INSTITUTO SUPERIOR DE ENGENHARIA DO PORTO

MESTRADO EM ENGENHARIA INFORMÁTICA

Desenvolvimento de uma Infraestrutura baseada em HL7® FHIR® para Interoperabilidade Clínica

TIAGO TAVARES GONÇALVES

Julho de 2018

POLITÉCNICO
DO PORTO

Infrastructure Development based on HL7® FHIR® for Clinical Interoperability

Tiago Tavares Gonçalves

Dissertation to obtain the Master Degree in Informatics Engineering, Area of Expertise in Software Engineering

Advisor: Paulo Alexandre Fangueiro Oliveira Maio
Supervisor: André Filipe Roque Silva

Porto, July 2018

Confidentiality Statement

Abstract

Throughout the years, the healthcare business knowledge, requirements, and the number of patients seeking medical attention has grown tremendously to a point where sensitive cases needed the input from multiple healthcare institutions in order to track the patient's medical history and make the most adequate decisions for each situation. Technology and digital information fulfils a great role in addressing these problems and improving healthcare provision. However, due to the immense number of organizations and systems in this business, sharing a patient's clinical information can be a major problem if the systems are not capable of understanding the data sent to each other. Ensuring interoperability between systems is crucial to guarantee the continuous flow of a patient's clinical history transmission and to improve the health professionals' work.

As a company working in the field of healthcare, ALERT's main goal is to help organizations improve in their health business and to help prolong life, by providing the necessary technology that is capable of benefiting the health professional's work management and sharing the necessary information with other organizations. Thus, the company seeks to constantly improve its product suite, ALERT®, by meeting the worldwide organizations requirements and assuring interoperability based on the existing health standards in the market.

This way, the company wants to add in the ALERT suite the latest standard, Fast Healthcare Interoperability Resources (FHIR ®), which brings great technological innovations for interoperability's improvement, provided by the standards developing organization, Health Level Seven International (HL7), being also considered to be a suitable standard for mobile applications thanks to its capabilities and ease of implementation.

Herewith, this thesis presents a development and architectural approach to apply FHIR features in the product suite, along with the problem and solution analysis, including the evaluation of suitable frameworks for the implementation phase. Considering the experiments' results, the implemented FHIR services actually improved the product's performance, and thanks to the standard's specification, the implementation of its core features proved to be simple and straightforward while respecting the key criteria for some of the developed services.

Keywords: HL7® FHIR®, Interoperability, Healthcare, Java, REST

Resumo

Ao longo dos anos, o conhecimento, as exigências, e o número de pacientes à procura de cuidados médicos na área de negócio de cuidados de saúde, tem vindo a aumentar drasticamente ao ponto de ser necessária a opinião de outras instituições para casos de maior sensibilidade, de modo a que o historial médico do paciente fosse acompanhado e que servisse para tomar as decisões mais adequadas para o problema em questão. A tecnologia e a informação digital representam um grande papel na resolução de problemas e promoção de entrega de cuidados de saúde. No entanto, devido à imensa quantidade de organizações e sistemas nesta área de negócio, a partilha de informação clínica relativa a um paciente pode vir a ser um grave problema caso os sistemas não sejam capazes de compreender os dados que estão a ser transmitidos entre eles. Deste modo, assegurar interoperabilidade entre sistemas é crucial para garantir um fluxo contínuo de transmissão de informação relativa ao historial clínico de um paciente, e para melhorar o trabalho dos profissionais de saúde.

Sendo uma empresa que trabalha na área de cuidados de saúde, a ALERT tem como principal objetivo ajudar as organizações a melhorar o seu negócio de saúde e ajudar a prolongar a vida, fornecendo a tecnologia necessária que beneficie a gestão de trabalho dos profissionais de saúde e que partilhe informação com outras organizações. Portanto, a empresa procura constantemente melhorar o seu produto ALERT®, procurando cumprir com os requisitos de organizações globais e garantindo interoperabilidade baseada nos standards de saúde existentes no mercado.

Assim, a empresa pretende adotar o último standard lançado, Fast Healthcare Interoperability Resources (FHIR®), que traz grandes inovações tecnológicas para o aperfeiçoamento da interoperabilidade, fornecida pela organização de desenvolvimento de standards, Health Level Seven International (HL7), sendo também considerado um standard adequado para aplicações móveis graças às suas capacidades e facilidade de implementação.

Com isto, esta tese apresenta uma abordagem arquitetural e de desenvolvimento para a aplicação de funcionalidades FHIR no produto, juntamente com a análise do problema e da solução, incluindo a avaliação de ferramentas adequadas para a fase de implementação. Os resultados de teste obtidos para os serviços FHIR implementados, demonstraram uma melhoria na performance do produto, e graças à especificação do standard, a implementação das principais funcionalidades provou ser simples e direta, respeitando os principais critérios para os serviços desenvolvidos.

Palavras-Chave: HL7® FHIR®, Interoperabilidade, Cuidados de Saúde, Java, REST

Acknowledgements

I want to start by thanking ALERT for the amazing support and help provided throughout this thesis, and for the opportunity to participate in this project along with an amazing team. A special thanks to my project supervisor, Engineer André Silva, for all the help provided during every stage of this project, and to all the other team members who also helped me with constructive feedback and support.

Next, I want to thank ISEP and all the teachers that contributed for my degree and for all the knowledge shared that helped me reach this stage of my life. A special thanks to Engineer Paulo Maio, who helped me tremendously during this project, for all the suggestions, analysis and discussions which continuously led me to improve and reach my objectives.

Last, but not least, I want to thank all the people that are dearest to me, my family and friends. Thanks to all of you, I was able to stay motivated and continue my studies, even with all the homesickness. I can't show enough gratitude for all your presence, conversations, laughter and support throughout these years.

To all, my sincere thanks.

Index

Index of Figures

Index of Tables

Acronyms and Glossary

Next, acronyms mentioned throughout the document are described and sorted alphabetically.

Acronym	Meaning
ARR	Audit Record Repository
ATNA	Audit Trail and Node Authentication
CDA	Clinical Document Architecture
CDR	Clinical Data Repository
DICOM	Digital Imaging and Communications in Medicine
DRG	Diagnosis-Related Group
DSTU	Draft Standard for Trial Use
EHR	Electronic Health Record
EIP	Enterprise Integration Patterns
FFE	Fuzzy Front End
FHIR	Fast Health Interoperability Resources
FMM	FHIR Maturity Model
HIE	Health Information Exchange
HIMSS	Healthcare Information and Management Systems Society
HL7	Health Level Seven
HTTP	Hypertext Transfer Protocol
ICD	International Classification of Diseases
IEEE	Institute of Electrical and Electronic Engineers
IHE	Integrating the Healthcare Enterprise
IHTSDO	International Health Terminology Standards Development Organization
ISO	International Organization for Standardization
IT	Information Technology
ITS	Implementable Technology Specification
IUA	Internet User Authorization
JSON	JavaScript Object Notation
JWT	JSON Web Token
LOINC	Logical Observation Identifiers Names and Codes

MHD	Mobile Access to Health Documents
mXDE	Mobile Cross-Enterprise Document Data Element Extraction
NCD	New Concept Development
NOC	Network Operations Center
OASIS	Organization for the Advancement of Structured Information Standards
OCP	Open Closed Principle
ONC	The Office of the National Coordinator for Health Information Technology
PCHA	Personal Connected Health Alliance
PDQ	Patient Demographics Query
PDQm	Patient Demographics Query for Mobile
PDQv3	Patient Demographics Query HL7 v3
PFH	Paper Free Hospital
PHD	Personal Health Devices
PHR	Personal Health Record
PIX	Patient Identifier Cross-Referencing
PIXm	Patient Identifier Cross-Referencing for Mobile
PIXv3	Patient Identifier Cross-Referencing HL7 v3
QEDm	Query for Existing Data for Mobile
QFD	Quality Function Deployment
RDF	Resource Description Framework
REST	Representational State Transfer
SAML	Security Assertion Markup Language
SNOMED CT	Systematized Nomenclature for Medicine Clinical Terms
SQL	Structured Query Language
SRP	Single Responsibility Principle
STU	Standard for Trial Use
VA	Value Analysis
WHO	World Health Organization
XCA	Cross-Community Access
XCPD	Cross-Community Patient Discovery
XDS	Cross-Enterprise Document Sharing

XML	Extensible Markup Language
XUA	Cross-Enterprise User Assertion

1 Introduction

This chapter labels the project, briefly describing its context, the problem related to it, the objectives to resolve the problem, the personal motivation to accept the project, the procedures adopted to concretize the goals, the project's value and contributions for future business goals, the results obtained, and finally, the document's structure.

1.1 Context

This work was developed in the context of Tese/Dissertação/Estágio (TMDEI), a course unit integrated in the Master in Informatics Engineering Degree (MEI) from Instituto Superior de Engenharia do Porto (ISEP). The main goal of TMDEI is challenging students to provide and develop a solution for a complex problem, adopting good engineering practices and tools.

