MESG

MESTRADO EM ENGENHARIA DE SERVIÇOS E GESTÃO

Developing Countries and ERP Implementations - Going Agile with Service Design

Miguel Marinho Mendes

Master Thesis

Supervisor at FEUP: Prof. Dra. Lia Patricio

Supervisor at Itgest: Doutor Sergio Pinto

U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

2020-07-07

Abstract

Following the software development industry, the ERP industry is shifting from traditional and bureaucratic waterfall implementation approaches to more agile approaches. The developing countries are no exception to this shift. Nevertheless, it is essential to understand if the developing countries are ready to follow Agile approaches and ensure that while this shift happens more human-centered, more iterative, more engaging implementation processes are delivered.

The present study is focused on integrating the Service Design into the SAP Activate implementation methodology, to develop a new method that brings together tools, practices, and principles from both approaches, improving the entire implementation experience in the developing countries such as Angola. The Activate Plus was developed following a Design Science Research methodology and supported on a qualitative study conducted to support the design and development process of the artifact, aiming to produce a relevant and valuable solution for a business problem.

As part of the Design Science Research, is it necessary to demonstrate and evaluate the implementation's results to understand if the proposed method solves the identified problems. Due to the timeline of the master thesis and current business context with the Covid-19 pandemic crises, it was impossible to implement the artifact on a real complete implementation project and present the evaluation results on this document. The demonstration of the artifact was done through the explanation and mapping through service blueprints of all phases of the new method, while the evaluation was done through a feedback session with some participants of the qualitative study. The feedback confirmed the value of the method and its significant applicability to the business environment, recognizing that several steps address can potentially minimize specific pain points and constraints identified during the implementation experience in developing countries.

Resumo

Seguindo as tendências da indústria de desenvolvimento de Software, a industria de ERP´s começa a adotar abordagens cada vez mais *Agile* em prol das abordagens tradicionais e burocráticas de implementação em *waterfall*. Os países em desenvolvimento não são exceção a essa mudança. No entanto, torna-se essencial perceber se os mesmos estão prontos para seguir abordagens Agile e garantir que, enquanto essa mudança acontece todo o processo de implementação seja mais centrado no ser humano "*Human-centred*", mais iterativo e que envolva o cliente e que minimize os típicos constrangimentos das implementações de ERP´s.

O presente estudo está focado na integração do *Service Design* à metodologia de implementação SAP Activate, com o objetivo de desenvolver um novo método que reúna ferramentas, práticas e princípios de ambas abordagens, tendo como grande objectivo melhorar toda a experiência de implementação em países em desenvolvimento, neste caso especifico Angola.

O Activate Plus é desenvolvido seguindo uma metodologia de Design Science Research, apoiado num estudo qualitativo, realizado para suportar o processo de design e desenvolvimento do artefacto, com o objetivo de cumprir o propósito do Design Science Research que é o de produzir uma solução relevante e válida.

Parte da metodologia, Design Science Resarch implica que haja uma demonstração e uma avaliação da aplicação do método na resolução do problema identificado. Devido à crise pandémica do Covid-19 e ao espaço temporal desta tese, não foi possível implementar o método num caso real, um projecto de implementação completa, e apresentar resultados do mesmo. A demonstração do método é feita através da explicação e mapeamento das fases de implementação através das *Service Blueprints* que ilustram a aplicação da nova metodologia às várias fases de implementação. A avaliação foi realizada através de uma sessão de feedback com alguns participantes do estudo qualitativo e confirmou o valor do método, sua significativa e aplicabilidade ao ambiente de negócios, reconhecendo que várias etapas abordam e potencialmente minimizam constrangimentos específicos e reconhecidos durante a experiência de implementação maioritariamente nos países em desenvolvimento.

Acknowledgments

First and foremost, I would like to offer my gratitude to my coordinating teacher, Prof. Lia Patrício, for the patience, support, time, and guidance through this study. I would like to thank also my other teachers and my colleagues from MESG.

A big thank you to Itgest that allowed me to do this master while working. I am thankful to everyone who makes this company a great place to work and develop academic studies, especially to Sergio Pinto and Lina Capela, for the full support and availability. To all interviewees, I must also communicate my word of gratitude for their time and willingness to help.

Finally, family and friends, the reason never to give up and the ones that always support and motivate me whenever I needed. Especially to my parents and sister that always encourage me in the most challenging situations and help me make the best decisions.

Table of Contents

List of Tables

List of Figures

List of abbreviations

ASAP - Accelerated SAP

BBP – Business Blueprint

BF – Business Function

CFO – Chief Financial Officer

CJM – Customer Journey Map

DSR – Design Science Research

DSRM - Design Science Research Methodology

ERP – Enterprise Resource Planning

GDP - Gross Domestic Product

IT – Information Technology

MESG – Mestrado em Engenharia de Serviços e Gestão

MS - Microsoft

MSD – Multi Level Service Design

SD – Service Design

SDL - Service-dominant logic

TDD – Test Driven Development

WTO – World Trade Organization

1 Introduction

As part of the master's in Services Engineering and Management of the Faculty of Engineering at the University of Porto, this master's thesis is focused on improving the ERP implementation methodology in a specific context: Angola. With the objective of boost the quality, the processes, and the "Implementation Experience" provided by ITGest, this research goes from a literature review about the major IT implementation approaches and Service Design to the application of the Design Science Research (DSR) to develop a new implementation method.

1.1 Project Background

Across all the globe, a large number of companies have transformed, or are transforming from a traditional "waterfall" development methodology to an "Agile" software development methodology (Winter 2014), the developing economies even with a much slower adoption are no exception on this shift. While Agile methodologies emerge as a new development model characterized by greater flexibility and responsiveness to new requirements (Almeida 2017), Waterfall methodologies are mostly a much linear way of delivering and schedule a project, with separate plans for risk mitigation, resourcing, budget and various other critical project functions (Grech 2015).

A common experience among companies that went into ERP (Enterprise Resource Planning) implementation projects is incurring high expenditures, which frequently overcome their initial budgets drastically. By this, frequently, the ERP projects seem to fail in their promises of cost-reduction (Francalanci 2001). Many factors can be the cause of such failure, although, as stated by (Ngai, Law et al. 2008), the ERP implementation methodology is one of the most relevant critical success factors in the adoption of an ERP. To specifically face this critical success factor, and pushed by the digital revolution that we are currently facing (Mahadevan et al. 2015), the ERP industry traditionally based on waterfall methodologies, is moving on to more agile approaches.

ERP is an integrated set of software packages to support the cross-functional operations of an enterprise. It provides a process-oriented information platform to enable an enterprise-wide operational efficiency (Asendorf, Gronau et al. 2010).

As a service, the ERP implementation commonly provided by IT consultancy firms, can be an object of research, improvements, and some innovations. Taking into consideration the objectives of this research (detailed in the following chapters), the chosen approach to foster this service innovation was the Service Design (SD).

As an interdisciplinary field that integrates operations management, information systems as contributions from marketing, interaction design, combining service science with design thinking the SD has the objective of design a service offering that enable customers to co-create valuable experiences (Patrício and Fisk 2013) The usage of Service Design may change the customer's and organization's role in value co-creation, by creating smart offerings, by changing the organization's processes of value integration, or by repositioning the organization in the value constellation (Patrício and Fisk 2013).

1.2 Problem Description

ERP implementations are notorious for taking a long time and costing more money than initially projected; most ERP implementations, have not lived up to their market expectations (Ehie and Madsen 2004).

Besides the challenge that is, perform an IT project following and specific methodology, in time, on budget and with a recognized value for the customer, the projects delivered by Itgest have this additional contextual challenge that is performing and deliver an IT project on a developing country, on this research more specifically Angola.

The characterization of a country as a "developing country" is not consensual, according to WTO (World Trade Organization) there are no definitions of "developed" and "developing" countries, the countries announce for themselves whether they are "developed" or "developing" countries. However, the International Monetary Fund is the more specific, taking into account a country's per capita income level, export diversification, and degree of integration into the global financial system. So, although the designation is not universal, according to the International Monetary Fund, Angola can be included in the group of the developing countries.

So, as a critical factor, the methodology can minimize delays and costs, ensure the customer adoption of an ERP and help companies improve their business processes. By this, the problem identified by this research is the difficulty of following the evolution of the ERP methodologies and implement them in developing countries like Angola.

1.3 Purpose of the Study

"Transition is not easy. The Agile approach is very different from the Waterfall approach in several ways" (Mahadevan, Kettinger et al. 2015). This master thesis aims to understand how Service Design and more Agile approaches can improve the implementation experience in developing countries. To overcome the problem, this research will focus on the below key points;

- Understand through literature review the difference between the Waterfall methodologies and Agile and identify the common principles present in the SAP Implementation methodologies (ASAP and ACTIVATE).
- Embed Service Design into the commonly used ERP implementation methodologies.
- Trough Design Science Research develop an adjusted ERP methodology framework that better fits the specifications of the developing countries;

Besides the focus on two methodologies on this master thesis, the literature gives us several other approaches to software development as Six-sigma, Spiral, or Rapid Application Development. Considering that the ERP traditional implementation models are actively build up in Waterfall principles (Kraljic, Kraljic et al. 2014) and together with the Agile methodologies, both, are the most popular software implementation methodologies (Pedersen 2013) the choose end up focusing on those two methodologies.

1.4 Research Questions

The following research questions aim to guide and focus this project.

- *Research question 1 (RQ 1)* – What are the advantages and pitfalls of the application of agile methodologies for the implementation of ERP´s in emerging countries?

- *Research question 2 (RQ 2)* – How, can Service Design support and improve an ERP implementation methodology applied in a Developing Country?

1.5 Study and Project Development at ITGest

This project will be developed in ITGest that is an implementation, integration, and systems management consulting firm specialized in SAP solutions with their headquarters in Matosinhos and with a total of 80 employees. It is one of the leading SAP partners in the African market with several successful implementations and regular customers.

SAP stands for "Systeme, Anwendungen und Produkte" is a German company created in 1972 and the world's largest enterprise software provider that, in 2019, had more than 437,000 customers in over 180 countries. Is the market share leader in enterprise resource planning (ERP), analytics, supply chain management, human capital management as well as experience management and customer experience (SAP Corporate Fact Sheet 2020).

The SAP ERP package provides software solutions for the full range of business functions in companies – from human resource management, back-office finance processes, the full sales order processing cycle and manufacturing, supply chain, and distribution functions. SAP ERP is usually installed on a database platform that handles several different business functions supported by the range of software modules that make up the SAP product suite.

ITGest is organized on a market and location logic with specific teams and chains of command for Angola Mozambique and Portugal, with some cross consultants over all locations their services portfolio include the implementation of new ERP systems, maintenance, remote support and certified training.

Also, regarding the company and relevant for the study was this pandemic crisis still ongoing, challenging times that ended up with the offices being closed and some employees (my case) sent to lay-off.

1.6 Report outline

This master thesis is organized following the DSR stages with a total of seven chapters.

In the first chapter, the introduction contextualizes the research with the motivations of this project, the objectives, and the research questions raised, as well as the problem this thesis intents to help to solve.

The following chapter, the literature review, start with the research of the existing literature about waterfall and Agile, the main characteristics and differences. The second sub-chapter of the literature review goes deeper into the service design and brings the literature review about the tools, methodologies, and approaches that can be used to achieve the thesis purpose.

In the third chapter designated methodology, it’s included the description of the method of research and the used work along with the proper justification of selection and the applicability in the context of the problem.

The fourth chapter contextualizes the problems, with a brief overview about Angola and a explanation of the standard SAP methodologies as the identification of the main Agile and waterfall principles on the SAP methodologies.

The following chapter presents the results of the qualitative analysis, with the results of the data analysis with the support of Nvivo 12 and the customer journey of the current implementation approach used.

Following the DSR stages the sixth chapter corresponds to the demonstration stage, presenting the new method Activate Plus, applying it through service blueprints, and presenting the results of the method evaluation. The last chapter, the conclusion, brings a summary of the obtained results and how the same results contributed to respond to the initial research objectives.

2 Literature Review

2.1 Waterfall project implementation approach

"Any successful project requires use of project methodology" (Nelson 2005). Large project as ERP implementation projects require formal project management methodologies or models, which establish a blueprint of business processes provide project planning process as handover plans. Among standard project management methodologies, one of the most common project methodologies used in the IT market is the waterfall project management model (Lapunka, Kolodziej et al. 2017). Developed by Winston Royce in 1970, the waterfall model has been famous for its relative ease of use.

Introduced by Royce as an idea/illustration of a flawed software development method that was weak because of its many inadequacies, ironically ended up as the first methodology developed for software development. The term waterfall arises as a metaphor for all processes, and it was not a word used by Royce. The initial model also included nonsequential processes. Royce believed that any well-designed software model should be interactive, allowing for advances and setbacks between phases which the standard waterfall model does not allow. Nevertheless, the software industry had adopted this model from the hardware manufacturing and construction industry used during the 1970s as a highly structured approach to software creation (Pedersen 2013).

The waterfall concept comes from the downward way of how the model goes from one phase to another. According with Othan et al. (2017) cited on Dima and Maassen 2018, the waterfall model implies one phase has to be finished before the other one starts, the risk being high that errors from the previous phase can be exported to the next phase, as verification occurs at the end of the software development or close to the end phase. With each stage being completed before moving on to the next, there is no way to turn back to a prior phase once the last one has been reached. This type of model sustains a more traditional type of management organization, where requirements circulate from top management to basis management and further on to employees and minimal or no contact with customers or other stakeholders occurs with developers' teams. Requirements are usually transmitted indirectly through organization members.

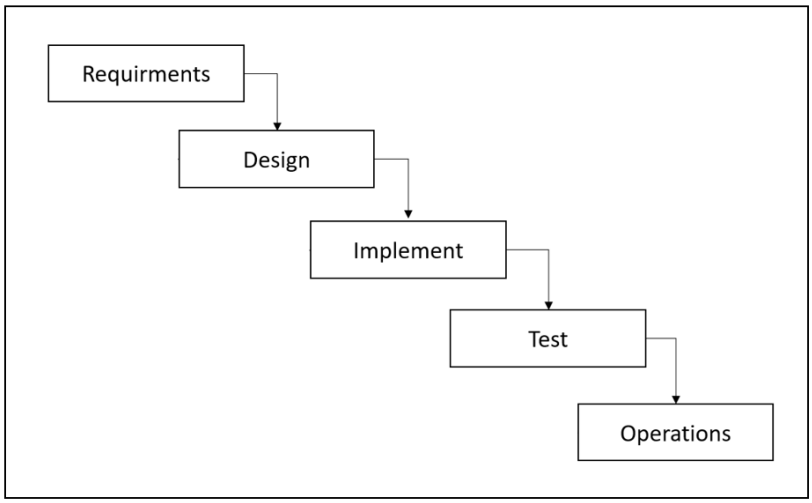


Figure 1 - Waterfall phases (Dima and Maassen 2018)

The Waterfall phases may vary by the numerous references of the process literature today. According to Pedersen (2013), the waterfall model may go from five to seven stages depending on the several authors. As shown in Figure 1, the Waterfall model (Winston, 1987) starts with the concept of inquiry or requirements analysis. The model follows a well-defined set of criteria and requirements indicators before the design phase and project implementation. The requirements phase includes setting out all possible requirements for the process/system, either for the software (Morgan 2018). Requirements are necessary system functionalities that are required for the end-user to use the system in operation (Thomas and Fernandez, 2008).

The system design phase follows the requirements, according to Royce (1970) cited on Morgan (2018) the design process should be;

1) Done with program designers not analysts or programmers
2) The design, the allocation, and definition of the data processing should be done at this phase, even taking the risk of being wrong.
3) An overview document about the system design, understandable for each stakeholder of the project. At least one person must have a deep understanding of the system, which partially comes from having to write an overview document.

In the current literature, a high-level program design is also known as solution architecture (Cusick 2013). In the implementation and testing phase, the project is outlined in the documents from phase two and separated into different units, and the actual coding of the software is begun. Before coding starts, the system is developed in small programs called units, which are incorporated in the following phase. Each small program, or unit, is created and then tested to make sure it works and serves the purpose it was developed. With the system separated into units, developed, and tested the integration and system-testing phase ends where the units are incorporated into the overall total system (Thomas and Fernandez, 2008).

The next phase, operations and maintenance, goes on indefinitely. Overall, problems with the system created that are not discovered during its development life cycle emerge after its use begins are solved after deployment of the system (Pedersen 2013).

