or similar kind, such as the data and metadata of both the top level and generic level data elements which, based on **Figure 6.4** were kept in database tables to ease verification or validation as part of the compliance process.

6.2.3 Design of the Enforcement Context

Phase 1: Design Level Activities: Enforcement Context

The enforcement context deals with the conditions for the lawful processing of personally identifiable information. To design this context, reference is made to the POPIA compliance domain discussed **Section 4.2.1.** Here the eight principles which are accepted and agreed upon in South Africa as the standard for implementing POPIA are listed and described, namely accountability, processing limitation, purpose specification, further processing limitation, information quality, openness, security safeguards, and data subject participation. In terms of design, these conditions are expressed and encoded in a business rules engine as the boundaries for the handling of personally identifiable information. Hence, adapting this to the Rauch (2012) model, the main component or module from the design of this context is the business rules engine.

Phase 2: Planning Level Activities: Enforcement Context

To achieve the design of the enforcement context, the following planning activities are undertaken. First, two types of business rules engines and business rules execution methods are examined. These two types of business rules execution methods are mainly distinguished by the way they implement the business rules that are scheduled for execution. The following are the types of business rules execution methods:

Forward chaining: Most business rules engines are based on the forward-chaining principles (Educba.com, 2019), which relies on an inference rule engine implementing one of the following algorithms (Linear, Rete, Treat, Leaps, etc.). In turn, it implements the IF … THEN logic or an event condition rule engine (Educba.com, 2019). **Figure 6.5** shows the architecture of a generic inference engine. The main characteristic is a knowledge base populated with learning from previous rule inferences and the inference engine itself, which stores a database of applicable rules. In this model, the knowledge base and the inference engine are separated from one another to facilitate maintenance. After all, in most cases, knowledge and policies will change over time, and one does not want to rewrite the inference engine (the program code) whenever a new rule is added.

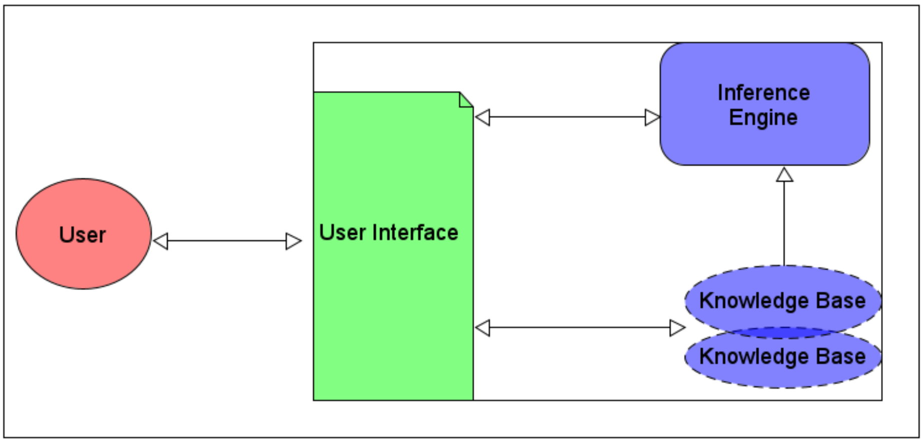


Figure 6.5. Architecture of Rules Inference Engine

(Source: JBOSS.Org)

Forward chaining is data driven and thus reactionary. It starts with a fact and ends with a conclusion. The architecture of a forward-chaining rules engine is depicted in **Figure 6.6.**

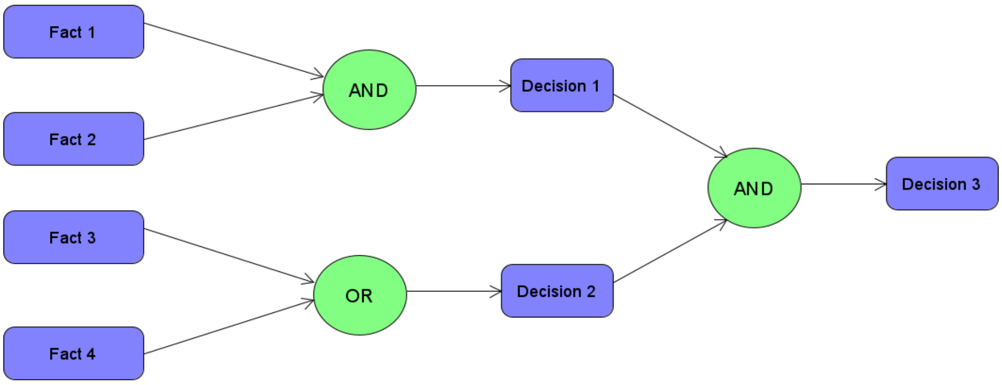
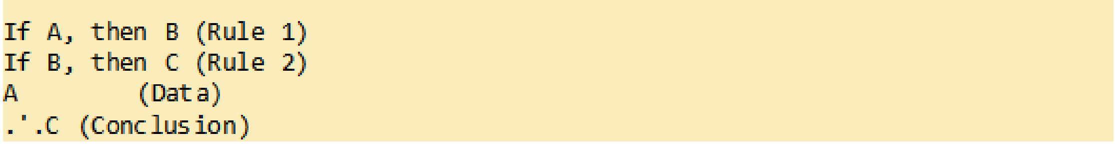


Figure 6.6. Architecture of Forward-Chaining Rules Engine

(Source: Adapted from Educba.com, 2019)

For example, a set of rules may be chained together, as in this example:



```
If A, then B (Rule 1)
If B, then C (Rule 2)
A         (Data)
.'.C  (Conclusion)
```

Figure 6.7. Class Model: Excerpt of Code

First, this example shows a standard rule represented by an IF condition, THEN consequent statement, where condition A fires the rule, and consequent B represents the conclusion. According to Graham (2006), such a rule can be interpreted in many ways, such as:

1. If some condition is fulfilled, then some action is performed;
2. If a condition statement is true, then another can be inferred.

This example, **Figure 6.7** shows data-directed inference because the data that is known (in this case, A) drives the inferences from left to right in rules, with rules chaining together to deduce a conclusion.

Backward chaining: This type of business rules execution method operates on a post facto basis and looks back to resolve a fact to fit a particular outcome (Educba.com, 2019). Backward chaining works primarily on an inference method of reasoning, for example, based on an end goal that one can backtrack and infer the steps which led to the end goal. For instance, if a pensioner has a policy to earn R1 million when they retire, the insurance company can infer, through a savings calculator, how much the pensioner needs to pay monthly, at the start of their pension policy.

Phase 2: Execution Level Activities: Enforcement Context

In light of the two business rules execution methods highlighted above, for execution, this study selects the forward-chaining business rules execution method (which implements the Rete algorithm) as the formal method to use in the design of the business rule engine to power the enforcement of POPIA in the framework. The choice of the forward-chaining rule is based on the premise that it is data driven and relies on facts to make conclusions. Hence, in the proposed framework, the enforcement context is represented by the business rules engine. The rationale of the business rules approach is captured by Moriarty (1993) when they suggest that systems analysts are still striving for a paradigm that can bridge the communication gap between businesspeople and information systems professionals. As a result, it is this gap that the business rules approach seeks to bridge. However, most significantly, the benefits of the business rules approach can be narrowed down to two points;

1) The business rules approach enables systems to be designed in such a way that they can accommodate changes of business rules easily with minimal disruption of the operation of the system. This is particularly important for modern and dynamic business organisations, where the time to market is very limited and change is constant, to cope with dynamic market conditions and technology trends.

2) The business rules approach introduces tracking, which is vital for large systems such as enterprise resource planning systems that host thousands of business rules and also requires that these rules be modified frequently to align with legislations and other business imperatives and directives. For example,

now with the Covid-19 pandemic, businesses were required to suspend taxes and rate changes on certain products and commodities, such as the deferment of payment of exercise taxes on alcoholic beverages and tobacco products. (SARS, 2020). This measure was a direct response to the restrictions placed on the sale of alcoholic beverages and tobacco products to ease the emergency section of the South African hospitals.

According to Theodoulidis and Youdeowei (2000), when talking of using a business rules approach in an information system, emphasis should be placed on the analytic methods and the system architecture that is relevant to support the business rules environment. In the context of this study at execution level, the standard notation of the UML modelling language is used formally to visualise, specify, construct, and document the business rules within the business rules engine. Hence, in the execution of this context, the rule-based system has the following design components:

1. A set of rules, which can modify the existing database and is applied as a condition on the current database.
2. A database of information.
3. A rule interpreter, which determines applicability and selection of the rules and possible conflict resolution between them, which constitutes a knowledge base.

Figure 6.8 illustrates the rules’ control structure and its components. The result is a rules-based knowledge base that is used to enforce the protection of privacy in this study.

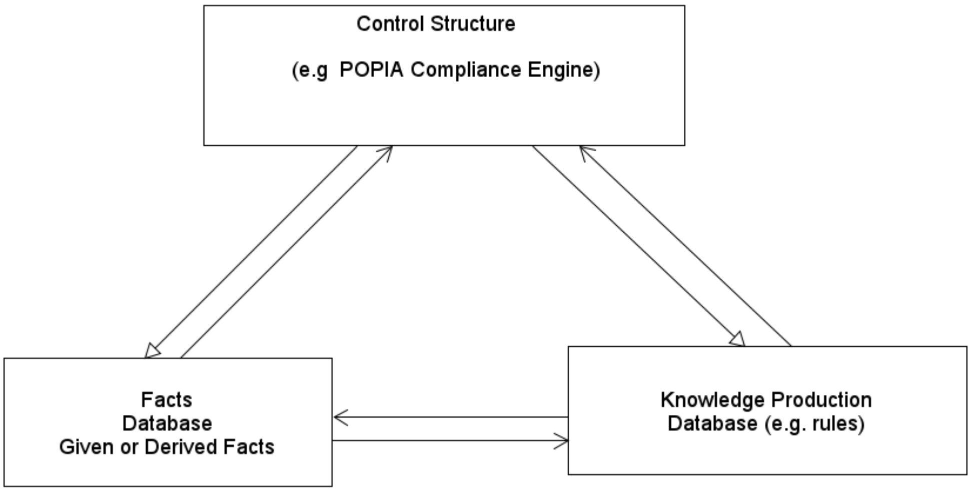


Figure 6.8. Enforcement Context Control Components

(Source: Adapted from Schalkoff, 1990)

Based on the Rauch (2012) model, UML language is used to show an abstraction of the inner processes, resources and organisations of the rules system, as indicated in **Figure 6.9.**

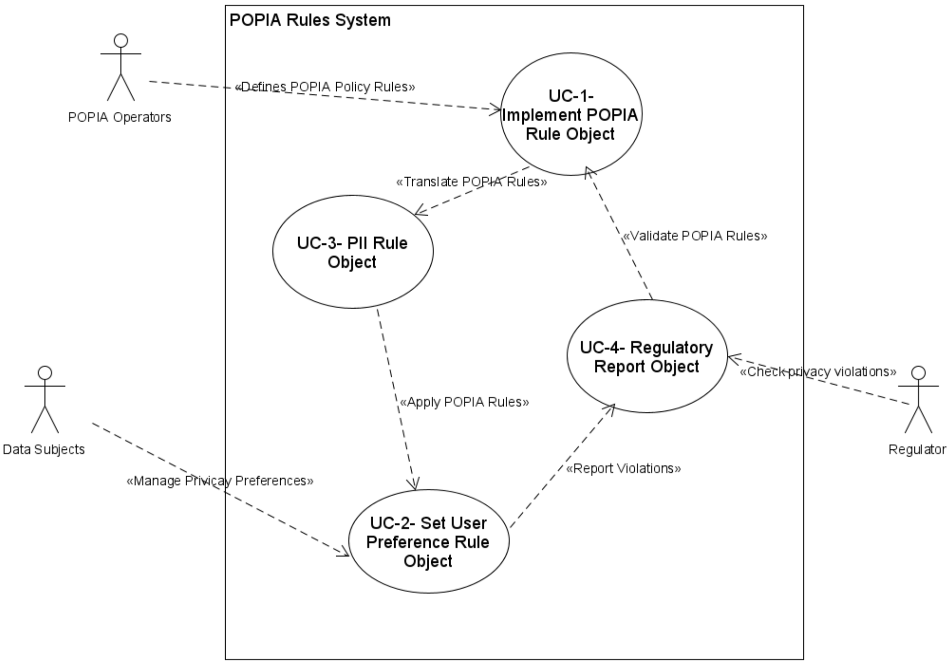

POPIA Rules System
POPIA Operators
«Defines POPIA Policy Rules»
UC-1- Implement POPIA Rule Object
«Translate POPIA Rules»
UC-3- PII Rule Object
«Validate POPIA Rules»
UC-4- Regulatory Report Object
«Check privacy violations»
Regulator
Data Subjects
«Apply POPIA Rules»
«Report Violations»
«Manage Privicay Preferences»
UC-2- Set User Preference Rule Object

Figure 6.9. Enforcement Context Control Components

(Source: Adapted from Schalkoff, 1990)

Based on **Figure 6.11**, three actors, namely the POPIA operator, the data subject and the regulator, all part of the organisation, are depicted interacting with different set of rules on the enforcement context. These rules themselves can be further broken down into the following rules components, based on the UML standard notation and definition of rules, for instance, the rule Manage Privacy Preference, from **Figure 6.11** is broken down into the following rules components, as per **Table 6.1**, using UML standard notations.

Table 6.1 Representing Rules using UML Standard Components

Rule component	**UML element**	**Example**
Event	message	verifyUserPreference (x:DataSubject)
Condition	guard	message.result=true
Action	message	POPIAPrototype^verifyUserPreference(x:DataSubject)

PostCondition	guard	self.user.oclInState(Verified)
		self.DataSubject=x

ElseCondition	guard	false

Next, the design of the technology context is explained.

6.2.4 Design of the Technology Context

Phase 1: Design Level Activities: Technology Context

The technology context serves as the main input to the information system context that is discussed in the next section. In this section, the choice of the design pattern / architectural synthesis and software development methodology is considered. As elaborated on in **Section 1.10**, an artefact is built as part of this study. The software development methodology and approach used to build the artefact is Agile software development methodology with rapid prototyping as the development framework. The rationale for adopting the Agile software development methodology as opposed to traditional software development methodologies such as the ‘waterfall model’, is based on the conclusions drawn by Edeki (2015), that Agile software development methods simplify the software planning and estimation process by breaking down large requirements into small, individual tasks. Small tasks enable both software engineers and business owners to better manage and control the software development efforts and results, thereby bringing about efficiency into the software development process. Hence, based on Rauch's (2012) model, the main components of this context are the software development methodology, the architectural design pattern and the relational database.

Phase 2: Planning Level Activities: Technology Context

In terms of planning, the Agile process brings about close integration, communication, and collaboration between the software development teams resulting in a more acceptable end product. Finally, and most importantly, the incremental nature of the Agile development method keeps the software development process on track even if the requirements / technologies might change in the course of developing the software system. (Edeki, 2015). Similarly, the rationale for using rapid prototyping is discussed in **Section 3.6.1**; however, to recap here, rapid prototyping is a system of engineering tools used to generate working models and design of systems quickly to test their functionality and fitness for purpose. This narrative fits the design science paradigm of this study, which requires the artefact to be built and evaluated in context through an iterative process. As illustrated in **Figure 3.4**, this rapid prototyping process allows for several cycles of testing and refining of the prototype until it is deemed fit for purpose and eventually implemented in a live environment for production.

Phase 2: Execution Level Activities: Technology Context

In terms of the execution of the design of the technology context, the building blocks identified are connected to create the technology context and input into the information system and validation context. Below is a list of the building blocks identified and a description of their role in the technology context of the framework. **Figure 6.10** illustrates the relationship and connection between these building blocks.

1. **Building Block 1: Software development methodology:** This block covers the software development methodology used and, in the case of this study, the rapid prototyping is used because of its agility and iterative nature and also

because the main research output is a software artefact taking the form of a prototype.

2. **Building Block 2: Design pattern and architectural synthesis:** This block deals with the programming paradigm used. In this case, the programming design pattern and architectural synthesis is the object-oriented paradigm. This programming paradigm is used because it provides for the creation of objects that can encapsulate both data and function in a loosely coupled arrangement.
3. **Building Block 3. Relational database system:** This blocks deals with the repository and knowledge base use in this study and involves the use of entity-relationship mapping and tables.

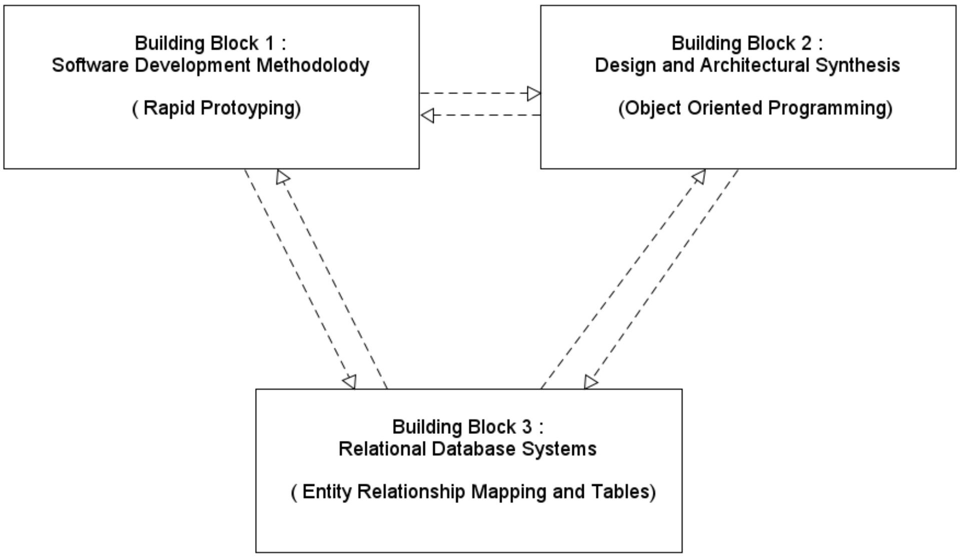

Building Block 1 :
Software Development Methodolody
(Rapid Protoyping)
Building Block 2 :
Design and Architectural Synthesis
(Object Oriented Programming)
Building Block 3 :
Relational Database Systems
(Entity Relationship Mapping and Tables)

Figure 6.10. Building Blocks of the Technology Context

All these technology building blocks (1, 2, and 3) serve as input into the information system and validation context of this study.

6.2.5 Design of the Information System and Validation Context

Phase 1: Design Level Activities: Information System and Validation Context

According to Baskerville et al. (2020), classical information systems were designed to be digital representations of the real world. In such representations, rules that are applicable to the real world are digitised and enforced in the technology system. This view is supported in previous research by Dourish (2001), in which the technology systems were shown to be a reflection of reality and a purposeful digital representation of the real world.

However, today technology systems are part of a digital world (Hui, 2016). These digital objects are not just a mirror of physical world objects but take a more prominent role in organising the physical world. For instance, we have chat bots operating call centres alongside human call centre agents. Similarly, we have drones delivering parcels to homes alongside human drivers from courier agencies. In essence, as observed by Yoo et al. (2012), these objects have an enormous capacity to sense, interact with, and record their physical surroundings actively, based on a set of rules operated by a rule engine. Adapting this to Rauch's (2012) model, the main components and modules in the information system and validation context are software-based objects.

Phase 2: Planning Level Activities: Information System and Validation Context

In terms of planning, the information system context is represented by software objects built to enforce the business rules elaborated on in the validation context of this framework. Generally, software objects are by-products of the object-oriented programming concept. According to Stephens and Sumner (1996), object-oriented programming is preferred over the other traditional programming methodologies, because it abstracts programming logic and data into objects that can be instantiated and reused multiple times within a software system or reference by other software systems. In fact, once all the individual software objects have been developed and implemented, they become connected in a coherent fashion to form a modular software system. According to Oracle Corporation (2020), an object, in the context of software engineering, is a software bundle of related state and behaviour. In its simplest form, software objects are used to model real-life objects that we find in everyday life. Object-oriented programming is loaded with many terminologies and concepts, such as encapsulation, polymorphism, and inheritance, just to mention a few.

Phase 2: Execution Level Activities: Information System and Validation Context

This study does not delve into the detail of theories or concepts guiding object-oriented programming but looks at software objects as vehicles to house programming instructions, such as business rules used in the validation context of this study. To design the information system and validation context, the building blocks identified in the technology context, notably software development methodology (rapid prototyping), design and architectural synthesis (object-oriented programming), and relational database systems in the form of entity-relationship mapping and tables are all combined to build these contexts. The entity-relationship mapping and table structure used in the design of this framework is shown in **Figure 6.11.**

The objects developed for this context should hold business rules encoded as software instructions and should communicate with data sitting in database tables. Therefore, the design of the information system and validation context entails analysing, specifying, and coding the business rules into the software-based objects of the information system.

In the context of this study, the information system and validation context is seen as the tool or artefact used to handle personally identifiable information within an organisation; consequently, helping the organisation to conform to information privacy laws through the use of configurable software objects.

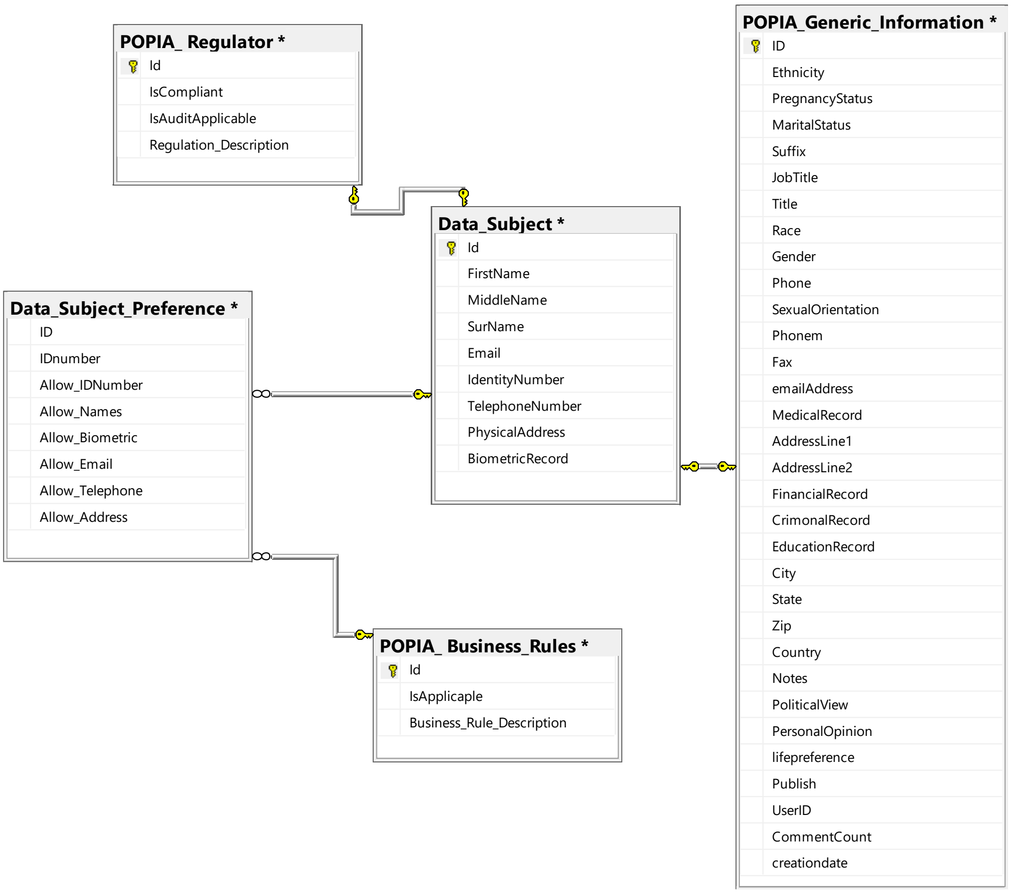


Figure 6.11. Entity-Relationship Diagram- Data Tables

In summary, in terms of the design of the information system and validation context, this study makes use of object-oriented programming principles and entity-relationship mappings and tables to design the objects and data repositories that represent the information system context, and finally, business rules to design and specify the validation context. **Figure 6.12** illustrates the different components of the information system and validation context.