The project "Infrastructure Development based on HL7® FHIR® for Clinical Interoperability", proposed by ALERT, focuses on the interoperability topic which already has a great impact in health-related business. Interoperability is a concept that, throughout time, started to be adopted by most of healthcare organizations, since it brought major advantages to the health business area. It promotes the business quality by ultimately offering a better service to their patients using standards that provide a common set of rules for the healthcare community, to share coherent information. Multiple organizations develop their own standards and improved them progressively, to cover several clinical aspects, such as nomenclature, clinical terminology and data structure, while others combined these standards along with complex integration approaches and clinical system implementations, to define worldwide approved clinical business processes, common to the numerous healthcare institutions.

To provide the means for the healthcare organizations to share their patients' information between each other, or between their inner departments, ALERT adopted a set of standards developed by the Health Level Seven International (HL7), which guarantees the exchange of information between the ALERT® software suite and the other existing healthcare information technology (IT) solutions in the market.

1.2 Problem

Technology is constantly improving, and the same can be said for healthcare standards which take into consideration these technological innovations to specify new approaches, improvements, or even brand-new standards that can benefit the healthcare business area. The healthcare IT organizations also keep track of these enhancements to deliver better quality products for the existing healthcare institutions, and the patients that seek medical attention. For this reason, and with the intention to simplify and ease the integration with mobile devices, HL7 developed a new standard named Fast Health Interoperability Resources (FHIR®).

FHIR ® is the latest standard in the market that has the potential to revolutionize interoperability in the healthcare business, by offering a set of standardized resources and services, based on the Representational State Transfer (REST) architecture defined by Fielding (2000), with easier methods to deliver clinical information with fine granularity, including a lightweight integration suitable for mobile devices.

With this, a new problem for ALERT's product emerges, related with the lack of support for FHIR services. This reduces ALERT® interoperability and ability to exchange information with other existing solutions in the market that adopt HL7's latest standard, which will also affect their current clients and capability to take advantage of new market opportunities related to FHIR®.

1.3 Objectives

In order for the ALERT software suite to fully support FHIR®, a set of objectives were established in accordance to the organizations' requirements:

- **Analyze the FHIR specification:** FHIR features and technical specification needs to be studied and analyzed, not only to follow correctly the standard's restrictions, but also to assist in future design and development decisions;

- **Development of FHIR web services:** depending on the required features, a set of FHIR web services should be developed to support existing and new functionalities with the standard's specification;

- **Development of security features:** the FHIR services to be developed should be enhanced with security features to guarantee secure communications and exchange of information;

- **Update or add product components:** to integrate the solution in the product, the existing architecture should be studied to add new behavior for existing components, or to create new ones depending on the architectural approach.

Herewith, the previous defined objectives should be accomplished for ALERT's current interoperability solution to be capable of performing FHIR services with new and efficient methods that can improve information sharing through ALERT® software. Overall, it is expected that the solution maintains the existing functionalities related to patient and document information sharing, follows FHIR® specifications, and that both services and architecture are documented to explain the development approach and the most relevant decisions made.

1.4 Personal Motivation

Regarding personal motivation, the major reasons that raise interest in this project, are the interoperability and integration aspects involving the whole problem. The healthcare business area is fairly complex, demanding a special care and attention regarding the needed IT system features and the data exchange operations, due to their diverse specifications. The fact that this project also involves the latest healthcare data standard in the market, with improved technology and features for information exchange between heterogeneous systems, is also a motivation factor and an interesting opportunity to obtain some knowledge and experience in the area.

1.5 Procedures

To promote the project requirements analysis and features development, a work methodology was established, along with the major tasks for the two project milestones, regarding its submission for the first and second stage of evaluation.

1.5.1 Work Methodology

The project development followed a work methodology similar to the Rational Unified Process (RUP) methodology defined by International Business Machines (International Business Machines [IBM], 2001), with the adoption of an iterative development throughout the project's phases, which involved the scope and major business concepts definition, the problem, requirements and business value analysis, the design of architectural artifacts, and the development.

1.5.2 Planning

Since the project's first delivery required a set of initial outcomes, to organize the work for both stages of evaluation involved, the most relevant tasks to be performed before each delivery milestone were planned and defined. For the first stage, the following tasks were established:

- **Understand the concept of interoperability:** one of the first tasks should be to understand what interoperability is and how it impacts the healthcare business area;

- **Analyze the major healthcare standards:** it is important to study a bit of other existing standards, to understand why they are necessary and what importance do they bring for worldwide healthcare organizations;

- **Analyze HL7 FHIR standard features:** an initial analysis of what FHIR® as to offer is crucial to plan and think about the solution's design and development;

- **Analyze clinical profiles for use cases:** along with FHIR®, integration profiles shall also be studied for a better understanding of the required business workflow and criteria for each desired use case;

- **Analyze multiple alternatives for the development phase:** the analysis phase shall cover some existing frameworks that can be used during the solution's implementation, to choose the most suitable one and maximize the development phase efficiency;

- **Analyze the problem:** to correctly approach a solution, the problem shall also be analyzed with detail to cover all the important factors that might influence the following stages for design and development;

- **Analyze the value for the customer regarding the standard's features:** the customer's point of view is also a relevant factor to take into consideration before deciding to prioritize some features over others, which can influence the development phase;

- **Define the project requirements:** the functional and nonfunctional requirements shall be defined to clearly establish the required features for development;

- **Analyze the product's software architecture:** the integration of these services in the product require an additional study of the product's current architecture, to plan a proper approach to extend the product with new features and components that shall apply the standard specification.

Regarding the second stage, the following tasks were defined:

- **Elaborate a detailed design:** the design for each required component, service, and business workflow regarding the product's new features shall be documented and in accordance with the development phase;

- **Develop the required features:** the required features and use cases shall be implemented, respecting the previous analysis and design decisions to correctly approach the problem and provide a solution;

- **Develop a set of test cases to evaluate the features:** the developed services shall be verified with suitable test cases depending on the defined hypothesis and on external factors regarding the standard and the integration profiles' criteria;

- **Analyze the test results:** at last the results shall be analyzed to establish some final conclusions regarding the implemented solution in comparison to previous existing services.

1.6 Contributions and Value

This project involves the development of specific standard features that will make part of one of ALERT's products. The use of FHIR® brings great innovations and promotes the product's value for worldwide healthcare organizations, by providing the means for healthcare professionals to communicate and share clinical information with improved system performance. It also extends the product's capability to answer incoming requests related to the exchange of information based on FHIR®, and its unique features which are only possible to achieve thanks to the standard's specification.

This project mainly contributes with the following aspects:

- **Product's support for FHIR®:** with the development of new services based on FHIR®, the product is capable of answering to incoming FHIR requests for features that were only available through previous HL7 standard versions;

- **Improvement for the product's services performance:** FHIR ® brings along the adoption of lightweight technologies which enhance the product's processing speed, promoting the overall performance;

- **Arrival of new market opportunities:** the standard's adoption might open new business opportunities for the organization, which were not possible to be achieved without the integration of FHIR services in the product;

- **Product's support for new clinical integration features:** the appearance of FHIR® brings along new clinical integration features, which will also be supported in the product, since most of the required use cases involve these profiles;

- **Enhancement of the overall service quality for the healthcare community:** besides the standard's lightweight nature, its ease of implementation and architectural approach promotes new innovative solutions and is suitable for platforms that are limited in their processing capabilities, such as mobile solutions.

Due to FHIR's fast capabilities of exchange, and considering the expected system's performance improvement, not only does it offer an enhanced system quality to assist the healthcare professionals' daily job, faster response and decision-making, but it also raises the patients' overall satisfaction and well-being, which are the primary individuals that will truly benefit from these innovations.

1.7 Document Structure

Excluding the current chapter, the document consists of six more chapters (plus the references and the annexes), namely:

- **State of the Art:** this chapter briefly describes the evolution of interoperability in healthcare, along with the clinical standards that caused a greater impact in this business area, highlighting the document's standard of focus. Additionally, relevant use case profiles, existing ALERT solutions involved with ALERT's interoperability product (which adopts some of the referred standards and profiles), and existing frameworks suitable for the development phase are also mentioned;

- **Project Context:** this chapter describes in detail the problem, involved stakeholders, requirements, and the value analysis. The problem addresses multiple aspects related to the market, existing engineering information, technologies, and major implications. The requirements describe the functional and nonfunctional aspects to be considered for implementation, considering the standard's specifications. The value analysis specifies the correlation between what brings the most value for the customer and the major features the product can offer, including the evaluation of the frameworks' overall capabilities, adopting a set of techniques to support the conclusions;

- **Analysis and Design:** this chapter includes a detailed analysis regarding the IHE profiles, which need to be considered for features' development, a domain model to agglomerate the major entities and their relationships, including other design artifacts that describe the solution's architecture and major use cases;

- **Development:** this chapter describes the implementation of the desired features considering the analysis and design previously depicted, providing a detailed explanation and evidences of implementation decisions regarding the clinical profiles, the product components and the standard's specifications;

- **Experiments:** this chapter describes a set of applied methodologies to test, experiment and evaluate the developed features, and, therefore, the raised hypothesis, by analyzing the results and elaborating some conclusions;

- **Conclusion:** this chapter describes the requirements' fulfillment, along with the future work and some final thoughts regarding the whole project analysis, design, development, and experiments.