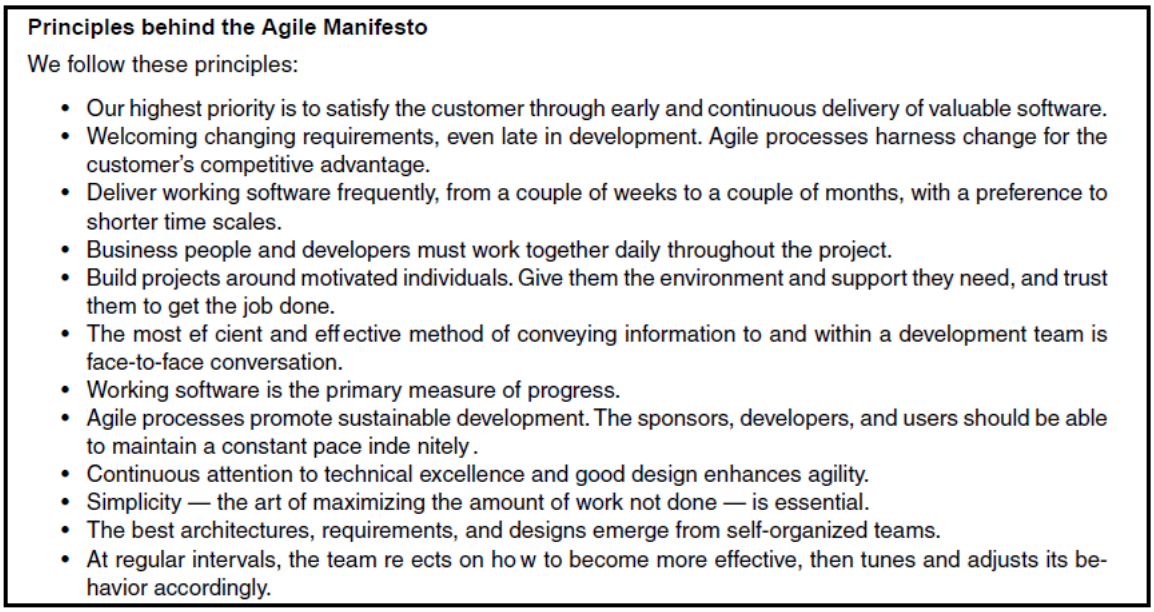
Table 1 - Advantages and Disadvantages of Waterfall (Pedersen 2013)

Advantages	**Disadvantages**
Easier to set a specific time/period for the tasks to be done;	Problematic when not all requirements are received at once.
No overlapping of phases;	Does not allow for alteration of previous phases
Errors in the software can be detected early as one phase must be complete before moving on to the next	The testing of the software is untimely; that is, it occurs late in the developmental process;
Due to its linear design, the associated costs are less than other models, which in turn can help reduce the cost of the overall project.	Not all of them are fulfilled, as the customer continues to add requirements
Formal documentation easy handover for customers and new workers	A large amount of time is potentially wasted on excessive documentation of the project.

Waterfall models are suitable for projects with unambiguous product requirements and where the requirements will not change dynamically during the project completion (Lapunka, Kolodziej et al. 2017). Waterfall represents the original software development model, but nowadays large companies are increasingly transitioning from Waterfall to Agile methods (Mahadevan, Kettinger et al. 2015)

2.2 Agile methodologies

Similar in popularity but quite different on the approach, Agile is different from the Waterfall in several ways (Mahadevan, Kettinger et al. 2015). The Agile approaches of software development started during the 90s when developers began to move away from highly traditional models with a lack of ability to provide flexibility during the software development lifecycle (Beznosov and Kruchten, 2004). An important milestone for Agile happened in 2001 here a group of software developers declared the Agile Manifesto (appendix A), a set of guidelines that were suggested as a framework for agile software development models (Lindstrom and Jeffries, 2004).



Principles behind the Agile Manifesto

We follow these principles:

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcoming changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to shorter time scales.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most ef cient and eff ective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace inde nitely.
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity — the art of maximizing the amount of work not done — is essential.
- The best architectures, requirements, and designs emerge from self-organized teams.
- At regular intervals, the team re ects on ho w to become more effective, then tunes and adjusts its behavior accordingly.

Figure 2 - Agile Manifesto (Lindstrom and Jeffries, 2004)

Currently, the Agile methods are the most popular models used in the IT industry. Customer interaction is the backbone of the agile methodologies, and open communication with minimum documentation are the typical features of an agile environment. (Lapunka, Kolodziej et al. 2017)

The Agile approach to software development do not have the pretension to know all the requirements that must be developed beforehand, but rather to focus on small sets of functionalities that frequently originate testable versions to the client (Almeida 2017). Agile begins with the assumption that not all the work is of equal criticality, and so the project's scope is broken out into discrete functionalities or features that are called user stories. Then management prioritizes these user stories into a backlog of work. (Grech 2015).

As explained by (Lindstrom and Jeffries, 2004) the responsibility is assigned to the software developers and business function (BF) areas. During the project, representatives from both

functions work “fiscally” close together. Agile team members jointly provide status reports daily. Each iteration cycle is only a few weeks long and goes through the design, coding, and testing. Agile design is often kept open to allow last-minute change and alterations involving business function areas and feedback at the end of each session.

Requirements are constantly evaluated, and feature priorities are either upgraded or downgraded depending on BF areas intervention (Mahadevan, 2015). The design concept allows for the evolution of new concepts as they come along. The importance of documentation is not emphasized, adaptability, flexibility and speed of the delivery are the drivers. BF areas are given demonstrations at the end of each iteration cycle, and feedback from the reviewers helps decide the next course of program development. This iterative cycle (flow presented in figure 3) keeps on going until the final product meets the specifications of the BF areas (Lindstrom and Jeffries, 2004).

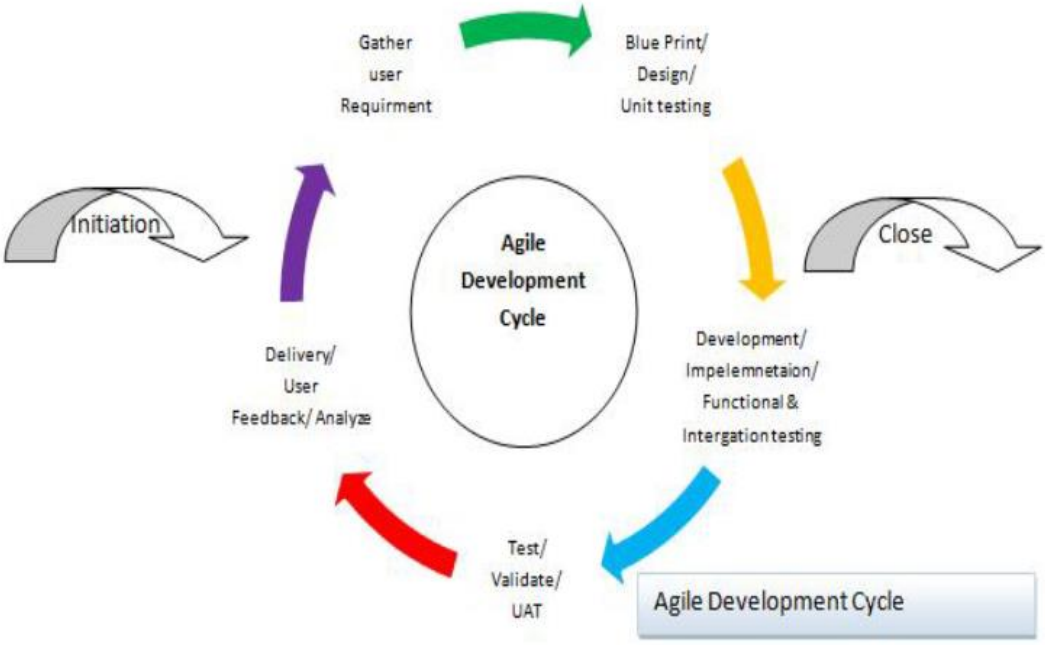


Figure 3 - The Agile Process (Pedersen 2013)

Miller (2001) list´s the following characteristics to agile software processes:

1. Modularity on development process level
2. Iterative with short cycles enabling fast verifications and corrections
3. Time-bound with iteration cycles from one to six weeks
4. Parsimony in development process removes all unnecessary activities
5. Adaptive with possible emergent new risks
6. Incremental process approach that allows functioning application building in small steps
7. Convergent (and incremental) approach minimizes the risks
8. People-oriented, i.e. agile processes favor people over processes and technology

9. Collaborative and communicative working style

Also, an essential factor detailed by (Pedersen 2013) is about the people besides teamwork, an agile methodology fit´s on the mindset, as requires that all staff members are recruited from the beginning of the project.

To summarize, according to Abrahamsson et al. (2010), the key aspects of agile methods are simplicity and speed. The software development is incremental (small software releases, with rapid cycles), cooperative (customer and developers constantly working together with close communication), straightforward (the method itself is easy to learn and to modify, well documented), and adaptive (able to make last moment changes).

The Agile model may include versions of software development management, “methods” as extreme programming (Beck 1999), scrum (Schwaber 1995), crystal family of methodologies (Cockburn 2006), Test-driven development (Beck 2003) and many other methods.

According to Dima and Maassen (2018), the Agile model has itself two main versions of software development management, namely the Scrum method and Test-driven development. As this master thesis is not about Agile and the software implementation methodologies are used as principles for the ERP implementations, the next following sub-chapters will just be focused on those two methods Scrum and Test-driven development.

2.2.1 Scrum

The term 'scrum' originally comes from a strategy in the game of rugby, where it denotes "getting an out-of-play ball back into the game" with teamwork. The software development term was first mentioned in Takeuchi and Nonaka (1986).

According to Schwaber and Beedle (2002), the Scrum approach has been developed to manage the software systems development process. It is an empirical approach putting in place the ideas of industrial process control theory to systems development resulting in an approach that reintroduces the ideas of flexibility, adaptability, and productivity. It does not define any specific software development techniques for the implementation of a software, instead it concentrates on how the team should work in order to produce the system flexibly in a continually changing environment.

On Schwaber (1997) cited by Abrahamsson (2002) the main idea of Scrum is that systems development involves several environmental and technical variables (e.g., requirements, time frame, resources, and technology) that are likely to change during the process. This makes the development process unpredictable and complex, requiring the flexibility of the systems.

Introduced by Schwaber and Beedle (2002) scrum process includes three phases: pre-game, development, and post-game (Figure 4).

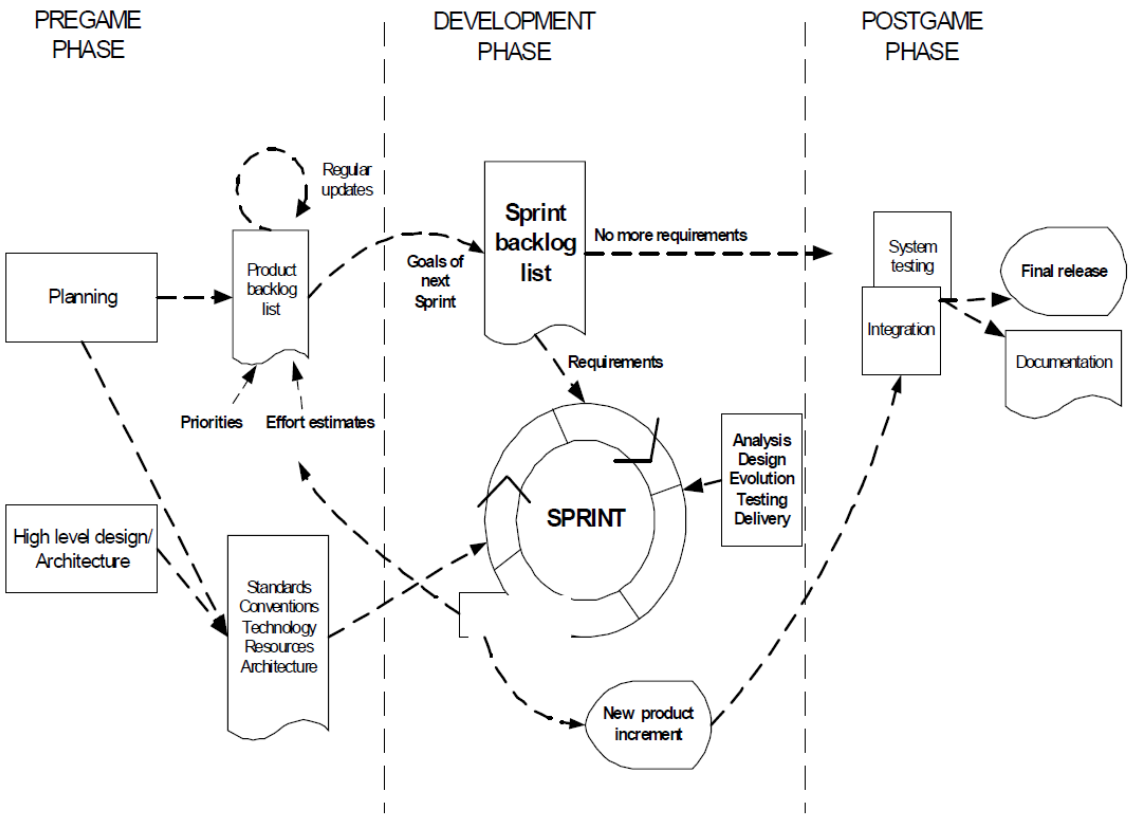


Figure 4 - Scrum process Schwaber and Beedle (2002)

Table 2 - Scrum phases are introduced according to Schwaber and Beedle (2002)

Pre-game phase includes two sub-phases: Planning and Architecture/High level design.
Planning includes the definition of the system being developed. A Product Backlog list is build containing all the requirements (currently known), prioritized, and estimated. At every iteration, the updated product Backlog is reviewed. In the architecture phase, the high level design of the system including the architecture is planned based on the current items in the Product Backlog.
The development phase is the agile part of the Scrum approach.
In the development phase the system is developed in Sprints enhanced to produce new increments Each Sprint includes the traditional phases of software development: requirements, analysis, design, evolution and delivery (Figure 4) phases.
The **post-game phase** contains the closure of the release.
Happens when the agreement that the requirements are completed

According with Abrahamsson (2002) there are six identifiable roles in Scrum that have different tasks and purposes during the process are they; Scrum master (management role focused on apply the scrum "way of work", interact with customers and the management, and assure that conditions are gathered to keep the team working), Product Owner (responsible for

the project and with the power of decision about the backlog), Scrum team (project team), customer and management (participates in the definition of goals and requirements, also on the selection of the Product Owner, following the progress and reducing the Backlog with Scrum Master).

Scrum does not require or provide any specific software development methods to be used. Instead, it requires certain management practices and tools in the various phases of Scrum that should be used (Pedersen 2013).

2.2.2 Test driven development

"The test represents an unambiguous requirement that the program must satisfy" James Newkirk on Marchesi and Succi (2003). The Test Driven Development (TDD) is another version of managing software development within the Agile method.

On TDD approach, the test is written before the actual start of software development through coding, thus, it starts with establishing the requirements for the new product feature with a testing of a first version of the product (Dima and Maassen 2018).

By satisfying the rules defined on the test and achieving a successful execution, the team can step back, proceed on the requirements and implementation of the coding to satisfy this feature (Marchesi and Succi, 2003). If, while the coding of the test, this one is not successfully passed, it implies that the feature will not be implemented (Dima and Maassen 2018).

Some professed benefits to TDD according with Maximilien and Williams, (2003):

- In any process, there exists a gap between decision (design developed) and feedback (functionality and performance obtained by implementing that design). The success of TDD can be attributed to reducing that gap.
- TDD require programmers to write code that is automatically testable, such as having functions/methods returning a value, which can be checked against expected results.
- The TDD test cases create a thorough regression test bed. By continuously running test cases, one can easily identify if a new change breaks anything in the existing system. This test approach should also allow smooth integration of new functionality into the code base.

2.2.3 User Stories

In an Agile approach user stories form a set of central work products that determine the software development processes (Rees 2002) they can be added to Scrum (Cohn 2004), and they are included on this literature review as user stories may also, according to Stickdorn and Hormess, et al. (2018) be used as Service Design methods on the several stages of the design process.

User stories are used in software development to define requirements from a user or customer perspective, they should be formulated without IT-specific language, keeping them as simple as possible example Figure 5 (Stickdorn, Hormess et al. 2018).

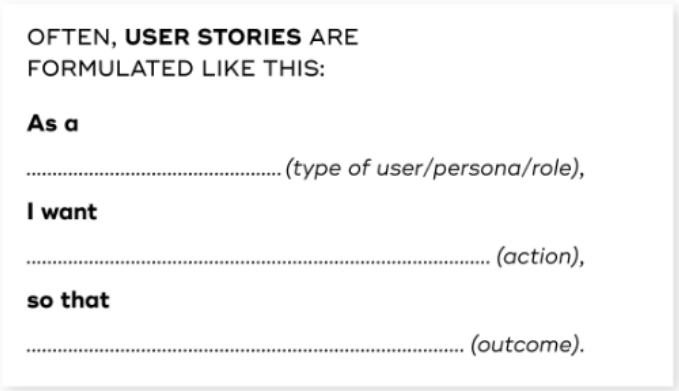


Figure 5 - User Stories (Stickdorn, Hormess et al. 2018)

Traditionally, user stories are handwritten on index cards (Figure 5) as they are easy to store, display, rearrange, and distribute to the co-located development team (Rees 2002). In this context, the requirements are recorded as a set of user stories developed jointly by the customer and the development team (Cohn 2004).

Furthermore, agile software development considers that production teams should start with simple, knowable approximations to the final requirements, and then continue to increment the details of these requirements throughout the life of the development being oriented to tasks and features (Rees 2002).

Once done the user stories should be clustered into epics, this means that development team before start (the development), should look for patterns, similar user stories and check if it is possible to combine/cluster and merge user stories into epics, then try to prioritize them (Stickdorn, Hormess et al. 2018).