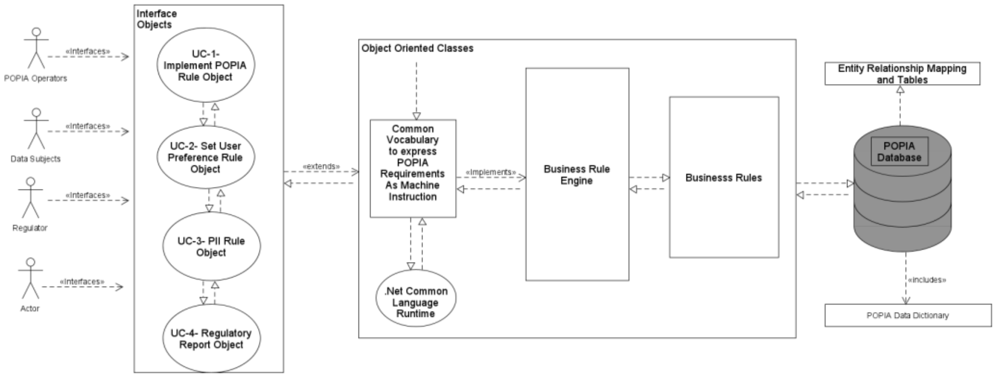


Figure 6.12. Architecture of the Information System and Validation Context

6.3 The Framework

To further illustrate the relationship between the conceptual framework contexts and the proposed framework, **Figure 6.13** shows the detailed connections and links between these conceptual framework contexts and the framework design elements. Based on the understanding of the detailed relationships and links established between the conceptual framework contexts and the frame design elements, **Figure 6.16** depicts a diagram of the final framework. This framework is used in the next chapter (**Chapter 7**) to implement the prototype for this study.

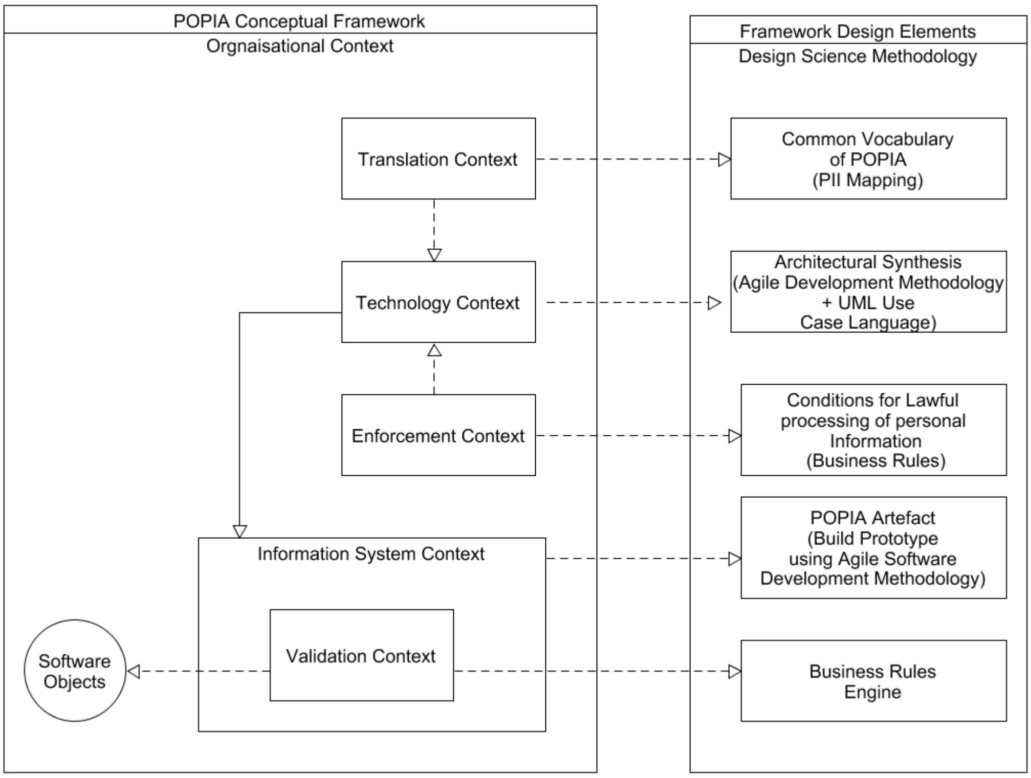

POPIA Conceptual Framework
Orgnaisational Context
Translation Context
Technology Context
Enforcement Context
Information System Context
Validation Context
Software Objects
Framework Design Elements
Design Science Methodology
Common Vocabulary of POPIA (PII Mapping)
Architectural Synthesis (Agile Development Methodology + UML Use Case Language)
Conditions for Lawful processing of personal Information (Business Rules)
POPIA Artefact (Build Prototype using Agile Software Development Methodology)
Business Rules Engine

Figure 6.13. Conceptual Framework Contexts vs Framework Design Elements

Figure 6.14 is a diagram of the final framework.

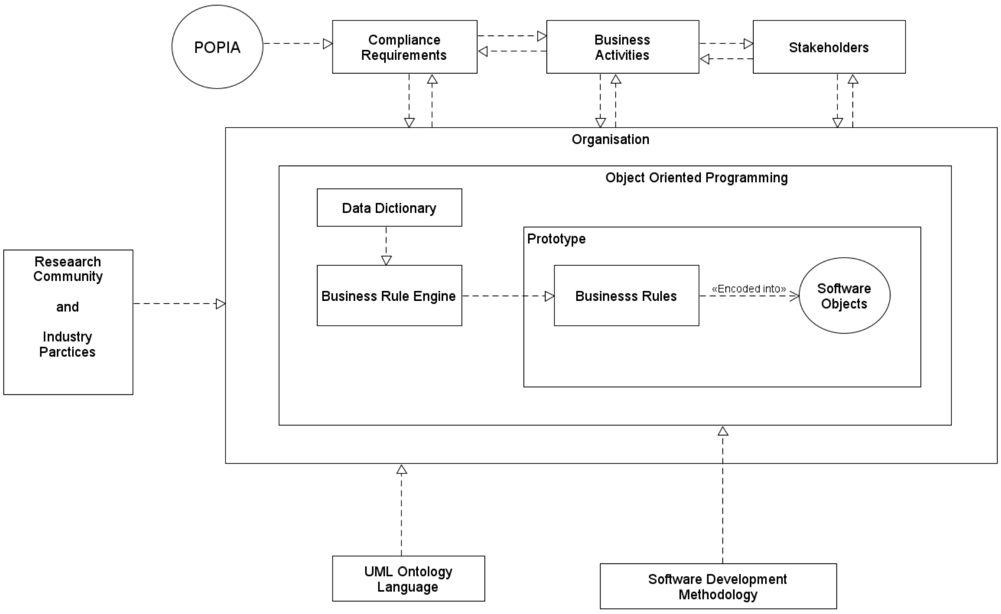


Figure 6.14. The Framework

Based on **Figure 6.14**, the framework is made up of the following building blocks: object-oriented programming, software development methodology, UML ontology language, data dictionary, business rules engine, business rules, software objects, POPIA compliance requirements, and the organisation made up of business activities and stakeholders.

6.4 Conclusion

In this chapter, the framework for the design of the POPIA prototype was developed. In designing this framework, the different contexts identified in the conceptual framework in Chapter 4, namely the organisational context, the translation context, the enforcement context, the information systems, and the validation context were all utilised.

The framework is illustrated in **Figure 6.14.** In this framework, the following key building blocks are highlighted, namely object-oriented programming, software development methodology, UML ontology language, data dictionary, business rules engine, business rules, software objects, POPIA compliance requirements, and the organisation made up of business activities and stakeholders. Finally, in conclusion, the objective of designing the framework for the design of the POPIA prototype was achieved. Hence, the next chapter focuses on using the building blocks of this framework to build, implement, and test the POPIA prototype.

CHAPTER 7

Implementation of the Prototype

7.1 Introduction

This chapter deals with the implementation of the framework developed in Chapter 6 to build and test the POPIA prototype in line with the research objectives of this study.

The implementation constitutes the last stage of the research design framework developed in Chapter 4 with **Stages 1 and 2** implemented in Chapters 5 and 6, respectively. **Figure 7.1** highlights the components of the implementation phase from the research design framework developed for this study.

In this chapter, the prototype used to test the assertions of this study was built using the components of the framework developed in Chapter 6. This prototype was built and scoped using four (4) use cases identified from the research objectives and research problem guiding this study. The four use cases are: implement POPIA rule engine object UC-1; Set User Preference Rules object UC-2; personally identifiable information rules object (PII Rule) UC-3; and regulatory report object UC-4. The main components of the prototype are: user interface, business rules, and business rules engine, data dictionary, entity relation model, database, and software objects.

Finally, the software objects hosting the business rules are built into a Microsoft.Net prototype with a SQL Server database serving as a data dictionary and implemented as a software system or POPIA prototype in this chapter.

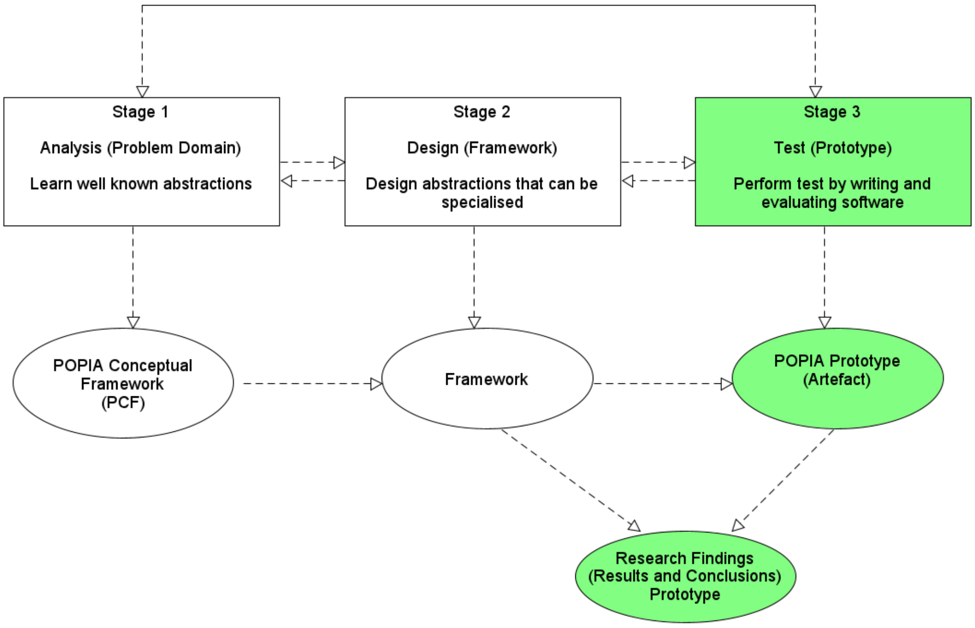

Stage 1
Analysis (Problem Domain)
Learn well known abstractions
Stage 2
Design (Framework)
Design abstractions that can be specialised
Stage 3
Test (Prototype)
Perform test by writing and evaluating software
POPIA Conceptual Framework (PCF)
Framework
POPIA Prototype (Artefact)
Research Findings (Results and Conclusions) Prototype

Figure 7.1. Research Design Framework: Stage 3

7.2 Implementation of the Prototype

To implement the prototype, The UML ontology language is used first to model the actors of the organisations, the business processes represented by the use cases and the activities of these actors within this organisation. The details of the implementation of the use cases is specified in the tables in **Appendix B** and dedicated for the specific use cases. They consist of the description of the use case, the primary actors, preconditions and post conditions, the main success scenarios, assumptions, notes and issues, the status, the owner of the use case and finally, the priority of the use case.

The UML ontology language prescribes standard sets of diagrams and notations to model object-oriented systems, with the notations describing the underlying semantics of these diagrams. According to Popkin (1998), there are nine (9) types of diagrams commonly used for the description of object-oriented systems as per the UML ontology language, as listed in **Table 7.1**.

Table 7.1 UML Diagrams and Purpose

Diagram type	**Purpose**
Use case diagram	For modelling the business processes
Sequence diagram	For modelling message passing between objects
Collaboration diagram	For modelling object interactions
State diagram	For modelling the behaviour of objects in the system
Activity diagram	For modelling the behaviour of use cases, objects, or operations
Class diagram	For modelling the static structure of classes in the system
Object diagram	For modelling the static structure of objects in the system
Component diagram	For modelling components
Deployment diagram	For modelling distribution of the system

These diagrams are used depending on the view that is sought from the system. In the context of this implementation and in the following sections, some of these diagrams and their underlying notations are used to model the POPIA prototype.

7.2.1 Identification of Key Actors and Use Cases (Business Activities)

To build and implement the prototype, it is important to understand the organisational structure. Hence, in this section, the UML ontology language is used to model the organisation responsible to handle the personal information of its data subjects. The rationale for modelling the organisation is to achieve the following:

1. To identify and define the key actors of the organisation.
2. To explain the roles of the key actors and their actions in the POPIA prototype.
3. To demonstrate the different scenarios and interaction points between the key actors and the POPIA prototype.
4. To showcase the major software objects (components) envisaged for the POPIA prototype.

First, in terms of the key actors, according to Campbell (2007), UML defines actors as anything that interacts with a system, in this case the POPIA prototype. These actors could be natural persons, roles played by different people, or another system (Campbell, 2007). Based on the scope of data manipulation and the PII data elements highlighted above, the following five (5) major actors of the proposed the POPIA rule engine prototype were identified. They are the data subject, the POPI Act represented in the rule engine by the personally identifiable information (data) rules object, the organisation, and the regulator. The organisation generally refers to the entity holding the personal information of the data subject and who, by virtue of POPIA, is responsible to safeguard the data under its custody. **Table 7.2** lists the main actors in the proposed POPIA prototype and provides a brief definition of their envisaged roles in the rule engine.

Second, based on the key actors and the description of their roles as well as tapping into the POPIA compliance domain discussed in **Section 4.2.1** and the information system context of the conceptual framework and its sub-contexts, the following four (4) use cases are modelled for the POPIA prototype.

1. **Use Case – Set User Preference Rules**: This use case draws from the research of Bélanger and Crossler (2011), who conclude that individual privacy can be presented by two instruments:
   a) CFIP, or concern for information privacy, which has four (4) dimensions, those being data collection, unauthorised secondary use, access control, and data quality, in addition to 15 privacy items.
   b) IUIPC, or Internet user's information privacy concerns, which has three (3) dimensions, those being awareness, control, and collection, in addition to 10 privacy items.
2. **Use Case – Personally Identifiable Information (Data) Rules :** This use case defines and sets the evaluation criteria for the POPIA rules in terms of when information is collected, processed, shared, and stored by organisations.
3. **Use Case – Set User Preference Rules Objects:** This use case verifies

that all the validation rules in the privacy rules object and the privacy concern object are adhered to in the organisation using the POPIA prototype within the validation and enforcement contexts of the POPIA conceptual framework.

4. **Use Case – Implement POPIA Rule Engine:** This use case links the privacy object with the reusable prototype protection object to determine compliance with POPIA regulations.

Table 7.2 briefly describes the key actors in the prototype for POPIA compliance and gives a brief definition of their envisaged roles in the system.

Table 7.2: Key Actors and Definition of Roles

Key actors	**Definition of role**
1. Data subject	The individual whose personal information has been collected and is in the custody of the organisation
2. POPIA	The law regulating the protection of personal information in South Africa
3. Organisation	The public or private entity responsible for safeguarding the personal data of its employees or customers
4. Regulator	The body responsible for enforcing compliance with the POPIA regulations
5. Information system	The computerised system responsible for processing and storing personal identifiable information

With a good understanding of the key actors of the organisation and their role in the POPIA prototype, **Table 6.3** describes the main use case scenarios and also describes the proposed software objects to represent the use case scenarios.

Table 7.3: Main Use Cases and Software Objects

Key actors	Use cases	Software objects
1. Data subject	Set User Preference Rules	Set User Preference Rules object
2. POPI Act	Personally identifiable information (data) rules	Personally identifiable information (data) rules object
3. Organisation	Set User Preference Rules	Set User Preference Rules object
4. Regulator	Regulatory report	Regulatory report object
5. Information system	Implement POPIA rule engine	Implement POPIA rule engine object

Based on the use cases identified and highlighted coupled with the associated software objects, **Figure 7.2** is a use case diagram, which shows the interaction between the different actors and the software objects in the POPIA prototype.

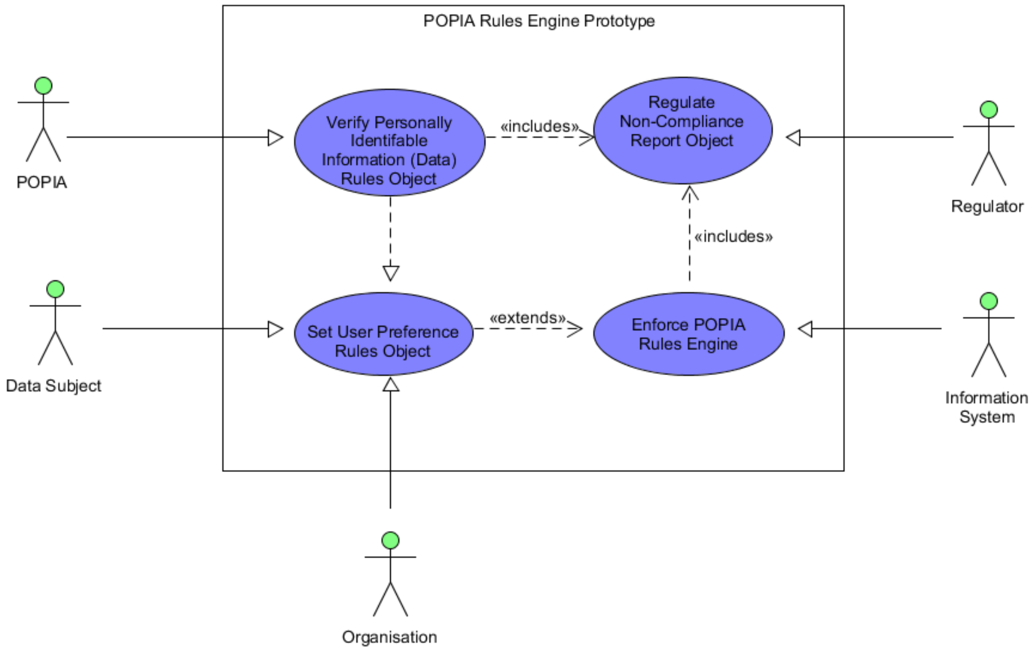

POPIA Rules Engine Prototype
POPIA
Verify Personally Identifiable Information (Data) Rules Object
«includes»
Regulate Non-Compliance Report Object
Regulator
«includes»
Data Subject
Set User Preference Rules Object
«extends»
Enforce POPIA Rules Engine
Information System
Organisation

Figure 7.2. Key Actors versus Software Objects

To further elaborate on the roles of the different actors in the POPIA prototype, **Table 7.4** shows the key actors, the software objects of the prototype with which they interact, and some of the key actions associated with the actors. In the context of this study it is important to note that these actions correspond to the business activities associated with the organisation, as illustrated in **Figure 5.10** in the previous chapter.

Table 7.4: Key Actors and Key Actions

Key actors	Rule engine objects	Key actions
1. Data subject	Set User Preference Rules objects	- Capture privacy concerns - Modify privacy concerns - Determine data to reveal - Determine data to conceal
2. POPI Act	Personally identifiable information (data) rules object	- Outline privacy rules - Set allowable action on data - Define data violation - Define remediation action
3. Organisation	Set User Preference Rules objects	- Verify privacy concerns - Seek data subject permission on data items - Classify data based on sensitivity - Notify data subject and regulator of data violation
4. Regulator	Regulatory report object	- Audit system policies - Identify non-compliance - Notify data subject of valuations - Determine privacy violation - Apportion fines for non-compliance

5. Information system	Implement POPIA rule engine object	- Constraint business processes - Validate data / information request - Trigger violation events - Send notifications and warnings

According to Vidgen (2003), the use case diagram is a formalised notation to showcase a system from the perspective of the actors (users). The focus of a use case diagram is to model what the system does – its behaviour rather than how it achieves it. For instance, a typical use case on a system will be 'create a new user' with a definite outcome of creating a new user on the system and with a definite business value of having one more user on the system. In the context of this study, to further elaborate on the actions of each of these actors, the following four (4) use cases are developed to examine the inner working of the POPIA prototype in more detail. **Table 7.5** shows these use cases and gives a description of their roles in the POPIA prototype. The full details of these use cases are presented in **Appendix B,** where the use cases, actors, software objects and conditions for implementation are laid out.

Table 7.5: POPIA Use Cases and Description

Use case name	Description of use case
1. Implement POPIA rule engine object	This use case examines how the POPIA rules are implemented within the information system used by both the organisation and the data subject to collect, share, process, and store personally identifiable information. Based on the theoretical framework and the POPIA rule engine prototype developed, business rules can be broken down into three rule parts used to implement the rules. Refer to Appendix 1 for more details.

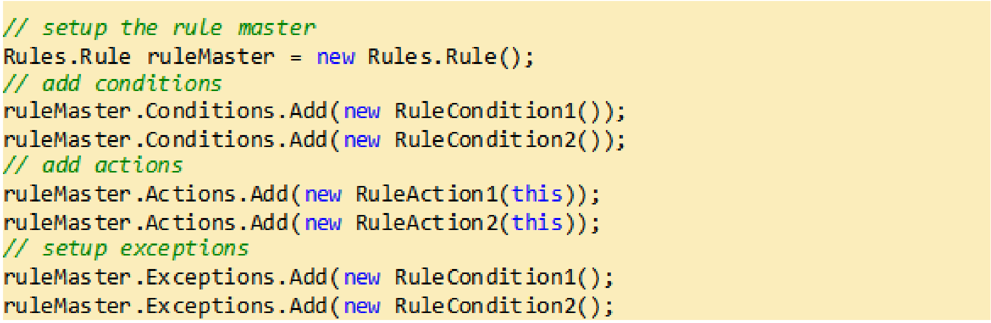
Use case name	Description of use case
2. Set User Preference Rules object	This use case allows the data owners and the organisation to set their preference on the POPIA rule engine to determine how PII data /information is collected, processed, shared, and stored by the organisation hosting and operating the information system. This use case leans heavily on Altman's (1975) privacy regulation theory highlighting the need for data subjects to manage private information and disclosure, based on their level of comfort, all within a privacy regulating mechanism.
3. Personal identifiable information rules object (PII Rule)	This use case is underpinned by POPIA, highlighting the different data/information elements constituting personally identifiable information (PII). According to POPIA, the data subjects have constitutional prerogatives to protect their personally identifiable information from exploitation by third parties and other commercial entities. This task requires organisations to classify data based on the conditions for the lawful processing of personal information, as stipulated in Chapter 3 of the POPI Act. As a result, every organisation that is a custodian of PII is bound to set allowable actions on PII data, report violations on PII data and, most importantly, to define remediation actions.
4. Regulatory report object	This use case links the privacy concern object with the system protection object to determine compliance with POPIA regulations.

In view of the identification of the key actors, their roles in the POPIA prototypes, and the main use cases within the POPIA prototype, the next sections focuses on the implementation of the system components of the POPIA prototype, starting with the business rules engine.

7.2.2 Implementation of Business Rules Engine

At the centre of the POPIA prototype are rules built around the proper handling of personally identifiable information. In **Section 6.2.3,** the forward-chaining business rules execution method, implementing the Rete algorithm, was select as the preferred business rules engine for this study. In **Figures 6.7, 6.8, and 6.9**, the architecture of the forward-chaining business rules engine and software code class models were illustrated.