2 State of the Art

In this chapter, the evolution of interoperability in healthcare is described, referring some of the most widely used standards in the market nowadays, with special attention to FHIR®, which is the topic of interest for this project. Additionally, it refers the most relevant use case profiles for use in this project, the ALERT solutions, and existing frameworks suitable for development.

2.1 Interoperability's Evolution in Healthcare

The project described throughout this document fits in the health business area, more specifically with health records' interoperability, which is a topic of interest discussed by Benson (2012) when addressing the impact of health digital information. In 2005, the Healthcare Information and Management Systems Society defined interoperability as "the ability of health information systems to work together within and across organizational boundaries in order to advance the effective delivery of healthcare for individuals and communities" (Healthcare Information and Management Systems Society [HIMSS], 2005), and according to Benson (2012, p. 3), "improved interoperability can help transform the efficiency and effectiveness of health services, to provide information when and where required, facilitate quicker and more soundly based decision-making, reduce waste by cutting out repeated work, and improve safety due to fewer errors". However, the adoption of digital means to share valuable documents and promote clinical interoperability was slow and protracted. The Electronic Health Record (EHR) system, which contains a collection of "information pertaining to the health of an individual or a health care provider to an individual" (Carter, 2008, p. 6), represents the core of digital information in healthcare. Nonetheless, due to the wide range of specialties for doctors, nurses and other professionals, the comprehension of the specific information from these branches compromised the success of the EHR usage in the healthcare organizations. Great examples are the hospitals that significantly suffered from this problem, since multiple systems with their own specialty needed to work together in order to perform some business tasks (Benson, 2012, p. 12).

Moreover, with the continuous growth of healthcare needs throughout the years, worldwide health facilities and organizations started to feel the necessity of obtaining information in a fast and efficient way, which led the exchange of data between different systems to become mandatory, bringing, in turn, various challenges for interoperability. Health institutions started to consider interoperability a goal concept, to embrace the exchange of well-structured data without the need to interpret information, by using EHR systems to promote the healthcare quality and semantic interoperability (HIMSS, 2013). Thanks to the improvement of interoperable systems capabilities and with the support of known standards, information started to be exchanged in an understandable way, regardless of the user or system who received it, stimulating an increase of Health Information Exchange (HIE), "the reliable and interoperable electronic sharing of clinical information" (Hersh et al., 2015, p. 1), which consequently helped professionals to retrieve more information about their patients, facilitating their decision-making and improving the management of their patients' condition (HIMSS, 2017). According to The Office of the National Coordinator for Health Information Technology (The Office of the National Coordinator for Health Information Technology [ONC], 2016a), and depicted in Figure 1, 41 percent of the hospitals in the United States exchanged electronic health information with external entities in 2008, which increased throughout the years reaching an 82 percent rate of exchange in 2015 (ONC, 2016b, p. 8).

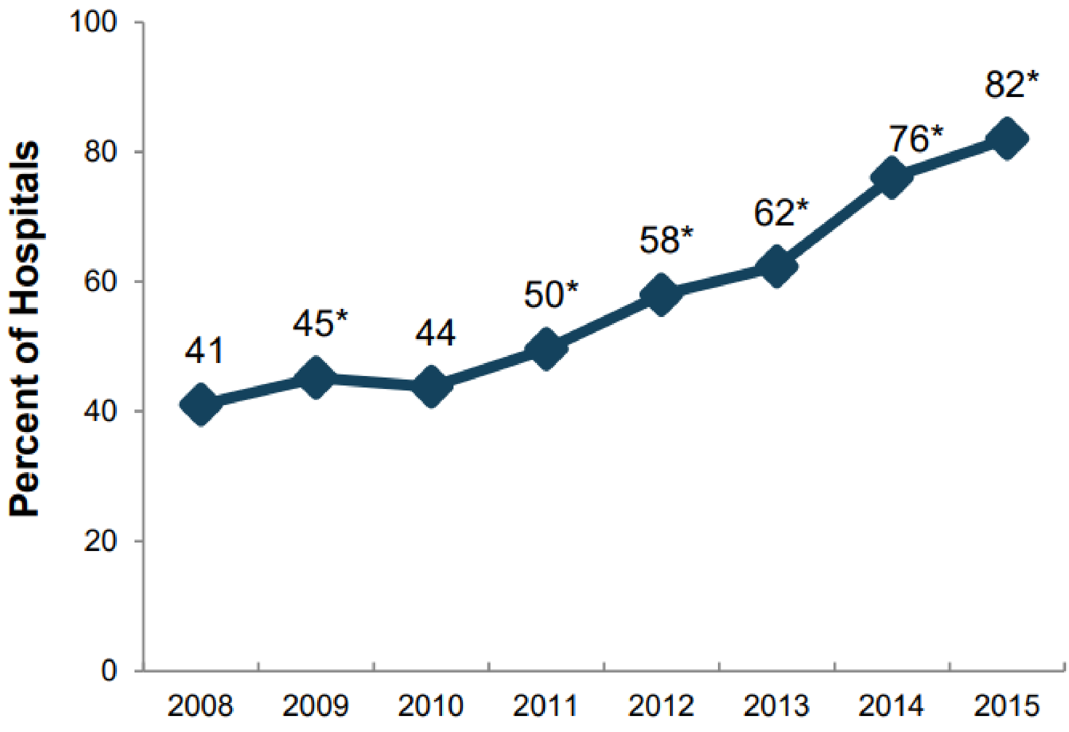

Figure 1 - Percent of HIE in the United States (ONC, 2016b, p. 8)

This increase with HIE occurred thanks to the standards provided by organizations such as HL7 and SNOMED International (SNOMED International, 2017a). Figure 2 shows an example of the advantages that standard adoption can bring for organizations, where the right side of the figure represents the linkage of heterogeneous systems through a single standard, whereas the left side represents the interfaces required to connect these systems without a standard (Benson, 2012, p. 22).

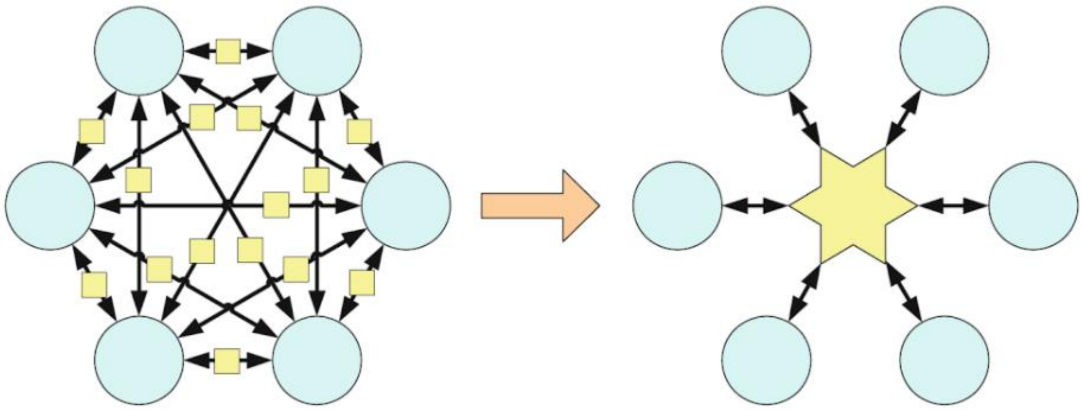

Figure 2 - Single standard application results (Benson, 2012, p. 22)

However, the wide range of available standards in the market led different organizations to follow some standards over others, which resulted in different implementations that affected interoperability. This way, organizations such as Integrating the Healthcare Enterprise (Integrating the Healthcare Enterprise [IHE], 2016a) and Personal Connected Health Alliance (Personal Connected Health Alliance [PCHA], 2016) produced a set of technical frameworks and guidelines to fight this problem, by selecting existing and worldwide-known standards, and specifying how these standards must be used to correctly address the clinical needs (Groupe Speciale Mobile Association [GSMA], 2016).

Nowadays, IHE frameworks are greatly used around the world thanks to their integration profiles that define common clinical use cases adopted by various institutions. Furthermore, with the healthcare mobile applications' emergence, IHE is creating new profiles suitable for mobile services that typically consume REST services. The major standard that contributed to this matter was FHIR®, thanks to its resource supportability for REST services and ease of implementation for mobile platforms (Parisot, 2016).

2.2 Healthcare Data Standards

Healthcare data standards can be distributed among three main categories: (i) data interchange, (ii) terminologies and (iii) knowledge representation (Committee on Data Standards for Patient Safety, 2004, p. 12).