2.3 Comparison between Waterfall and Agile

Comparing both methodologies is possible to identify some similarities and differences and end up with some conclusions. The first thing that we can establish clearly by the two models is the common goals, which serve as delivery method across project teams and customers. Also, the objective and syntactic comparison between Waterfall and Agile can be found in the research of Dima and Maassen, (2018) transcribed below in table 3.

Waterfall model	Agile model
-formal, hierarchical organisation and transfer of information (top- down)	- informal organisational culture (frequent communication managers, employees, departments)
-customer receives product at the end of the software development process, then feedback	-customer involved in production
-customer feedback received after delivery of end product	-customer feedback on product releases and improvement suggestions (usually monthly) required
-product released at the end of all production phases	-frequent releases (usually monthly)
-often no cooperation with other departments	-often cooperation with other departments to achieve customer insight and satisfaction
-less frequent presentation of results	-pressure on employees to present (weekly, every two weeks, etc.) progress in project development

Table 3 - Differences in project management in the Agile vs. Waterfall (Dima and Maassen 2018)

Hybrid approaches are also becoming more and more popular among large-companies, that in one side cannot discard some control procedures and formalities of the Waterfall approach and in other side seek more Agile approaches, according with (Lapunka, Kolodziej et al. 2017) methods as SCRUM the introduction of daily follow-ups, the constant introduction of system demos and tests, can be Agile approaches introduced on traditional Waterfall implementations. On the Figure 6 an example from Mahadevan, Kettinger et al. (2015) of an integration of SCRUM into a Waterfall project.

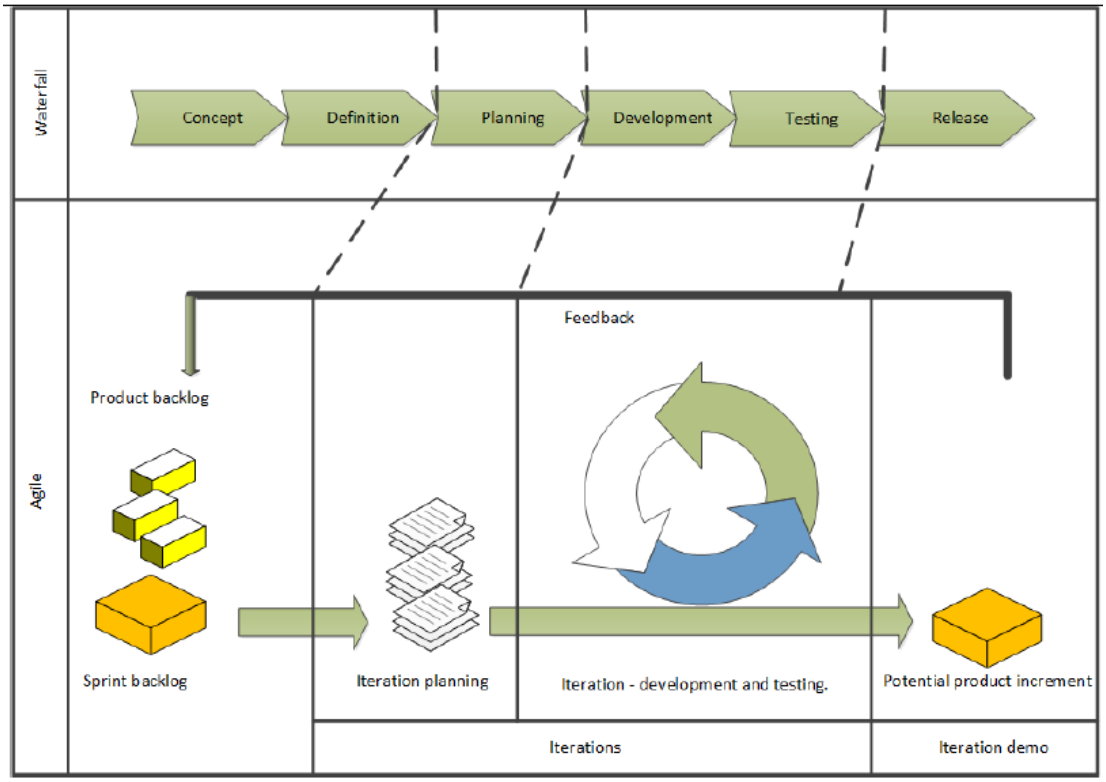


Figure 6 - Scrum Activities/Concepts Mapped to Waterfall (Mahadevan, Kettinger et al. 2015)

The introduction of Agile approaches can minimize typical issues of the waterfall, e.g. limited opportunities for interaction between the customer and de project team, regular follow ups, regular demos for testing, can boost the customer engagement and involvement on the project, as provide to the customer an greater visibility into the working of performed by the project team (Mahadevan, Kettinger et al. 2015).

Some literature seems to present the Agile as an alternative to replacing the more traditional Waterfall approach, however, Pells (2017) states that Agile should not be understood as a replacement to waterfall. The effectiveness of an Agile approach will depend on several factors, as an example; the size of the project (for larger projects documentation becomes more important), the type of industry, the geographies and teams locations, so in some contexts the Waterfall approach it is still the most recommendable.

2.4 Service Design

Service Design is a multidisciplinary approach that brings together contributions from fields such as service marketing, operations, interaction design, through a holistic approach and making use of design thinking (Figure 7). Different methods and tools are applied to improve existing services or create new ones (Patrício and Fisk 2013), the process being an iterative one.

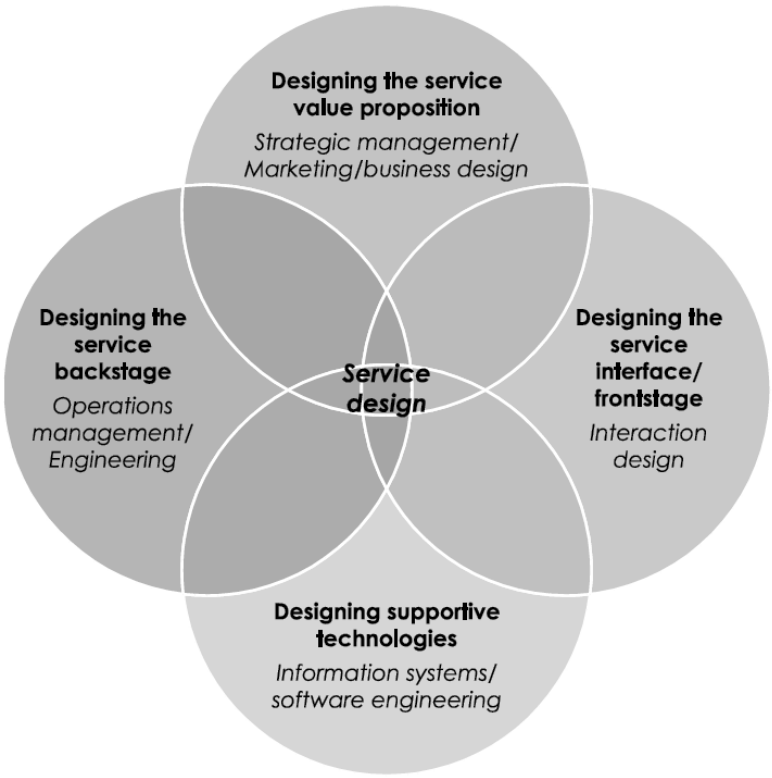


Figure 7 - Creating new services (Patrício & Fisk 2013)

SD can help solve important challenges faced by the organizations; it´s about customer experience, innovation, and collaboration (Stickdorn and Hormess et al. 2018). By understanding customer experiences, SD can origin new service ideas (Mahr et al. 2013) (Joly et al. 2018).

SD is becoming more visible than ever, as more recently, customer experience has become overwhelmingly important for many organizations, and design (usually "design thinking") has become a key innovation and management methodology. Sitting at the intersection of design thinking and customer experience can potentially be applied to shaping much of human activity as "design" is the process of making sure that something fits its purpose (Stickdorn and Hormess et al. 2018).

SD follows a human-centered vision that starts from understanding the customer, the context, and proceeds to the orchestration of different elements, to find new opportunities of value-creation (Patrício and Fisk 2013).

The human-centered approach SD draws on a variety of methods enhancing and prototyping not just *how* the service can be performed, but also with *which* qualities should be part of the

services. By being human-centered, SD enhances people's lives and helps organizations better connect with their existing networks of stakeholders while discovering new opportunities to co-create value (Meroni and Sangiorgi 2011).

SD, in this sense, aims to create services that are useful, usable, desirable, efficient, and effective (Holmlid and Evenson, 2008).

SD follows a design thinking approach (Stickdorn and Hormess et al. 2018) that builds upon a creative, collaborative, and iterative process of understanding customers and their context, envisioning future service solutions, and prototyping them. The SD process involves stages as: Inspiration (understanding the experiences of multiple actors and their contexts), ideation (designing the service offering), reflection (through prototyping and testing), and implementation, as shown in Figure 8.

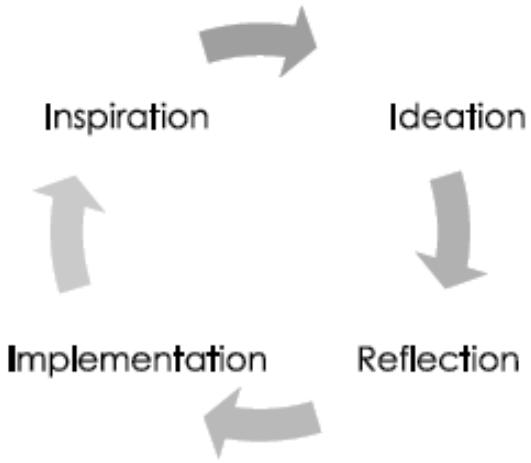


Figure 8 - Service design process (Patrício & Fisk 2013)

The SD methods have also evolved from focusing on only one service level to multilevel (concept, system, encounter) (Patrício, Fisk et al. 2011) and from single-channel to multichannel service interfaces and designing for value networks. (Patrício, de Pinho et al. 2018). On this basis, new research questions have raised, such as how SD methods should continue evolving to support organizational change, including the transition from concept towards implementation (Ostrom, Parasuraman et al. 2015).

Furthermore, (Ostrom, Parasuraman et al. 2015) identify several research priorities that have the potential to advance the service field and benefit customers. The authors highlight the development of service networks and systems, defining several research areas that emerge from it, such as building service systems that are adaptive and flexible enough to respond to dynamic environments or creating modularized service system architectures that can be used for value networks. Also identified on the study, some critical areas that need further research for leverage SD, as participatory design and codesign to enhance service experience, the usage of SD approaches to innovate complex service systems and value networks, the involvement of multidisciplinary teams, and advance in service prototyping to support rapid customer-centric service innovation.

2.4.1 Basic Service Design Tools

The current sub-chapter will present the theoretical framework of the SD tools used in this master thesis. It´s important to start by differentiating between tools and methods. According to Stickdorn and Hormess, et al. (2018), tools are concrete models such as CJM, storyboards, service blueprints, or customer value constellations. They follow a specific structure or are built on given templates. Methods are particular procedures to accomplish or approach something, for example, Multilevel Service Design (Patrício, Fisk et al. 2011) or the MINDS (Grenha Teixeira, Patrício et al. 2017). Tools represent "What" we use, while methods usually describe "how" we create and work with certain tools in SD project.

Personas

A persona is a profile representing a particular group of people, such as a group of customers or users. This profile is not a stereotype, but is an archetype built after an exhaustive observation (Stickdorn and Hormess et al. 2018).

According to Long (2009), using personas as a design tool in the product development process can lead to more user-centered designs. Personas strengthen the focus on the end-user, their tasks, goals, and motivation.

Personas make the needs of the end-user more explicit and, thereby, direct decision-making within design teams more towards those needs. (Long 2009).

There are many templates to create personas or guidelines that can be used to build comprehensive personas. One often-used approach is empathy maps that identify potential pains and gains and include topics like "What does the customer think & feel/hear/see/say & do?" (Stickdorn, Hormess et al. 2018).

Customer Journey Map

Understanding customer experience and the customer journey over time is critical for firms and is becoming more and more at the top priority of the top management. This focus arises because customers now interact with firms through myriad touchpoints in multiple channels and media, resulting in more complex customer journeys (Lemon and Verhoef, 2016).

Customer experiences are more social, in nature, and peer customers influence experiences as well what leads to less control from the firms, overall, of the customer experience and the customer journey (Lemon and Verhoef, 2016).

Customer Journey Maps (CJM) are an effective way of visualizing either existing experiences or new experiences that are planned but do not yet exist (future-state journey maps) (Stickdorn, Hormess et al. 2018). They are a straightforward idea: a diagram that illustrates the customer's steps in engaging with our company, whether it be a product, an online experience, retail experience, or a service (Richardson 2010).

As a human-centered tool, CJM not only include steps where a customer is interacting with a company, but reveal all the key steps of an experience. CJM helps to find gaps in the customer experiences and explore potential solutions (Lemon and Verhoef, 2016).

The basic structure of CJM consists of steps and stages defining the scale of the visualized experience. The scale can range from a high-level journey map that shows the entire end-to-

end experience to a very detailed journey map showing only a few minutes. CJM make intangible experiences visible and facilitate a common understanding between the SD team (Stickdorn, Hormess et al. 2018).

Elements that construct a customer journey may vary depending on the followed author, as an example, Richardson (2010) consider elements as; activities, motivations, the timeline, questions a barrier while Stickdorn and Hormess, et al. (2018) embed on this base, storyboards, emotional journeys and dramatic arcs.

Service Blueprint

Service Blueprinting provides a diagrammatic tool for designing the service delivery (Patrício et al. 2011). It can be understood as an extension of the customer journey map (Stickdorn, Hormess et al. 2018). The actual concept evolved to connect customer experiences with both frontage and backstage and the support process and the physical evidence (Patrício et al. 2011).

“Frontstage” refers to people and process with which the user has direct contact. ”Backstage” people and processes that are invisible to the user while support processes are activities by the rest of the organization or external partners (Stickdorn, Hormess et al. 2018).

On Bitner, Ostrom, and Morgan (2008) study, they identified five components of a typical service blueprint (see Figure 9)

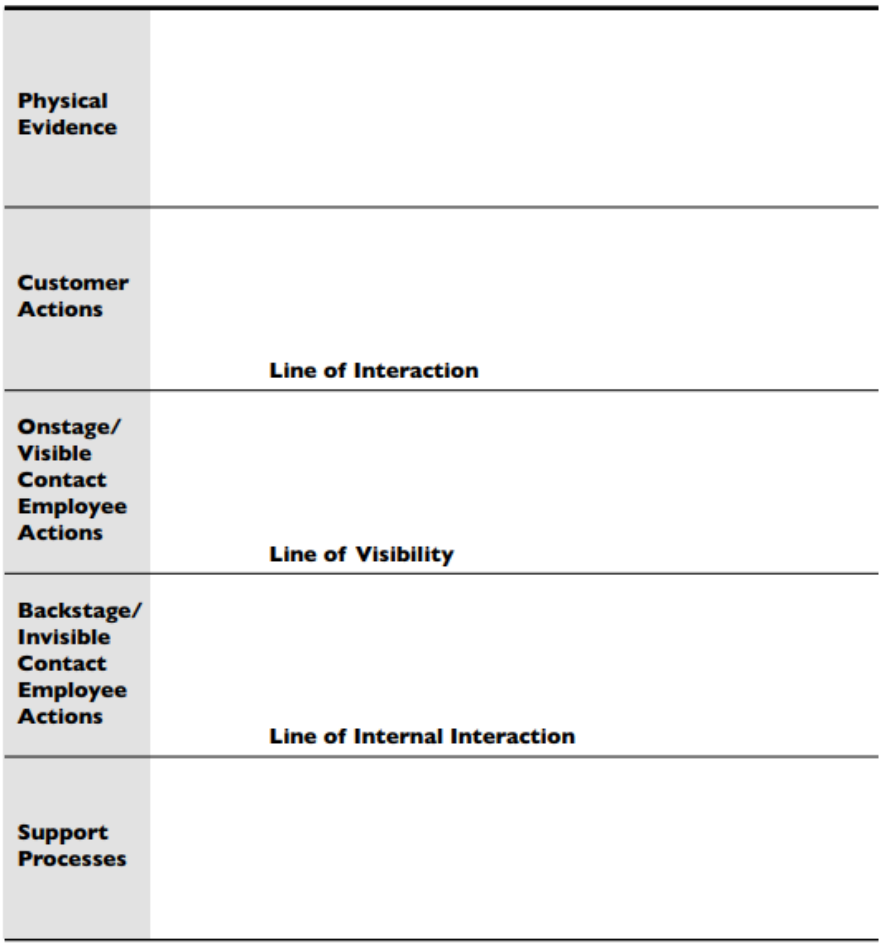


Figure 9 - Service Blueprint components (Bitner, Ostrom, and Morgan, 2008)

"Customer actions" include all of the sequential steps that customers take as part of the service delivery process. The next critical component is the "Frontstage," separated from the customer by the line of interaction. Every time the line of interaction is crossed via a link from the customer to a contact employee, a moment of truth has occurred. The "backstage" separated from the onstage actions by the very important line of visibility. Everything that appears above the line of visibility is seen by the customer, while everything below it is invisible. Below the line of visibility, all of the other contact employee actions are described, both those that involve non-visible interaction with customers (e.g., telephone calls) as well as any other activities that contact employees do in order to prepare to serve customers or that are part of their role responsibilities. The fourth critical component of the blueprint is "support processes" separated from contact employees by the internal line of interaction. These are all of the activities carried out by individuals and units within the company who are not contact employees but that need to happen in order for the service to be delivered. Vertical lines from the support area connecting with other areas of the blueprint show the inter-functional connections and support that are essential to delivering the service to the final customer. Finally, for each customer action, and every moment of truth, the physical evidence that customers come in contact with is described at the top of the blueprint. These are all the tangibles that can influence the customer quality perceptions (Bitner, Ostrom, and Morgan 2008).