To implement the rules in code, a <<Rules>> class is created, and the class is broken down into three (3) software objects or component classes, namely (Rule.Conditions, Rule.Actions and Rule.Exceptions). To consume the <<Rules>> class, an instance of it is created. **Figure 7.3** shows a code excerpt demonstrating how to instantiate a <<Rules>> class and create the corresponding Rules Conditions, Actions and Exceptions, collectively called a RulePart for a specific Rule.



```
// setup the rule master
Rules.Rule ruleMaster = new Rules.Rule();
// add conditions
ruleMaster.Conditions.Add(new RuleCondition1());
ruleMaster.Conditions.Add(new RuleCondition2());
// add actions
ruleMaster.Actions.Add(new RuleAction1(this));
ruleMaster.Actions.Add(new RuleAction2(this));
// setup exceptions
ruleMaster.Exceptions.Add(new RuleCondition1();
ruleMaster.Exceptions.Add(new RuleCondition2();
```

Figure 7.3. Class Model: Excerpt of Code

Figure 7.4 shows how a POPIA rule is broken down into these software objects or components, as illustrated in the code extract in **Figure 7.3**.

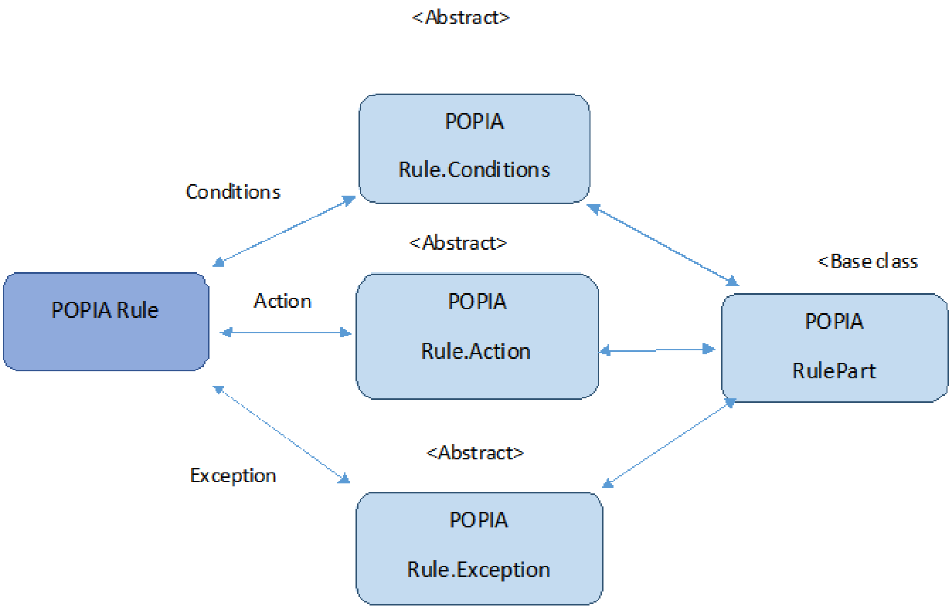


Figure 7.4. Structure of POPIA Rules Engine POPIA Conditions

Adapting this model to one of our use cases, the use case: Set User Preference Rules object, produces **Figure 7.5.**

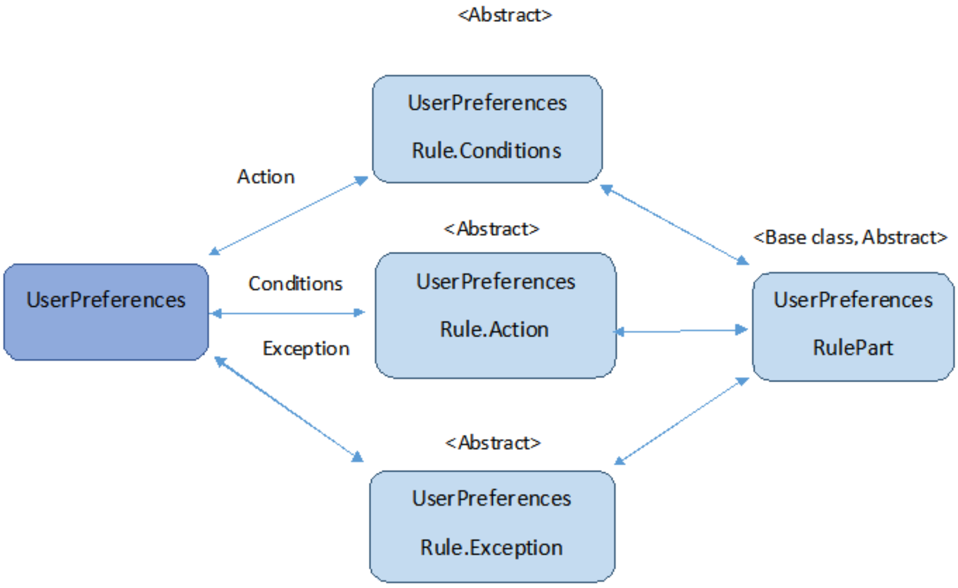
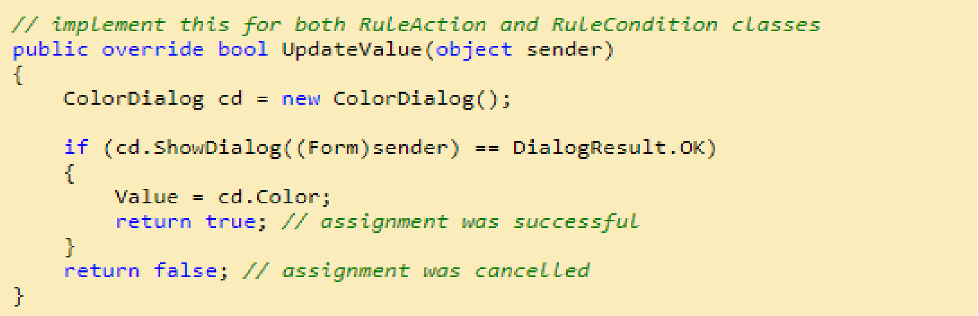


Figure 7.5. Structure of POPIA Rules Engine User Preferences

Each Rule.Condition and Rule.Action has a Value attribute. This Value attribute is set by the user at run time and it specifies when a condition is met, or how the action is to be executed. When the user clicks on the value at run time the UpdateValue() method is triggered in the subclass. This method requires supplying the user with a dialog to set the Value for the respective action or condition. This could be a text dialog, a colour picker, a file dialog, or any other type of input dialog. Note also that it is possible to set only one value per action or condition, as per **Figure 7.6.**

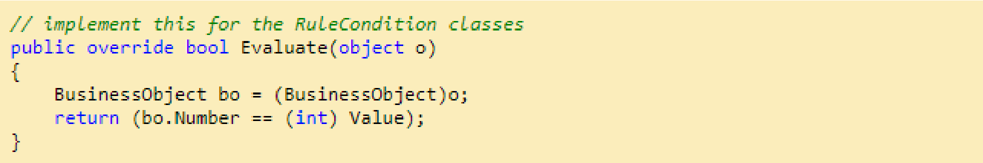


```
// implement this for both RuleAction and RuleCondition classes
public override bool UpdateValue(object sender)
{
    ColorDialog cd = new ColorDialog();

    if (cd.ShowDialog((Form)sender) == DialogResult.OK)
    {
        Value = cd.Color;
        return true; // assignment was successful
    }
    return false; // assignment was cancelled
}
```

Figure 7.6. Value per action or condition

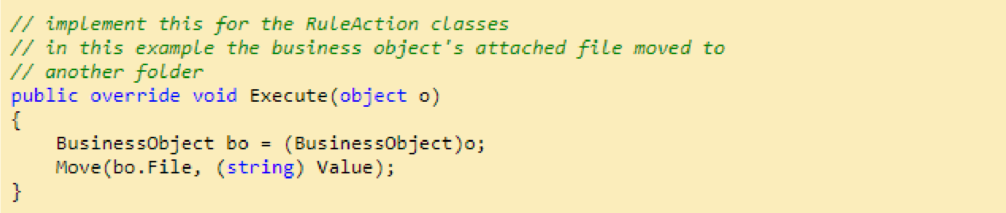
When a rule is executed, the conditions are first evaluated. Therefore, you must implement the Rule.Condition.Evaluate() method. The Evaluate() method gets an object as parameter. This will be your own defined business object. So, you need to cast it first, as per **Figure 7.7**.



```
// implement this for the RuleCondition classes
public override bool Evaluate(object o)
{
    BusinessObject bo = (BusinessObject)o;
    return (bo.Number == (int) Value);
}
```

Figure 7.7. Casting of Value

When all conditions are met and there are no exceptions, the actions are executed one by one. Therefore, the Rule.Action.Execute() method needs to be implemented, as in the example, illustrated in **Figure 7.8**



```
// implement this for the RuleAction classes
// in this example the business object's attached file moved to
// another folder
public override void Execute(object o)
{
    BusinessObject bo = (BusinessObject)o;
    Move(bo.File, (string) Value);
}
```

Figure 7.8. Rule Execute Method

7.2.3 Implementation of Prototype Objects

To implement the POPIA prototype components, the object-oriented programming paradigm and the Agile software development methodology was used in conjunction with the UML ontology language to model and design the different components and modules of the prototype. The reasons for this choice are based on conclusions drawn by Edeki (2015), who states:

> Agile software development methods simplify the software planning and estimation process by breaking down large requirements into small individual tasks. Small tasks enable both software engineers and business owners to better manage and control the software development efforts and results. Thereby bringing about efficiency into the software development process. The Agile process brings about close integration, communication and collaboration between the software development teams resulting into a more acceptable end product. Finally, and most importantly, the incremental nature of the Agile development method, keeps the software development process on track even if the requirements / technologies might change in the course of developing the software system. (Edeki, 2005, p.).

Mindful of the four benefits listed above and taking into cognisance the use of UML as the ontology language for the design of the POPIA prototype, the following UML class diagram Figure 7.9 is developed. According to Ambler (2011), a UML class diagram is a graphical notation used to construct and visualise object-oriented systems. A UML class diagram is essentially made up of the following:

- A set of classes;

- A set of relationships between the classes.

With reference to the UML notation, a class consists of three (3) parts, namely class name, class attributes, class operations (methods), as illustrated in **Figure 7.9**

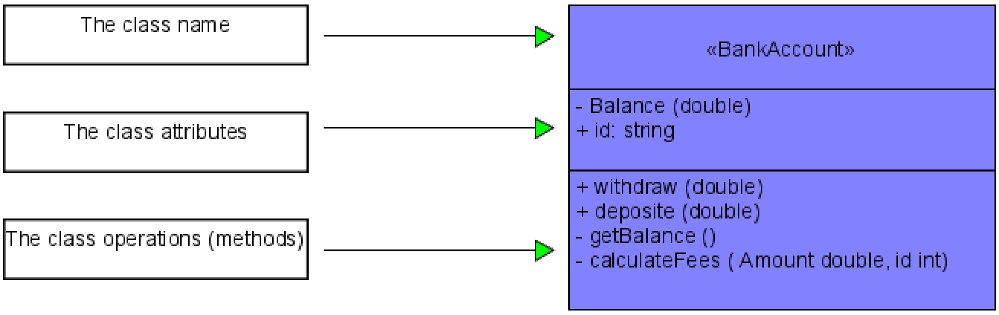


Figure 7.9. Graphical Representation of a UML Class

- **Class name:** The name of the class appears in the first partition.
- **Class attribute:** The attributes are shown in the second partition. The attribute has a data type and visibility possibilities. UML class diagram allows using four different visibility levels, like - for private, + for public, # for protected, ~ for packaged.
- **Class operations (methods):** The class operations are shown in the third partition. Essentially, they are services that the class provides. The return type of the method is shown after the colon at the end of the method signature. The method parameters also have a return type shown after the colon following the parameter name.

Applying the class diagram principles to the POPIA prototype generates the following class model, as illustrated in **Figure 7.10**.

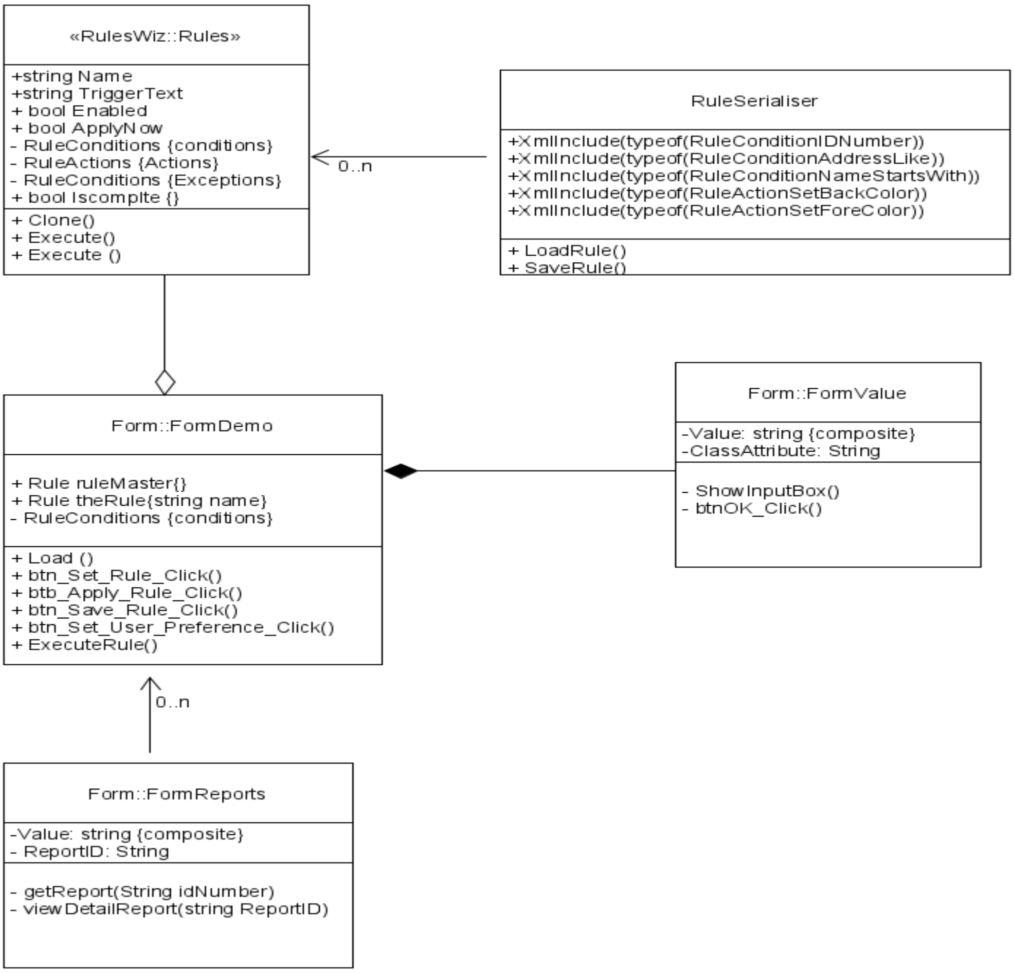

«RulesWiz::Rules»
+string Name
+string TriggerText
+ bool Enabled
+ bool ApplyNow
- RuleConditions {conditions}
- RuleActions {Actions}
- RuleConditions {Exceptions}
+ bool Iscomplte {}
+ Clone()
+ Execute()
+ Execute ()
0..n
RuleSerialiser
+XmlInclude(typeof(RuleConditionIDNumber))
+XmlInclude(typeof(RuleConditionAddressLike))
+XmlInclude(typeof(RuleConditionNameStartsWith))
+XmlInclude(typeof(RuleActionSetBackColor))
+XmlInclude(typeof(RuleActionSetForeColor))
+ LoadRule()
+ SaveRule()
Form::FormDemo
+ Rule ruleMaster{}
+ Rule theRule{string name}
- RuleConditions {conditions}
+ Load ()
+ btn_Set_Rule_Click()
+ btb_Apply_Rule_Click()
+ btn_Save_Rule_Click()
+ btn_Set_User_Preference_Click()
+ ExecuteRule()
Form::FormValue
-Value: string {composite}
-ClassAttribute: String
- ShowInputBox()
- btnOK_Click()
0..n
Form::FormReports
-Value: string {composite}
- ReportID: String
- getReport(String idNumber)
- viewDetailReport(string ReportID)

Figure 7.10. POPIA Prototype: Class Diagram

In **Figure 7.10**, the rules engine class diagram exposed five classes or functions available in the rules engine prototype, namely: RulesWiz::Rules, RulesSerialiser, Form::FormDemo, Form::FormValue and finally, Form::FormReports. In accordance with the UML model of Ambler (2011), each of these classes has a number of attributes (properties) and operations (methods). For instance, the class Form::FormValue has one composite attribute ‘value’ of type string, followed by two methods, namely ShowInputBox() and btnOK_Click().

To better design the inner working of the proposed POPIA rules engine, a sequence diagram is developed. According to Fowler (n.d.), a sequence diagram is typically used during the analysis and design phase of a software system to document and understand the logical flow of the system. In summary, a sequence diagram describes the flow of messages, events, and actions between objects of a system. A standard sequence diagram is made up of the following three essential parts:

1. **The participant:** which is the object or entity that acts in the diagram. The diagram starts with an unattached ‘found message’ arrow.
2. **The message:** Communication between participants and objects.
3. **The axes:** horizontal: which object /participant is acting; vertical: time (down -> forward in time).

Based on the sequence diagram notations and principles, the following sequence diagram is developed for the POPIA prototype.

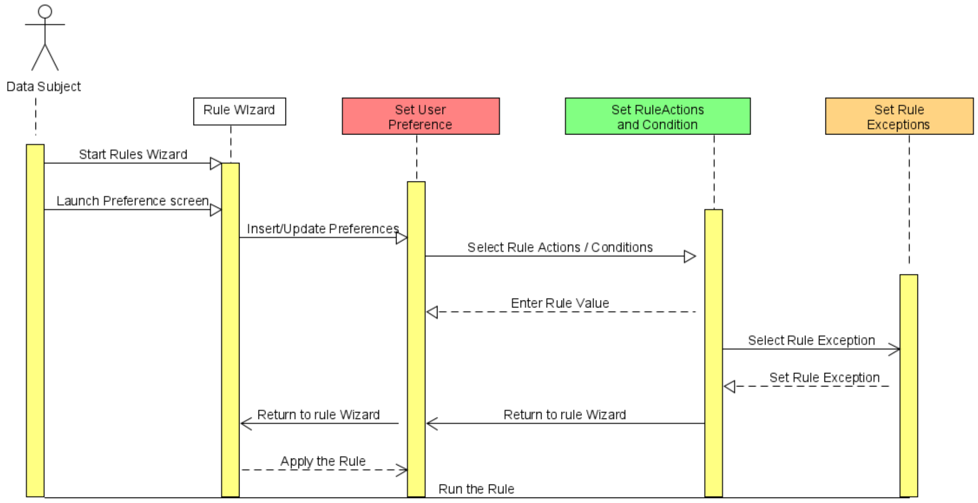


Figure 7.11. POPIA Rules Engine: Sequence Diagram

Based on **Figure 7.11**, the main participant is the data subject. The data subject can also be the business user who is interacting with the system to manage the personally identifiable information of the data subject. In the sequence diagram, the data subject launches the rules wizard and sends a message to set their user preferences, which triggers both the rules actions and conditions and finally, the rules exceptions at the different stages or verticals. At every stage of the sequence flow, messages are sent back and forth between the objects and participants, completing a forward and backward flow, as shown in **Figure 7.11**. Next is the ‘use cases’ identified for this prototype.

The detailed design of the ‘use cases’ are illustrated in **Appendix B**. However, in this section, highlights of the design and implementation of the use cases as software-based objects within the POPIA prototype are discussed in context. The first use case represented as a software object to be considered is the ***UC1: Implement POPIA Rule Engine Object***. This use case examines how the POPIA rules are implemented within the POPIA prototype used by both the organisation and the data subject to collect, share, process, and store personally identifiable information. Based on the framework used in the design of the artefact discussed in Chapter 6 and the design of the enforcement context and business rules engine in **Section 6.2.3**, business rules can be broken down into three rule parts:

1. **Rule condition**: The rule condition is essentially a statement of the rule expressed in the syntax of some programming language. In the context of this study and the POPIA prototype, the rule condition is expressed using the extensible markup language (UML) syntax.
2. **Rule action**: The rule action specifies the decision to be taken by the information system running the POPIA rule engine based on the evaluation of the rule condition, whether it is true or false.
3. **Rule exception**: The rule exception is a special type of rule condition, which is triggered only if an unexpected error occurs when the POPIA rule engine is busy processing the prescribed rule condition. For example, the rule exception may be as simple as to abort the operation and send notification of failure to the parties involved.

These three parts are associated together in the software bases object and deployed in the POPIA prototype to enforce POPIA compliance. **Figure 7.12** illustrates how these three RuleParts are constructed within the POPIA prototype.

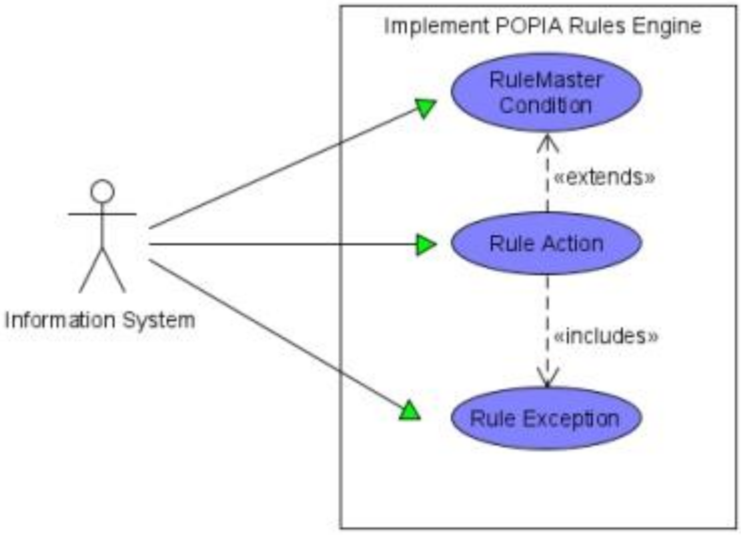


Figure 7.12. Use Case 1: Implement POPIA Rules Engine Object

Based on **Figure 7.12**, the implement POPIA rules engine object operates as follows:

1. All POPIA rule conditions are coded as XML instructions within the rule engine object.
2. All POPIA rule actions are coded as XML instructions within the rule engine object.
3. Exception conditions coded as XML instructions within the rule engine object.
4. Rule engine is able to listen to the user preference constraints setup in the Set User Preference Rules Object.
5. The resultant POPIA prototype developed is able to manipulate data based on the rules conditions and rule actions setup in the POPIA rule engine.
6. The POPIA rule engine is able to generate exceptions and pass them through as report entries.

All in all, the essence of the implement POPIA rules engine object is to help in the lawful manipulation of personally identifiable information within the prototype.

The next software-based use case object to be considered is the ***UC2: Set User Preference Rules Object***. This use case allows the data owners and the organisation to set their preference on the POPIA prototype to determine how PII data /information is collected, processed, shared, and stored by the organisation hosting and operating the information system. This use case heavily leans on Altman's (1975) privacy regulation theory highlighting the need for data subjects to manage private information and disclosure based on their level of comfort within a privacy regulating mechanism (the POPIA rule engine prototype). Based on the theoretical framework developed for this study, two (2) distinct sets of preferences were identified as follows:

1. **Data subject preferences**: These are preferences set either by the end users or employees of the organisation hosting the information system or processing the personally identifiable data identified in the model as stakeholders of the organisation.
2. **Organisation preferences**: These preferences vary from organisation to organisation and from industry to industry. For instance, government organisations such as the Police and Intelligence Services will have different levels of allowance and approval to manage PII as opposed to non-governmental organisations such as banks and insurance companies. However, all the stakeholders of the organisation are required to interact with the Set User Preference objects to capture and maintain their detailed information privacy preferences. In terms of the data subjects, they will interact with the system to achieve the following:

1. To determine the personally identifiable data to reveal or to conceal;
2. To capture privacy concerns on the POPIA rule engine; and
3. To verify whether privacy concerns have been captured.

For the organisation, it is as follows:

1. To seek data subject permission on data items to be revealed or concealed;
2. To classify data based on sensitivity and security best practices; and
3. To notify data subject and regulator of data violation.