For the first category, the standards are mainly adopted for (i) message formats, which improve interoperability thanks to the defined specifications, structure and relationship definitions between data elements, (ii) document architecture, to offer a standardized format for electronic clinical documents and assist on the consultation of valuable information, (iii) clinical templates, to constrain specific clinical data that needs to be included to promote standardization and precision of information, (iv) user interface design, to help define a set of rules, specifications, or architectural patterns to improve user-friendliness and safety, independently of the technologies involved, (v) and patient data linkage, which represents the core of interoperability and capability of transmitting healthcare data in an understandable way between organizations (Committee on Data Standards for Patient Safety, 2004, pp. 132–142).

Regarding the second category, the standardization of clinical terminology brings along great importance for health, although, compared to other areas such as chemistry and biology, the medical area is the one that lacks the most on formal terminology, carrying major risks for patient safety (Benson, 2012, p. 201). The wide range of clinical areas and specifications hinders the use of a single terminology that can be suitable for each one of them, which can be related to medication, laboratory, demographics, and other domains. To resolve this problem, core terminology groups were defined to join multiple terminologies together, in order to provide clinical granularity, decision support, interoperability and other relevant functionalities (Committee on Data Standards for Patient Safety, 2004, pp. 142–157).

The third category, knowledge representation, is related to any kind of information repository that holds medical research, such as disease registries, databases with medical knowledge for drug reactions, and so on. Data standards can be used to link these knowledge bases with decision support systems for clinicians and physicians, which enables them to access precise medical information to avoid undesired actions, for example, regarding drug contraindications (Committee on Data Standards for Patient Safety, 2004, pp. 158–162).

2.2.1 Organization for the Advancement of Structured Information Standards

Organization for the Advancement of Structured Information Standards (OASIS) is a nonprofit consortium composed by several companies' representatives that together define worldwide standards, which fit in the first category (data interchange), regarding design rules, messaging services, security, data exchange languages and format, protocol specifications, and other standards related to information technology areas (Organization for the Advancement of Structured Information Standards [OASIS], 2017). Although OASIS is not directly related to the healthcare area, their standards are usually used along with healthcare standards from other organizations such as HL7.

2.2.2 Logical Observation Identifiers Names and Codes

Logical Observation Identifiers Names and Codes (LOINC®) is a terminology standard (second category) usually adopted by other standards to provide universal names and identification codes for clinical information contained in electronic health records. This standard was created in 1994 at the Regenstrief Institute, holding nowadays more than 84.500 terms and more than 55.500 registered users (Regenstrief Institute, 2017).

2.2.3 Systemized Nomenclature for Medicine Clinical Terms

Systemized Nomenclature for Medicine Clinical Terms (SNOMED CT) is a terminology standard (second category) defined by International Health Terminology Standards Development Organization (IHTSDO), which nowadays is known by the trading name SNOMED International. It provides a comprehensive clinical vocabulary, with more than 320.000 concepts, which improves the quality of electronic health records and enables a unified global language between international health systems (SNOMED International, 2017b).

2.2.4 Diagnosis-Related Group

Diagnosis-Related Group (DRG) is a patient classification system developed by Professor Robert Fetter along with his colleagues that, based on the discharge data, assigns patients to economically homogeneous groups, combining them with overall similar features. The DRGs establish the required services for each group, in order to provide fair prices and decision-making assistance, with enhanced and efficient payment systems that may vary depending on the respective countries' payment policies (Busse et al., 2011, pp. 3–37).

2.2.5 International Classification of Diseases

International Classification of Diseases (ICD) is a terminology standard (second category) developed by the World Health Organization (WHO), which contains information regarding the global population's health situation. This standard is used for clinical and research purposes to serve as a diagnostic source of known diseases for improved analysis and accurate decision making (World Health Organization [WHO], 2017).

2.2.6 Digital Imaging and Communications in Medicine

Digital Imaging and Communications in Medicine (DICOM®) is a data interchange standard (first category) for medical images which offers services to promote interoperability between image devices related to radiology, cardiology and radiotherapy, and other systems capable of receiving data in DICOM format, being a widely adopted standard for equipment vendors and healthcare IT organizations (DICOM Standard, 2017).

2.2.7 Health Level Seven

Health Level Seven International is a global organization who created the HL7 standards (Health Level Seven International [HL7], 2017a), which fit in the first category regarding data interchange. The organization was founded in 1987, and throughout their existence, HL7 produced a set of standards that evolved over the years, namely, HL7 Version 2 (V2), HL7 Version 3 (V3) and HL7 Reference Information Model (RIM), HL7 Version 3 Clinical Document

Architecture (CDA®), and most recently, Fast Healthcare Interoperability Resources (FHIR®) (HL7, 2017a).

2.2.7.1 HL7 V2

HL7 V2 was marked by allowing the exchange of electronic clinical data between heterogeneous systems in the healthcare industry, using a messaging paradigm with ASCII characters (cf. Example 1).

Example 1 – HL7 V2 message (Ringholm bv, 2011)

```
PID|||555-44-4444||PATIENT^ONE^P^^^^L|JONES|19620320|F||||||||||AC555444444||
```

This standard is adopted by 95% of healthcare organizations in the United States, and implemented in a wide range of countries due to its supportability with the most used interfaces in the industry, being currently in Version 2.7 since 2011 (HL7, 2017b).

2.2.7.2 HL7 V3 RIM

HL7 V3 suffered major modifications in implementation compared to the previous one (HL7 V2). It allowed the production of electronic documents in XML syntax and the use of V3 messages (cf. Example 2), adopting a model driven methodology that depends on a set of normative specifications, regarding data types, the Implementable Technology Specification (ITS) for XML encoding rules, messages and transport protocols (HL7, 2017e).

Example 2 – HL7 V3 partial message (Ringholm bv, 2011)

```
<recordTarget>
    <patientClinical>
    <id root="2.16.840.1.113883.19.1122.5" extension="444-22-2222"
        assigningAuthorityName="GHH Lab Patient IDs"/>
    <statusCode code="active"/>
      <patientPerson>
        <name use="L">
           <given>One</given>
           <given>P</given>
           <family>Patient</family>
        </name>
        <asOtherIDs>
           <id extension="AC555444444" assigningAuthorityName="SSN"
               root="2.16.840.1.113883.4.1"/>
        </asOtherIDs>
      </patientPerson>
    </patientClinical>
 </recordTarget>
```

The HL7 V3 specifications also adopted RIM, which is an object model created to support all the concepts' representation in the HL7 domain, serving as a base reference for documents, data, and messages (HL7, 2017f).

2.2.7.3 HL7 CDA®

CDA® is a type of HL7 V3 specification which adopts a standard with a set of rules to be followed regarding clinical documents' structure based on XML syntax, and semantics, using existing code systems and terminologies to reference specific clinical content (cf. Example 3).

Example 3 – HL7 CDA document section

```
<section>
    <!-- Allergies and Intolerances Section (entries required) (V3) -->
    <templateId root="2.16.840.1.113883.10.20.22.2.6.1"/>
    ...
    <code code="48765-2"
          codeSystem="2.16.840.1.113883.6.1"
          codeSystemName="LOINC"/>
    <title>ALLERGIES AND ADVERSE REACTIONS</title>
    <text>No Known Allergies</text>
    <entry typeCode="DRIV">
        <!-- Allergy Concern Act -->
        <act classCode="ACT" moodCode="EVN">
            ...
            <entryRelationship typeCode="SUBJ">
                ...
                <observation classCode="OBS" moodCode="EVN" negationInd="true">
                    ...
                    <value xsi:type="CD" code="419199007"
                           displayName="Allergy to substance (disorder)"
                           codeSystem="2.16.840.1.113883.6.96"
                           codeSystemName="SNOMED CT"/>
                    <author>
                        ...
                    </author>
                    ...
                </observation>
            </entryRelationship>
        </act>
    </entry>
</section>
```

It is mostly used for HIE, since it promotes the interoperability and transmission of documents among facilities, organizations, and enterprises respecting a group of standards, improving the delivery and retrieval of information (HL7, 2017g).

2.3 FHIR®

FHIR® combines the major characteristics from HL7 V2, HL7 V3 and HL7 CDA®. Unlike the previous ones, this standard adopts a RESTful architecture using resources that cover multiple clinical, administrative and patient information, which improves ease of implementation regarding the exchange of data for messages, documents and services (HL7, 2017h).

2.3.1 Resources

FHIR® resources are identified with a URL and can be represented in three different formats, namely, XML, JSON, and RDF serialized through Turtle language. Although RDF is a possible format for data representation, it is still not recommended for implementation due to its current state of maturity in the current standard's specification version, which means it will suffer major updates in future releases. Each supported format needs to be declared in FHIR's Capability Statement resource, in order for a FHIR® server to respond to incoming requests in one of these three formats (HL7, 2017g). The Capability Statement resource specifies the server's capabilities in terms of supported formats, security service adopted, supported use case profiles, and other specifications (HL7, 2017h), which, in earlier versions, was specified by the Conformance resource.