Value Constellation & Affinity Diagrams

Following the MINDS method (Grenha Teixeira, Patrício et al. 2017) the current study integrates affinity diagrams over the customer value constellation to enable the service concept, the following paragraphs present the literature about both tools separately.

The customer value constellation portrays service offerings and their relationships from a customer point of view, focusing on the services that support a customer's overall activity independent of the service provider. It represents the network of actors and their relationships that jointly create an offering (Grenha Teixeira, Patrício et al. 2017).

Developed by Normann and Ramirez (1993), the value constellation framework provides an insightful approach for positioning the firm's service concept in the value-creating system (Patrício et al. 2011).

The affinity diagram organizes a large number of ideas into their natural relationships. It is useful to creatively explore new concepts through brainstorming sessions (Grenha Teixeira, Patrício et al. 2017). Use it to generate, organize, and consolidate information related to a product, process, complex issue, or problem. After generating ideas, group them according to their affinity or similarity. This idea creation method taps a team's creativity and intuition. It was created in the 1960s by Japanese anthropologist Jiro Kawakita, known as the KJ method (Nancy 2005).

3 Methodology

A critical success factor of a “value-added” research that fosters solutions for real-world problems and tries to generate new knowledge is the research methodology. The Cambridge Dictionary lays down the meaning of methodology as “a system of ways of doing, teaching, or studying something” while the definition of research as “the study of a subject in order to discover new information.” According to Kothari, C. R. (2004), research methodology can be synthesized as a way of systematically solving a research problem. It may be understood as a science of studying how research is done scientifically.

Not just focused on understanding the research problem, but in developing a solution, through a construction of an artifact to solve a specific problem, Design Science Research builds on both quantitative and qualitative strategies, and was the research approach followed to develop this master thesis.

3.1 Design Science Research (DSR)

DSR relies on the understanding of organizational phenomena in context, advancing on research by creating and evaluating dual-purpose artifacts that solve organizational and real-world problems, and consequently advance on the existing knowledge base (Hevner et al. 2004). It supports the development of robust and relevant knowledge that can directly or indirectly support managers' problem-solving efforts while urging scholars away from the "ivory tower" towards research with increased relevance (Van Aken, et al., 2016). It has three objectives main objectives; it is consistent with the prior literature, it provides a nominal process model for doing DS research, and it provides a mental model for presenting and evaluating DS research (Peffers et al. 2007).

DSR is a discipline oriented for the creation of successful artifacts (Peffers et al. 2007). According to Hevner et al. (2004), artifacts can be constructs, models, methods, and implementations that are innovative and valuable in such a way that they contribute to advance the field. Furthermore, constructs are the vocabulary that defines problems and solutions; models are representations of the problem; methods define processes; and Instantiations are applications showing that the artifacts can be implemented and solve the identified problem.

The purpose of this dissertation is the development of a new *Method,* thus extending the DSR theoretical foundations and contributing to the DSR knowledge base. The method will be enabled by the application of the Service Design on existing SAP implementation methodologies.

The development of a research methodology (DSRM) required the design of a DSRM process (Figure 10). The DSRM process consists on the following six steps: problem identification (including motivation), definition of objectives, design and development of a solution, demonstration, evaluation, and communication (Peffers et al. 2007).

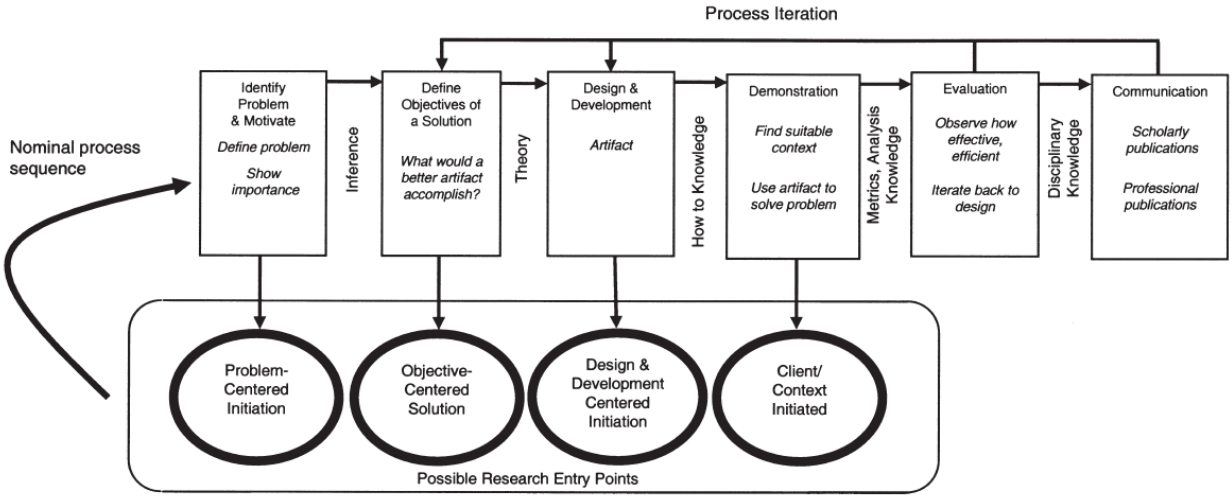


Figure 10 - DSR Process (Peffers et al. 2007)

The application of the DSR methodology (Peffers et al. 2007) to the current study is described on the below in Table 4. Also, our DSR process has the starting point at the *Problem Identification* (activity 1), we consider our research a problem-centered approach, as it emerges from a specific set of issues/problems identified on our SAP implementation methodology that we want to research and improve with the design of an artifact to minimize or even completely solve the identified problem.

Table 4 - DSR stages

Design Science Research Methodology Process (Peffers et al. 2007)	
Problem identification and motivation - Define the specific research problem - Justify the value of a solution - Atomize the problem conceptually so that the solution can capture its complexity	ERP implementation methodologies are following the shifting on the software development industry, from waterfall approaches to Agile approaches. In our case, implementing this shift is challenging due to the geographical context (chapter 4). Current standard SAP implementation methodologies are designed for European and US reality.
Define the objectives for a solution - Infer the objectives of a solution from the problem definition - Knowledge of what is possible and feasible	As a critical success factor of a successful implementation, the methodology must be executable, valuable, and adjusted to the context. Develop a new implementation methodology for developing markets embedding Service Design

Design and development - Create the artifact and determine the desired functionality and it´s architecture - Artifacts are constructs, models, methods, and/or informational resources	The artifact is a new Activate implementation methodology for developing countries. This new *method* was built integrating the Service Design tools into implementation processes. To develop the *method*, a qualitative study was conducted. The approach is detailed on the next three sub-chapters and the results presented in chapter 5.
Demonstration - Demonstrate the use of the artifact to solve one or more instances of the problem - Could involve its use in experimentation, simulation, case study, proof, or other appropriate activity	Due to the COVID-19 crises and the no existence of full implementation projects during the dissertation timeline, it was not possible to apply the method into a business environment. Instead, the output of the research process would be and service blueprint.
Evaluation - Observe and measure how well the artifact supports a solution to the problem - Compare the objectives of the solution with the observed results	The evaluation of the artifact was made through two final interviews, where we collected the feedback and general perception about the methodology and understood the relevance for the business context.
Communication - Communicate the problem and its importance, the artifact, its utility and novelty, the rigor of its design - Communicate the effectiveness to researchers and other relevant audiences	The present paper represents the communication and documentation of the research process conducted, the research results and the application results obtained, their importance, and the future research possibilities generated by it.

3.2 Qualitative Resarch

"Qualitative research of all sorts relies on those who conduct it" (Charmaz 2006). The qualitative approach is based on subjective questions and interpretations of a phenomenon. Thus, reality concerning an aspect or situation under study is socially constructed and is continuously changing. In this type of research, there is a relationship between the researcher and participants that allows the results to be mutually created (Sale et al. 2002). Aligned with the DSR methodology, a qualitative approach was used.

Understanding the entire "implementation experience" of its main actors, it´s main phases is the critical condition to modeling a valuable solution. This chapter refers to the qualitative approach used to gather information to design and develop and evaluate our artifact. The below Table 5 highlights the main activities followed that go from the observation and field notes, semi-structured interviews to co-design, and participatory design in the feedback session.

Table 5 - Qualitative approaches followed

Data Collection	Data Analysis	Purpose
Semi-structured interviews (eight interviews with different actors of the implementation process)	Data analysis with the support of NVivo software (coding, querying, cross-querying data)	• Collect relevant information about implementation projects, main challenges, interactions, best approaches, main tools, and interactions, pain points and success factors
Observations and field notes along 4 months of being part of the SAP consulting team	Based on observation as employee	• Identify and understand the problem • Additional sources of information beside the interviews
Company Internal Documentation and SAP Documentation	Analysis of existing internal documentation and SAP documentation available to SAP Partners	• Interview preparation • Support the interviews information and analysis • Match the SAP Best Practices and methodology with the company process
Two feedback session (interviews with competence leader and Operational Manager (thesis supervisor)	Gathering and analysing the feedback of the meeting	• Present the new *method* • Validation of the process

The results of the above qualitative approach described in Table 5 are presented in chapter 5.

3.2.1 Sample definition

To gather information and better understand the "implementation experience," eight semi-structured interviews were conducted, followed by two feedback sessions.

The participants' definition followed the sample guidelines of qualitative approaches (Charmaz 2006), and the relational behind the definition was the convenience sampling nonprobabilistic (Etikan et al. 2016). The first criterion was the relevant and proven experience on IT projects in the African market, and then it took into consideration the significant and different actors who participated directly or indirectly on the ERP implementation projects.

The first phase included two project managers interviews, two competence leaders' interviews, and four interviews with consultants (two from the company and two outside of the company either working for other consultancy firms either working for final customers).

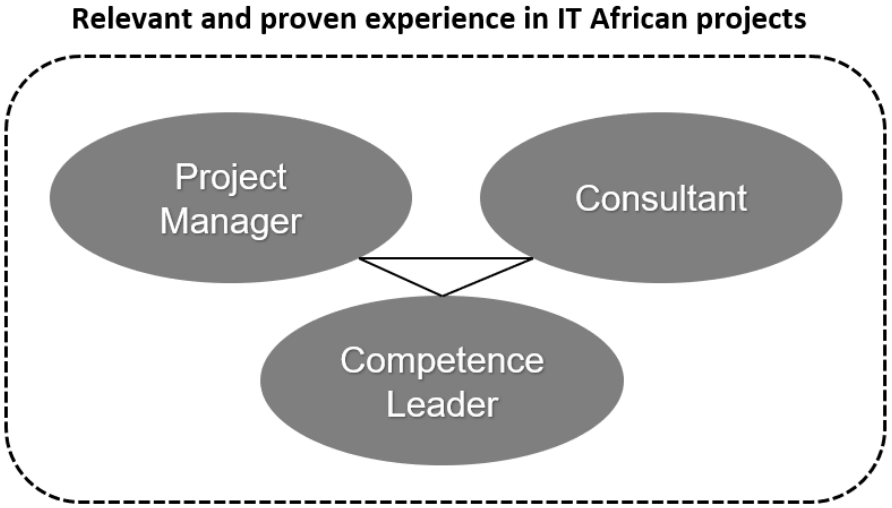


Figure 11 - First phase interviews sample

Competence leaders are the hierarchical superiors of consultancy teams split among functional areas (e.g., Financial team, Logistics, Technical team, Systems maintenance). Each competence leader is responsible for managing a team with specific responsibilities. The participation in the implementation projects can be indirectly, on the choice of the project teams and supporting the teams remotely or directly involved in the implementation projects. The Consultants are responsible for executing the entire projects, and the Project Managers are responsible for the coordination and to guarantee that the projects run smoothly on time and budget.

In the second phase, the feedback session was realized with the operational manager (company thesis supervisor and Competence leaders hierarchical superior) and one competence leader. The interactions between the teams will be further detailed in the 5 chapter of this study.

3.2.2 Data collection

Due to the Covid-19 pandemic crisis and consequently restrictions to people flow, all eight semi-structured interviews had to be done and recorded via Microsoft Teams, between 15.06.2020 and 19.06.2020, the feedback meeting happened in 03.07.2020 with the thesis supervisor, and one competence leader.

After the recording of the first round of interviews, the transcription was made on an open source app otranscribe.com and the analysis was made using a computer-assisted qualitative data analysis software in our case NVIVO 12. Before each interview, a brief contextualization and scope of the study was presented. To ensure compliance and the formal procedures, an authorization for recording the conversation, an “Informed Consent,” was delivered to each participant. The interviews were performed only if the participants read and agreed to sign the form (available in Appendix C). The duration of the interviews went from 25 minutes and 40 minutes. The guidelines used for conducting the interviews are available in Appendix B.

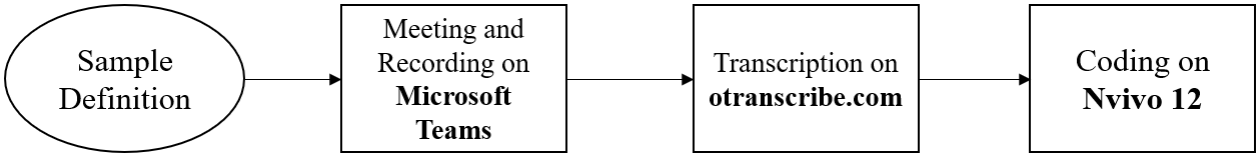


Figure 12 - Data collection process

3.2.3 Data Analysis

Coding is the fundamental analytic process used by the researcher (Corbin, Strauss 1990). As mentioned on the previous the sub-chapter the interviews occurred via Microsoft Teams a virtual conference room software that allows the recording of the meetings, regarding the transcription, it was made with the otranscribe.com a web app that allows the upload of the video/audio making much easier the transcription with easy to use short cuts, finally the tool used to code the semi-structured interviews was the NVivo 12, a computer-assisted qualitative data analysis software.

The interviews content was categorized into segments, represented by nodes in NVivo and can be organized in a structure that would illustrate the significance of the data (see Figure 13). Coding is the pivotal link between collecting data and developing an emergent theory to explain these data. Through coding, is possible define what is happening, define patterns and understand the meaning of the data (Charmaz 2006). The coding of the text can be done word-by-word, line-by-line or incident-by-incident (Charmaz 2006) and one piece of text can be coded to multiple categories, enabling furthermore complex cross analysis of the nodes.

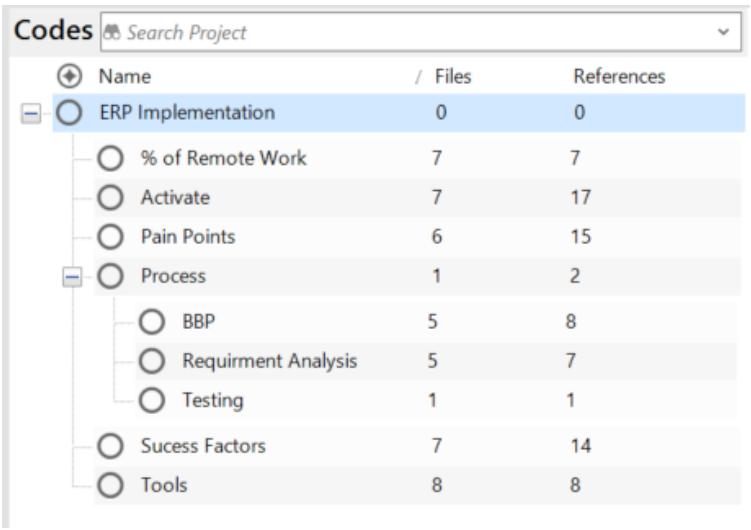


Figure 13 - Example of Coding followed on the project

Visualizations play an essential role in qualitative research; they are used to explore ideas and present the connections in the data. Nvivo allows the creation of several visualizations such as mind maps or project maps or Concept maps. Therefore, in this next chapter, we will use mind maps to confirm the structure of our nodes and project maps to explore and present the connections in our data.

The generic results of the Nvivo qualitative research and data analysis are explained in chapter 5.

4 Challenge contextualization: Developing Countries and SAP methodologies

4.1 Angola: Brief overview

The objective of this project is not to do a dissertation about the socio-economic and demographic situation in Angola, but Angola is indeed part of the challenge, so this chapter will highlight some specificity of the country, based on some industry reports and some existing literature.

In Angola, the political, economic, and social transformations marked by the different historical contexts as colonialism, socialist politics, civil war, economic liberalization and multiparty system had strongly impacted several sectors of the country as education, the economy the public policies and ended up with a country with sever issues corruption (Liberato, 2012).