In summary, the activities that the data subject carries out on the Set User Preference Object is maintenance in nature, which is summarised in the use case diagram illustrated in **Figure 7.13**.

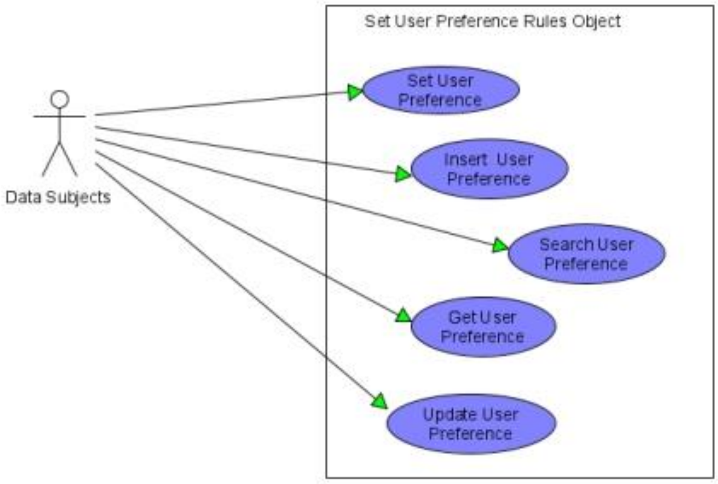


Figure 7.13: Use Case 2: Set User Preference Rules Object

Based on **Figure 7.13**:

1. Data subject selects the Set User Preference Rules Object from the menu.
2. System displays list of User Preference Rules available for both users and organisations.
3. Data subject selects one or more User Preference Rules and engine lists the different privacy dimensions for that rule.
4. Data subject configures validation criteria for that specific POPIA rule based on the user preference and privacy dimension.
5. Data subject clicks Submit button.
6. The system stores the user preference for that data subject and displays a confirmation message.

All in all, the rationale for this object is to enable every data subject to configure their privacy preferences on the POPIA prototype in order to help the system to decide which data to reveal or conceal about the data subject. Moving on from this, the next software-based object use case is the ***UC3: Personally Identifiable Information Rules Object***. This use case is underpinned by POPIA, highlighting the different data/information elements constituting personally identifiable information (PII). According to POPIA, the data subjects have constitutional prerogatives to protect their personally identifiable information from exploitation by third parties and other commercial entities. This task requires organisations to classify data based on the conditions for the lawful processing of personal information, as stipulated in Chapter 3 of the POPI Act. As result, every organisation that is a custodian of PII should set allowable actions on PII data, report violations on PII data and, most importantly, define remediation actions. The use case diagram below illustrates the inner workings of the Personally Identifiable Information Rules Object.

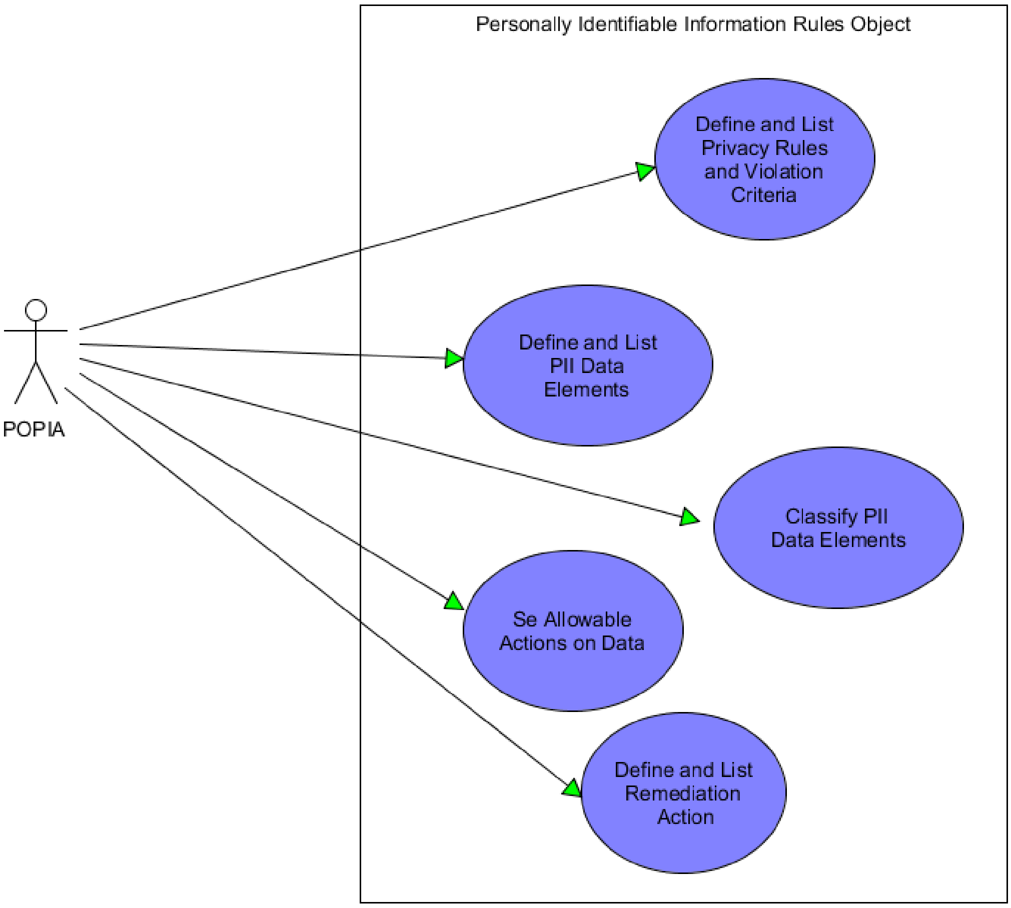

Personally Identifiable Information Rules Object
Define and List Privacy Rules and Violation Criteria
Define and List PII Data Elements
Classify PII Data Elements
Se Allowable Actions on Data
Define and List Remediation Action
POPIA

Figure 7.14: Use Case 3: Personally Identifiable Information Rules Object

1. Define and list all POPIA information privacy rules. (The eight rules of the lawful processing of personal information).
2. Define and list all personally identifiable information, as defined by POPIA.
3. Classify personally identifiable information, according to their sensitivity.
4. Set allowable actions on personally identifiable information.
5. Define personally identifiable information violation criteria.

Define remediation actions on the violation of personally identifiable information. This role will typically be done by the administrator and the software-based object here functions as the broker object to link the Set User Preference Object, the POPIA rule engine object and the regulatory report object.

Finally, the last, software-based object use case is the ***UC4: Regulatory Report Object***. This use case links the privacy concern object with the system protection object to determine compliance with POPIA regulations. In the context of POPIA, the regulator is known as the Information Regulator and is the mandated authority to enforce POPIA compliance. The use case diagram below illustrates the inner working of the regulatory report object.

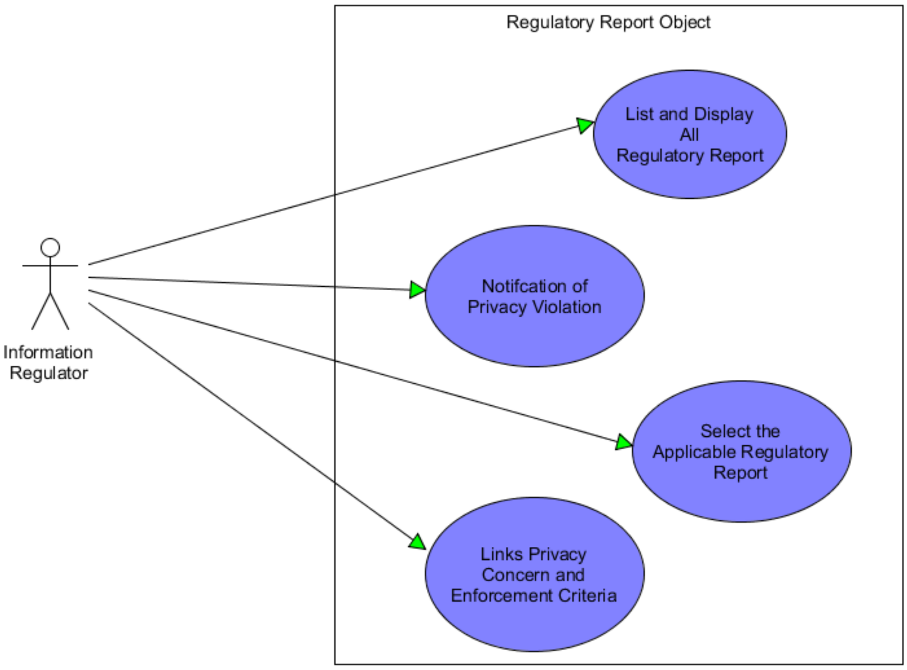

Regulatory Report Object
List and Display All Regulatory Report
Notifcation of Privacy Violation
Select the Applicable Regulatory Report
Links Privacy Concern and Enforcement Criteria
Information Regulator

Figure 7.15: Use Case 4: Regulatory Reports Object

The activities performed on the regulatory report object are listed below:

1. Regulator is notified of a particular violation of privacy concern.
2. Information regulator selects Regulatory Report Object from the menu.
3. System displays list of All Regulatory Reports available.
4. Regulator selects one or more Privacy Enforcement Criteria and system lists items for that criteria.

5. Regulator links the privacy concern with the privacy enforcement criteria.
6. Regulator links privacy enforcement criteria with a particular organisation's system.
7. Regulator clicks the Check button.

System scans information system objects to determine compliance. This object is administrative in nature and occurs once a violation has been reported and enforcement needs to happen.

To this point, the inner working of the POPIA prototype and the selected use cases for implementation have been described and documented using the UML language. **Figure 7.16** shows a schematic version or architecture of the prototype to be set up and implemented for this study.

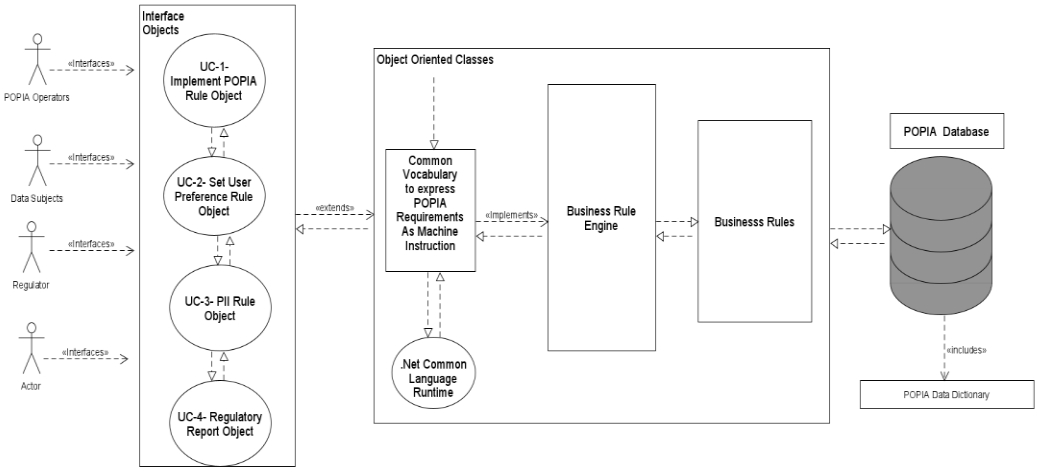
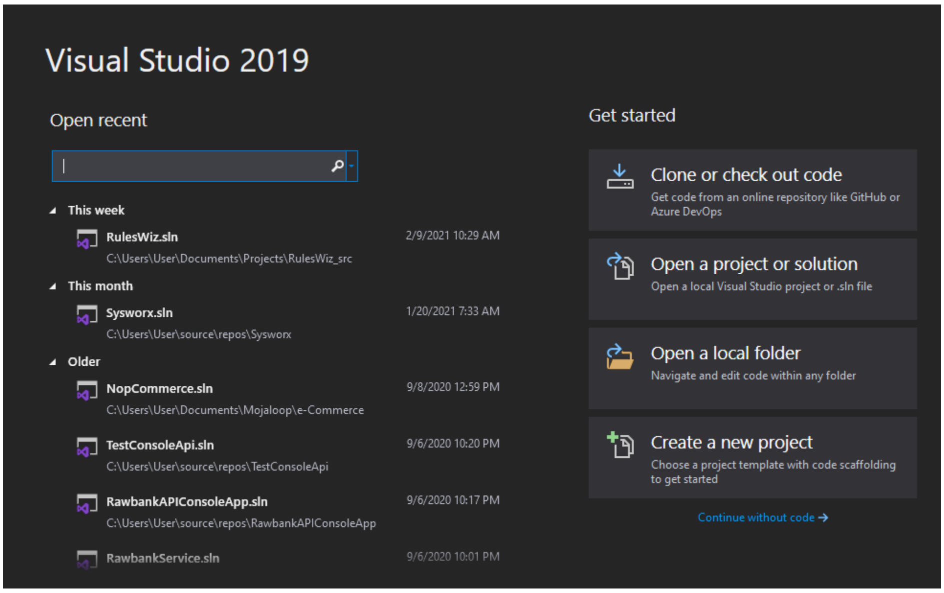


Figure 7.16: POPIA Prototype

The next section focuses on actually setting up and testing the POPIA prototype, as documented in **Figure 7.16.**

7.3 Setup and Testing of POPIA Prototype

To test and simulate the prototype, an environment was setup using Microsoft Visual Studio 2019 Professional. The community version was used because of licence implications since this version is free. **Figure 7.17** shows a screenshot of the Visual Studio 2019 project interface.



This week
RulesWiz.sln C:\Users\User\Documents\Projects\RulesWiz_src 2/9/2021 10:29 AM
This month
Sysworx.sln C:\Users\User\source\repos\Sysworx 1/20/2021 7:33 AM
Older
NopCommerce.sln C:\Users\User\Documents\Mojaloop\e-Commerce 9/8/2020 12:59 PM
TestConsoleApi.sln C:\Users\User\source\repos\TestConsoleApi 9/6/2020 10:20 PM
RawbankAPIConsoleApp.sln C:\Users\User\source\repos\RawbankAPIConsoleApp 9/6/2020 10:17 PM
RawbankService.sln 9/6/2020 10:01 PM

Clone or check out code
Get code from an online repository like GitHub or Azure DevOps

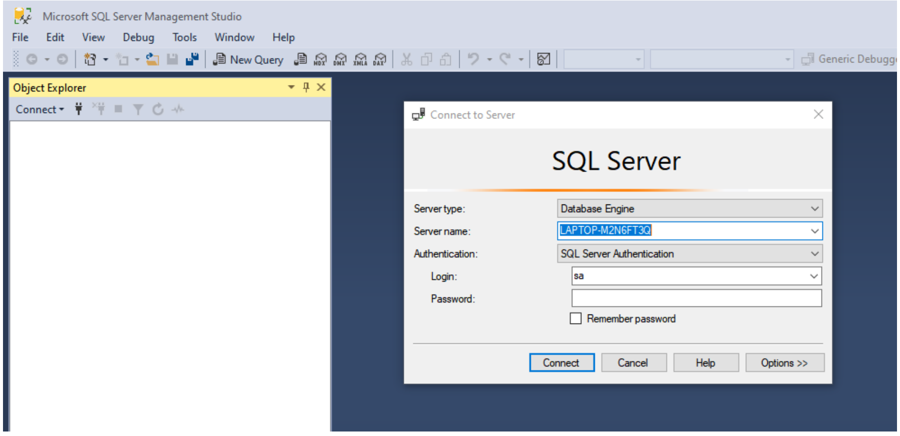
Open a project or solution
Open a local Visual Studio project or .sln file

Open a local folder
Navigate and edit code within any folder

Create a new project
Choose a project template with code scaffolding to get started

Figure 7.17: Use Case: Visual Studio 2019-Project

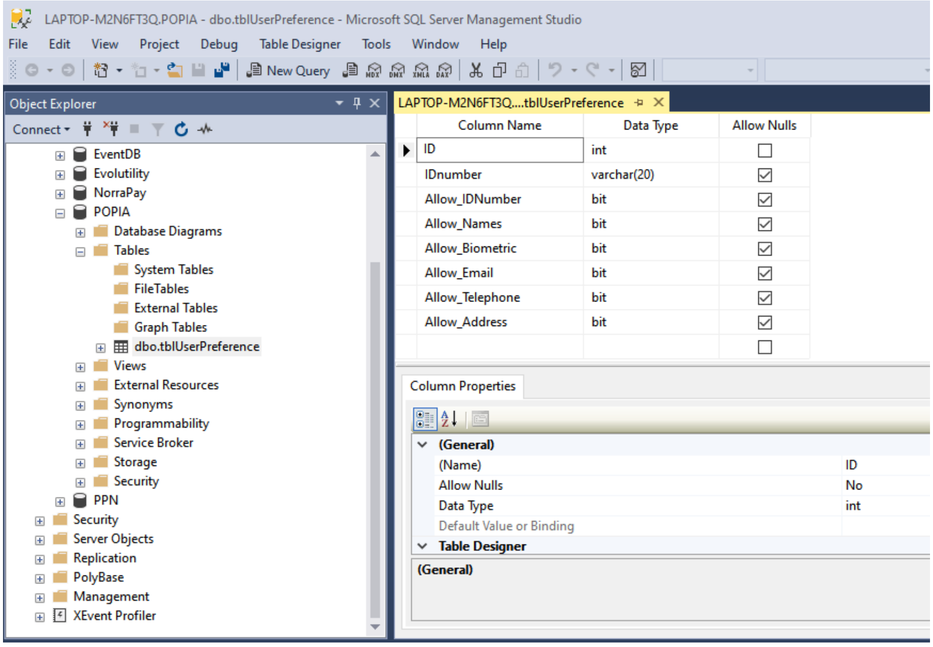
The POPIA prototype software-based objects were coded using Microsoft Visual C# as the code-behind programming language targeting the Microsoft Dot.net Framework 4.0. Similarly, the POPIA prototype's database was created using Microsoft SQL Server 2016 and was accessed using the SQL Server management Studio 1.7. **Figure 7.18** shows the SQL Management Studio 1.7 interface used to access the prototypes database.



Server type: Database Engine
Server name: LAPTOP-M2N6FT3Q
Authentication: SQL Server Authentication
Login: sa
Password:
□ Remember password

Figure 7.18: SQL Server Management Studio Interface

In the SQL Management Studio administrative interface, a database called POPIA was created with tables to capture the PII data required to simulate the prototype, as shown in **Figure 7.18.**



Column Name	Data Type	Allow Nulls
ID	int	☐
IDnumber	varchar(20)	☑
Allow_IDNumber	bit	☑
Allow_Names	bit	☑
Allow_Biometric	bit	☑
Allow_Email	bit	☑
Allow_Telephone	bit	☑
Allow_Address	bit	☑
		☐

(General)
(Name) ID
Allow Nulls No
Data Type int
Default Value or Binding
Table Designer
(General)

Figure 7.19: POPIA Table to Capture PII

Based on **Figure 7.19**, some of the PII data captured are emails, ID number, telephone number, address data, biometric data, name and surname, just to mention a few. This data is used by the prototype to build metadata for the POPIA inference engine during the simulation. The interface to the POPIA database was built using Visual C# Windows forms showing the look and feel and controls to carry the programming logic in the code-behind files. The prototype is fully object oriented and built using the agile software development methodology (rapid prototyping). More will be discussed about the prototype in Chapter 8 on the evaluation of the ‘live prototype’.

7.4 Conclusion

In this chapter, the framework developed in Chapter 6 was used to implement the prototype to enforce POPIA in technology systems. In implementing this framework, first, the UML ontology language was used to model the different components of the organisation in which this prototype will be used. The main components modelled were the actors, their actions and use cases, and finally, their relationship with one another. Second, the object-oriented paradigm was used to build the business rules and rules engine classes required by the objects of the information systems. Lastly, the prototype was set up for testing and evaluation using the Microsoft.Net C# programming language and the Microsoft SQL Server database technologies. Hence in conclusion, this chapter achieved the goal of developing the live POPIA prototype (Artefact) required by the design science methodology to test the assertions defined for this study.

CHAPTER 8

Evaluation of Prototype

8.1 Introduction

Stepping further from the implementation of the prototype covered in Chapter 7, this chapter focuses on the observation and evaluation of the ‘live prototype’ in line with the expectations and conditions defined in the research questions, problem, and objectives of this study.

The STEP principle derived from the business rules approach is used to evaluate the POPIA prototype. Furthermore, the evaluation of the live prototype is done using other criteria and findings elaborated on in **Section 8.5** of Chapter 8.

8.2 STEP Principles

STEP is the acronym for the following business rule terms (Separating, Tracing, Externalising and Positioning) within information systems. According to von Halle (2002), it is a tool derived from the business rules approach or methodology and is used in information systems to evaluate business rule design. According to Gougeon (2003), there are several definitions of a business rule but the most simplistic definition holds that a business rule is a constraint placed upon the business.

Meanwhile, von Halle and Sandifer (1993) provide a more practical and technical definition, which suits the context of this study, by defining a business rule as a natural language sentence that describes the data requirements to the business user. This definition is particularly useful to this study as it touches on data which, in this study, is represented as the personally identifiable information residing in information systems. According to von Halle (2002), STEP elements can be subscribed in full as follows:

1. S is for separating rules from the rest of the system so that they can be isolated and reused.

2. T is for tracing rules to determine the rationale for their existence and to get an inventory of where they are being used so that they can be changed if the need arises.

3. E is for externalising rules so that they are not hard coded in systems and so that anyone with business knowledge and expertise can access and change the rules easily.

4. Finally, P is for positioning to enable quick and easy change to the rules.

With a full understanding of the STEP acronym, the next section uses the STEP principles to evaluate the business rules built into the POPIA prototype

8.3 Evaluating Prototype Using STEP

Table 8.1 shows how to adapt the STEP principles defined above as assessment criteria for the evaluation of a business rules system. In the analysis phase of the POPIA rule engine prototype, it is tested and analysed to see whether it conforms to the principles of the von Halle (2002) business rules system methodology. The analysis of the POPIA rule engine prototype using the von Halle (2002) STEP principles is based on the assertion by von Halle that a business rules approach is best suited to deliver that guidance system with externalised rules, automated as an integral and active component in systems architecture.

This brings a new knowledge-focused way of designing new systems into light. In such a new system, it is no longer acceptable to bury knowledge deep in code where no one knows what it is. It is equally no longer acceptable to have that knowledge locked up where it cannot change on demand. However, in a business rules approach, technology is deployed to externalise and manage rules which, in turn, empowers the organisation by improving its capacity to make decisions and change systems more quickly, using the technology as an extension of its intellectual power.

Table 8.1: Rules Assessment Criteria

Assessment principles (STEP)	**Evaluation criteria**	**Response (Yes/No)**	**Comments**
1. (S) Separate rules	To reuse rules To apply special techniques to optimise rule quality To change rules independently of other system aspects		
2. (T) Trace rules	To determine, over time, whether the rule remains a correct rule for guiding the business. To assess the impact of rule changes		
3.(E) Externalise rules	To allow everyone to know where the rules can be known To allow everybody to know what the rules are To allow everyone to challenge the rules		
4. (P) Position rules	To enable quick rule change		

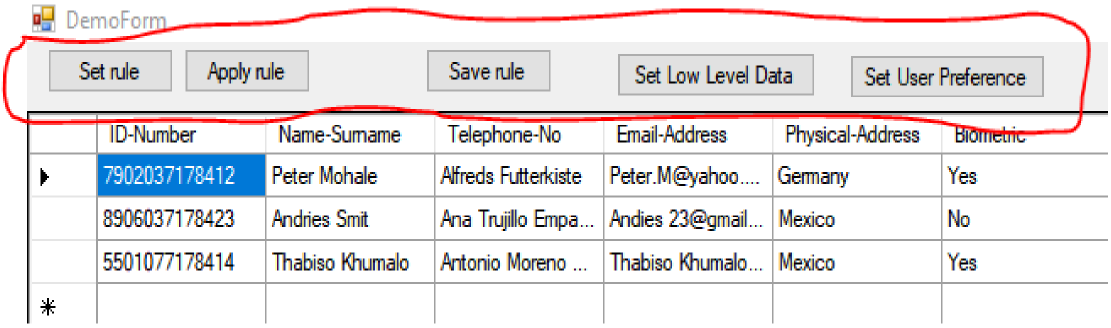
8.4 Additional Findings Based on the STEP Evaluation

Based on the von Halle (2002) STEP principle discussed in the design section, the following evaluation criteria were developed. **Table 8.2** shows findings from the evaluation of the POPIA rule engine prototype, based on the STEP Criteria.