The standard possesses more than 115 resources. However, regarding the FHIR Maturity Model (cf. section 2.3.2), only eleven resources are at level 5 of maturity, namely: Binary, Bundle, CodeSystem, DomainResource, Observation, OperationOutcome, Parameters, Patient, Resource, StructureDefinition, and ValueSet. Each resource has its own specification, which defines the relationships with other resources, its content, constraints, and other relevant aspects that are related to it. The specification about all the existing resources can be found in the official FHIR® specification for the current release (HL7, 2017i).

2.3.2 FHIR Maturity Model

HL7 defined a maturity model named FHIR Maturity Model (FMM) to specify the FHIR resources' current level of maturity. FMM describes seven levels of maturity based on a set of criteria that defines the stability of the artifact, similar to the Capability Maturity Model (CMM), in which a certain level is reached only if the criterion from the current level is met, including the criteria from the previous levels (Humphrey, 1987). FMM comprehends the levels presented in Table 1, which content was based on the criteria defined by HL7 regarding the maturity levels from FHIR® specifications (HL7, 2017j).

Table 1 – FHIR Maturity Level (HL7, 2017j)

FMM Level	Criteria
0	• Published on current build
1	• Respects FMM Level 0 criteria • Build with no warnings • Considered substantially complete by the work group
2	• Respects FMM Level 1 criteria • Tested and successfully exchanged between three or more systems respecting some data content information and scopes • Interoperability results accepted by the FHIR Management Group
3	• Respects FMM Level 2 criteria • Trial Use Quality Guidelines verified by the work group • Subject to a round of formal balloting • Had at least ten implementer comments from three or more organizations • Result of one or more substantive changes
4	• Respects FMM Level 3 criteria • Tested across its scope • Published in a formal publication • Implemented in multiple prototype projects • Stability to require implementer consultation for subsequent non-backward compatible changes agreed by the work group
5	• Respects FMM Level 4 criteria • Published in two formal publications release cycles at FMM1+ • Implemented in five or more independent production systems, at least in two countries
Normative	• Stable

2.3.3 Extensibility

Resource instances can be defined along with an extension field, which is a FHIR specification that enables the instance to contain additional information that is not part of the resource's base definition, by using a URL that points to its meaning. Each application should deal accordingly with receiving resource instances that contain an extension, by ignoring unknown extensions and processing extensions that are defined and published. The Capability Statement resource is also used to indicate which servers do not accept unknown extensions (HL7, 2017k).

2.3.4 FHIR Exchange Paradigms

FHIR defines a set of possible exchange paradigms to be applied for development, which are also documented with their own specification for the current standard's version.

2.3.4.1 RESTful API

FHIR is marked by its simple exchange mechanism, based on a client/server API, to manipulate medical records following a RESTful design. FHIR respects the third level of the Richardson Maturity Model, regarding the correct use of HTTP verbs for each operation (Richardson, 2008), thanks to its core specification and resource distribution throughout the different healthcare entities, providing a vast range of HTTP operations to manipulate resources. Either way, by applying extensions to the resources, it is possible to reach the fourth level, regarding the hypermedia controls, although the use of extensions is only taken into consideration if there is a need to include additional requirements that are not part of the resource's basic definition, since it raises the level of complexity (HL7, 2017l).

2.3.4.2 Messaging

Besides the RESTful API, FHIR® also supports a messaging exchange framework, which is like HL7 V2's messaging paradigm and doesn't require a specific transfer mechanism. It is an exchange mechanism between a sender and a receiver where the messages' content is known by both applications, following synchronous or asynchronous exchange patterns. Typically, the request message is composed by a Bundle resource of type "message", which contains a MessageHeader resource that identifies the event. Based on the receiving ids from the request message, the receiver can verify if there was a problem with the message's processing and act accordingly to respect the transactions' exchange patterns (HL7, 2017m).

2.3.4.3 Documents

Like HL7 CDA®, FHIR® can create XML and JSON documents based on a set of resources, which can either contain clinical information, regarding patient's healthcare data, or other type of information such as guidelines. Each document is composed by a Bundle resource of type "document" with an inner Composition resource and other resources that specify the documents' content and information (cf. Example 4).

Example 4 – FHIR XML Document example

```xml
<Bundle xmlns="http://hl7.org/fhir">
  <id value="father"/>
  <meta>
    <lastUpdated value="2013-05-28T22:12:21Z"/>
  </meta>
  <identifier>
    <system value="urn:ietf:rfc:3986"/>
    <value value="urn:uuid:0c3151bd-1cbf-4d64-b04d-cd9187a4c6e0"/>
  </identifier>
  <type value="document"/>
  <!--      The Composition resource      -->
  <entry>
    <fullUrl value="http://.../Composition/..."/>
    <resource>
      <Composition>
      ...
      </Composition>
    </resource>
  </entry>
  <!--      The Patient resource      -->
  <entry>
    <fullUrl value="http://.../Patient/..."/>
    <resource>
      <Patient>
      ...
      </Patient>
    </resource>
  </entry>
  ...
```

The Composition resource identifies the document, provides information about the document's subject and author, and contains references to the resources that are to be added in the document's sections upon its creation, among other valuable information (HL7, 2017n).

2.3.4.4 Services

The services exchange paradigm suggests the adoption of a Service Oriented Architecture (SOA) along with FHIR® to implement the desired web services. The application of SOA brings advantages regarding modularity, error handling, orchestration of services, security and other relevant aspects, since it reduces the dependencies between the client and the FHIR server, by distributing the responsibilities among components that shall deal with these concerns. This frees the client from additional responsibilities, promotes loose-coupling, and, considering future changes to some FHIR resources' maturity, mitigates the risk of future problems to several components due to one's modification (HL7, 2017o).

2.3.4.5 Considerations

Currently, the defined FHIR exchange paradigms specification regarding Messaging, Documents and Services are still in need for improvement and approval. Some of its features and approaches need further evaluation from HL7 in order to reach a stable level of maturity for public usage in an interoperable environment (HL7, 2017p). Nonetheless, these paradigms' specification does not impact the solution's development, since the development follows the RESTful API paradigm defined in the FHIR specification, which is well matured.

2.3.5 Comparison with previous HL7 standards

HL7 V2 uses a specific messaging structure based on message delimiters (for example, the pipe) to exchange information, while HL7 V3 exchanges data in XML format, including the documents based on the CDA standard. FHIR®, on the other hand, uses REST resources to exchange this information with REST services in a finer granular way. Its improved capabilities also allow FHIR to create documents for exchange, without the need to use CDA documents (which can still be used for exchange in FHIR®).

Table 2 presents a features comparison between the HL7 standards.

Table 2 – Comparison between HL7 standards

	HL7 V2	**HL7 V3**	**HL7 CDA**	**HL7 FHIR**
Messaging paradigm	✓	✓	×	✓
RIM Reference Model	×	✓	✓	×/✓
Standard's Extensibility Feature	✓	✓	✓	✓
Human Readable Content and Linkages	×	×	✓	✓
XML encoding	×	✓	✓	✓
JSON encoding	×	×	×	✓
RESTful paradigm	×	×	×	✓

As it can be seen in Table 2, FHIR® contains new features regarding JSON data encoding along with the RESTful paradigm, holding major features from the previous standards, such as the human readable content with the resource's informative narrative, and XML data encoding. However, some of these features are not fully implemented in FHIR®, which is the case for the RIM Reference Model that is still not supported by some of the existing resources, due to their current level of maturity.

2.4 IHE Profiles

Integrating the Healthcare Enterprise (IHE, 2016a) is an organization which delivers a set of profiles to address clinical needs and improve the exchange of healthcare information (IHE, 2016a). The IHE profiles are complete and documented implementation approaches for specific clinical requirements and common use cases, providing developers with a detailed explanation on how to use several standards in a structured manner to develop clinical features in healthcare interoperable products. It also serves as a global definition of interoperable features for both vendors and purchasers to discuss integration requirements for healthcare institutions' systems.

This organization annually arranges an event called *Connectathon*, where hundreds of healthcare IT organizations test their IHE Profiles implementations with other systems, using IHE's test environment, to obtain certifications of interoperability for their products (IHE, 2016b). Figure 3 defines the overall process from the profiles definition to the implementation of these.