Angola has just opened the economy to a "market economy" in 1991. This opening allowed the capture of some foreign investment, increasing the need for qualified workers and consequently boosting the numbers of students as employment rates (Liberato, 2012).

The economic context is still seen as one of the major risks for international companies operating in Angola. Despite the efforts made by the current government to diversify the economy, the country remains heavily influenced by the oil industry. Indicators from the National Bank of Angola (2020) show a retraction of -0.8% on the GDP with an unemployment rate of 30% as a rampant inflation rate of 20%.

Furthermore, on the CFO Survey Angola, an industry report conducted by Deloitte ended up highlighting the significant concerns about the external environment, related to Angola context, and the internal constraints that each company faces. Some external concerns about the high-interest rates, high costs of financing, and the government policies effectiveness but more relevant to this work are the internal concerns. At the top of the first one was, "Find and retain talent" immediately followed by "insufficient skills of support staff" additionally this lack of talented people with the lack of technical skills among the existing teams according with the inquiries lead to heavy dependency on the top employees, low autonomy and consequently a low productivity.

About IT skills, the Information Technology is part of the political strategy for the next years with the creation of "Livro Branco das Tecnologias de Informação e Comunicação" a set of policies and guidelines across all sectors to modernize the country. A relevant issue is that the IT learning is still not universal in the Angolan national education system (Júnior, 2013).

4.2 ERP project methodologies

With the contextual challenge briefly described in chapter 4.1, the second part of our problem is shifting from ASAP to ACTIVATE. This chapter will highlight the main phases and objectives of each SAP implementation methodology, as end up connecting those two methodologies to the approaches explored in the literature review.

4.2.1 ASAP

ASAP Methodology (Accelerated SAP) is a standard SAP related project preparation and systems implementation method, developed and updated by consultants that are involved in implementing SAP software products. The purpose of ASAP is to help project management design SAP implementation in the most efficient manner possible, by effectively optimizing time, people, quality, and other resources. It divides the implementation process into six phases (Figure 14) briefly explained below (SAP SE).

Figure 14 - ASAP Methodology (source: SAP SE)

Project preparation - This phase provides initial planning and preparation for the project. Each project has its unique objectives, scope, and priorities. The deliverables included in this phase are; setup of project governance, project plan and project schedule (SAP SE).

Scope validation (Business Blueprint) - This phase aims to achieve a common understanding of how the company intends to run SAP to support their business. Through the Business Blueprint, a detailed document containing all requirements derived during initial brainstorming and initiation meetings. This is also where the project team revises its objectives and goals while setting a schedule for each (SAP SE).

Realization - The purpose of this phase is to implement all the business process delta requirements defined during the Scope Validation phase. The team configures, develops, tests, and documents the solution. Before the release to the next phase, the solution is fully end-to-end integration tested and accepted by end-users (SAP SE).

Final preparation - This phase's purpose is to complete the cutover activities (including technical and load testing, end-user training, system management, and cutover rehearsal activities) to finalize your readiness to go live. The Final Preparation phase also serves to resolve all remaining critical issues. If successful, the customer is ready to run their business in a live SAP System (SAP SE).

Go-live support - This phase aims to move from a project-oriented, pre-production environment to live production operation and provide sustained support to business users to aid their transition into the new environment (SAP SE).

Operate - The purpose of this phase is to fine-tune the application lifecycle standards, processes, and procedures established during the project and align them with operation needs (SAP SE).

Table 6-Waterfall principles in ASAP Methodology

Waterfall Principles in ASAP
- Emphasis on “formal” and structured documentation - No overlapping of phases - Requirements are gathered and agreed on the signature of the Business Blueprint - Strict time/periods planning - Strict “Scope Management” requirements raised after the Blueprint stage are considered out-of-scope, and by this discarded, postponed, or charged and developed separately to the project - There´s a clear border between the consultancy team and the customer

4.2.2 ACTIVATE

SAP Activate is a methodology used in S/4 HANA (last product version of the SAP ERP). It is a combination of SAP Best Practices, Methodology, and SAP Guided Configuration delivered with a reference solution. As a successor of the ASAP, SAP Activate is a transparent process of structured and solution-specific practices, carried out in a standardized environment, that puts business users in the driver’s seat. It reduces project lifecycle only in 4 phases (Figure 15), reducing the implementation time, and follows a much closer approach with the business user delivering up to 10 deliverables per phase. The above model represents the SAP Activate version (on-permise) being followed on this specific research. There´s also the SAP Activate for cloud products that include more two phases Discover and Run (Denecken 2020).

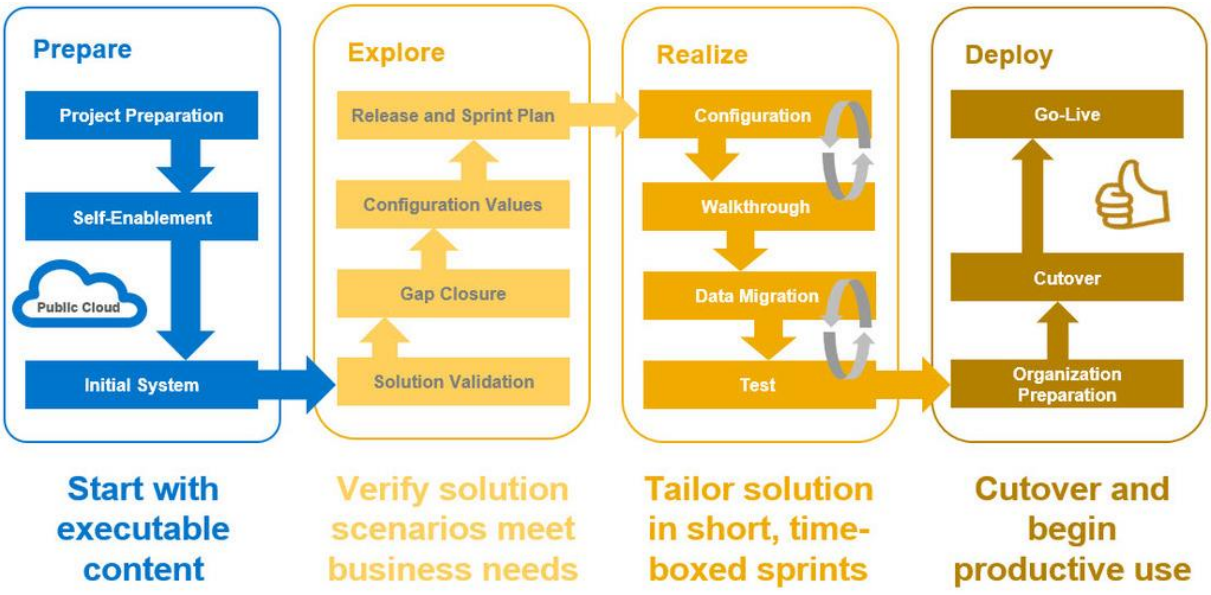


Figure 15 - SAP ACTIVATE (on-permise)

In the **Prepare** phase, the project governance is defined, the resources and the definition of roles and responsibilities are shared among the project team. The details of the management plans for running the project are presented (Denecken 2020).

The **Explore** phase has the objective of identifying the fit of the best practices based solution. So making use of a pre-configured SAP machine based on the best practices, the fit/gap

analysis is performed capturing the delta configuration requirements, identifying gaps, and configuration values. Following an inverse of idea of the previously presented *Test Driven Development* approach, the requirement and the business processes are tested by the business user, and in case of success the standard solution is adopted, the need of software development will just occur if the test does not fulfill the business needs. All these are gathered by the project team into one backlog that will be handled in the next phase (Denecken 2020).

During the **Realize** phase, the project team will use an agile approach to iteratively and incrementally build the functionality from the backlog. This build follows the prioritization given by the product owner (e.g., the business users) that continue to be involved with the project team during the sprints. During the build, the team completes the remaining configuration and development on top of the best practices content. Additionally, the team performs the unit and string testing during the sprints. The Realize phase covers all the build and test activities required to prepare a release of functionality into production. This includes a full integration test and user acceptance test before the team can continue the Deploy phase (Denecken 2020).

The objective of the **Deploy** phase is to launch the production environment and confirm readiness to switch into business operation with the new solution. Once the customer has switched over into the new environment, the project ream supports the customer IT and business. The team stay on-site for a pre-determined period of time, which is usually defined in the contract or agreed as part of the plan to the transaction to operations (Denecken 2020).

Table 7 - Agile Principles in Activate

Agile Principles in ACTIVATE
- The deliverables are divided among the phases, - " Puts business users in the driver's seat," the customer get access to a standard system on a first stage, - Test driven development, - The Gap Analysis from the Explore phase, generates the product backlog that should be developed with sprints following a SCRUM approach on the Realize stage, - There is no focus on the documentation, the focus is in the processes

Why change from ASAP to ACTIVATE?

- A new product (S/4 Hana) brings a new implementation approach.
- IT Market, as described on the literature review all IT market is moving from Waterfall approaches to more Agile approaches.
- Activate can lower the costs as the timeline of the implementation projects (SAP SE).
- Activate tries to address some common pain points of the traditional implementations for example, the "big-bang" approach the lack of involvement of the business users on all the implementation processes.

5 Results of the qualitative study

This chapter will present the results of the qualitative study performed to Design and Develop our new *method*. The first sub-chapter focus on the results of the interview coding (explained in chapter 3.2) and the modelling of the implementation process in its current state.

Based on the information gathered an analysed on the qualitative study, we have developed a customer journey and a summary containing all identified opportunities of improvement.

5.1 Understanding the Implementation Experience

All qualitative study was conducted to obtain as much information about the "Implementation experience," the primary process and steps, the most common issues, pain points, success factors, tools, and potential improvements.

The interview sample is attached to Appendix B. Also, the chosen questions for the interview attempted to be aligned with the objectives of the research. There were not two equal interviews all add something new and unexpected, and the order of the interview questions dependent on the interviewee answers.

The eight semi-structured interviews were transcribed and coded into the NVivo software. Therefore, the text was broken into categories, following the principles of coding explained on the sub-chapter 3.2.3. The followed coding on our qualitative research is mapped and connected on a mind map on the below figure 16. The first level of nodes represents the main topics followed in our interview (Appendix B); the second level detailed on the further chapters is a consequence of the coding followed in categorizing the interview answers.

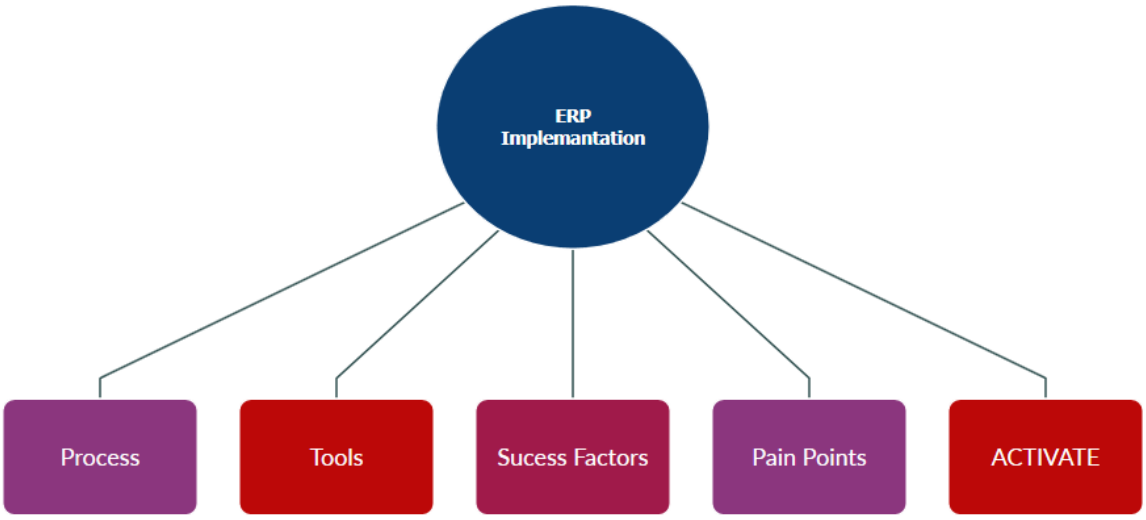


Figure 16 - NVivo Mind Map for node structure, first level

The above categorization in figure 16 reflects the main topics discussed in the semi-structured interviews. On the category "Tools," the answers were unanimous in the eight interviews seven used exclusively as implementation tool the SAP ERP, as a base of work and object of all implementation activities, the SAP official documentation and forums for research on SAP webpage and the Microsoft Office 365 ecosystem, with some variances depending on the responsibility of each participant, as example, Visio for consultants and Microsoft Project for Project Managers, all other Ms 365 applications as Teams, Word, PowerPoint, are transversal to all participants. One of the participants outside of the organization referred to a task/project manager called ClickUp.

The next category refers to "Success Factors" factors that on the opinion of the participans and based on their implementation projects experience were critical for the success of the projects. The results of this coding are presented on a findings summary by Saldaña, J. (2013) on table 8.

Table 8 - Findings summary "Success Factors" (Saldaña 2013)

Findings	References	Meaning	Evidence
1. Customer Consultant relation	**4**	Mutual support, constant and precise communication to get customer commitment and availability	"Undoubtedly, the consultancy and client involvement went well, and the consultancy was always very available, alongside the client" (A2)
2. Requirement Analysis	**3**	Broader understanding of the Business and about the ERP capabilities	"A team that does a good review of requirements, and that understands business processes has to know the business" (A8)
3. Systems experience at an early stage	**3**	Preparation of a Demo / Testing machine at the beginning of the project	"the client puts his hands on the system and goes on looking for cases to certify that thing, and he will even realize the need for specific developments" (A1)
4. Project Management	**2**	An effective team/ or Project Manager planning, scheduling and tracking all project activities	"Strong management both on the side of the consultant and the client, making a substantial monitor for both planning and engagement on the customer side." (A5)
5. Change Management	**2**	An effective team, preparing, supporting, and helping the customer to adapt to the new ERP	"Change management, changing the mindset, in the beginning, because this effectively takes time and dedication from the client" (A2)

6. Business expertise	**1**	Full understanding of the business and the customer needs	"The service order does everything you want, but if you don't know and business, forget!" (A8)
7. ERP Standard Approach	**1**	Have a different ERP just on the core process, keep normal and common process as SAP standard	"At least start from the standard" (A1)

By opposition to the "success factors," we have also analyzed the "pain points" of the implementation process. The main highlights of this coding phase were the "negotiation process" between the consultants and the key users on the requirement analysis. Convince the key users to adapt the standard SAP process or present the development specification, and costs are one of the most stressful situations in the implementation process. Also, the consequence of the current pandemic context, performing some activities as the requirements analysis, cause some insecurity on the customers. In some big projects, the bureaucratic procedure of relationship, of governance, and working was considered a pain point on the implementation projects. By last other common pain points more related to the ERP itself are the integrations with other systems.

When speaking about the "Activate" the implementation methodology, the master thesis proposes to "update" to the company context, positive and negative opinions raised, so we coded both insights into sentiment analysis. The results of the NVivo matrix coding query on the table 9.

Table 9 - NVivo Matrix Coding Query (Activate)

	A : Activate	B : Positive	C : Negative
1 : Experience:Activate = Already used Activate	9	7	2
2 : Experience:Activate = Never used Activate	8	3	5

From each interview file, a case on Nvivo was created with the attribute "Experience" with the values "Already used Activate" or "Never used Activate." With this attribute, we were able to better understand the sentiment about the Activate on the participants. Through the analysis of the matrix coding query, as an example, we could see that the majority of the negative references (5) raised from participants that never used the methodology.

The first level of the coding end´s up with the category "Process," the objective of this category was to understand how the consultants divided the implementation process, Figure 18. This category was subdivided into the "Testing" highlighted as a critical phase of the projects, mainly due to the integration of different modules and different software and, in some cases, the first shock of the user with the system. The "Requirement Analysis" also coded on the category "Success Factors" and "BBP" that stands for Business BluePrint (the formal document that closes the Requirement Analysis phase into the desired future ERP scenario).