Table 8.2: POPIA Rule engine: STEP Evaluation

Assessment principles (STEP)	**Evaluation criteria**	**Response (Yes/No)**	**Comments**
1. (S) Separate rules	• To reuse rules • To apply special techniques to optimise rule quality • To change rules independently of other system aspects	Yes	From the rule engine design, the rules are kept separate from the rest of the application logic. Any business user can use the rules Wizard Interface to create and apply rules without knowing the full inner working of the rules system.
2. (T) Trace rules	• To determine, over time, whether the rule remains a correct rule for guiding the business. • To assess the impact of rule changes	Yes	The rules are saved and can run with a trace from the Interface. Hence, there is full auditability of the rules as implemented in the system.
3. (E) Externalise rules	• To allow everyone to know where the rules can be known • To allow everybody to know what the rules are • To allow everyone to challenge the rules	Yes	The rules are visible on the interface of the rule engine and can be saved and exported into an XML file for full visibility / audit trail. `RuleSerializer.SaveRule(theRule, Application.StartupPath + "\\rule.xml");`
4. (P) Position rules	• To enable quick rule change	Yes	Entirely in user-friendly Interface. The interface is wizard driven and gives menus to perform the following: 1. set rule 2. apply rule 3. save rule 4. set user preference 5. set lower-level data Figure 8.1 shows the POPIA rule engine - Interface Menu Items

Figure 8.1 shows the top menu of the POPIA rule engine Interface. In this menu there are items to Set Rules, which allow the system user to create and set new rules on the Rule Engine. Next to the Set rule menu is the Apply rule menu. This menu enables the user to apply the rule on the system immediately as soon as the rule is created. Further on is the menu to Save rule. This menu allows the user to save rule after it has been created for use at a later stage. The Save rule menu item creates an entry in an XML file on the system where the rules are stored and can be retrieved for later use.



	ID-Number	Name-Surname	Telephone-No	Email-Address	Physical-Address	Biometric
▶	7902037178412	Peter Mohale	Alfreds Futterkiste	Peter.M@yahoo....	Germany	Yes
	8906037178423	Andries Smit	Ana Trujillo Empa...	Andies 23@gmail...	Mexico	No
	5501077178414	Thabiso Khumalo	Antonio Moreno ...	Thabiso Khumalo...	Mexico	Yes
*						

Figure 8.1: POPIA Rule Engine - Interface Menu Items

8.5 Findings on Utilising the Rule Engine Prototype

In using the POPIA prototype, the user can immediately focus on implementing the business logic for specific Rule Conditions, Rule Actions and Rule Exceptions without worrying about how to manage the underlying software instructions written in code. The prototype provides a wizard, which guides the user in terms of setting up the rules. The screens of the wizard implement easy-to-use controls such as Listview, which contains text with hyperlinks leading to other menu criteria and items. In addition, the wizard also summarises the rules options which the user has selected cascading it with the rules conditions linked with the exception conditions.

The prototype also makes provision for the user to save their rules for future use or reference; this is achieved by the Serialise / de-Serialise class, which converts the rules object into an XML file and can also load the XML file. Based on the use case definitions above, there are three (3) major actors (the data subject: the person whose personal information is at risk; the information system: the software system using the rule engine prototype to enforce POPIA; and the organisation: which is the entity responsible for safeguarding the personal information of the data subject).

In the POPIA rule engine prototype, there is direct association between the data subject and the mechanism to set its user preferences on the one hand, and on the other hand, responsibility for system-specific rules lies with the organisation as the principal actor in the enforcement of the POPIA rule engine, since by law (according to the POPIA) the organisation becomes the custodian of said information/data. In utilising the Set User Preference Object of the POPIA rule engine prototype, we note the following findings:

1. The capability to set rules for user preferences is done upfront as every other behaviourism of the POPIA rule engine prototype is intricately linked to the data subject preferences. This pre-configuration enables the system to behave in way that is aligned to POPIA prescribes and best practices upfront.
2. If the user does not set any preference upfront, the basic rules of the POPIA will apply. This will serve as the lowest common denominator where specific privacy rules are applicable. By virtue of this, the system is termed POPIAfriendly from the onset.
3. Where the personally identifiable information is not properly defined or in doubt or where the hierarchy of personal information is not defined, setting the user preference becomes a problem and will be difficult to implement in the POPIA rule engine. This scenario is common with quasi-identifiers that do not directly refer to the data subject but when combined together using intelligent software can easily help in identifying the underlying data subject.
4. This same problem exists from an organisational perspective. It will occur if, as an organisation, an attempt is made to set the user preference when the organisational data is not classified based on sensitivities such as top secret, secret, confidential, and public sensitivities.
5. Every user’s preferences are unique and set to change as per their perceived level of social interaction as well as per their desired privacy outcome. This is illustrated in Figure 8.2, which shows a diagrammatic representation of

the Altman Privacy Regulation Theory (1975). According to Elprama et al. (2011), who based their argument on Altman's (1975) privacy regulation theory, privacy is seen as a balance between social isolation and desired exposure.

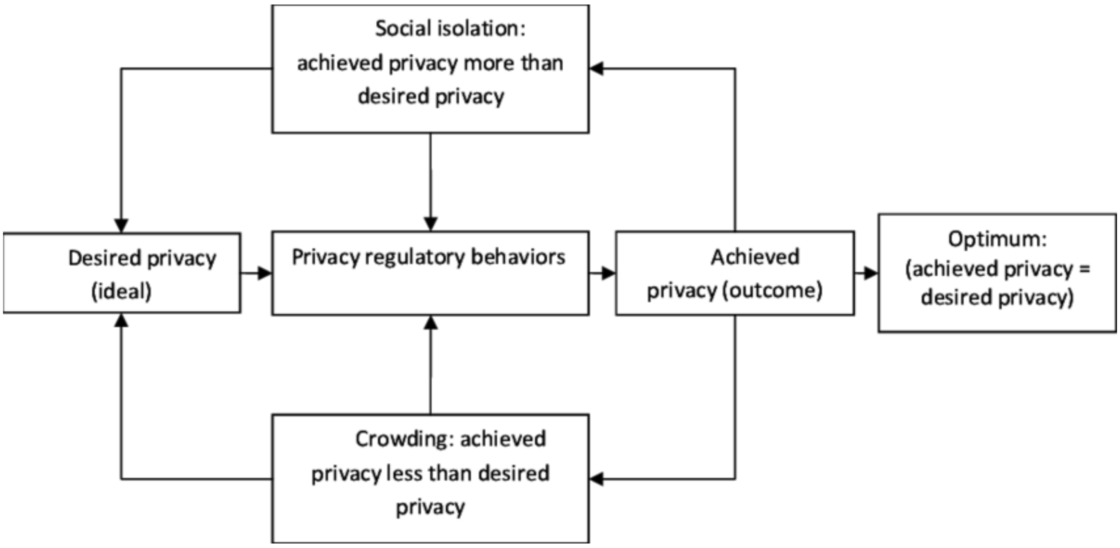
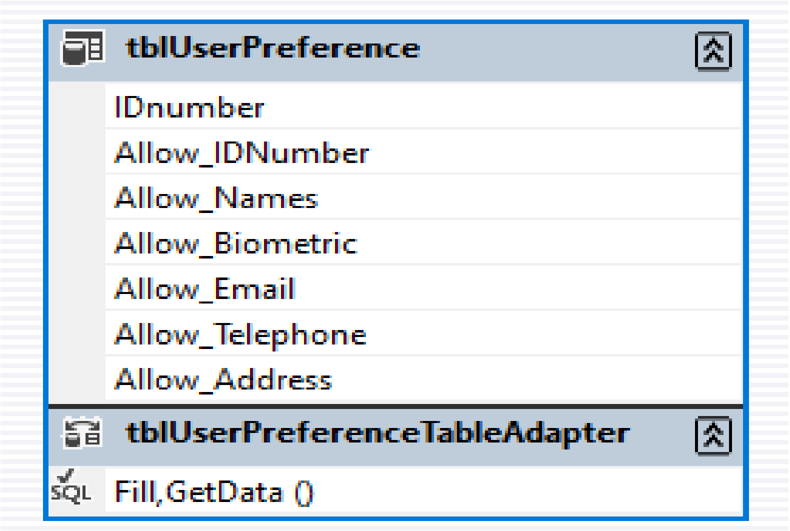

Social isolation: achieved privacy more than desired privacy
Desired privacy (ideal)
Privacy regulatory behaviors
Achieved privacy (outcome)
Optimum: (achieved privacy = desired privacy)
Crowding: achieved privacy less than desired privacy

Figure 8.2: Privacy Regulation Theory (Adapted from Altman 1975)

(Source: Elprama et al., 2011)

6. To populate the data subject preferences into the POPIA rule engine, the following table will be called upon to fetch the data containing the selected preferences. This table has a data dictionary of the top level information of the data subject which, when combined with any generic level information, can be used to identify the data subject precisely. For instance as per **Figure** 8.3, if the Allow_IDNumber is set to true, then the identification number of the data subject will be visible, and if this identification number is paired with a generic informal such as a criminal record or medical record, then you can know precisely the criminal record or medical record of the data subject.



tblUserPreference

IDnumber

Allow_IDNumber

Allow_Names

Allow_Biometric

Allow_Email

Allow_Telephone

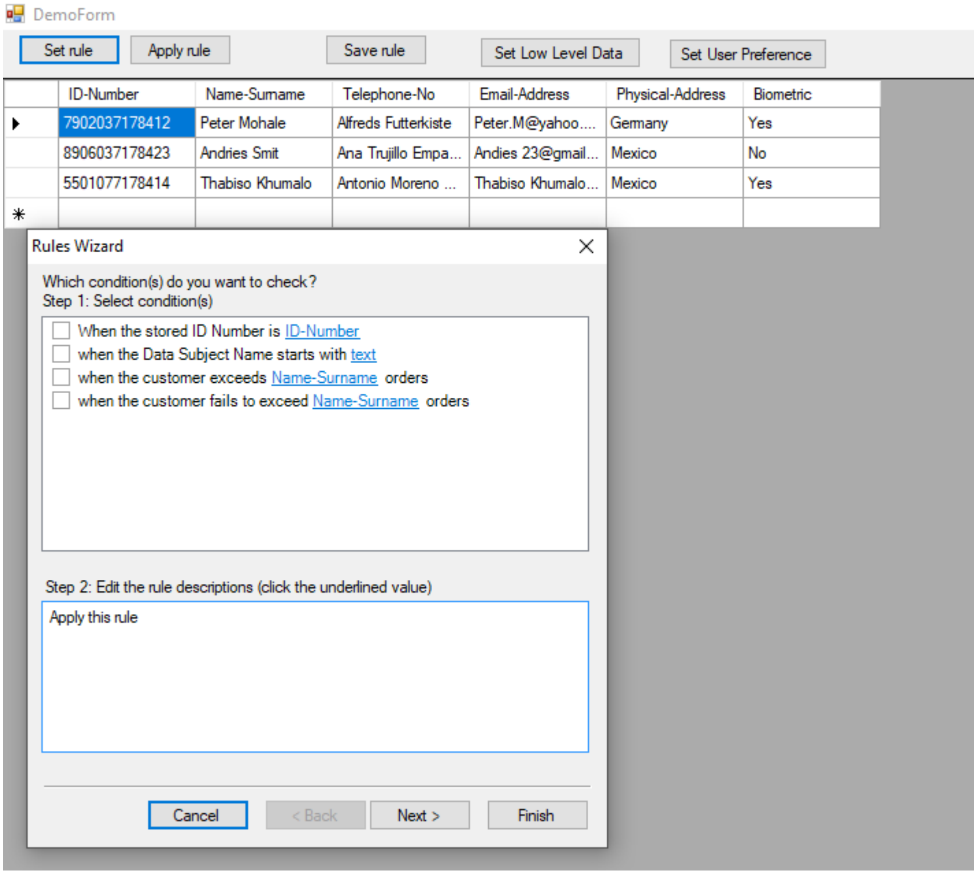
Allow_Address

tblUserPreferenceTableAdapter

Fill,GetData ()

Figure 8.3: Database Table: Set User Preference

7. The Enforce POPIA rule engine includes the Set User Preference Rules object. This association is an include association, meaning the behaviour of the Set User Preference use case is inserted into the behaviour of the including Enforce POPIA Rule engine object use case.
8. The POPIA rule engine prototype presents a very visual, intuitive and user-friendly user interface in the form of a wizard. Through this wizard, the data subjects / organisational users can easily set up POPIA rules about their data elements without any need for coding. **Figure 8.4** shows the Visual Wizard Interface used to setup rules on the POPIA rule engine. It is important to highlight that, through this single interface, the data subject can set Rule.Condition, Rule.Action and Rule.Exception for the same rule, meeting the theoretical design criteria of the rule engine. **Figure 8.4** also shows the Wizard Interface screen to configure the Rule.Exception for a specific rule.



ID-Number	Name-Surname	Telephone-No	Email-Address	Physical-Address	Biometric
7902037178412	Peter Mohale	Alfreds Futterkiste	Peter.M@yahoo....	Germany	Yes
8906037178423	Andries Smit	Ana Trujillo Empa...	Andies 23@gmail...	Mexico	No
5501077178414	Thabiso Khumalo	Antonio Moreno ...	Thabiso Khumalo...	Mexico	Yes

Which condition(s) do you want to check?
Step 1: Select condition(s)

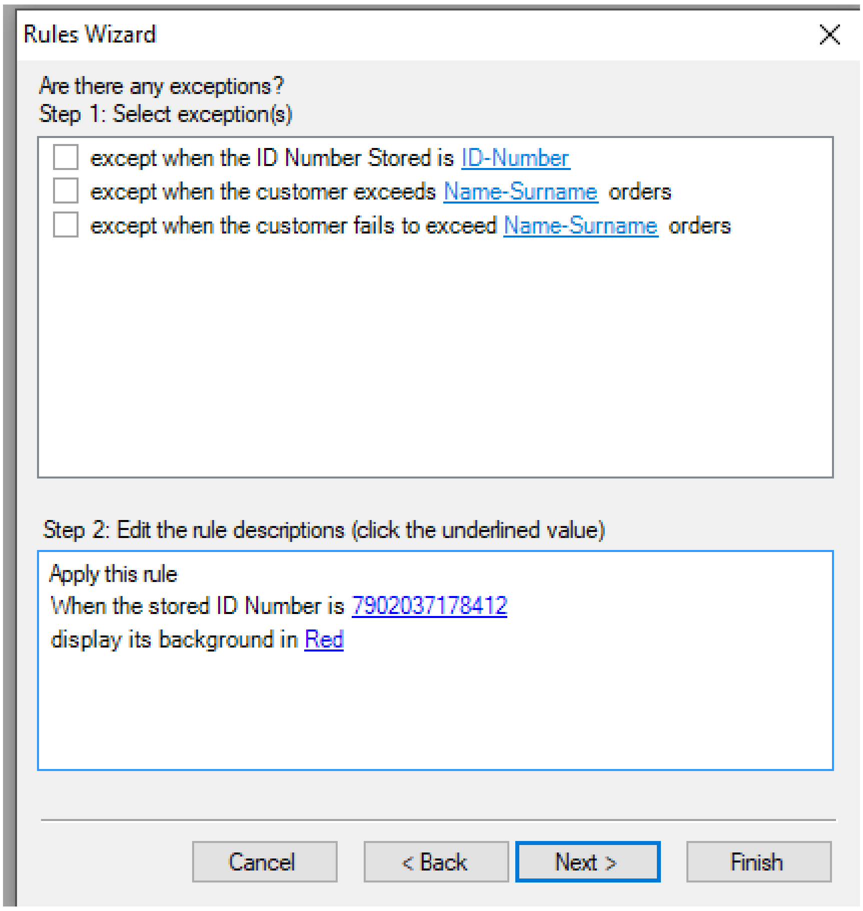
□ When the stored ID Number is ID-Number
□ when the Data Subject Name starts with text
□ when the customer exceeds Name-Surname orders
□ when the customer fails to exceed Name-Surname orders

Step 2: Edit the rule descriptions (click the underlined value)

Apply this rule

Figure 8.4: POPIA Rule Engine - Visual Wizard Interface

Figure 8.5 shows the rules exceptions in both the select criteria and the edit criteria. It allows the user to select and edit the exception criteria of the rules that they are configuring on the software object with the POPIA rule engine prototype.



Rules Wizard

Are there any exceptions?
Step 1: Select exception(s)

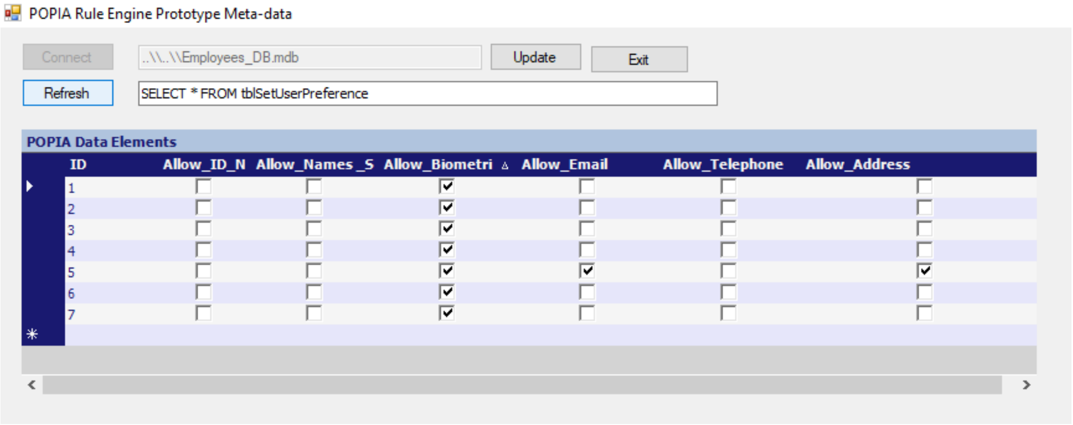
□ except when the ID Number Stored is ID-Number
□ except when the customer exceeds Name-Surname orders
□ except when the customer fails to exceed Name-Surname orders

Step 2: Edit the rule descriptions (click the underlined value)

Apply this rule
When the stored ID Number is 7902037178412
display its background in Red

Figure 8.5: POPIA Rule Engine - Rule.Exception Interface

9. By design and architecture, the POPIA rule engine prototype stores minimal subject data elements. Instead, it stores more configuration data (metadata) of the data elements. That way, the inference engine can make decisions based on the data elements presented to it. **Figure 8.6** shows some of the configuration fields that the user checks or unchecks; this will allow them to make preferences about their PII metadata without actually interfering with the data elements. This mode of setting the system configuration through a graphic user interface (GUI) is highly interactive and makes it easy for system users to operate and update configuration changes.

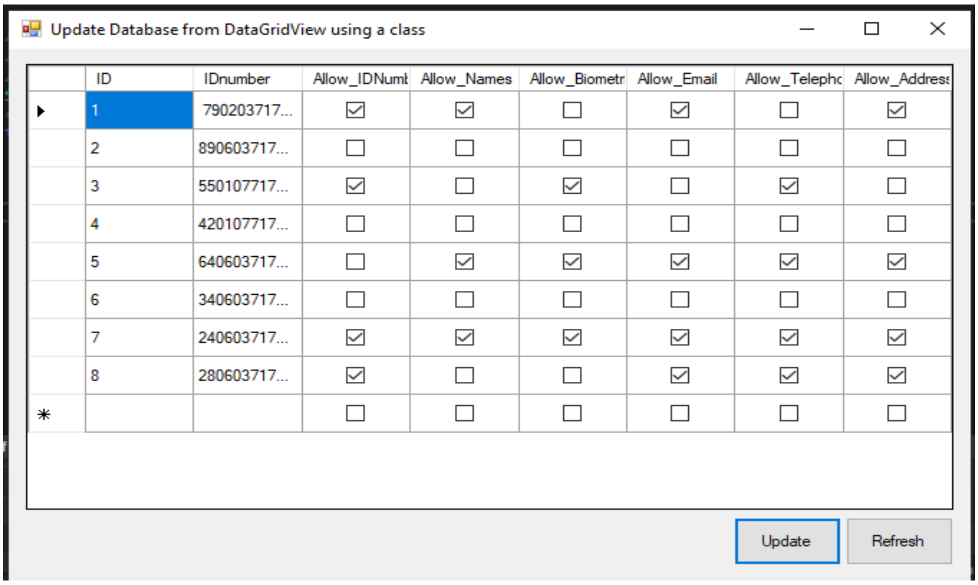


POPIA Rule Engine Prototype Meta-data
Connect
..\\..\\Employees_DB.mdb
Update
Exit
Refresh
SELECT * FROM tblSetUserPreference
POPIA Data Elements

	ID	Allow_ID_N	Allow_Names_S	Allow_Biometri ▵	Allow_Email	Allow_Telephone	Allow_Address
▶	1	□	□	☑	□	□	□
	2	□	□	☑	□	□	□
	3	□	□	☑	□	□	□
	4	□	□	☑	□	□	□
	5	□	□	☑	☑	□	☑
	6	□	□	☑	□	□	□
	7	□	□	☑	□	□	□
*							

Figure 8.6: POPIA Rule Engine - PII Metadata Repository

From a system interface perspective, the interface presents the user with options to select which of their personally identifiable information data they can use as metadata for setting the rules, as illustrated in **Figure 8.7**.



	ID	IDnumber	Allow_IDNumt	Allow_Names	Allow_Biometr	Allow_Email	Allow_Telephc	Allow_Address
▶	1	790203717...	☑	☑	□	☑	□	☑
	2	890603717...	□	□	□	□	□	□
	3	550107717...	☑	□	☑	□	☑	□
	4	420107717...	□	□	□	□	□	□
	5	640603717...	□	☑	☑	☑	☑	☑
	6	340603717...	□	□	□	□	□	□
	7	240603717...	☑	☑	☑	☑	☑	☑
	8	280603717...	☑	□	□	☑	☑	☑
*			□	□	□	□	□	□

Figure 8.7: POPIA Rule Engine - Options to Select Metadata.

8.6 Discussion of Findings and Observations

The process to conceptualise, design, and utilise the POPIA rule engine prototype generated a number of key findings and observations, as highlighted in the sections above. The findings and observations are structured into two parts, namely general findings and observations, and findings observations based on the STEP evaluation criteria. What follows is a discussion of these observations and findings.

First, from a conceptual framework perspective, it was observed that the enforcement of POPIA rules within an organisation is context driven. This position was derived and inspired by the work of Regoniel (2005). In this vein, a conceptual model was formulated in Chapter 5 and the following six (6) contexts were identified, namely research context, organisational context, translation context, technology context, enforcement context, and information system context. In exploring all of these contexts and their purpose, it was seen that some of these contexts are sub-contexts to the main context. The POPIA prototype is the output of the information system and validation context, which is a sub-context of the organisation context. As a result, the POPIA prototype is central to the enforcement of the conditions of the lawful processing of personal information within an organisation through an information system and validation context.

Second, from a design perspective, the forward-chaining business rule execution method was adopted owing to its ability to build a knowledge base and draw inferences from data in a loosely coupled design architecture. This design model is supported by the finding that the POPIA rule engine prototype presents a Wizard Interface, by which application logic and business rules are loosely coupled and separated from one another. Furthermore, the visual, intuitive, and user-friendly user interface presented by the POPIA rule engine prototype makes it very easy for data subjects / organisational users to set up POPIA rules about their data elements without any need for coding. In addition, based on the von Halle (2002) STEP principle and the derived evaluation criteria outlined in **Table 8.1**, the POPIA business rules prototype demonstrates sufficient evidence of separation of rules and application logic as well easy rule set up through a Wizard Interface.