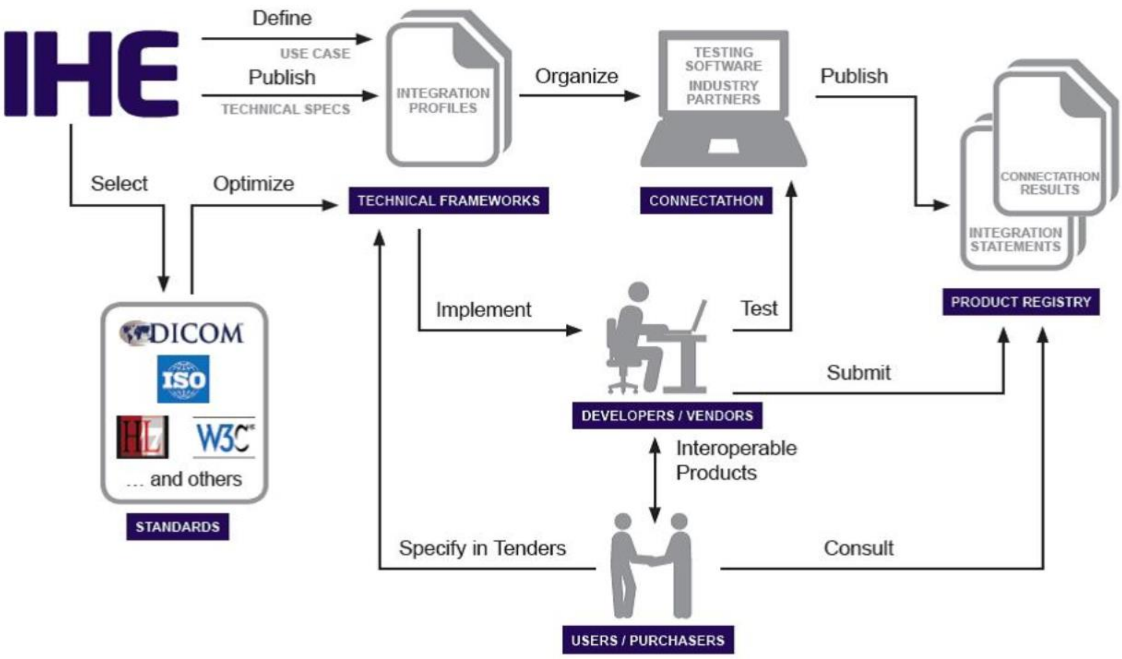

Figure 3 – IHE Process (IHE, 2016c)

IHE defines use cases with major relevance for healthcare organizations and publishes the technical specifications for those use cases. These specifications include the detailed explanation on how to implement the necessary features, using a set of the most suitable standards for application in the given scenario. The healthcare IT organizations test their profiles' implementation in *Connectathon*, which are then evaluated by IHE and published.

IHE profiles are related to multiple clinical domains, namely: Anatomic Pathology; Cardiology; Dental; Endoscopy; Eye Care; IT Infrastructure; Laboratory; Patient Care Coordination; Patient Care Devices; Pharmacy; Quality, Research and Public Health; Radiation; Oncology; and Radiology. Currently, ALERT interoperability product adopts IHE profiles in the IT Infrastructure domain, plus, although IHE possesses several amounts of profiles, since FHIR® is the standard of focus for this project, the profiles of interest are:

- **Patient Identifier Cross-Reference for Mobile (PIXm):** being similar to IHE's PIX and PIXv3 profile, PIXm defines a RESTful interface to query patient identifiers from different domains, retrieving the correlated identifiers to the requesting application (IHE ITI Technical Committee, 2017a);

- **Patient Demographics Query for Mobile (PDQm):** being similar to IHE's PDQ and PDQv3 profile, PDQm defines a RESTful interface to query for patients' demographic information, retrieving that information to a requesting application (IHE ITI Technical Committee, 2017b);

- **Mobile access to Health Documents (MHD):** being similar to IHE's XDS profile, MHD defines a RESTful interface with transactions to submit a set of documents and metadata to a document receiver, and to query for document submission sets and entries, in order to retrieve the associated document (IHE ITI Technical Committee, 2017c);

- **Mobile Cross-Enterprise Document Data Element Extraction (mXDE):** defines an approach to share and access documents (Document-Level Granularity) composed by multiple data elements, or to access these data elements (Data Element-Level Granularity) from shared structured documents (IHE ITI Technical Committee, 2017d);

- **Audit Trail and Node Authentication (ATNA):** defines how to establish security measures to support patient information confidentiality, data integrity and user accountability (IHE ITI Technical Committee, 2017e, sec. 9);

- **Internet User Authorization (IUA):** defines an approach to manage authorization tokens, in order to perform HTTP RESTful transactions (IHE ITI Technical Committee, 2015);

The technical specifications for these profiles are described with more detail in section 4.1.

2.5 ALERT Solutions

This section describes the ALERT products along with the architecture from the project's product of focus, ALERT® HIE.

2.5.1 ALERT® Products

ALERT has a wide range of products, but for the scope of this document, only ALERT® PAPER FREE HOSPITAL (ALERT® PFH), MyALERT® PERSONAL HEALTH RECORD (MyALERT® PHR), ALERT® PRIVATE PRACTICE, ALERT® PRIMARY CARE and ALERT® HEALTH INFORMATION EXCHANGE (ALERT® HIE) are referred since these are the main solutions that directly involve exchange of health information between multiple institutions.

ALERT® PFH is a solution typically used by the whole hospital community, that contains a set of specialized applications for different areas, with information about the patients' health conditions, medication, clinical processes, monitoring and other hospital operations (ALERT Life Sciences Computing, 2017a).

MyALERT® PHR is a solution created for the patients to manage their Personal Health Record. It allows them to monitor their health conditions, consult their personal medical history and share medical information which can then be read by physicians, providing valuable material for clinical encounters (ALERT Life Sciences Computing, 2017b).

ALERT® PRIVATE PRACTICE is a smaller solution suitable for private clinics and smaller institutions, which possesses designated templates that vary upon the clinic's needs and specialties, including relevant functionalities to document and manage the patient's clinical information and medical history (ALERT Life Sciences Computing, 2017c).

ALERT® PRIMARY CARE is a solution aimed for primary care centers, that provides the means to manage individual patient records and consult the patients' information from various facilities (ALERT Life Sciences Computing, 2017d).

Last, but not least, ALERT® HIE is the core solution that connects all the clinical information from multiple systems. It is an integration infrastructure that adopts international standards for information exchange, which guarantees interoperability even if the connected systems are not part of ALERT's product suite. The solution contains a set of modules that follow IHE profiles (IHE, 2017) to address clinical needs and promote an efficient use and sharing of information (ALERT Life Sciences Computing, 2017e). Figure 4 demonstrates a generic view of the centralized system that ALERT® HIE represents, and the other systems connected to it, to highlight the product's role in exchanging information, both among ALERT's products and among external institutions' products.

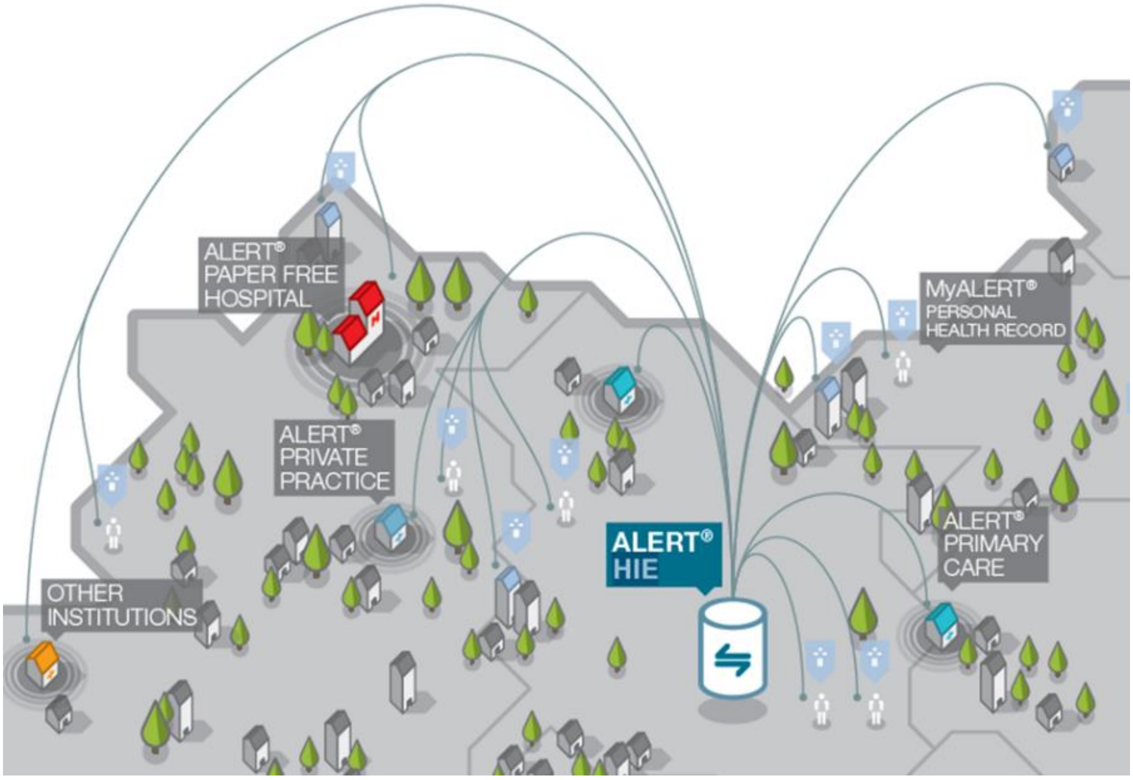

Figure 4 - ALERT® HIE connections (ALERT Life Sciences Computing, 2017e)

2.5.2 ALERT® HIE Architecture

ALERT® HIE is the product of focus in this project, since all the development stages impact its architecture and internal components. Currently, ALERT® HIE is composed by mainly seven modules, which are related to the current implemented IHE profiles, namely:

- **ALERT® PIXPDQ (IHE's PIX/PIXv3 profile and IHE's PDQ/PDQv3 profile):** correlates patient identifiers registered in different healthcare information systems, inside the same community;
- **ALERT® XDS (IHE's XDS profile):** enables document sharing and storage by multiple healthcare facilities, being managed by a Document Registry, which contains the list of published documents in existing repositories, and a Document Repository, that holds the patient's related clinical documents;
- **ALERT ® XDS Affinity Domain (IHE's XDS profile):** enables data share between healthcare enterprises that agreed upon a set of policies to share a common infrastructure;
- **ALERT® XCA (IHE's XCA profile):** enables the retrieval of patients' medical data from other communities;

- **ALERT® XCPD (IHE's XCPD profile):** enables the cross-referencing of patient identifiers across different communities;

- **HIE-Security (IHE's XUA profile):** provides security features and enables the identification of authenticated entities for transactions that exceed the enterprise boundaries.