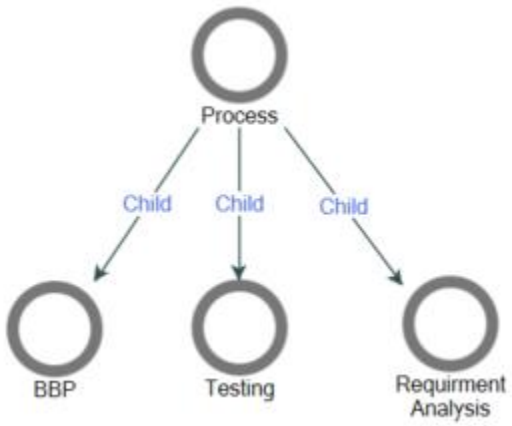


Figure 17 - NVivo project map "Process"

The sub-category “BBP” related to the mother category “Process” was highlighted in the interviews with several different perspectives. The term BBP refers to the Busines BluePrint, one of the implementation documents that document the company's business processes that will be implemented in the system. The findings of this sub-category are presented in table 8 with a findings summary (Saldaña 2013)

Table 10 – Findings Summary “BBP” (Saldaña 2013)

Findings	References	Meaning	Evidence
1. Ambiguous	**2**	By using natural language, even on the same cultural groups, the business blueprint can be a source of ambiguity.	“It is not the BBP that makes things happen, and the BBP signed without showing a machine, will be worth very little because later, what will happen is that people have different interpretations of the process even when written” (A1)
2. Time consuming	**2**	On a big project, it can take up to 3 months	“it takes more time to do the requirements review and then it takes more time to do the document, and then you have to wait for the customer to read ask questions and so on, don't move on until you have the BBP signed” (A2)
3. Protection	**2**	Some consider it as a security for who is doing the implementation.	“the client sometimes says one thing, and it is another, definitely the BBP gives this safeguard” (A4)

4. Dispute root	**1**	Later on the project, it will be a source of conflict between consultants and customers.	"And that's not all, we're going to waste ourselves out as follows; "Ah, but you told me that it did this, but it doesn't. - that is not in the scope in the scope is this "in our understanding ... oh no but I thought..." (A1)
5. Signed but object of further changes	**1**	Even being a milestone on the project and object of signature for both parts involved, it is common later one the project to do the process in a different way.	""When they really understand what it's capable of and what we're saying, there are always changes at a stage before Go-live." (A7)

5.2 Customer Journey Map

Based on the information gathered from the qualitative study, the interviews further analysis through Nvivo, the field notes, the observations all the documentation (internal and from SAP), we build up this Customer Journey Map Figure 18.

The CJM based on the Richardson, A. (2010) model, illustrates the steps, feelings, barriers, and main activities of our customers during the implementation experience. Additionally, we´ve included a line with an illustrating the "*feelings*" of the customer

On the current project, the CJM illustrates the current point of view of the customers (AS-IS) regarding the delivered and ongoing projects. The objective of using this map is to better understand the customer experience and build up the base to improve the implementation experience.

The "*Implementation timeline*" refers to the main stages of the implementation process currently applied to the company projects that are still directly connected with the ASAP methodology. The "*Activities*" highlight the main activities performed by the customer on each stage, as the activities performed to advance to the next stage. This map also brings the "*Motivations*" of the customer and the main "*Questions*" and "*Barriers*" that are preventing the customer from moving to the next phase.

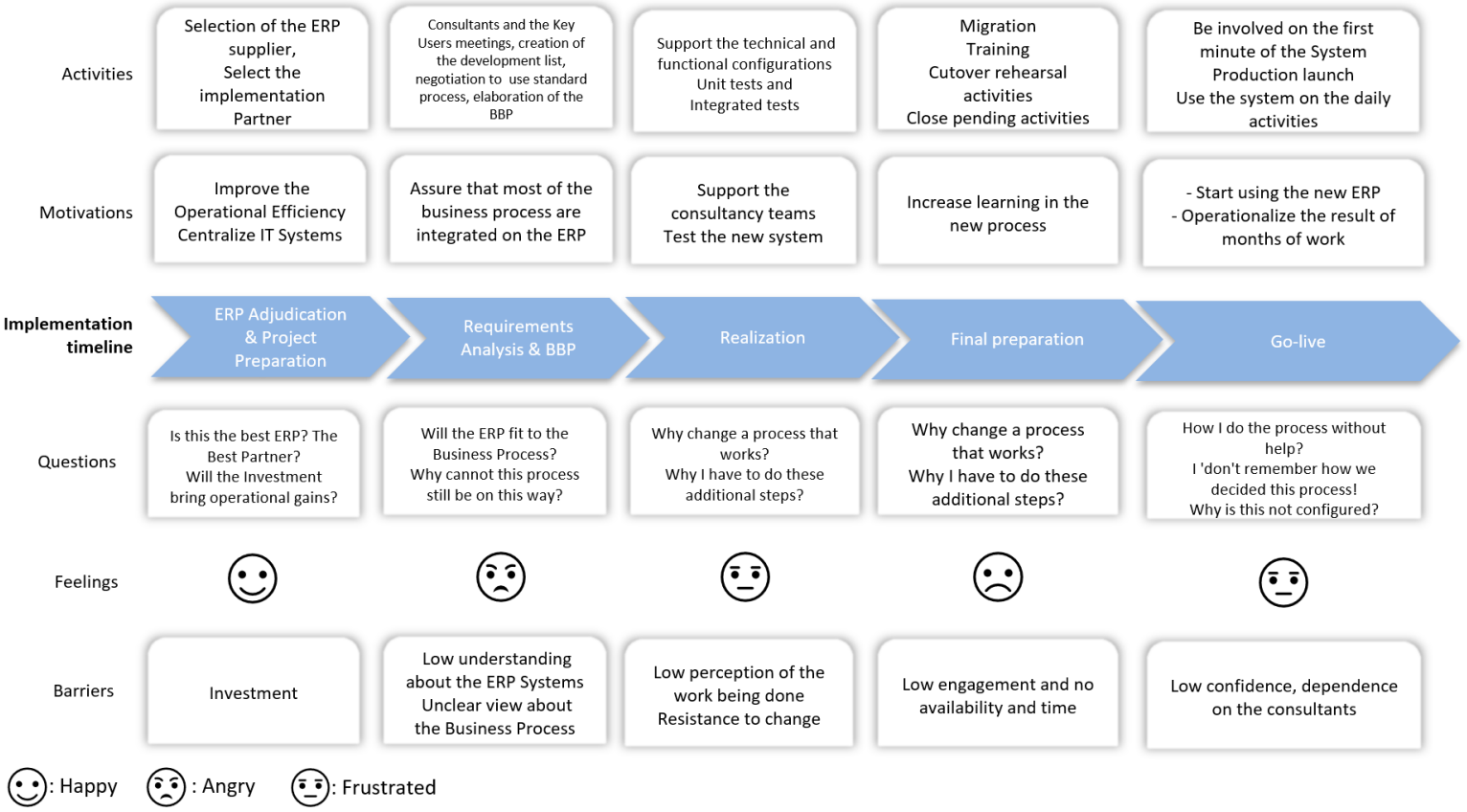


Figure 18 - Customer Journey Map

Our journey timeline included the first engaging with the ERP that is the "ERP Adjudication," the first moment where the customer assists the commercial Demos and takes the decision of which ERP and which consultancy company we will sign with.

5.3 Opportunities of Improvement: Summary

With the qualitative study and the Customer Journey Map done, to close this chapter, we have summarized the "opportunities of improvement" raised from the two previous exercises on the below Figure 19. To have a clear overview of the problems that our method will address and try to solve, we have considered the implementation phases and categorized them as related with;

- **Context Angola:** common issues consequence of the cultural, demographic, and social characteristics of this developing country.
- **Methodology:** common issues that the methodology does not avoid or does offer a solution.
- **Company:** common issues consequence of the company process, people, and management.

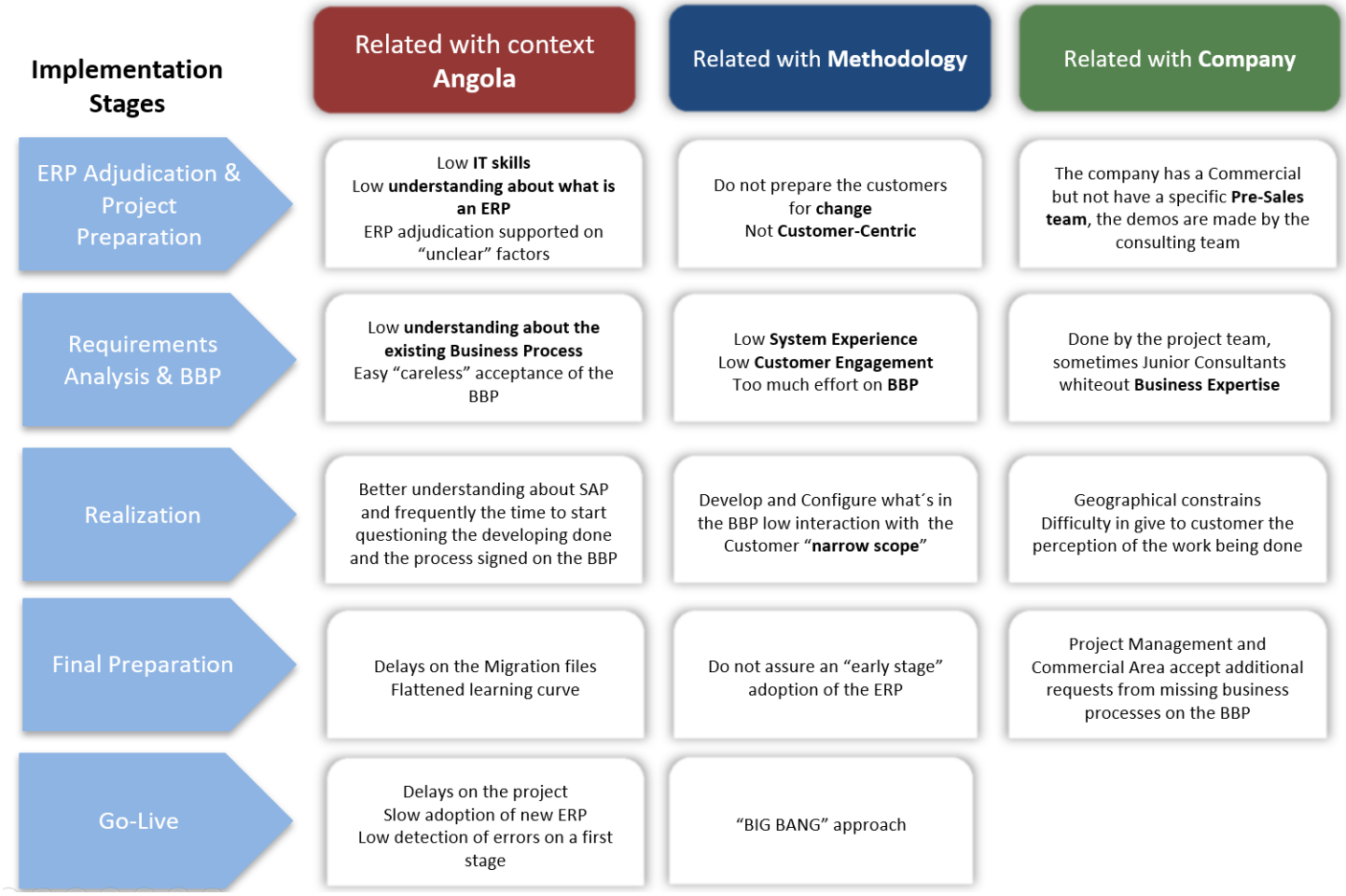


Figure 19 – Opportunities of improvements: Summary

Despite characterizing the problems as "*Related with the company*," our artifact will be built, taking into consideration the "*Context Angola*" and the "*Methodology*" issues just by that we can generalize and provide a solution that can be scaled and used in other contexts.

The main issues related with Angola are mostly related with the lack of IT skills and not a systemic view about the business process, this fragility leads to that, for example, the Business Blue Print is signed unconsciously, without understanding the impacts on their daily operations, what on later stages lead to disputes.

Regarding the methodology that nowadays is ASAP, the waterfall principle does not engage the customer into the implementation project. The ASAP focused on the processes rather than the customers, and does not prepare the customer for the change. The system experience occurs at a later stage.

The new method Activate Plus is build-up integrating Service Design, taking into the consideration all results presented in this chapter, and demonstrated in the next chapter.

6 Activate Plus

This chapter will present the DSR demonstration results phase, introducing the new proposed method, and the application to the business environment.

Activate Plus is build-up integrating Service Design into the new pre-delivered SAP Activate methodology. As a starting point we consider five principles to build up our method;

- **User-Centred:** "Puts business users in the driver's seat," the implementation should focus and be adapted to the customer needs, starting from an extensive user blueprint for defining specific strategies for each user.
- **Agile:** Customer interaction, open communication, minimum documentation, Scrum the Agile principles should be on the core of the Activate Plus.
- **Iterative** implementation process with more flexibility for change
- **System Experience instead documents the system:** "get hands dirty" the user should experience the system from the first moment.
- **Standard approach on universal processes, develop or "tailor-made" on core process:** An ERP implementation is much more an adoption project than a software development project.

The new method illustrated in Figure 20, where the "origin" of the activities is colour marked, as SAP Activate or Service Design. The following chapters present each stage, the method appliance, and the method evaluation.

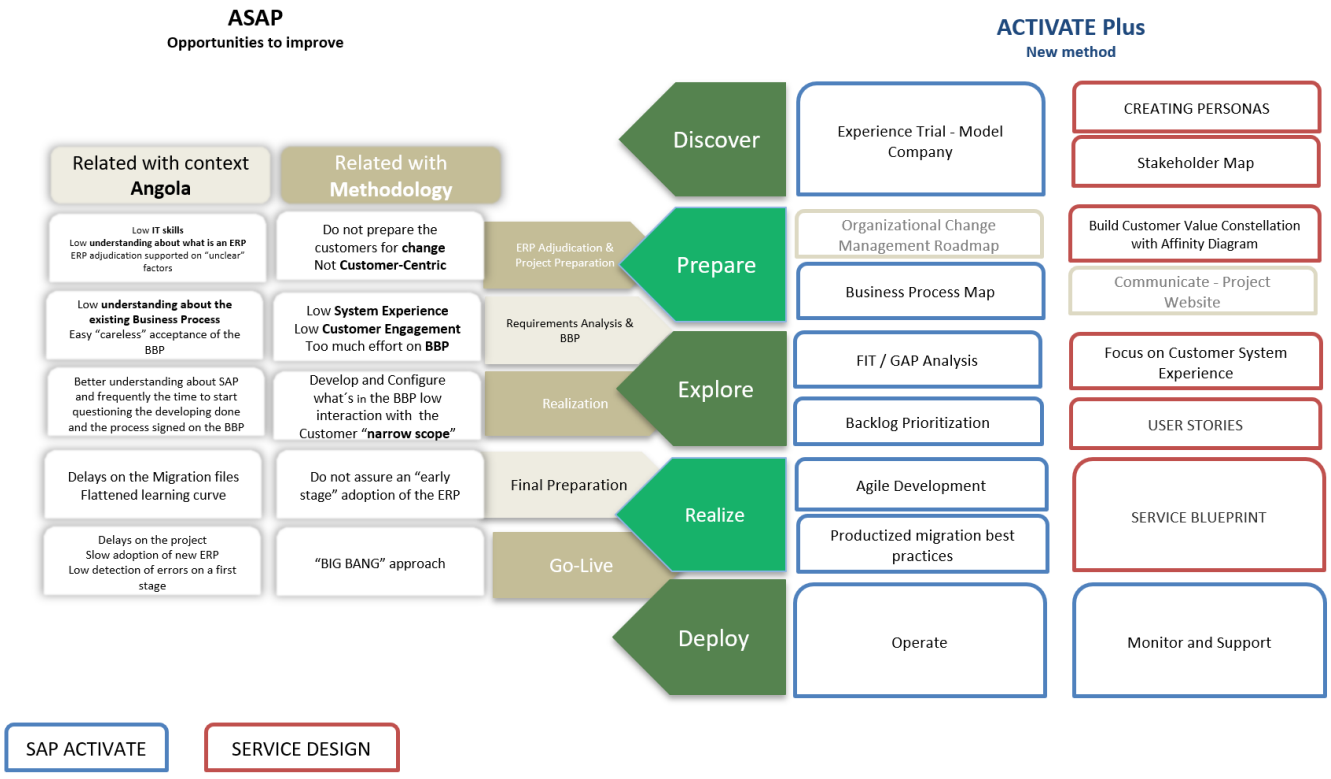


Figure 20 - New ACTIVATE Plus

6.1 Activate Plus: Phases, Objectives, and Tools

Discover

The Discover phase aims to discover the solution capabilities, understand the business value and benefit for the customers, this phase is inspired on other models as SAP Activate for cloud solutions, and fit's precisely on the objectives and principles defined for our new method. Experience the ERP before starting to implement, in parallel with better understanding and engagement of the users through the utilization of Service Design tools as "Personas," and the "stakeholders map" are the objectives of this phase.

The Discover phase is operationalized through three activities, the first the installment of an SAP Model Company and the second Personas elaboration and then the stakeholder map.

Table 11- Discover phase

Experience Trial – SAP Model company	**Objectives**
SAP Model Company is a prepacked, environment that offers a ready-to-use end-to-end processes from specific industry's. A very powerful demo that works an accelerator that can squeeze the delivery time, increase the customer engagement, improve the "fit to standard" (for example, reducing custom development efforts), and be the base of the *Explore* stage.	• Simplify and increase the speed of the implementation project • Experience SAP from the first moment • Better understand the Standard functionalities • Boost the curiosity and the willingness to learn on a very early stage • Engage the user on the project • Smooth training gaps in the Angolan context
Creating Personas	**Objectives**
As the consultancy team is mainly from Portugal, a different country requires a deeper understanding of the users. Personas are a powerful tool to represent a group of people with shared interests, common behavior patterns.	• Identify behaviour patterns that can impact the ERP implementation • Identify training gaps, to further develop training strategies • Define change management strategy • Persona per functional role
Stakeholder's Map	**Objectives**
Understand the customer ecosystem and in which way SAP can be embedded in this ecosystem. Stakeholder's Map illustrates the various stakeholders in an experience/ service and also on this case, the Stakeholder's Map of the project.	• Map the various stakeholders involved in the user experience • Identify business opportunities • Understand the future stakeholder´s map after the SAP implementation (which will remain which will change)

To conclude the Discover phase, customers and consultancy teams (project management) agree on;

- Implementation scope and solution model
- Project timelines
- Personas
- Stakeholders Map

Prepare

The purpose of this phase is to provide the initial planning and preparation for the project. In this phase, the project is started, plans are finalized, the project team is assigned, and work is underway to start the project optimally. Regarding the activities Business process maps are part of the SAP Activate existing method, the Customer Value Constellation and Affinity Diagrams are Service Design tool embed on this stage.