Third, based on the practical design and modelling of the different components of the POPIA prototype, challenges surrounding data classification and identification of personally identifiable information were encountered by the prototype in attempting to handle the data. The UML ontology language was instrumental as a tool to model the hierarchical data structure pertaining to the personally identifiable information as well as to model the different use cases, class diagrams, and activity diagrams showing the internal working of the POPIA prototype's components. It also appears, from running the prototype, that it utilises more metadata than actual data to make inferences to apply to the POPIA rule. In fact, according to the findings, where PII or its hierarchy is not properly defined or is in doubt, setting rules within the prototype becomes problematic.

Fourth, from a theoretical perspective, and leaning on the Altman (1975) privacy regulation theory, the privacy preference of each user or employee is unique and based on the balance between their social isolation and desired exposure. However, in the context of today, where most data is stored in cloud systems or managed by third parties, the data subjects do not have control over the enforcement of this balance. As such, new variables need to be introduced to the privacy regulation theory to extend the theory to cover the context of the cloud or third-party hosted data.

Finally, from a governance perspective, rules can create unintended consequences if there is no formal governance process in place for use in creating and approving business rules prior to their implementation in production systems. As a finding from this prototype, any business user can call up the Wizard Interface and proceed to create a rule, which they can apply or save and apply later on the rule engine. Another angle to this problem is the process of determining which rules go into the application login and which go into the rule engine. The recommendation is that organisations adopt the decision model notation (DMN) as a formal standard to manage their business rule engine.

8.7 Evaluation of the Live Prototype

Based on the observation and evaluation of the POPIA prototype in action, the following points were noted, in summary:

1. The rule inference engine is powered by metadata and a knowledge base built from the configurable rules in addition to the personally identifiable information created by the data subject or users when they set their preferences on the prototype.

2. The knowledge base and rules specific to a data subject or user constitute a privacy

context, which is specific to that particular user or data subject. In turn, this creates a context-driven reality for compliance with POPIA relative to that specific user or data subject.

3. The POPIA rules used by the prototype were configurable and could be set or changed on the fly, using the intuitive wizard-like user interface and also without hard coding into the application logic.

4. The forward-chaining business rules execution method employed in the prototype has a decision structure which enables it to make inference from a knowledge base or from metadata before applying constraints on the business process.

5. In the prototype there is direct association between the data subject and their privacy context. This connection is framed in the rule engine and implemented in the information system handling the data subject’s personally identifiable information.

6. Evaluating the prototype using the von Halle (2002) STEP principle confirmed that the prototype implemented adequate separation of rules from application logic

7. In evaluating the prototype, a big gap was seen in that the ease of implementing and changing rules can create unintended consequences for the organisation if proper internal governance processes and controls are weak or absent.

8.8 Conclusion

In this chapter, the POPIA prototype was evaluated using the criteria of the von Halle (2002) STEP principles to determine how well it can handle business rules deployed as a business rules engine. Furthermore, other findings and observations on the utilisation of the prototype were outlined and discussed in context. Finally, a summary of the findings and observations was recorded as input into the next chapter, which deals with the conclusion of the research study. Hence, in conclusion, sufficient findings and observations were extracted from the activities undertaken in this chapter to enable the research study to make valuable conclusions in the next chapter.

CHAPTER 9

Conclusions and Implications

9.1 Introduction

This is the final chapter of this study, focusing solely on the conclusions and implications of the research study. This chapter first covers the general conclusions reached from the observation and evaluation of the findings of this study, as discussed in **Section 9.5**.

Second, this chapter delves into the specific conclusions based on the research question and problem identified and highlighted in the study.

Concluding this chapter, the implications of this study are highlighted and discussed in context, especially as they pertain to information privacy industry practitioners as well as to the broader research community. Finally, suggested recommendations will be made for future research.

In terms of the value proposition of this research project, while there are several proposals in literature on how to build an effective informational privacy control mechanism, at the same time, according to Wu et al (2020), there is still a gap in building a one-size-fits-all solution for information privacy needs that spans multiple contexts. Hence, Wu et al. (2020), propose that, although investigating privacy using a contextual approach is more difficult, it nonetheless provides a more accurate reflection of the privacy context.

As a result, the information technology community should lead the enquiry into the development of privacy-sensitive systems that take into consideration the needs and requirements of a wider range of users and communities (Wu et al., 2020). To effect this research challenge and to understand the privacy context in depth, this study explored one of the information privacy legislations – POPIA – as a use case, and developed a conceptual framework and a design framework through which the POPIA rules and knowledge domain is represented in software objects as a context-driven reality.

In utilising these frameworks, the POPIA rules, or the rules of any similar information privacy law such as India's Personal Data Protection Bill (PDPB), the European General Data Protection Regulation (GDPR), and the California Consumer Privacy Act of 2018 (CCPA), are encapsulated into the business rule engine of a technology system prototype developed to ease the enforcement of information privacy laws. In this study, the design science methodology, complemented by Agile software development methodology and the object-oriented programming paradigm was used to design and develop the artefactual prototype. Hence, the principal value proposition of this prototype is to test how best to enforce information privacy rules and guidelines into technology systems used in handling personally identifiable information (PII).

9.2 Research Significance

The significance of this study primarily bears on its contribution to the body of knowledge in the domain of information privacy and privacy engineering. Furthermore, it has significance for organisational practitioners of information privacy, both technical and non-technical. It is expected that the outcome of this study will:

- **With regard to the body of knowledge** – add to the existing literature on the use of software objects to enforce information privacy control within information systems by adapting a metadata inference model for just-in-time privacy (JIT) and for the creation of context-aware privacy zones (CAP) using software objects. It takes theoretical privacy concepts to a more practical dimension and improves the effectiveness and utility of IT artefact(s) used to enforce information privacy.
- **With regard to industry practitioners** – recommend a practical and theoretical model to ease the implementation of information privacy rules as configurable software objects in information systems used in handling personally identifiable information. It will also provide guidance on how organisations can use software objects to enforce information privacy controls in a manner which will not impede their business opportunities or aggrieve their customers.

- **In terms of privacy engineering –** this study proposes a reusable contextual design model and a POPIA rule engine prototype through which POPIA rules, or the rules of any similar information privacy law, such as India’s Personal Data Protection Bill (PDPB), or the European General Data Protection Regulation (GDPR), can be encapsulated into business software objects or technology systems to ease enforcement of compliance with information privacy laws. The artefact created extends organisational capabilities and problem solving boundaries surrounding information privacy control and compliance by providing intellectual as well as computational tools for the purpose.

To develop such a model, Van Vuuren (2015) suggests that POPIA compliance requires a focused, systematic, and formalised approach to the management of information. However, many companies have not taken this aspect seriously in the past, thus giving rise to the research objectives, questions, and problem, which were addressed in this study.

9.3 Recap of the Research Objectives, Question, and Problem

This study was initiated to answer the following primary research question: How can the POPIA regulations be implemented as software objects used to enforce information privacy and compliance in organisations? This question emanates from the main research problem identified from the review of literature concerning the rapid adoption of technology by organisations and the impact it has on information privacy and regulations. From the in-depth literature review, it was observed, primarily, that the rapid technology adoption is opening doors of opportunity to organisations on the one hand. On the other hand, it is creating tremendous information privacy and compliance risks with respect to the personally identifiable information residing in systems owned and managed by these organisations.

This view was supported by Bélanger and Crossler (2011), who conclude that the advances in information technology have greatly expanded opportunities as well as technical solutions for organisations to address information privacy concerns. This thus sets the stage for information system researchers to take a leading role in the practical implementation of technology solutions to mitigate information privacy concerns. An example is the initiative undertaken in this study.

Bélanger and Crossler (2011) further observe that information privacy regulators are doing everything they can to keep up, but as the pace of technology evolution accelerates, regulators continue to fall behind. Yet, according to Ernst and Young (2013), regulations remain a useful tool to improve information privacy protection. Hence, globally, many countries and regional authorities have passed legislation aimed at enforcing information privacy protection by organisations operating within their jurisdiction. Similarly, in South Africa, and in the context of this study, the South African government passed the Protection of Personal Information Act (POPIA) in 2013 and to date (eight years later), organisations are still struggling to enforce the provisions of the act.

This background led to the main problem of this study: to formulate a coherent approach for companies to enforce the provisions of the POPI Act, in particular, or any other similar information privacy legislation in technology systems responsible for the handling of personally identifiable information. To solve this problem and to answer the primary research questions highlighted here, this study addressed the following sub-questions:

1. How can the common vocabulary of the POPIA regulations be expressed as machine-interpretable instructions?
2. Which software architectural synthesis can best meet the organisational goal of POPIA compliance?
3. How should individual privacy concerns be validated against the POPIA regulatory controls within an organisation?

The foregoing questions linked to the corresponding research objectives, as listed below:

1. To facilitate the translation of the POPIA requirements into machine-interpretable language.
2. To determine the best design pattern to meet the technical and operational requirements for the POPIA compliance in organisations.

3. To formulate a model to use for the validation and verification of personally identifiable information against the POPIA regulations.

9.4 Meeting the Research Objectives

Based on the research objectives created for this study, and the use of the design science research methodology, the research process culminated in a practical model or artefact (the POPIA rule engine prototype). With reference to the design, implementation and evaluation of the live prototype, it can be concluded that:

1. The objective to facilitate the translation of the POPIA requirements into machine-interpretable language was met. To justify this conclusion:

- Based on the conceptual and design framework developed for this study, the research objective to translate POPIA requirements into machine-interpretable language was met, by using the UML use case ontology language; first, as a conceptual construct to represent POPIA as a knowledge domain and second, as a tool for modelling and representing the POPIA compliance domain as software objects encapsulating POPIA rules.

- From the implementation of the practical design principles of the study to build the POPIA rule engine prototype, using the Agile software development methodology and the object-oriented programming paradigm. POPIA rules were constructed as software objects or UML class diagrams with methods (functions) and properties (attributes) to visualise POPIA as an object-oriented system. Lastly, the POPIA UML sequence diagram was developed to show the flow and inner working of the POPIA rule engine prototype.

2. The objective of using the best design pattern to meet the technical and operational POPIA compliance requirements was also met in this study.

- To justify the conclusion that this objective was met, it is argued that the forward-chaining business rules execution method was used. It served, first, as a technical architectural pattern to represent POPIA rules as declarative statements within a Rete-based rules inference engine. Second, from an operational perspective, the inference engine benefits from a knowledge base that is metadata driven and reactionary as it can infer and deduce conclusions based on actual data, in this case personally identifiable data.

3. The objective of formulating a model to use for the validation and verification of personally identifiable information against POPIA regulations was also met.

- To justify this point, it is argued that, through the POPIA rule engine prototype developed, and using the business rules approach and the forward-chaining business rules execution method, POPIA rules were represented as configurable software objects. This is a Rules class that is made of three sub-components namely: Rule.Conditions, Rule.Actions and Rule.Exceptions all within a RulePart. This model was adapted to one of the use cases called Set User Preference Object to demonstrate how the verification attributes can be set at run time; how to specify the conditions to be met for validation of the rules; how the action is executed; and finally, how to handle exceptions in the rule if any occur.

9.5 General Conclusions

In conclusion, this study was initiated to answer the fundamental question: What is the best way to enforce the POPIA privacy rules and guidelines in technology systems responsible for the handling of personal indefinable information? Drawing from the discussion of the findings and observations recorded from the evaluation of the live prototype in Chapter 8, the conclusion is outlined in the following five (5) points for both public and private organisations to implement:

1. Adopt a formal and systematic business rule-implementation approach based on a business rule development life cycle (RDLC) methodology to manage the rules life cycle efficiently from rule creation to retirement (end of useful life of the rule).

2. Employ a standardised rules-based modelling technique with formal notations to represent POPIA business rules both visually and programmatically. It is of interest to develop and standardise privacy-specific notations to represent concepts within the privacy domain of knowledge.

3. Deploy a suitable business rule algorithm (engine) that complies with the von Halle (2002) STEP principle within the technology system responsible to handle the data, resulting in what can be referred to as just-in-time privacy (JIP).

4. Extend the boundaries of privacy beyond the Altman (1975) privacy regulation theory to cover the cloud (private and public multi-tenancy) and third-party hosted- or managed data.

5. Extend the context of privacy to include information privacy regulations (laws) and data classification, creating what can be referred to as context-aware privacy (CAP) or privacy compliance zone. It is of interest to further studies to break up the world into different privacy compliance zones.

In summary, if an organisation implements these five (5) points as highlighted here, then the POPIA rules and guidelines can effectively be enforced in the technology system handling personal identifiable data.

In addition to these general conclusions, the following further conclusions and remarks from this study are noteworthy:

First, leveraging on the UML ontology language underpinning this study, it was easy to identify and describe the roles of the main actors of the POPIA rule engine prototype. Drawing from the actors and their roles, it was easy to synthesise the main use case diagrams showing the system-user interaction points, as illustrated in **Figure**

7.2, showing the relationship between the actors and the sub-systems of the prototype. Similarly, a detailed use case was developed for all the sub-systems, as illustrated in **Section 7.2,** which deals with the implementation of the POPIA rule engine prototype. Drawing from the Data Subject vs Enforce POPIA Rule engine illustration in **Figure 7.2** and the detailed use cases implemented **Appendix B**, a better appreciation was established of the POPIA rule engine.

Hence, with the UML ontology, it was easy to model the POPIA rule engine prototype to show its structure and behaviour both at process level and at technical level with the class, sequence, and use case artefacts. Judging from the ease with which the UML ontology principles were adapted to develop the POPIA rule engine prototype in the different sections of this study, it can be concluded that applying the same UML ontology principles will help to create similar rule engine prototypes for other international information privacy laws, such as India's PDPB and the European GDPR.

Second, regarding which business rules approach to implement, this study concluded that, based on business agility requirements and the need for frequent changes to the business rules logic, a declarative and forward-chaining business rule methodology will be the most suitable. The rationale is drawn from the observation of the POPIA rule engine prototype. It is seen in this prototype that the forward-chaining business rules execution method adapts well with the wizard-like interface designed for the POPIA rule engine prototype and the sequential nature of the rule configuration, which is based on an event condition rule-implementation model.

9.6 Contribution of the Study

The outcomes recorded in this study are expected not only to add to the body of knowledge but also to influence the practices of both public and private sector organisations, as well as information privacy practitioners in the following ways:

1. This study provides guidance on how companies can use software objects to enforce information privacy regulations in general. More specifically, it shows how companies, who are required by law to comply with the South African Protection of Information Act (POPIA), can implement said Act in a manner that does not jeopardise their business opportunities or aggrieve

their customers.

2. The artefacts created from this study also extend the boundaries of organisational capabilities and problem solving by providing intellectual as well as computational tools to organisations interested in using business rule algorithms and design science best practices in building and integrating their business rules into the technology systems used to handle personally identifiable information.

3. In terms of the design science research methodology used in this study, the output of this research contributes as follows to the steps of the design science research methodology as proposed by Vaishnavi and Kuechler (2005) and presented in **Table 9.1**

Table 9.1 Research Contribution as per Design Science Guidelines
(Source: Peffers et al., 2008)

No.	DSR Phases	Contributions of this study to DSR
1	Problem identification and motivation	• Analysis of the problem domain using the Regionel (2015) four (4) broad Steps model to build the POPIA Conceptual Framework as the main output of this step.
2	Solution design (design artefact)	• Utilisation of the Rauch (2012) model to build the design framework with the POPIA conceptual framework serving as the main input. • Utilisation of the design framework to develop the POPIA framework.

		• Implementation of the POPIA framework by building and testing the POPIA Prototype
3	Evaluation and demonstration	• Observation and evaluation of the life POPIA prototype using the von Halle (2002) STEP principles

Based on Table 9.1, this study contributed to the three phases of the DSR research process with most of the contribution visible in the solution design (design artefact) phase.

9.7 Implication of Research

In terms of implications, notably, drawing from the specific conclusions regarding the research objectives, and also tapping into the general conclusions recorded in this study, it can be seen that this study has significant implications.

To the research community: This research project will take the theoretical privacy concepts into a more practical dimension and will improve the effectiveness and utility of IT artefacts used to enforce information privacy within organisations in the public and private sector.

To the community of information privacy industry practitioners globally: This study is trend setting and will ease their compliance journey by their adopting the processes and methodologies utilised in this study to develop the POPIA rule engine prototype in order to develop similar rule engines and prototypes for other international information privacy legislations, such as India’s Public Data Protection Bill, and the European General Data Protection Regulation.

9.8 Recommendations for Future Research

Considering that compliance with information privacy laws and regulations is an ongoing challenge for many organisations and that the technologies utilised to handle personally identifiable information are evolving constantly and mostly reside in the cloud, it is recommended that further research be conducted to build globally acceptable and standardised privacy tags and notations. Such tags and notations could be similar to Hyper Text Markup Language (HTML) 5 notations and tags, or the Unified Markup Language (UML) 2.5 tags and notations. This approach will change the trajectory of information privacy compliance and privacy engineering significantly.

This is especially significant in the present era of cloud and cyberspace where data is generated at an accelerated pace and resides in different public and private cloud environments across the globe, often far from where the data subject resides. More specifically, this approach has the potential to improve the general compliance with information privacy laws and to create privacy-aware zones within the cloud and cyberspace. To industry practitioners, such as system designers and software developers practising the disciplines of privacy engineering / software engineering, it will help to simplify and standardise practices and ultimately, to build systems that are privacy aware from inception.

In addition, this study recommends that any organisation looking at implementing the POPIA-styled prototype, should adopt a formal rules governance process and a rule life cycle methodology. In conclusion, this study raises the question of how to delineate which rules to implement in the rule engine and which rules to implement in the application programming logic in the context of protecting information privacy and PII data. To answer this question, further research will be required.

Finally, this study recommends a separate external evaluation of the conceptual framework and design framework developed in this study to allow for more generalisation and reuse.

REFERENCES

Abrahamson, P., Salo, O., Ronkainen, J., & Warsta, J. (2002). Agile software development methods: Review and analysis. *VTT Publications*, 112. http://www.vtt.fi/inf/pdf/publications/2002/P478.pdf

Agile Alliance (2020). Agile 101. (Accessed 1 February, 2020). https://www.agilealliance.org/agile101/

Akter, S., Wamba, S. F., Gunasekaran, A., Dubey, R., & Childe, S. J. (2016). How to improve firm performance using big data analytics capability and business strategy alignment? International Journal of Production Economics, 182, 113–131. https://doi.org/10.1016/j.ijpe.2016.08.018

Allen, A. L. (2016). Penn Law: Legal Scholarship Repository Protecting One ’ s Own Privacy in a Big Data Economy protecting one ’ s own privacy. https://scholarship.law.upenn.edu/cgi/viewcontent.cgi?article=2718&context=faculty_scholarship

Al-Slais, Y. (2020). Privacy engineering methodologies: A survey. In *2020 International Conference on Innovation and Intelligence for Informatics, Computing and Technologies (3ICT).* https://ieeexplore.ieee.org/document/9311949/

Alter, S. (2008). Defining information systems as work systems: Implications for the IS field. *European Journal of Information Systems, 17*(5), 448–469.

Altman, I. (1975). *The environment and social behavior*. Brooks/Cole. https://www.scirp.org/(S(lz5mqp453edsnp55rrgjct55))/reference/ReferencesPapers.aspx?ReferenceID=1358927

Agre, P. (1997). Introduction. In: Agre, P., Rotenberg, M. (Eds.), Technology and Privacy: The New Landscape. MIT Press, Cambridge, MA, pp. 1±28.

Ambler, S. W. (2005). UML class diagrams. *The elements of UML™ 2.0 style* (pp. 47–72). Cambridge University Press. https://doi.org/10.1017/CBO9780511817533.006

Barbosa, P., Brito, A., & Almeida, H. (2020). Privacy by evidence: A methodology to develop privacy-friendly software applications. *Information Sciences, 527*(1), 294–310. https://doi.org/10.1016/j.ins.2019.09.040.

Baskerville, R. L., Myers, M. D., & Yoo, Y. (2020). Digital first: The ontological reversal and new challenges for information systems research. MIS Quarterly, 44(2), 509–523. https://doi-org.uplib.idm.oclc.org/10.25300/MISQ/2020/14418

Bayswater (2016, December 15). What is compliance engineering? EMC-Bayswater https://www.emcbayswater.com.au/blog/certifications/what-is-compliance-engineering/

Bednar, K., Spiekermann, S., & Langheinrich, M.(2021). Engineering Privacy by Design: Are engineers ready to live up to the challenge. *The Information Society*, 35(3),122–142. https://doi.org/10.1080/01972243.2019.1583296

Bélanger, F., & Crossler, R. (2011). Privacy in the digital age: A review of information privacy research in information systems. *MIS Quarterly*, 35(4), 1017–1041. doi:10.2307/41409971

Bell, D. (2003). UML basics: An introduction to the Unified Modeling Language IBM. (Accessed 5 February, 2020). https://cs.nyu.edu/~jcf/classes/g22.2440-001_sp06/handouts/UMLBasics.pdf

Black, J., & Steel, J. (2017). Privacy developments: Private litigation, enforcement actions, and settlements. *The Business Lawyer*, 73(1), 177–190. (Retrieved 31 July, 2021). https://www-jstor-org.uplib.idm.oclc.org/stable/26419196

Blackwell, A.H. & Manar, E. P. (2015). *UXL Encyclopaedia of science*. (2nd ed.). UXL.

Biselli, T., & Reuter, C. (2021). On the Relationship Between IT Privacy and Security Behavior: A Survey Among German Private Users. Lecture Notes in Information Systems and Organisation. https://doi.org/10.1007/978-3-030-86797-3_26

Boell, S. K. & Cecez-Kecmanovic, D. (2015). What is an information system? In *Proceedings of the Annual Hawaii International Conference on System Sciences*, 2015 (March), 4959–4968 . https://doi.org/10.1109/HICSS.2015.587

Breaux, T. (2014), Privacy engineering: Examples of system design strategy. (Retrieved August 1, 2015). http://csrc.nist.gov/news_events/privacy_workshop_040914/breaux_nist_pew_07apr14.pdf.

Brenton, M. (1964). The privacy invaders. New York: Coward-McCann.

Brown, K. E., (2020), The Fourth Industrial Revolution: Will Africa be ready?. ACET. (Retrieved 1 February, 2021). https://acetforafrica.org/publications/policy-briefs-and-discussion-papers/the-fourth-industrial-revolution-will-africa-be-ready/?gclid=Cj0KCQiAvbiBBhD-ARIsAGM48bxv0Dxx8b4UXj65pwadaITmI1toFBWfnnoSzuYhomUhRLFzjBgc4CoaAhHHEALw_wcB

Brocke, J. von, Hevner, A., & Maedche, A. (2020). Introduction to design science research. In *Design science research cases*. 978-3-030-46780-7. pp.1–13. https://doi. 10.1007/978-3-030-46781-4_

Bu, F., Wang, N., Jiang, B., & Liang, H. (2020). Privacy by Design implementation: Information system engineers' perspective. *International Journal of Information Management*, *53*(10). https://doi.org/10.1016/j.ijinfomgt.2020.102124.

Burman, A. (2020). Will India's proposed Data Protection Law protect privacy and promote growth? (pp. 3-8, Rep.). Carnegie Endowment for International Peace. (Retrieved 4 August, 2021). http://www.jstor.org/stable/resrep24293.4

Campbell, J. (2007). Lecture 9, Part 1: Modelling Interactions. University of Toronto. (Retrieved on 12 February, 2020). http://www.cs.toronto.edu/~nn/csc340h/winter07/lectures/w10/L9-part1-6up.pdf

Canedo, E. D., Do Vale, A. P. M., Patrão, R. L., de Souza, L. C., Gravina, R. M., Dos Reis, V. E., Mendonça, F. L. L., & de Sousa, R. T. (2020). Information and communication technology (ICT) governance processes: A case study. *Information (Switzerland), 11*(10), 1–28. https://doi.org/10.3390/info11100462

Cavoukian, A. (2009). Privacy by design—the 7 Foundation Principles. Online: https://www.ipc.on.ca/wp-content/uploads/ Resources/7foundationalprinciples.pdf. Accessed: 2019-12-09

Centre for International Governance Innovation. (2018). Data governance in the digital age (pp. 54–69, Rep.). Centre for International Governance Innovation. (Retrieved 31 July, 2021). http://www.jstor.org.uplib.idm.oclc.org/stable/resrep26128.6

Chatterjee, R., & Roy, S., (2017). Cryptography in cloud computing: A basic approach to ensure security in cloud. *International Journal of Engineering Science and Computing, 7*(5), 11818–11821. http://ijesc.org/.