Figure 5 presents a high-level diagram with HIE's modules, along with the service endpoints provided by each one of them.

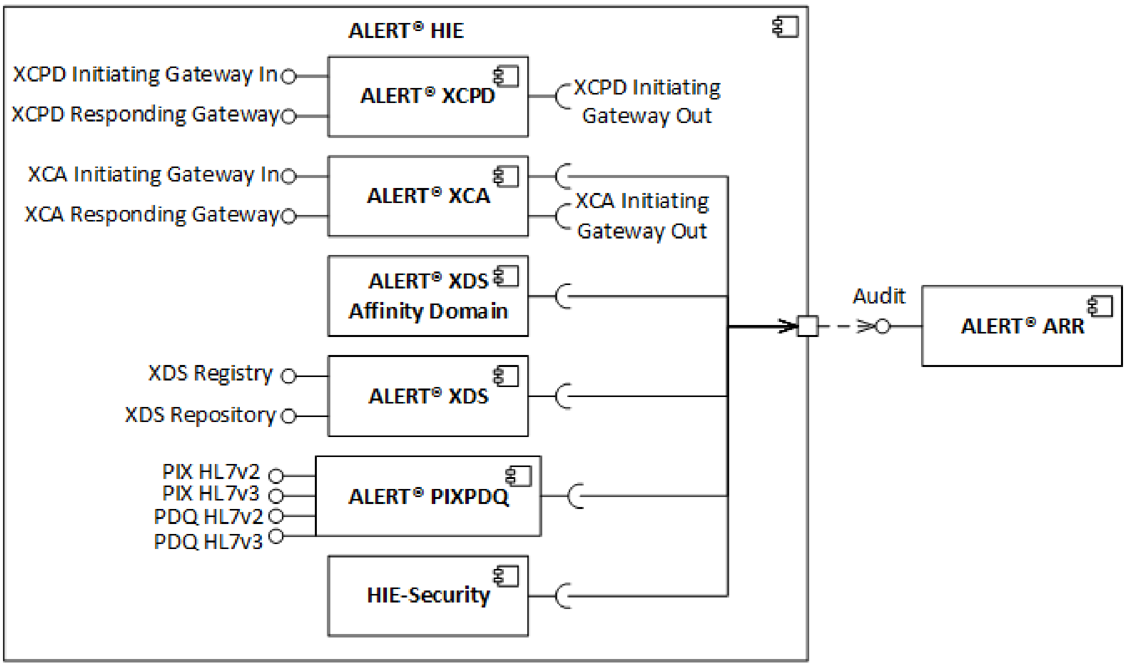

Figure 5 – HIE high level architecture diagram

Each module is developed using HIE frameworks, which also provide a set of core features and APIs to build each module's services, transactions, handlers, and additional business logic to define the overall workflow. Additionally, ALERT® ARR is also represented in the diagram, since it applies IHE's ATNA profile, and acts as a log repository for auditing.

2.6 Frameworks for REST services

Since FHIR® fully adopts REST, existing technologies that make use of REST services or FHIR® specifications, including some security measures, might be suitable to aid on the FHIR® services' implementation. This way, a set of existing frameworks and tools on the market are identified and described in this section.

2.6.1 JAX-RS

JAX-RS is a Java API developed by Oracle Corporation (Oracle, 2017a) which is suitable for implementing RESTful web services. It contains a set of annotations to define resources and actions, which are the core features to perform and respond to HTTP requests, among other features (Oracle, 2013, pp. 381–401).

2.6.1.1 Security

In terms of security, JAX-RS possesses specific features that can restrict the access to services depending on the authenticated user's role, which specifies if the user has permission, or not, to invoke those methods in the application (Oracle, 2017b). Security restrictions can be applied using three methods, namely:

- Web.xml;
- SecurityContext;
- Annotations.

With web.xml, the security feature can be configured with a `<security-constraint>` element in a xml file by indicating the service URL and adding a constraint of access to it with defined security roles. A `<login-config>` element, indicates the type of authentication to be applied for the defined security constraints.

With SecurityContext, the security measures can be applied directly in the implementation of the services, by injecting an instance of SecurityContext using Context annotation, which can then be used to check if the user requesting that service has permissions or not to request it, for example:

Example 5 – JAX-RS SecurityContext (Oracle, 2017b)

```
public String sayHello(@Context SecurityContext sc) {
       if (sc.isUserInRole("admin"))  return "Hello World!";
       throw new SecurityException("User is unauthorized.");
}
```

Lastly, annotations provide the means to define which roles may access certain methods and classes, regarding the services implementation. For example, with the annotation @RolesAllowed("admin"), only the users that belong to the role *admin* can access or request any services that hold that annotation.

2.6.1.2 Ease of Implementation

The JAX-RS annotations to map HTTP requests into Java methods, provide an easy approach to implement web services. It provides annotations for the common HTTP methods (@GET, @PUT,

@DELETE and @POST), annotations to define the resources' URI path (@Path), annotations for the message type that a server resource can consume, the type of message response possible, and other annotations which can also be included in the services.

2.6.1.3 Supportability

Considering the whole scenario regarding FHIR®, which will mainly involve the implementation of server-side services along with the standardized resources, although JAX-RS annotations for Java web services ease the implementation process, the fact that it is not directly related to FHIR® requires extra work to correctly implement the necessary resource representations for the requests, according to the current release of FHIR®.

2.6.1.4 Overview

JAX-RS provides the necessary features to implement RESTful web services, which is a demanding point to implement FHIR. As previously mentioned, JAX-RS doesn't support or follow FHIR specifications, which means that, to guarantee that the REST services consume standardized FHIR resources, the profiles, value sets and the remaining specification would need to be integrated in the product to validate the resources and requests. Luckily, HL7 already provides the JSON and XML resources' schemas, including necessary definitions and a FHIR validator that can be used to validate these resources, although only supported in XML format.

2.6.2 HAPI FHIR

HAPI FHIR is a library with a FHIR® specification for Java, developed by the University Health Network (University Health Network, 2017), which contains features for the implementation of RESTful clients and servers, along with the adoption of FHIR® model objects.

2.6.2.1 Security

HAPI FHIR provides specific approaches for security implementations regarding FHIR®, using Interceptors for authentication and authorization measures to verify the user's permissions to perform certain operations in a FHIR server. HAPI provides an interceptor called AuthorizationInterceptor, which examines the client requests, verifying if write operations (for example, create and update) can be performed by the user before creating or modifying any resource. It also examines the responses from read operations (such as read and search), to verify if the data retrieved can be read by the user, especially in situations where the client requests additional resources related to the ones from the search results. Custom interceptors can also be created by simply implementing the IServerInterceptor interface, or extending the InterceptorAdapter class, in order to apply a customized crosscutting concern to the server methods (University Health Network, 2018a).

2.6.2.2 Ease of Implementation

The library possesses two types of client, namely a generic client with a simple approach to implement the operations, and an annotation-driven client like the JAX-RS approach. HAPI already contains a set of functions which are necessary to be implemented to communicate with FHIR® servers.

HAPI uses mainly annotations to create a RESTful server (Servlet) and resource providers to deliver a supported resource in FHIR's current specification, including plain providers, which are not directly associated with a specific FHIR resource. RESTful operations can then be added to each resource or plain provider created, which HAPI also supports by using a set of annotations for operations such as, @Read, @Search, @Create and @Update, and for parameters such as, @IdParam, @ResourceParam, among others. It also includes JAX-RS providers which can be used as an alternative for the HAPI FHIR clients that use the Apache HTTP provider by default, and extension features regarding the FHIR® specification (University Health Network, 2018b).

In general, the services are very simple and easy to develop, although some operations such as @Transaction might require challenging implementations due to its complexity and capability to perform multiple operations for multiple resources in one action (University Health Network, 2018c).