Table 12 - Prepare phase

Business Process Map	**Objectives**
Drive and agree on the scope for the start of the next phase. During this work, the process map builds the foundation for the process hierarchy - a decomposition of the process design - which is reflected as scenarios, processes.	• Mapping Business Needs • Map Business Processes and match with existing SAP best practices best practices
Customer Value Constellation and Affinity Diagrams	**Objectives**
Through some brainstorming sessions, understand form the user point of view what he expects from the SAP, which service offerings, which value will an ERP bring to the company? Collaboratively built the Value Constellation with affinity diagrams for each functional area.	• Map service offerings from user point of view • Evaluate the understanding of the ERP from the customers • Understand if on the next phase it will be needed to adjust/balance expectations

At the end of the Prepare phase the below objectives should be achieved;

- Project goals, a high-level scope, and a project plan, organization, and governance
- Executive sponsorship
- Definition of roles and responsibilities for the project team
- Project team training strategy and start project team training

Also, as a result of the qualitative study, Activate Plus needs to embed the methodology, elements of change management, and ensure proper communication channels within the project.

Explore

With intensive solution validation workshops, this phase aims to perform a fit/gap analysis to validate the solution functionality included in the project scope and confirm that the business requirements can be satisfied. Identified gaps are specified with user stories (SD improvement) and are added to the backlog for use in the next phase.

Table 13 - Explore phase

Fit /gap analysis	**Objectives**
Validate predefined scenarios, processes, and enhancements; identify potential gaps between delivered product and customer requirements. The deliverable only captures requirements for gaps. It follows an iterative approach.	• Experience the system • Validate the standard solution • Identify gaps that will need further custom development • Refine business requirements • Construct the Backlog (Gap list)
User Stories	**Objectives**
Identified needs from the customers that are not supported by the standard can be addressed with this tool that is either used on Service Design either on Agile. User stories are a common tool used to define requirements.	• Specify requirements • User point of view • Be the base for the further service blueprint developed by the functional consultants for the future custom developments
Backlog Prioritization	**Objectives**
A possible approach is to use the MSCW prioritization framework (Must-Have, Should-Have, Could-Have, Would-Have) for an initial grouping of the backlog requirements. The business process owner is responsible for determining the relative priority of each requirement.	• Prioritize of each requirement by business importance • Provide the sequence of requirements for the next Realize stage

This phase should also used to identify master data and organizational requirements and confirm business process.

Realize

This phase uses a series of iterations to incrementally build and test an integrated business and system environment based on the business scenarios and process requirements identified in the previous phase. During this phase, data is loaded, adoption activities occur, and operations planned.

Besides the Agile approach that should be followed on this stage, already as the recommended approach from SAP Activate, as a support for the development on the custom code, we ´ve included the Service Experience do design the service interfaces of each development.

Table 14 - Realize phase

Service Blueprint	**Objectives**
In collaboration with the customers, and based on the user stories, the functional consultants can design the user interface for future custom development through Service Experience Blueprints, that can be a huge support tool for the development team.	• Build up better and customer centred interfaces • Map interaction experience and get a broader view. • Provide additional support for the custom developments
Agile Development	**Objectives**
SCRUM-based approach to building solutions; this allows for an iterative implementation process with more flexibility for change. Include on the project team an Product Owner Development team fiscally on the project locations	• Resolve the Backlog through sprints • Close to customers and functional consultants on evolutive process • Ensure good fit of the built solution to the requirements from the backlog

The objective of the realization phase is;

- Implement the solution in the development environment using the incremental build in time-boxed iterations
- Conduct overall end-to-end testing of the solution on the quality environment
- Setup production environment
- Prepare data migration
- Finalize user training

The following stage on the Activate Plus it´s the **Deploy** phase. The purpose of this phase is to resolve all crucial open issues, set up the production system, confirm customer organization readiness, and switch business operations to the new production system. This stage is not presented on o sub-chapter as it is resumes to the previous mentioned activities and in fact our method does not changes anything to the original SAP activate methodology.

6.2 Activate Plus applied to the company process

Due to the COVID-19 crises and the no existence of full implementation projects during the dissertation timeline, it was not possible to apply the method into a business environment. So, the demonstration results were made through the service blueprint of the tree first phases Activate Plus.

The Service Blueprint follows the Bitner, Ostrom, and Morgan (2008) model with five components (Physical Evidence, Customer actions, Frontstage and Backstage actions and support process) with the draw of tree lines, line of interaction, line of visibility and line of internal interaction.

The first Service Blueprint (see Figure 22) refers to Discover phase. The Blueprint map the main activities done by the customer on this stage, the physical evidences in this case Model Company (an SAP demo system) and the SAP Use Cases, a pre-delivered SAP documentation to execute and test the pre-configured business scenarios, as identify the activities and actors that are part of this phase.

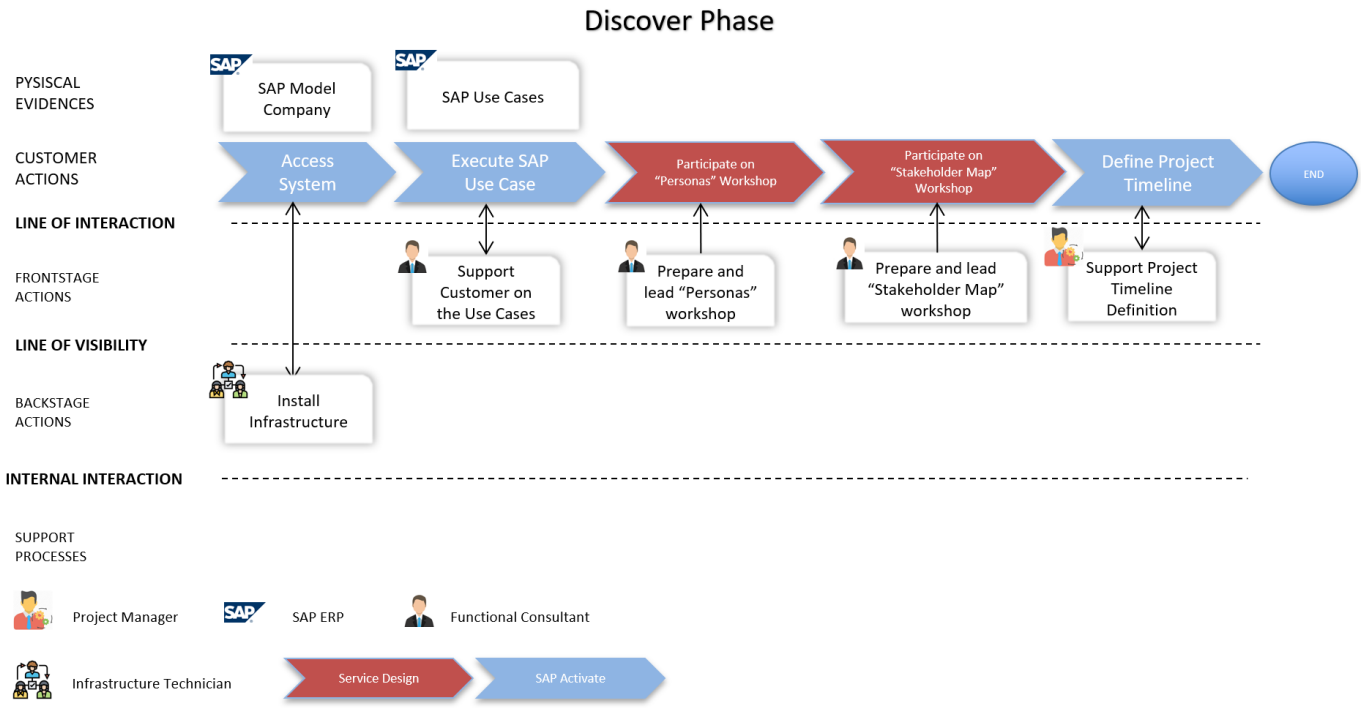


Figure 21 - Service Blueprint "Discover phase"

Aligned with the major objectives of our method, by delivering a more User-Centered more engaging a more supported on the standard practices, giving access at the first moment to an SAP Model company will make the user more familiar with the system, with the standard configuration and with the full potentialities of SAP. This will be a huge facilitator for the Explore phase (as it will not be the first contact with the system) and will definitely enhance the customer's learning curve. To support this activity, use cases are provided, the functional consultants are available to answer questions and explain the process. Following the first system experience, is time to better understand the users, and this is done through a workshop to create the "Personas" that involves functional consultants and the customers. The next step also performed by the functional consultants together with the users; it's the identification of the stakeholders of the role as the project.

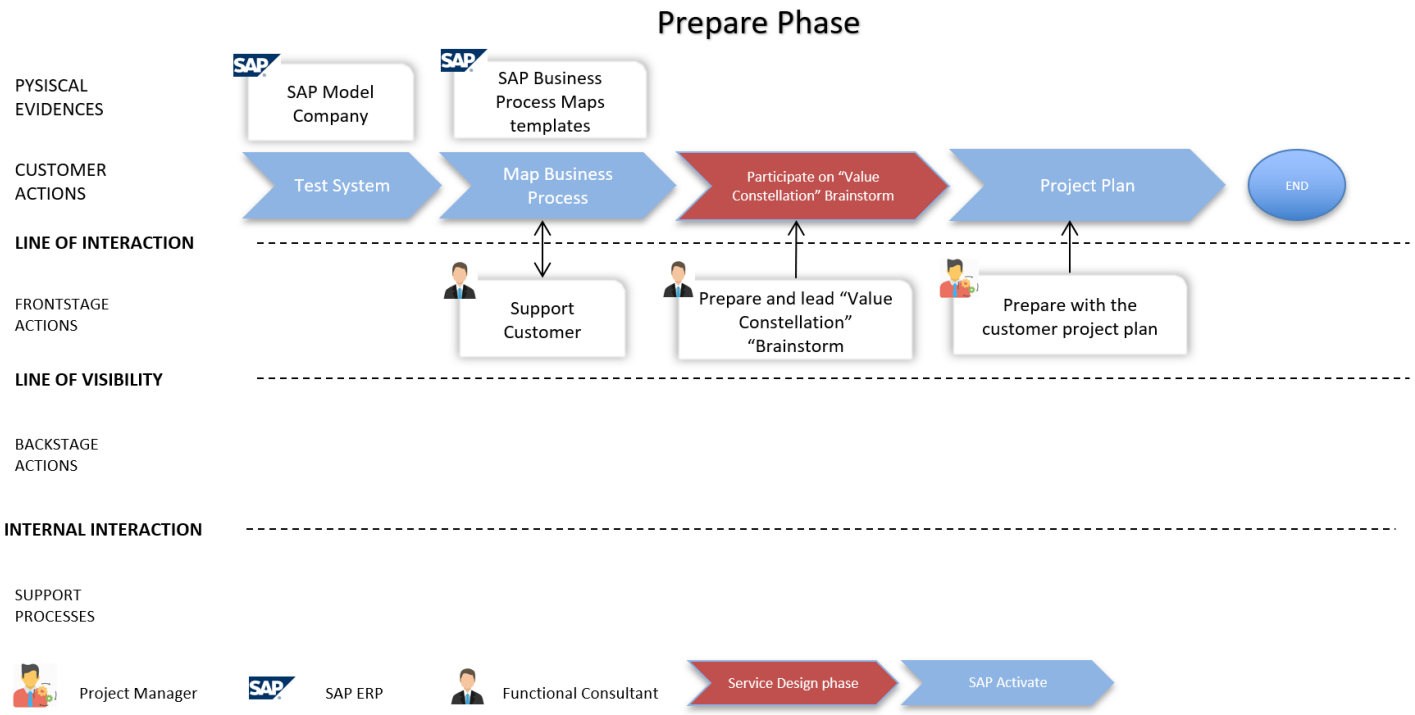


Figure 22 - Service Blueprint "Prepare phase"

Following the system experience approach, the "Prepare" phase keeps the testing and experience approach on the SAP Model Company, the first activity of the method phase. The second activity is mapping of the busines processes on a pre-delivered template (Physical evidence) the user will map all business process with the support of the functional consultant. Then the phase introduces the service design tool "Customer Value Constellation," this activity would be done through a team brainstorm where the objective is mapping the service needs, understand the customer expectations and even evaluate the understanding of the ERP from the customers. The conclusion of this stage end up with a project plan made together with the customer and project manager.

The Explore (Figure 23) phase is characterized by the intensive solution validation. The execution of the testing scripts is made together with the functional consultants, process by process. All identified business process that the customer considers that are not covered by the previous step is demonstrated to the functional consultant, and further discussed, if there´s not an agreement the customer should create a User story the result of this entire process will be the Gap List, the list with all custom development that will have to build on the next stage This gap list will be prioritized on the next step, the "Backlog Prioritization," on this step, product owner, eventually supported by someone of the development team, will prioritize the development list together with the customer, evaluate the effort, the price, and the importance for the business. The rational behind this stage in particular way it´s the opposite to the Test driven development presented on the literature review, on this case if the test is successful there will not a custom development

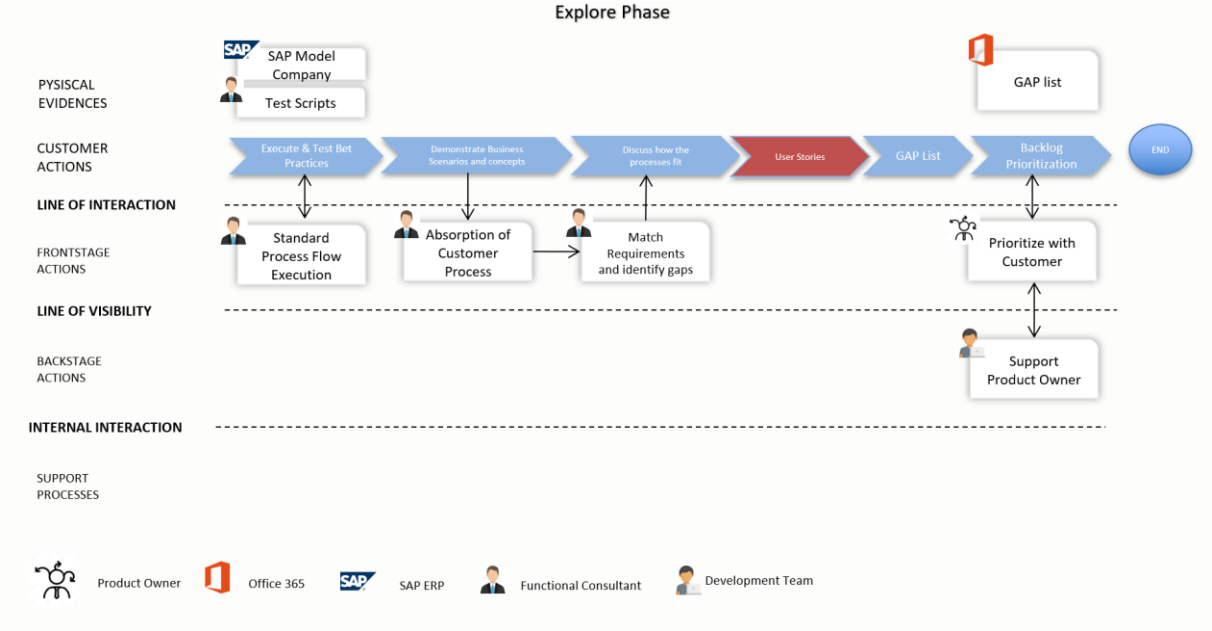


Figure 23 - Service Blueprint "Explore phase"

Those service blueprints present the implementation stages on high level approach, it would be possible drill-down and create a new service blueprint for each one of those activities.

6.3 Method Evaluation

Following the DSR methodology (Peffers et al. 2007), it is necessary to evaluate it before communicating the method. Activate Plus was evaluated in terms of a process of work and relevance\efficiency for the business environment. Between the interviews and the method demonstration, on those weeks, all participants showed satisfaction in participate and contributing to built up a solution. The final feedback was gathered through two interviews, one with the operations manager responsible for the SAP team, and my thesis supervisor and the other one with my competence leader responsible for one of the functional teams.

According to the feedback session:

- Start doing projects using the Activate methodology; it´s a milestone for the SAP team, Activate Plus brings that.
- Tools as user stories, personas, and stakeholders map make very sense on the implementation context, considering the usual problems that we face while doing projects in Angola, other´s tools presented would need additional training for the consultancy team
- In a general way, it fits to Angola, with the correct placement and preparation of the team, the methodology is possible to follow.