Cimpanu, C. (2018, December 10). *Google+ hit by second API bug impacting 52.5 million users.* ZDNET. https://www.zdnet.com/article/google-hit-by-second-api-bug-impacting-52-5-million-users/

Cockburn, A. (1997). Goals and Use Cases. *Journal of Object-Oriented Programming, 10* (5), 35–40.

Coelho, M.D, Vasconcelos, A., & Sousa, P. (2021). Privacy by design enterprise architecture patterns. INESC-ID, Instituto Superior Tecnico. https://www.scitepress.org/Papers/2021/104735/104735.pdf

Collins, C. S., & Stockton, C. M. (2018). The Central Role of Theory in Qualitative Research. International Journal of Qualitative Methods. https://doi.org/10.1177/1609406918797475

Computting.co.uk (2019, April 4). *540 million Facebook records exposed by app developers on insecure AWS server*. Computing.co.uk. https://www.computing.co.uk/news/3073677/540-million-facebook-records-exposed-by-app-developers-on-insecure-aws-server

Conger, S., Pratt, J., & Loch, K. (2013). Personal information privacy and emerging technologies. *Information Systems Journal.* 23. https://doi.org/10.1111/j.1365-2575.2012.00402.x.

Clifton C. (2009) Privacy metrics. In: Liu L., & Özsu, M.T. (eds.) *Encyclopedia of database systems.* Springer. https://doi.org/10.1007/978-0-387-39940-9_272

Creswell, J. (2009). *Research design: Qualitative quantitative and mixed methods approaches*. (3rd ed.).Sage.

Culnan, M. J., & Bies, R. J. (2003). Consumer privacy: Balancing economic and justice considerations. *Journal of Social Issues*, *59*(2), 323–342.

Cronan, B. (2014). Have we lost control of our online privacy? Americans think so. (Retrieved 25 March, 2015). http://www.csmonitor.com/Technology/Tech/2014/1112/Have-we-lost-control-of-our-online-privacy-Americans- think-so.

Dennedy, M.F., Fox, J., & Finneran, T. (2014). A vision of the future: The privacy engineer's manifesto. In *The privacy engineer's manifesto*. Apress. pp. 299–320. https://library.oapen.org/handle/20.500.12657/28156

Deloite. (2021, July 21). POPIA Go Live : What should your organisation be managing in the first 100 days post POPIA compliance. Deloite. https://www2.deloitte.com/za/en/pages/risk/articles/popia-go-live.html

Dias Canedo, E., Toffano Seidel Calazans, A., Toffano Seidel Masson, E., Teixeira Costa, P. H., & Lima, F. (2020). Perceptions of ICT practitioners regarding software privacy. *Entropy, 22*(4), 429. https://doi-org.uplib.idm.oclc.org/10.3390/e22040429

Diksha, S. (2020). Organisation: Meaning, process and principles. (Retrieved 1 February, 2020). https://www.businessmanagementideas.com/organisation/organisation-meaning-process-and-principles/3430

Dourish, P. (2001). *Where the action is: The foundation of embodied interaction*. MIT Press. pp.1–245.

Edeki, C. (2015). Agile software development methodology. *European Journal of Mathematics and Computer Science, 2*(1), 22–27.

Educba.com.(2019). Chapter 1. Difference Between Forward Chaining and Backward Chaining. (Retrieved on 2 February, 2020). https://www.educba.com/forward-chaining-vs-backward-chaining/

Elprama, S., Haans, A. l., & De Kort, Y. (2011). Relation between privacy and place attachment in student housing. In *9th Biennial Conference on Environmental Psychology*. Eindhoven, Netherlands.

EFF (2015). New technologies are radically advancing our freedoms but they are also enabling unparalleled invasions of privacy. (Retrieved 26 March, 2015). https://www.eff.org/issues/privacy.

Ernst & Young. (2013). Insights on governance, risk and compliance. (Retrieved 1 August, 2015). http://www.ey.com/Publication/vwLUAssets/Privacy_trends_2013_The_uphill_climb_continues/$FILE/Privacy%20trends%202013%20%20The%20uphill%20climb%20continues.pdf

Ey.com. (2014). Privacy protection in the age of technology. (Retrieved 18 August, 2105). http://www.ey.com/Publication/vwLUAssets/EY_Privacy_trends_2014:_Privacy_protection_in_the_age_of_technology/$FILE/EY-Insights-on-GRC- Privacy-trends-2014.pdf.

Fennessy, C. (2019). Privacy engineering: The what, why and how. (Retrieved 1 November, 2019). https://iapp.org/news/a/privacy-engineering-the-what-why-and-how/

Fenghua, L., Hui. L., Ben, N., & Jinjun C.(2019). Privacy Computing: Concept, Computing Framework, and Future Development Trends Engineering, 5(6), 1179–1192. https://doi.org/10.1016/j.eng.2019.09.002.

Fowler, M. (n.d.). Sequence diagram tutorial. UML Distilled. http://csis.pace.edu/~marchese/CS389/L9/Sequence%20Diagram%20Tutorial.pdf

GAN Integrity. (2020). Culture of compliance. (Retrieved 1 February, 2020) https://www.ganintegrity.com/compliance-glossary/culture-of-compliance/

Gerber, N., Gerber, P., & Volkamer, M. (2018). Explaining the privacy paradox: A systematic review of literature investigating privacy attitude and behavior. Computers&Security,77(2),226–261. https://doi.org/10.1016/j.cose.2018.04.002.

Giles, J. (2020). When is the POPIA deadline in South Africa? (Retrieved on 20 February, 2019).https://www.michalsons.com/blog/when-is-the-popia-deadline-in-south-africa/39672

Gonzalez, R. (2012). Validation and design science research in information systems. (Retrieved 23 August, 2105). http://www.igi-global.com/chapter/validation-design-science-research-information/63275

Gougeon, A. (2003). Everything you ever wanted to know about business rules. 1–16. (Retrieved on 31 January, 2021). http://ww.bptrends.com/publicationfiles/07-03 ART Everything About Bus Rules - Gougeon.pdf

Greenleaf, G. (2012). Global data privacy laws: 89 Countries, and Accelerating. *Privacy Laws & Business International Report.*

Gregor, S., & Hevner, A. R. (2013). Positioning and presenting design science research for maximum impact. *MIS Quarterly, 37*(2), 337–356.

Graham, I. (2006). *Business rules management and service oriented architecture*. John Wiley and Sons.

Gray, J., & Shenoy, P. (2000).Rules of thumb in data engineering. In *Proceedings of 16th International Conference on Data Engineering* (Cat. No.00CB37073), 2000. pp. 3–10, doi: 10.1109/ICDE.2000.839382.

Gundu, T. (2019). Big data, big security, and privacy risks: Bridging employee knowledge and actions gap. *Journal of Information Warfare,* 18(2), 15–30. (Retrieved 31 July, 2021). https://www-jstor-org.uplib.idm.oclc.org/stable/26894668

Gurses, S., & Del Alamo, J. M. (2016). Privacy engineering: Shaping an emerging field of research and practice. *IEEE Security and Privacy,* 14 (2016), 40–46.

Hevner, A. R. (2007). A three cycle view of design science research. *Scandinavian Journal of Information Systems, 19*(2), 87–92.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. MIS Quarterly, 28(1), 75–105. https://doi-org.uplib.idm.oclc.org/10.2307/25148625.

Hu, H. (2013). Conceptualizing and measuring information privacy. (Retrieved 20 July, 2015). https://networkedprivacy2013.files.wordpress.com/2013/03/xu_cscw_privacy_workshop.pdf

Holmes. A. (2021). 533 million Facebook users' phone numbers and personal data have been leaked online. *Business Insider*. (Retrieved 23 July, 2021). https://www.businessinsider.co.za/stolen-data-of-533-million-facebook-users-leaked-online-2021-4?r=US&IR=T

Hosein, G., & Altshuller, M. (2017). Privacy and security in a digital Age: An interview with Dr. Gus Hosein. *Harvard International Review*, 38(3), 67–71. (Retrieved 4 August, 2021). https://www.jstor.org/stable/26528687

Hosseini, M.B., Breaux, T.D., Slavin, R., Niu, J., & Wang, X. (2021). Analyzing privacy policies through syntax-driven semantic analysis of information types. Information and Software Technology, 138, 106608. https://doi.org/10.1016/j.infsof.2021.106608.

INCOSE (2010). A basic introduction to measurement concepts and use for systems engineering. INCOSE. (Retrieved 16 November, 2021). https://www.incose.org/docs/default-source/ProductsPublications/systems-engineering-measurement-primer---december-2010.pdf

ISO/IEC JTC. (2014) ISO/IEC 24744:2014 Software engineering – Metamodel for development methodologies.

Interaction Design Foundation. (2020). What is Prototyping? (Retrieved 1 February, 2020). https://www.interaction-design.org/literature/topics/prototyping

Jaikaran, C. (2016). Encryption: Frequently asked questions. (Retrieved 19 February, 2020). https://fas.org/sgp/crs/misc/R44642.pdf

Jain, A., & Mahajan, N. (2017). Introduction to cloud computing. *The Cloud DBA-Oracle*, 3 –10. https://doi.org/10.1007/978-1-4842-2635-3_1

Jeroen, V., R. (2014). Designing Privacy-by-Design. *Lecture notes in computer science,* 8319, 55–72. doi:10.1007/978-3-642-54069-1_4. ISBN 978-3-642-54068-4.

Johnson, R. E., & Deutsch, P. (1993). How to design frameworks. *Notes for OOPSLA ’93.* (Retrieved 12 February, 2020). http://web.cse.msu.edu/~cse870/Materials/Frameworks/how-to-design-fw-tutorial.ps

Kalloniatis, C., Mouratidis, H., Vassilis, M., Islam, S., Gritzalis, S., & Kavakli, E. (2014). Towards the design of secure and privacy-oriented information systems in the cloud: Identifying the major concepts. *Computer Standards & Interfaces, 36*(4), 759–775. https://doi.org/10.1016/j.csi.2013.12.010.

Kindler, E., & Krivy, I. (2011). Object-oriented simulation of systems with sophisticated control. *International Journal of General Systems*, 313–343.

Klopfer, P. H., and Rubenstein, D. I. (1977). The Concept Privacy and Its Biological Basis. *Journal of Social Issues*, (33:3), 52–65.

Kogut, P., & Cranefield, S. (2002). UML for ontology development. *The Knowledge Engineering Review*, 17, 61–64. https://doi.org/10.1017/S0269888902000358

Koneru, L. (2018). Using Design Science Research to Develop a Conceptual Solution for Improving Knowledge Sharing in a Virtual Workspace. ProQuest Dissertations and Theses, December, 1–194. http://ezproxy.lib.ucalgary.ca/login?url=https://search.proquest.com/docview/2209772213?accountid=9838%0Ahttp://ucalgary-primo.hosted.exlibrisgroup.com/openurl/01UCALG/UCALGARY??url_ver=Z39.88-2004&rft_val_fmt=info:ofi/fmt:kev:mtx:dissertation&genre=disse

Khosrowshahi, D. (2017, November 21). *2016 Data Security Incident.* Uber Newsroom. https://www.uber.com/en-CA/newsroom/2016-data-incident/

Krebs, A. (2006). Methods and software for a batch processing framework for wizard-based processes. *IFS Claims Patent Service.* https://patents.google.com/patent/US7757234B2/en

Laprie, J.C., & Randell, B. (2001). Fundamental concepts of dependability. *LAAS-CNRS: Tech. Report No. 1145*. http://www.cs.cmu.edu/~{}garlan/17811/Readings/avizienis01_fund_concp_depend.pdf (Accessed on 2 December, 2020).

Latour, B., & Woolgar, S. (1986). Laboratory life: The construction of scientific facts. (Retrieved 22 August, 2015). http://www.nuffieldfoundation.org/sites/default/files/files/Argumentation%20research%20summary(1).pdf

Le Grand., H.C. (2021). Building a culture of compliance. *QualityMag*. (Retrieved 5 February, 2020). https://www.qualitymag.com/ext/resources/files/white_papers/BuildingaCultureofCompliance-IBS.pdf

Leshner, R. (2012). Online privacy: How to control your personal data. (Retrieved 26 March, 2015). http://mashable.com/2012/05/21/online-personal-information-protect.

McAfee (2018). *Key Findings from our Survey on Identity Theft, Family Safety and Home Network Security.* McAfee. https://www.mcafee.com/blogs/privacy-identity-protection/key-findings-from-our-survey-on-identity-theft-family-safety-and-home-network-security/

Mai, J. E. (2016). Big data privacy: The datafication of personal information. *Information Society, 32*(3), 192–199. https://doi.org/10.1080/01972243.2016.1153010

Malhotra, N. K., Kim, S. S., & Agarwal, J. 2004. Internet users' information privacy concerns (IUIPC): The construct, the scale, and a causal model. *Information Systems Research*, *15*(4), 336–355.

Mania, N. (2020). A four-step guide to privacy engineering. (Retrieved 26 January, 2021). https://www.truata.com/2020/02/05/a-four-step-guide-to-privacy-engineering/

Martin, K.D., Kim, J.J, Palmatier, R.W., Steinhoff, L., Stewart, D.W., Walker, B.A., Wang, Y., & Weaven, S.K. (2020). Data privacy in retail. *Journal of Retailing, 96*(4),474–489. https://doi.org/10.1016/j.jretai.2020.08.003.

Martín, Y. S., & Del Álamo, J. M. (2017). A metamodel for privacy engineering methods. *CEUR Workshop Proceedings,* 1873(731711), 41–48.

Martin, K, D., & Murphy, P. E. (2016), "The Role of Data Privacy in Marketing," Journal of the Academy of Marketing Science, (published electronically September 22), DOI: 10.1007/ s11747-016-0495-4.

Mason, Q. (2005). 38.1 Introduction: The meaning of privacy. (Retrieved 26 March, 2021) https://constitutionallawofsouthafrica.co.za/wp-content/uploads/2018/10/Chap38.pdf

Manzo, V., & Bergamo, M. (2020). From information privacy to emergency privacy. *European Journal of Privacy Law & Technologies*, 2020(1), 83–97.

Maple, C., Epiphaniou, G., & Bottarelli, M.(2021). Trustworthy digital infrastructure for identity systems: Why should privacy matter to security engineers?. *Computer Fraud & Security*, 2021(6),6–11. https://doi.org/10.1016/S1361-3723(21)00063-4.

Margulis, S. (2003). On The Status and Contribution of Westin's and Altman's Theories of Privacy. *Journal of Social Issues*, 59. 411 - 429. https://doi.org/10.1111/1540-4560.00071.

Matuszewska, K., Sweeney, M. & Lubowicka, K. (2021, April 13). *What is PII, non-PII, and personal data?.* PIWIK. https://piwik.pro/blog/what-is-pii-personal-data/

McKendrick, J. (2016). Public cloud computing growing almost 50 percent annually, Cisco says. Forbes.

McGinnis, D. (2020), What is the Fourth Industrial Revolution? *Salesforce*. (Retrieved 1 February, 2021). https://www.salesforce.com/blog/what-is-the-fourth-industrial-revolution-4ir.

Microsoft (2019). What is cloud computing?. Microsoft (Retrieved 31 July, 2021). https://azure.microsoft.com/en-us/overview/what-is-cloud-computing/#benefits

Moore, A. (2008). Defining Privacy. *Journal of Social Philosophy*. 39(3), 411–428.,

Milà, M., & Abellán, F.(2019). General Data Protection Regulation. What is new?. *Revista Española de Medicina Nuclear e Imagen Molecular* (English Edition), 38(2), 69–71. https://doi.org/10.1016/j.remnie.2018.10.018.

Michalsons. (2019). Overlap between data protection laws | Global law - Michalsons. https://www.michalsons.com/focus-areas/privacy-and-data-protection/overlap-between-data-protection-laws

Mitra, R. (2020, April 24). *What is Facebook Libra Cryptocurrency?* [Most Comprehensive Guide]. https://blockgeeks.com/guides/understanding-facebooks-cryptocurrency-libra/

MITRE (2013). Privacy systems engineering. (Retrieved 26 March, 2015). http://www.mitre.org/publications/systems-engineering-guide/enterprise-engineering/engineering- information intensive-enterprises/privacy-systems-engineering

Nuseibeh, B., & Easterbrook, S. (2000). Requirements engineering: A roadmap (PDF). ICSE '00. In *Proceedings of the Conference on the future of Software Engineering.* pp. 35–46. CiteSeerX 10.1.1.131.3116. doi:10.1145/336512.336523. ISBN 1-58113-253-0.

OECD, C. (2021). *Personal Data Protection at the OECD.* OECD. https://www.oecd.org/general/data-protection.htm

Padmanabhan B. (2012). Principles of software engineering. (Retrieved 5 February, 2020). https://people.eecs.ku.edu/~hossein/Teaching/Fa16/810/Readings/UML-diagrams.pdf.

Pavlou, P. (2011). State of the information privacy literature: Where are we now and where should we go? *MIS Quarterly, 35*(4), 977–988. doi:10.2307/41409969

Peffers, K., Tuunanen, T., Gengler, C., Rossi, M., Hui, W., Virtanen, V., & Bragge, J. (2006). The design science research process: A model for producing and presenting information systems research. In *Proceedings of the 1st International Conference on Design Science Research in Information Systems and Technology,* (pp. 83–106).

Perera, C., Barhamgi, M., Bandara, A. K., Ajmal, M., Price, B., & Nuseibeh, B.(2020). Designing privacy-aware internet of things applications. *Information Sciences,* 512, 238–257.https://doi.org/10.1016/j.ins.2019.09.061.

Petronio, S. (2010). Communication privacy management theory: What *Do We Know About Family Privacy Regulation?* (Retrieved 23 July, 2015). http://www.pewinternet.org/Reports/2010/Mobile-Access-2010.aspx

Pfitzmann, A. & Hansen, M. A. (2010). Terminology for talking about privacy by data minimization: Anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management. Tech. Univ. Dresden 2010.

Pew Research Center (2020). Online privacy and security statistics you should know in 2020. Ledgeview. (Retrieved 23 July, 2020). https://ledgeviewpartners.com/blog/online-privacy-and-security-statistics-you-should-know-in-2020/

Protection of Personal Information Act. (2013). *Government Gazette (No. 37067)* https://www.gov.za/sites/default/files/gcis_document/201409/3706726-11act4of2013protectionofpersonalinforcorrect.pdf

Rainie, L., & Anderson, J. (2014). The future of privacy. (Retrieved 23 March, 2015). http://www.pewinternet.org/2014/12/18/future-of-privacy/.

Ravitch, S. M., & Riggan, M. (2017). *Reason & rigor: How conceptual frameworks guide research* (2nd ed.).: SAGE Publications.

Regoniel, P. (2015). Conceptual Framework: A Step by Step Guide on How to Make One [Blog Post]. In Simply Educate Me. https://simplyeducate.me/2015/01/05/conceptual-framework-guide/

Roberg-Perez, S. (2017). The future is now: Biometric information and data privacy. *Antitrust*, 31(3), 60–65.

Royce, W. W. (1970). Managing the development of large software systems: concepts and techniques (PDF). ICSE'87. In *Proceedings of the 9th international conference on Software Engineering*. pp. 1–9.

Sanders, J. (2016). Defining terms: Data, information and knowledge. In *Proceedings of 2016 SAI Computing Conference*, July 2016, 223–228. https://doi.org/10.1109/SAI.2016.7555986

SARS. (2020). Tax relief measures. 9Retrieved 1 February, 2020). https://www.sars.gov.za/Media/Pages/Tax-Relief-measures.aspx

Schalkoff, R.J. (1990), *Artificial intelligence: An engineering approach.* McGraw-Hill College, 1990.

Schulze, E. (2019). Everything you need to know about the Fourth Industrial Revolution.(Retrieved13February,2021). https://www.cnbc.com/2019/01/16/fourth-industrial-revolution-explained-davos-2019.html

Scribante, N. P., Pretorius, L., & Benade, S. (2019). The design of a research tool for conducting research in a complex socio-technical system. *South African Journal of Industrial Engineering, 30*(4), 143–155. https://doi.org/10.7166/30-4-2191

Slane, A. (2018). Information brokers, fairness, and privacy in publicly accessible information. *Canadian Journal of Comparative and Contemporary Law*, 4, 249–292.

Slepchuk, A.N., & Milne, G.R.(2020). Informing the design of better privacy policies. *Current Opinion in Psychology*,31(1),89–93. https://doi.org/10.1016/j.copsyc.2019.08.007.

Shapiro, B., & Baker, C. R. (2001). Information technology and the social construction of information privacy. Journal of Accounting and Public Policy, 20(4–5), 295–322. https://doi.org/10.1016/S0278-4254(01)00037-0

Singla, S., & Singh, J. (2013). Cloud data security using authentication and encryption technique. *Global Journal of Computer Science and Technology 13*(3), 1–6.

Smith, H. J., Milberg, J. S., & Burke, J. S. (1996). Information privacy: Measuring individuals' concerns about organisational practices. *MIS Quarterly, 20*(2), 167–96.

Smith, H., Dinev, T., & Xu, H. (2011). Information privacy research: An interdisciplinary review. *MIS Quarterly, 35*(4), 989–1015. https://doi.org/10.2307/41409970

SolutionsIQ. (2019). What is business agility & why it matters for your enterprise. 9Accessed on 15 June, 2020).

Sommerville, I. (2009). Software engineering (9th ed.). Addison-Wesley. https://www.studocu.com/row/document/comsats-university-islamabad/software-engineering/ian-sommerville-software-engineering-9th-edition-addison-wesley-2011/13935203

Stephens, K.P., & Sumner, J. P. (1996). Software objects : A new trend in programming and softw
are patents. 12 Santa Clara High Tech. L.J. 1 (Accessed on 12 June, 2019). https://digitalcommons.law.scu.edu/chtlj/vol12/iss1/1

Stringer, J. (2011, October). Protecting personally identifiable information. (Retrieved 15 March, 2019). https://www.sophos.com/en-us/medialibrary/pdfs/other/sophosprotectingpii.pdf

Swartz, M, P., & Solove, J. D. (2011). The PII problem: Privacy and a new concept of personally identifiable information. New York University Law Review, *86*(6), 1814–1894.

Szalvay, V. (2004). An introduction to Agile software development. 9Accessed on 1 February, 2018). http://www.danube.com/docs/Intro_to_Agile.pdf.

Taylor, D. (2015). The literature review: A few tips on conducting it. (Retrieved 23 August, 2015). http://www.writing.utoronto.ca/advice/specific-types-of-writing/literature-review

TechTarget. (2020). Information security (infosec) (Retrieved February 1, 2020). https://searchsecurity.techtarget.com/definition/information-security-infosec

Teravainen., T. (2020). Information security (infosec). TechTarget. (Retrieved 1 February, 2020). https://searchsecurity.techtarget.com/definition/information-security-infosec

Theodoulidis, B., & Youdeowei, A. (2000). Business rules: Towards effective information systems development. *Business Information Systems--Uncertain Futures*, November, 313–321.

Umhoefer, C.A.F. (2014). How EU data protection laws impact cross-border FCPA investigations. (Retrieved 28 March, 2015). https://www.dlapiper.com/en/global/insights/publications/2014/09/the-global-anticorruption- perspective-q3-2014/how-eu-data-protection-laws-impact.

Van Vuuren, J.J. (2015). Cybercrime: A serious challenge to the protection of personal information. (Retrieved 26 March 26, 2015). http://www.insurance-times.net/article/cybercrime-serious-challenge-protection-personal-information.

Vaishnavi, V., & Kuechler, W. (2005). *Design science research in information systems: Innovating information and communication technology*. CRC Press.

Valatkaite, I., & Vasilecas, O. (2004). On business rules approach to the information systems development. In Linger, H., Fisher, J., Wojtkowski, W. G., Zupančič, J., Vigo, K., & Arnold, J. (Eds). *Constructing the infrastructure for the knowledge economy*. pp.199–208 https://doi.org/10.1007/978-1-4757-4852-9_14

Vidgen, R. (2003). Requirements analysis and UML: Use cases and class diagrams. *Computing & Control Engineering,* 14(2), 12 – 17. https://doi-org.uplib.idm.oclc.org/10.1049/cce:20030202

Vimercati S..C.., Foresti S. (2011) Quasi-Identifier. In: van Tilborg H.C.A., Jajodia S. (eds) Encyclopedia of Cryptography and Security. Springer, Boston, MA. https://doi.org/10.1007/978-1-4419-5906-5_763

Vissser, A. (2021, June 28). How areday are you?, FIN24. https://www.news24.com/fin24/finweek/how-ready-are-you-20210628

Wagner, I., & Eckhoff, D. (2018). **Technical Privacy Metrics: A Systematic Survey**. ACM Computing Surveys (CSUR), vol. 51 (3), pp. 57:1-57:38

Wang, J. (2001). Object-oriented analysis. (Retrieved 16 July, 2019). https://www.umsl.edu/~sauterv/analysis/488_f01_papers/wang.htm

Wang, S.J., Liu, J., Shon, T., Vaidya, B., & Chen, Y.-S. (2015). Security and privacy information technologies and applications for wireless pervasive computing environments. *Information Sciences, 321*(2),147–149. https://doi-org.uplib.idm.oclc.org/10.1016/j.ins.2015.07.037

Warren, S. D., Brandeis, L. D., Review, H. L., & Dec, N. (1929). The Right to Privacy Today. *Harvard Law Review, 43*(2), 297. https://doi.org/10.2307/1330091

Westin, A. (1976). Privacy and freedom. (Retrieved 10 February, 2015). http://heinonline.org/HOL/LandingPage?handle=hein.journals/hulr6&div=23&id=&page=

Wieringa, R. (2013). Introduction to design science methodology. In *Proceedings of the REFSQ Doctoral Symposium.* pp.1–17. (Retrieved 15 March, 2019). https://wwwhome.ewi.utwente.nl/~roelw/microtutorial.pdf.

Wigmore, I. (2020). Fourth Industrial Revolution. (Retrieved 15 February, 2021). https://whatis.techtarget.com/definition/fourth-industrial-revolution

Wu, P.F., Vitak, J., & Zimmer, M.T. (2020). A contextual approach to information privacy research. *Journal of the Association for Information Science & Technology*, *71*(4), 485–490. https://ideas.repec.org/a/bla/jinfst/v71y2020i4p485-490.html

Zave, P. (1997). Classification of research efforts in requirements engineering. *ACM Computing Surveys*, 29(4), 315–321.

Zimmer, M. (2014). Mark Zuckerberg's theory of privacy. (Retrieved 10 February, 2015).http://www.washingtonpost.com/lifestyle/style/mark-zuckerbergs-theory-of-privacy/2014/02/03/2c1d780a-8cea-11e3-95dd-36ff657a4dae_story.html

APPENDIX A:
Privacy Research Literature Contributors

Contributors to the Information Privacy debate and highlights of their contributions

	Names of Contributors	**Key Privacy Concepts**	**Highlight of Contribution**
1.	Dias Canedo et al. (2020)	Privacy by design	Privacy requirements : Information privacy violation can be prevented if privacy requirements are properly elicited at the early stages of a system development process that exists both at the functional and non-functional requirements gathering phases of the SDLC.
2.	Papacharissi & Gibson (2011)	Privacy by design	Regrettably, many current systems and platforms still fail to protect user privacy because privacy is an afterthought of system design.
3.	Cavoukian et al. (2010)	Privacy by design	Seven (7) fundamental principles of PbD were proposed. The highlight of PbD rests on the principles that information privacy protection should be preventative rather than remedial and privacy should be embedded in the design of privacy systems and tools.
4.	Wu et al. (2019)	Privacy engineering	The need exists for a practical and innovative privacy enhancing framework and solutions.
5.	Bélanger & Crossler (2011)	Privacy engineering	Better engineering mechanism for privacy enforcement in technology systems. The opt-in and opt-out mechanism is not effective and context sensitive.
6.	Liu et al.(2011); Wu (2019)	Privacy engineering	The need exists for an alternative technical mechanism for information privacy enforcement.
7.	The National Institute of Standards. NIST(2020)	Privacy engineering	Definition of privacy engineering and guidelines for implementation. NIST information privacy framework as a tool to guide the protection of information privacy.
8.	Fennessy (2019)	Privacy engineering	Characterises privacy engineering as the technical side of privacy by which privacy considerations are integrated into privacy design.
9.	Martin & Del Alamo (2017)	Privacy engineering	Information privacy engineering is multidisciplinary, and thus subject to multiple reference frameworks and paradigms. Some are: social, legal, risk, or technical.
10.	Li et al. (2019)	Privacy standard	Most information privacy schemes are focused on relatively isolated software applications scenarios and technical points and there is a need for privacy standards in large data environments such as cloud, big data, social networks and cyberspace.
11.	Biselli & Reuter (2021),	Privacy standard	There is a need to grow the body of knowledge on the level of technology required to protect information privacy.

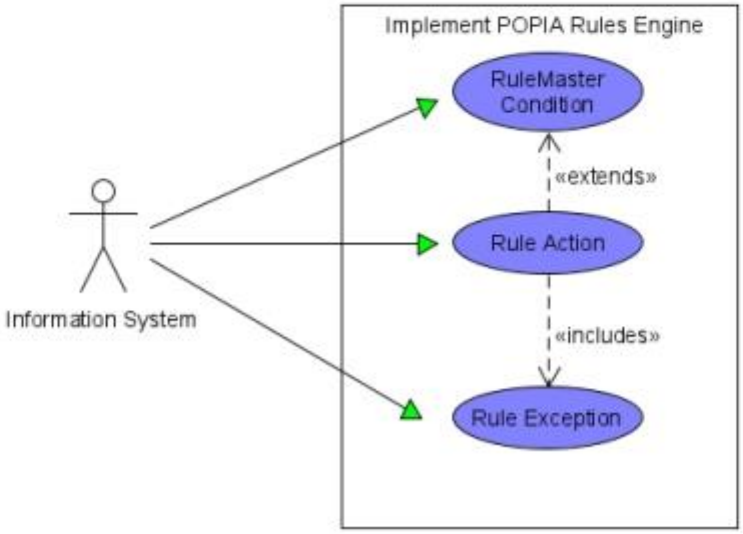
12.	Wu et al (2020)	Privacy concept	Advocates for a more contextual approach to information privacy with emphasis on the conditions and context guiding the disclosure of privacy.
13.	Slepchuk & Milne (2020)	Privacy concept	The goal of the overall improvement of information privacy through technology.
14.	Shaar (2010)	Privacy regulation	The legal and regulatory frameworks established to protect information privacy are not enough to ensure the protection of personal data in the rapidly growing information industry.
15.	Gerber et al. (2018)	Privacy paradox	A dichotomy exists between privacy attitude and privacy behaviour of data subjects, by which they advocate the importance of their privacy publicly and willingly give it away privately.
16.	Wu (2020)	Privacy paradox and Social networks	How to resolve the privacy paradox through technological design. Social network shifts and the collapse of privacy boundaries as social circumstances change.
17.	Conger (2020); Lomas (2019); Holmes (2021)	Data protection	That social networks are a threat to global data protection, leading to data leakages.
18.	Perera et at. (2020)	Internet of things (IoT) and Data protection	The growing ability of devices to connect with each other to collect and share data, which is a threat to privacy.
19.	Kalloniatis et al. (2013)	Cloud computing and Data privacy	A new generation of technology has invaded our lives positively providing a number of capabilities that have made our digital behaviour much easier than before.
20.	Microsoft (2020)	Cloud computing	Definition and models of cloud computing.
21.	Bellman et al. (2004),	Fair information practices	Origin and purpose of fair information practices.
22.	Slane (2018),	Fair information practices	The EU, Canada, and the US have, in recent times. grappled with the issue of what is fair information practice when collecting processing, transmitting, and storing data.
23.	Akter et al. (2016)	Privacy regulation and Big data	Businesses, assisted by advanced information and communication technologies, rely heavily on customer data and advanced technologies such as big data analytics to shape their products and services.
24.	Allen (2016)	Big data	Frames big data as a threat to data protection.

		and Data protection	
25.	ISO/IEC	Privacy standard	Proposes the ISO/IEC 27701 standard to manage information privacy risk.
26.	Teravainen (2020)	Information security	Proposes information security as one of the tools to manage information privacy.
27.	Cole (2015)	Information security	Information has become one of the biggest business assets in recent times and organisations across the world are scrambling to build a large pool of secured information to get competitive advantage over their competitors.

APPENDIX B:
POPIA Detailed Use Cases

Use Case 1

ID:	UC-1
Title:	Implement POPIA rule engine object
Description:	This use case examines how the POPIA rules are implemented within the information system used by both the organisation and the data subject to collect, share, process, and store personally identifiable information. Based on the theoretical framework and the POPIA rule engine prototype developed, business rules can be broken down into three rule parts: 1. Rule Condition: The Rule Condition is essentially a statement of the rule expressed in the syntax of some programming language. In the context of this study and the POPIA rules Prototype, the Rule Condition is expressed using the Extensible Markup Language (XML) syntax. 2. Rule Action: The Rule Action specifies the decision to be taken by the information system running the POPIA rule engine based on the evaluation of whether the Rule Condition is true or false. 3. Rule Exception: The Rules Exception is a special type of Rule Condition, which is triggered only when an unexpected error occurs when the POPIA rule engine is busy processing the prescribed Rule Condition. For example, The Rule Exception may be as simple as to abort the operation and send notification of failure to the parties involved. These three parts are associated together in the POPIA rule engine object and deployed in the information system to enforce POPIA compliance.
Primary Actor:	Information System
Preconditions:	POPIA rules (conditions, actions, exceptions) are translated as programming instructions using the UML programming language syntax.

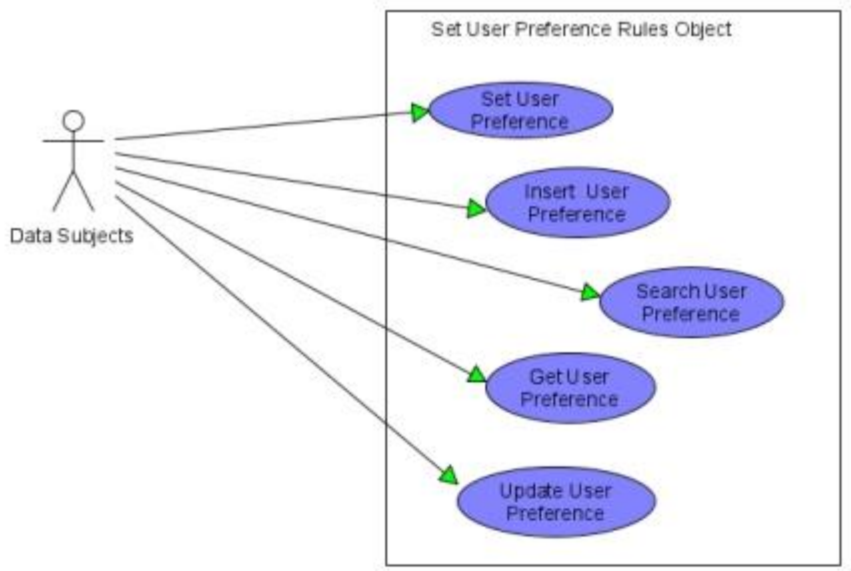
Post conditions:	The POPIA rule engine is implemented in the information system used to manipulate personally identifiable information (PII)
Main Success Scenario:	The UML interaction between the information system and the POPIA rule engine object is illustrated in **Figure 6.1**.  Implement POPIA Rules Engine RuleMaster Condition «extends» Rule Action «includes» Rule Exception Information System **Figure 1. Use Case: Data Subject vs Enforce POPIA Rule Engine** (Source: Author's own work) 1. All POPIA rule conditions coded as XML instructions within the rule engine object. 2. All POPIA rule actions coded as XML instructions within the rule engine object. 3. Exception conditions coded as XML instructions within the rule engine object 4. Rule engine is able to listen to the user preference constraints setup in the Set User Preference Rules Object. 5. The information system is able to manipulate data based on the rules conditions and rule actions setup in the POPIA rule engine 6. The POPIA rule engine is able to generate exceptions and pass them through as report entries.

Assumptions	The proposed privacy instruments are representative of privacy concerns in South Africa
Notes and issues	South African users might not agree with the privacy instruments and the dimensions captured
Frequency of use:	Used frequently by the information system to manipulate data and information
Status:	Prototype developed
Owner:	XXXXXX
Priority:	P1 - High

Use Case 2

ID:	UC-2
Title:	Set User Preference Rules object
Description:	This use case allows the data owners and the organisation to set their preferences on the POPIA rule engine to determine how PII data /information is collected, processed, shared, and stored by the organisation hosting and operating the information system. This use case leans heavily on Altman's (1975) privacy regulation theory, highlighting the need for data subjects to manage private information and disclosure based on their level of comfort, all within a privacy regulating mechanism (The POPIA rule engine prototype) Based on the theoretical framework developed for this study, two (2) distinct sets of preferences were identified as follows: 1. Data subject preferences: These are preferences set by either the end users or employees of the organisation hosting the information system or processing the personal identifiable data. 2. Organisation preferences: These preferences vary from organisation to organisation and from industry to industry. For instance, government organisations, such as Police and Intelligence Services will have different levels of allowance and approval to manage PII as opposed to non-governmental organisations, such as banks and insurance companies.

	This use case will also tap into the two (2) data privacy concern models proposed by Bélanger and Crossler (2011). Below is a list of the proposed privacy models and their dimensions of privacy. 1. CFIP (Concern for information privacy): four dimensions and 15 items with dimensions, those being collection of data, unauthorised secondary use of data, improper access to data, and errors in data 2. IUIPC (Internet user's information privacy concerns): three dimensions and 10 items with dimensions, those being control, awareness, and collection
Primary actor:	The organisation and the data owners, simply called the data subjects
Preconditions:	The data subjects (organisation and data owners) are aware of the conditions and rules guiding the legal handling and processing of personally identifiable information, as mandated by POPIA
Post conditions:	The data subjects have set up their user preferences on the Set User Preference Rules object
Main success scenario:	The main actors of this use case (The data subject and the organisation) will interact with the Set User Preference objects to capture and maintain their detailed information privacy preferences such as in terms of data owners: 1. To determine the personally identifiable data to reveal or to conceal. 2. To capture privacy concerns on the POPIA rule engine. 3. To verify whether privacy concerns have been captured. For the organisation, it is as follows: 1. To seek data subject permission on data items to be revealed or concealed. 2. To classify data based on sensitivity and security best practices. 3. To notify data subject and regulator of data violation. In summary the activities that the data subject carries out on the Set User

	Preference Object is maintenance in nature which is summarised in the use case in **Figure 2**.  Set User Preference Rules Object Set User Preference Insert User Preference Search User Preference Get User Preference Update User Preference Data Subjects **Figure 2 : Use Case: Set User Preference Rules Object** (Source: Author’s own work) 1. Data subject selects the Set User Preference Rules Object from the menu. 2. System displays list of User Preference Rules available for both users and organisations. 3. Data subjects selects one or more User Preference Rules and engine lists the different privacy dimensions for that rule. 4. Data subject configures validation criteria for that specific POPIA rule based on the user preference and privacy dimension. 5. Data Subject clicks Submit button. 6. System stores the user preference for that data subject and displays a confirmation message.

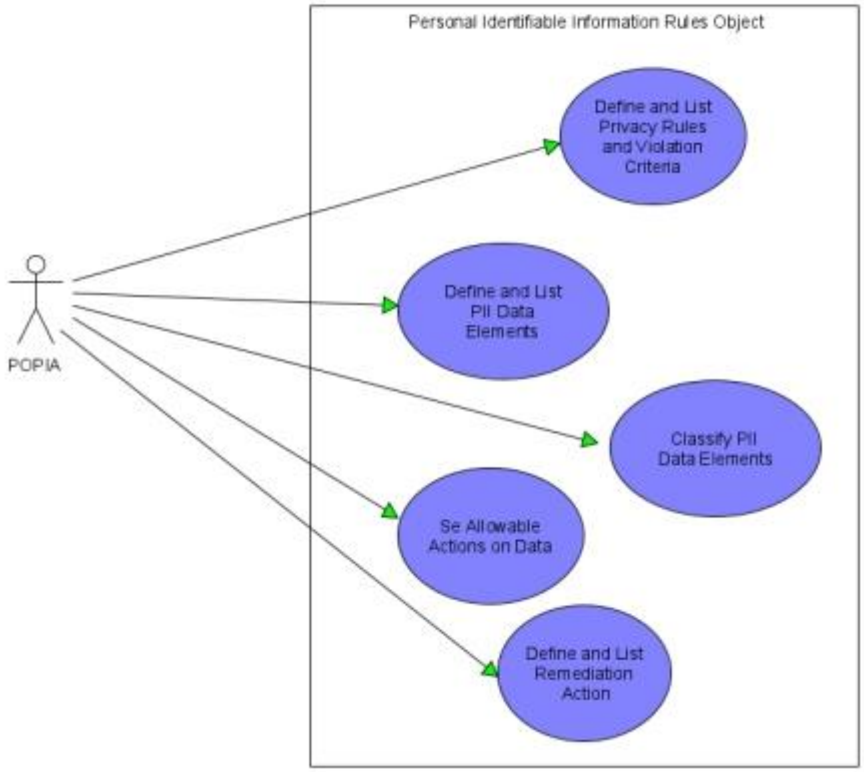
Assumptions:	The configured user privacy preferences are in line and in accordance with the POPI Act.
Notes and issues:	If any of these rules are not clear then it would be logged as an exception and published in the outcome of this project
Frequency of Use:	All the time. In fact, it should be used by every data subject/owner to protect the privacy of their personal identifiable information
Status:	Prototype developed
Owner:	XXXXXX
Priority:	P1 - High

Use Case 3

ID:	UC-3
Title:	Personal Identifiable Information Rules Object (PII Rule)
Description:	This use case is underpinned by POPIA, highlighting the different data/information elements constituting personally identifiable information (PII). According to POPIA, the data subjects have constitutional prerogatives to protect their personally identifiable information from exploitation by third parties and other commercial entities. This task requires organisations to classify data based on the conditions for the lawful processing of personal information, as stipulated in Chapter 3 of the POPI Act. As a result, every organisation that is a custodian of PII should set allowable actions on PII data, report violations on PII data and, most importantly, define remediation actions.
Primary actor:	POPIA and the Regulator
Preconditions:	The data subjects (organisation and data owners) are aware of the conditions and rules guiding the lawful processing of personally identifiable information, as mandated by POPIA.
Post conditions:	Organisation systems meet all the validation criteria required by the POPIA object and the privacy concern object.

<table>
<tr>
<td>Main
Success
Scenario:</td>
<td>

The use case diagram below illustrates the inner workings of the Personal Identifiable Information Rules Object.



Personal Identifiable Information Rules Object
Define and List Privacy Rules and Violation Criteria
Define and List PII Data Elements
POPIA
Classify PII Data Elements
Se Allowable Actions on Data
Define and List Remediation Action

Figure 3: Use Case: Set User Preference Rules Object
(Source: Author's own work)

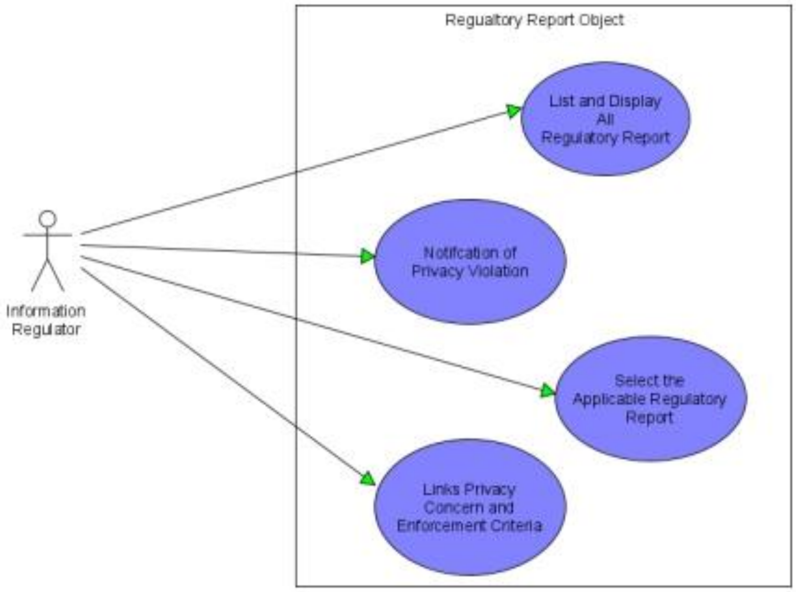
1. Define and list all POPIA information privacy rules. (The eight rules of the lawful processing of personal information)
2. Define and list all personally identifiable information, as defined by POPIA.
3. Classify personally identifiable information according to their sensitivity
4. Set allowable actions on personally identifiable information.
5. Define personally identifiable information violation criteria.
6. Define remediation actions on the violation of personally identifiable information.

</td>
</tr>
<tr>
<td>Assumptions:</td>
<td>This will be the broker object and should provide interfaces to link the Set User Preference Object, the POPIA Rule Engine Object and the Regulatory Report Object.</td>
</tr>
</table>

Notes and issues:	The administrator would be the POPIA champion operating the organisation under review. This role can also be played by the system administrator of the company.
Frequency of Use:	All the time. In fact, it should be used by every data subject/owner to protect the privacy of their personal identifiable information
Status:	Prototype developed
Owner:	XXXXXXX
Priority:	P1 – High

Use Case 4

ID:	UC-4
Title:	Regulatory report object
Description:	This use case links the privacy concern object with the system protection object to determine compliance with POPIA regulations.
Primary actor:	The Regulator (Also known as the Information Regulator)
Preconditions:	The Regulator is notified of POPIA violations through the regulatory report object
Post conditions:	Organisation systems meet all the validation criteria required by the POPIA object

Main Success Scenario:	The use case diagram below illustrates the inner working of the Regulatory Report Object.  Regualtory Report Object List and Display All Regulatory Report Notifcation of Privacy Violation Select the Applicable Regulatory Report Links Privacy Concern and Enforcement Criteria Information Regulator **Figure 4: Use Case: Set User Preference Rules Object** (Source: Author’s own work) 1. Regulator is notified of a particular violation of privacy concern. 2. Information regulator selects ‘Regulatory Report Object’ from the menu. 3. System displays list of All Regulatory Reports available. 4. Regulator selects one or more Privacy Enforcement Criteria and system lists items for that criteria. 5. Regulator links the privacy concern with the privacy enforcement criteria. 6. Regulator links privacy enforcement criteria with a particular organisation’s system. 7. Regulator clicks the Check button. 8. System scans information system objects to determine compliance.

Assumptions	This object is more for administrative purposes and should only activate once a violation has been reported.
Notes and issues	This object is not a requirement for this research project although it is a nice one to have.
Frequency of Use:	Used frequently by the regulator to monitor and administer/enforce privacy non-compliance
Status:	Proof of concept (POC)
Owner:	XXXXXXX
Priority:	P1 – High

APPENDIX C:

Publications from this Research

Journal	**Publication Title**	**Status**
South African Journal of Information Management **(SAJIM)**	**Enforcement of the Protection of Personal Information (POPI) Act: Perspective of data Management Professionals**	Published in the SA Journal of Information Management | Vol 20, No 1 | a917 | DOI: **https://doi.org/10.4102/sajim.v20i1.917** |
South African Computer Journal **(SACJ)**	**A Compliance Prototype for POPIA using UML Use Case Ontology Language**	Under review for resubmission @ **SAJC**