The objective of embedding Service Design into the Activate methodology, which can address specific constraints of the developing countries, bring the customer to the center, allow the delivery of better implementation services, can be considered successful either the feedback was positive, either we can state that we have an artifact that can be the solution for the identified problems and for future implementations.

7 Conclusion and future research

Implementing an ERP is much more an adoption project, guiding the customer to standard industry best practices than a software development project. Traditional implementation models are quite the opposite of the "human-centered," "iterative" and "co-creative" approach of the Service Design, and that's precisely the way how Service Design can support projects in developing countries. Bringing a Service Design approach to the SAP Activate methodology can clearly boost the "implementation experience" by addressing the pain points either for customers either from consultancy teams (RQ.2).

Following a DSR methodology, the method represents the generated artifact, meant to help solve an organizational challenge. By applying it in empirical cases, the method itself enables the generation of artifacts that support organizational processes, improving them.

The exploratory study conducted within the company to support the design process and development of our method brought a deeper understanding of the implementation process, success factors, and pain points. The treatment and analysis of this research data provided meaningful data for identifying improvement opportunities as map the customer journey and to confirm that, yes, it is possible to use agile approaches in developing countries, and there are clear benefits in using it (RQ.1).

The new method was built upon the SAP Activate methodology, resulting from the qualitative study, and by embedding a Service Design approach either by re-defining the new method principles and by introducing the best-fit Service Design tools on the several stages of the Activate methodology, ending up with the Activate Plus. The new method runs on five stages with elements of Service Design an SAP Activate.

Due to the timeline of the master thesis and current business context with the Covid-19 pandemic crises, it was impossible to implement the artifact and evaluate the results. The demonstration of the artifact was done through the mapping and explanation of all phases into this document. As implementation, the study includes service blueprints to demonstrate how the activities and stages would be developed during an implementation project. Also part of DSR methodology was the evolution of the artifact, on this research was made through two feedback interviews, here the feedback was positive, and the artifact was perceived as valuable and relevant for the business context.

It is necessary to implement the method to an implementation project and measure if the objectives are fulfilled as a future approach. To conclude, it is clear that an Agile approach can improve the way how an ERP is delivered, either on the operational side either on the mindset, the service design can work on the "pain points" side by enabling a better "Implementation Experience."

Refrences

Abrahamsson, P., Oza, N., & Siponen, M. T. (2010). Agile software development methods: a comparative review. In Agile software development (pp. 31-59). Springer, Berlin, Heidelberg.

Almeida, F. (2017). Challenges in migration from waterfall to agile environments. World Journal of Computer Application and Technology, 5(3), 39-49.

Asendorf, S., Gronau, N., Kurbel, K., Loos, P., & Schubert, P. (2010). Enterprise Resource Planning und Transformation 2010, 299.

Beck, K. (1999). Embracing change with extreme programming. Computer, 32(10), 70-77.

Beck, K. (2003). Test-driven development: by example. Addison-Wesley Professional.

Beznosov, K., & Kruchten, P. (2004, September). Towards agile security assurance. In Proceedings of the 2004 workshop on New security paradigms (pp. 47-54).

Bitner, M. J., Ostrom, A. L., & Morgan, F. N. (2008). Service blueprinting: a practical technique for service innovation. California management review, 50(3), 66-94.

Charmaz, K. (2006). "Constructing grounded theory: A practical guide through qualitative analysis", Sage.

Cockburn, A. (2006). Agile software development: the cooperative game. Pearson Education.

Cohn, M. (2004). User stories applied: For agile software development. Addison-Wesley Professional.

Corbin, J. and A. Strauss (2008). "Strategies for qualitative data analysis." Basics of Qualitative Research. Techniques and procedures for developing grounded theory 3.

Corbin, J. M. and A. Strauss (1990). "Grounded theory research: Procedures, canons, and evaluative criteria." Qualitative sociology 13(1): 3-21.

Cusick, J. J. (2013). Durable ideas in software engineering: concepts, methods and approaches from my virtual toolbox. Bentham Science Publishers.

Deloitte CFO Survey Angola Retrived from: https://www2.deloitte.com/ao/pt/pages/finance/articles/cfo-survey-2018---deloitte-angola---finance---article.html

Denecken, S., Musil, J., Santhanam, S., (2020). SAP Activate Project Management for SAP S/4HANA. Rheinwerk Publishing, Bonn, Boston.

Dima, A. M., & Maassen, M. A. (2018). From Waterfall to Agile software: Development models in the IT sector, 2006 to 2018. Impacts on company management. Journal of International Studies, 11(2), 315-326.

Ehie, I. C., & Madsen, M. (2005). Identifying critical issues in enterprise resource planning (ERP) implementation. Computers in industry, 56(6), 545-557.

Etikan, I., Musa, S. A., & Alkassim, R. S. (2016). Comparison of convenience sampling and purposive sampling. American journal of theoretical and applied statistics, 5(1), 1-4.

everything in between. *MIS Quarterly Executive*, *4*(3), 361–372. Retrieved from http://www2.commerce.virginia.edu/cmit/Research/MISQE%209-05.pdf

Francalanci, C. (2001). Predicting the implementation effort of ERP projects: empirical evidence on SAP/R3. Journal of information technology, 16(1), 33-48.

Fraser, S., Beck, K., Caputo, B., Mackinnon, T., Newkirk, J., & Poole, C. (2003, May). Test driven development (TDD). In International Conference on Extreme Programming and Agile Processes in Software Engineering (pp. 459-462). Springer, Berlin, Heidelberg.

Grech, T. (2015). The intersection of agile and waterfall. Industrial Engineer: IE, 47(8), 47-49.

Grenha Teixeira, J., et al. (2017). "The MINDS method: integrating management and interaction design perspectives for service design." Journal of Service Research 20(3): 240-258

Hevner, A., March, S. T., Park, J., & Ram, S. (2004). Design science research in information systems. MIS quarterly, 28(1), 75-105.

Holmlid, S., & Evenson, S. (2008). Bringing service design to service sciences, management and engineering. In Service science, management and engineering education for the 21st century (pp. 341-345). Springer, Boston, MA.

Joly, M. P., Teixeira, J. G., Patrício, L., & Sangiorgi, D. (2019). Leveraging service design as a multidisciplinary approach to service innovation. Journal of Service Management.

Júnior, M. O. (2013). As políticas das tecnologias de informação e comunicação em Angola e a sua implementação para o desenvolvimento socioeconómico (Doctoral dissertation).

Kothari, C. R. (2004). Research methodology: Methods and techniques. New Age International.

Kraljic, A., Kraljic, T., Poels, G., & Devos, J. (2014). ERP implementation methodologies and frameworks: a literature review. In 8th European Conference on IS Management and Evaluation (ECIME) (pp. 309-316). Academic Conferences and Publishing International Limited.

Lapunka, I., Marek-Kolodziej, K., & Jagoda-Sobalak, D. (2017). Agile-waterfall hybrid approach in innovation project management. in 4th international multidisciplinary scientific conference on social sciences and arts sgem 2017 (pp. 41-48).

Lemon, K. N., & Verhoef, P. C. (2016). Understanding customer experience throughout the customer journey. Journal of marketing, 80(6), 69-96.

Lindstrom, L., & Jeffries, R. (2004). Extreme programming and agile software development methodologies. Information systems management, 21(3), 41-52. Retrieved from https://www.tandfonline.com/doi/pdf/10.1201/1078/44432.21.3.20040601/82476.7

Long, F. (2009, May). Real or imaginary: The effectiveness of using personas in product design. In Proceedings of the Irish Ergonomics Society annual conference (Vol. 14, pp. 1-10).

Mahadevan, L., Kettinger, W. J., & Meservy, T. O. (2015). Running on hybrid: Control changes when introducing an agile methodology in a traditional “waterfall” system development environment. Communications of the Association for Information Systems, 36(1), 5.

Marchesi, M., & Succi, G. (2003). Extreme programming and agile processes in software engineering. Proceedings of XP.

Maximilien, E. M., & Williams, L. (2003, May). Assessing test-driven development at IBM. In 25th International Conference on Software Engineering, 2003. Proceedings. (pp. 564-569). IEEE.

Meroni, A., & Sangiorgi, D. (2011). Design for services. Gower Publishing, Ltd.

Miller, G. G. (2001, July). The characteristics of agile software processes. In Technology of Object-Oriented Languages, International Conference on (pp. 0385-0385).

Morgan, J. (2018). "Applying 1970 Waterfall Lessons Learned Within Today's Agile Development Process" PM World Journal Vol. VII, Issue VII – July 2018

Nancy R. Tague. (2005). The quality toolbox. Asq Press.

Nelson, R. (2005). Project retrospectives: Evaluating project success, failure, and

Ngai, E. W., Law, C. C., & Wat, F. K. (2008). Examining the critical success factors in the adoption of enterprise resource planning. Computers in industry, 59(6), 548-564.

Normann, R., & Ramirez, R. (1993). From value chain to value constellation: Designing interactive strategy. Harvard business review, 71(4), 65.

Ostrom, A. L., Parasuraman, A., Bowen, D. E., Patrício, L., & Voss, C. A. (2015). Service research priorities in a rapidly changing context. Journal of Service Research, 18(2), 127-159.

Patrício, L. and R. P. Fisk (2013). "Creating new services." Serving customers globally: 185-207.

Patrício, L., de Pinho, N. F., Teixeira, J. G., & Fisk, R. P. (2018). Service design for value networks: enabling value cocreation interactions in healthcare. Service Science, 10(1), 76-97.

Patrício, L., et al. (2011). "Multilevel Service Design: From Customer Value Constellation to Service Experience Blueprinting." Journal of Service Research 14(2): 180-200.

Patrício, L., et al. (2018). "Service Design for Value Networks: Enabling Value Cocreation Interactions in Healthcare." Service Science 10(1): 76-97.

Patrício, L., et al. (2018). "Upframing service design and innovation for research impact", Sage Publications Sage CA: Los Angeles, CA.

Pedersen, M. (2013). A quantitative examination of critical success factors comparing agile and waterfall project management methodologies (Doctoral dissertation, Capella University).

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2007). A design science research methodology for information systems research. Journal of management information systems, 24(3), 45-77.

Pells, D. (2017). Devil's Advocate: Agile from a Distance Vol. VI, Issue IV – April 2017 Welcome to this Edition

Rees, M. J. (2002, December). A feasible user story tool for agile software development?. In Ninth Asia-Pacific Software Engineering Conference, 2002. (pp. 22-30). IEEE.

Richardson, A. (2010). Using customer journey maps to improve customer experience. Harvard business review, 15(1), 2-5.

Royce, W. W. (1987, March). Managing the development of large software systems: concepts and techniques. In Proceedings of the 9th international conference on Software Engineering (pp. 328-338).

Saldaña, J. (2013) .The coding manual for qualitative researchers. London:Sage

Sale, J. E., Lohfeld, L. H., & Brazil, K. (2002). Revisiting the quantitative-qualitative debate: Implications for mixed-methods research. Quality and quantity, 36(1), 43-53.

SAP Corporate Fact Sheet 2020. Retrieved from https://www.sap.com/documents/2017/04/4666ecdd-b67c-0010-82c7-eda71af511fa.html

SAP SE. Retrived from: https://archive.sap.com/documents/docs/DOC-8032

Schwaber, K. (1997). Scrum development process. In Business object design and implementation (pp. 117-134). Springer, London.

Schwaber, K., & Beedle, M. (2002). Agile software development with Scrum (Vol. 1). Upper Saddle River: Prentice Hall.

Stickdorn, M., Hormess, M. E., Lawrence, A., & Schneider, J. (2018). This is service design doing: applying service design thinking in the real world. " O'Reilly Media, Inc.".

Takeuchi, H., & Nonaka, I. (1986). The new new product development game. Harvard business review, 64(1), 137-146.

Thomas, G., & Fernández, W. (2008). Success in IT projects: A matter of definition?. International journal of project management, 26(7), 733-742.

Westfall, L. (2010). The Certified Software Quality Engineer Handbook. Milwaukee, Quality Press.

Winter, R. J. (2014). Agile software development: Principles, patterns, and practices: Robert c. martin with contributions by james w. newkirk and robert s. koss. Performance Improvement, 53(4), 43-46.

APPENDIX A: Agile Manifesto

TABLE 1 Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others to do it. Through this work we have come to value:

Individuals and interactions over processes and tools
Working software over comprehensive documentation
Customer collaboration over contract negotiation
Responding to change over following a plan

That is, while there is value on the items on the right, we value the items on the left more.

Principles behind the Agile Manifesto

We follow these principles:

- Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- Welcoming changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to shorter time scales.
- Business people and developers must work together daily throughout the project.
- Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- The most ef cient and eff ective method of conveying information to and within a development team is face-to-face conversation.
- Working software is the primary measure of progress.
- Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace inde nitely .
- Continuous attention to technical excellence and good design enhances agility.
- Simplicity — the art of maximizing the amount of work not done — is essential.
- The best architectures, requirements, and designs emerge from self-organized teams.
- At regular intervals, the team re ects on ho w to become more effective, then tunes and adjusts its behavior accordingly.

Agile Methods

The number of methods that claim to align the Agile Manifesto will continue to grow with the popularity of the agile software methodologies. The early initial methodologies include:*

- Extreme Programming
- SCRUM
- Crystal
- Feature Driven Development
- Lean Development
- Adaptive Software Development
- DSDM

* Jim Highsmith, *Agile Software Development Ecosystems*, Addison-Wesley, 2002, provides a comparison of these methodologies.

APPENDIX B: Semi-structured Interview guidelines

1. Identificação

1.1 Função e tempo na função

1.2 Em quantas implementações completas já participou

1.3 Já teve alguma formação em metodologias de implementação SAP?

2. Projecto de Implementação

2.1 Baseado na sua experiência, consegue descrever um processo de implementação em que tenha participado?

Bem sucedido/ Mal sucedido (razões)

Faria alguma coisa diferente

2.2 Num processo de implementação, quais são as principais atividades que desempenha? Sozinho(a) /Equipa.

2 .3 Das atividades que referiu quais as mais simples e quais as mais complexas?

2.4 Durante o projeto, qual considera ser a fase mais critica

2.5 E em que fase costumam surgir mais issues / atrasos.

2.6 Durante uma implementação, qual a percentagem de trabalho que executa localmente e qual a percentagem de trabalho que faz remotamente.

2.7 Que fatores na sua opinião levam ao sucesso de um projecto, e que fatores podem comprometer o projecto.

3. Ferramentas

3.1 Durante uma implementação que ferramentas utiliza.

3.2 Das ferramentas que referiu, considera serem as melhores para o trabalho?

4. Metodologias

4.1 Qual na sua opinião sobre a metodologia ACTIVATE

4.2 Vê algum benefício na adoção da mesma?

4.3 Existe alguma das fases que acha especialmente complexa ou difícil de executar

4.4 Vê algum risco na adoção da metodologia ACTIVATE

4.5 Faria alguma melhoria no modelo?

4.6 Dado o contexto onde opera a empresa, qual considera a metodologia mais indicada para elaborar um projecto.

APPENDIX C: Informed consent

CONSENTIMENTO INFORMADO

Estamos a solicitar a sua participação para um estudo no âmbito da dissertação "Developing Countries and ERP Implementations - Going Agile with Service Design", realizada na Faculdade de Engenharia da Universidade do Porto e na ITGest.

Este estudo tem como objetivo a construção de um modelo de implementação de ERP ajustado a mercados emergentes. A entrevista pretende, recolher informação, avaliar necessidades, dificuldades e melhores abordagens à utilização do modelo SAP ACTIVATE na ITGest. Para tal, consideramos os vários atores envolvidos no processo de implementação, gestores de projecto, competence leaders e consultores funcionais/técnicos. A sua participação fornecerá informações importantes para o projeto.

A entrevista é gravada para possibilitar a transcrição e análise. A gravação é apenas iniciada após o consentimento do entrevistado, expresso através da assinatura deste consentimento informado.

A informação recolhida é estritamente confidencial e será apenas utilizada no âmbito deste estudo. Os resultados serão reportados de forma agregada, sem identificar individualmente os entrevistados. A informação poderá ser usada para relatórios, apresentações ou artigos científicos, mas o seu nome não será identificado.

A sua participação neste estudo é voluntária, pelo que a poderá interromper a qualquer momento. Nesse caso toda a informação recolhida até ao momento será inutilizada.

Para qualquer esclarecimento adicional, poderá contactar o estudante Miguel Mendes através de email (miguel.mendes@ideiasdinamicas.com) ou de telemóvel (917735221).

O estudante:

Nome: Miguel Marinho Mendes

Assinatura: _____ Data ___ / ___ / _____

O participante:

Declaro ter lido e compreendido este documento, bem como as informações verbais fornecidas e aceito participar nesta investigação. Permito a utilização dos dados que forneço de forma voluntária, confiando em que apenas serão utilizados para investigação e com as garantias de confidencialidade e anonimato que me são dadas pelo investigador.

Nome: _____

Assinatura: _____ Data ___ / ___ / _____

ESTE DOCUMENTO É FEITO EM DUPLICADO: UM PARA O PARTICIPANTE E OUTRO PARA O INVESTIGADOR.