2.6.2.3 Supportability

HAPI FHIR resources and operations are all according to FHIR's current specification in Java language. HAPI also possesses an automatic feature to export a Conformance Statement, which, as previously mentioned in section 0, is required by FHIR specification to verify the server's supported resources and capabilities, by verifying the annotations implemented in the server. Additionally, all the operations it uses are from FHIR specification, which enables the server to implement operations such as @Validate, that can verify if a resource is valid or not for saving to the server.

2.6.2.4 Overview

The fact that HAPI adopts full FHIR support (Figure 6), is, at first hand, a major advantage to implement FHIR solutions since it provides some useful features.

HAPI library is composed by mainly five modules:

- **Core Libraries:** possesses the core features that are required to use the framework;
- **Structures:** contains the model classes from the multiple FHIR versions;
- **Client Framework:** includes HTTP implementations for the FHIR client framework;
- **Validation:** contains the standardized profiles from the multiple FHIR versions, and a validator to approve the resource instances according to these profiles;

- **Server:** includes a server framework to develop FHIR compliant servers.

The core FHIR features are already implemented, which not only saves effort and time, but also leaves space to focus on the business logic and use cases development.

HAPI Version	DSTU1	DSTU2	DSTU2.1	DSTU3	R4
HAPI FHIR 1.1	0.0.82	Draft 0.5.0-5843			
HAPI FHIR 1.2	0.0.82	Draft 0.5.0-5843			
HAPI FHIR 1.3	0.0.82	1.0.2			
HAPI FHIR 1.4	0.0.82	1.0.2		Draft 1.3.0-7602	
HAPI FHIR 1.5	0.0.82	1.0.2		Draft 1.4.0-8138	
HAPI FHIR 1.6	0.0.82	1.0.2		Draft 1.4.0-8636	
HAPI FHIR 2.0	0.0.82	1.0.2		Draft 1.6.0-9663	
HAPI FHIR 2.1	0.0.82	1.0.2		Draft 1.7.0-10129	
HAPI FHIR 2.2	0.0.82	1.0.2	Draft 1.4.0	Draft 1.8.0-10528	
HAPI FHIR 2.3	0.0.82	1.0.2	Draft 1.4.0	Draft 1.9.0-11501	
HAPI FHIR 2.4	0.0.82	1.0.2	Draft 1.4.0	3.0.1	
HAPI FHIR 2.5	0.0.82	1.0.2	Draft 1.4.0	3.0.1	
HAPI FHIR 3.0.0		1.0.2	Draft 1.4.0	3.0.1	Draft 3.1.0-12370
HAPI FHIR 3.1.0		1.0.2	Draft 1.4.0	3.0.1	Draft 3.1.0-12370
HAPI FHIR 3.2.0		1.0.2	Draft 1.4.0	3.0.1	Draft 3.2.0-12917

Figure 6 – FHIR versions supported by HAPI FHIR (University Health Network, 2018d)

The next examples shows some of the existing features that HAPI possesses considering FHIR specifications, namely a JSON and XML parsing (Example 6), manipulation of FHIR model objects (Example 7), and FHIR RESTful operations (Example 8), considering the specification version in use.

Example 6 – FHIR resource XML and JSON parsing (University Health Network, 2018b)

```
FhirContext ctx = FhirContext.forDstu2();
String xmlEncoded = ctx.newXmlParser().encodeResourceToString(patient);
String jsonEncoded = ctx.newJsonParser().encodeResourceToString(patient);
```

Example 7 - FHIR Patient model object (University Health Network, 2018b)

```
Patient patient = new Patient();
patient.addIdentifier().setUse(OFFICIAL).setSystem("urn:fake:mrns").setValue("
7000135");
patient.addIdentifier().setUse(SECONDARY).setSystem("urn:fake:otherids").setVa
lue("3287486");
patient.addName().addFamily("Smith").addGiven("John").addGiven("Q").addSuffix(
"Junior");
patient.setGender(AdministrativeGenderEnum.MALE);
```

Example 8 – FHIR search operation (University Health Network, 2018c)

```
@Search()
public List<Patient> searchByIdentifier(@RequiredParam(name=Patient.SP_IDENTIFIER)
TokenParam theId) {
   String identifierSystem = theId.getSystem();
   String identifier = theId.getValue();
   List<Patient> retVal = new ArrayList<Patient>();
   // ...populate...
   return retVal;
}
```

Considering that the previous HL7 services are also developed in Java, the adoption of this framework would provide an efficient way to extend the product with FHIR services.

2.6.3 Smile CDR

Smile CDR is a Clinical Data Repository (CDR) integration tool which is powered by the HAPI FHIR library, providing a long-term support for the published versions of FHIR®.

2.6.3.1 Security

Smile CDR server is composed by nodes with multiple modules, which includes an Inbound Security Module that handles authentication actions for incoming requests to the server, and authorizes the requests based on the associated user. The authentication is applied using user accounts and salted password hashes in an existing database, which can also be Smile CDR's administration database. Moreover, additional security measures can be added to Smile CDR, for example, FHIR endpoints can be configured to verify OAuth2 bearer tokens to apply authentication.

2.6.3.2 Ease of Implementation

With Smile CDR, the services implementations are mainly based on configurations, since the overall platform is a local or cloud server that provides a friendly user interface with multiple options to define FHIR REST Endpoints, database configurations, security modules with authentication, authorization and auditing, including other configurations related to FHIR specifications.

2.6.3.3 Supportability

Smile CDR supports the creation of the storage and endpoints that correspond to the current FHIR version, regarding the resources specifications. Since Smile CDR adopts HAPI FHIR, the modules are expected to follow FHIR specifications, with the option to use the most recent release of FHIR®, or previous released versions.

2.6.3.4 Overview

Smile CDR provides a set of configurations which can be used to manage existing users, storage, endpoints, security, search parameters for FHIR resources, among other features. Figure 7 presents the existing modules that can be created and configured in the server, for example, FHIR REST Endpoint (R3) defines an endpoint for FHIR services in the latest release, which is dependent from persistence and security modules, due to the established architecture (Figure 8).

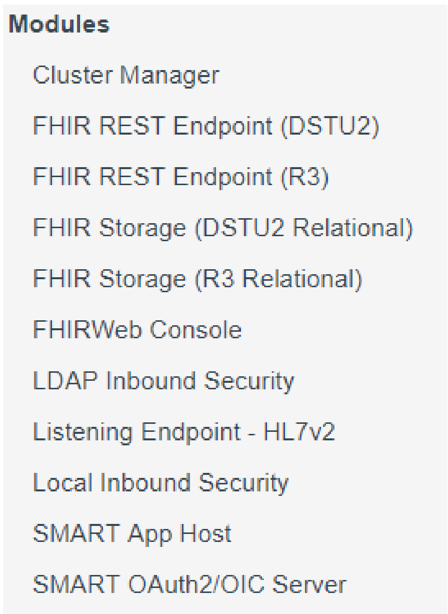

Figure 7 – Smile CDR Modules (Simpatico Intelligent Systems, 2018)

One of the great advantages of this framework is the fact that is developed based on HAPI FHIR, which means that the defined resources, and search parameters used by the FHIR endpoint to perform the requests, correspond to the current FHIR specification, since HAPI monitors FHIR releases. Moreover, the configuration page offers an easy method to configure the endpoints with security measures and connections to databases, reducing the amount of effort and time to apply these features.

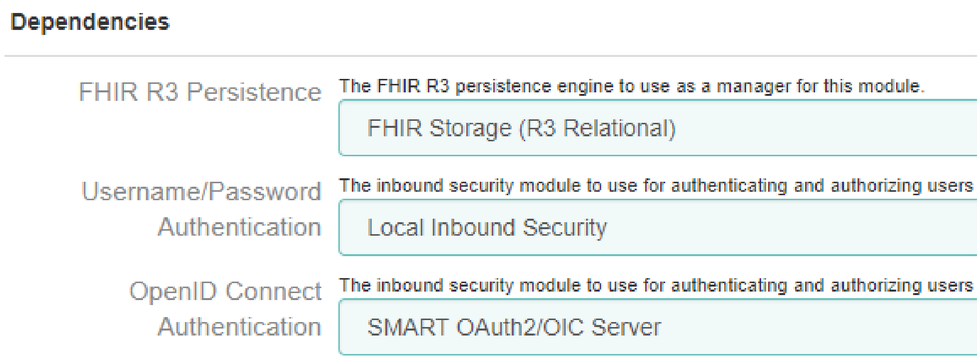

Figure 8 – FHIR endpoint dependencies (Retrieved from Smile CDR Module Configuration)

However, the architecture defined for Smile CDR (Figure 9), partially limits implementation approaches since the main purpose of this framework is to provide FHIR services for resources, where the defined endpoint is directly connected to a database that will store the standardized resources. The use of an endpoint like this can be useful for certain situations to store clinical data that correspond to a standardized FHIR resource, but since the workflow of operations sometimes only require the adoption of FHIR services to respect the resources profile and structure for the exchange of information, not involving a direct storage of that resource, its adoption would limit some complex workflows.

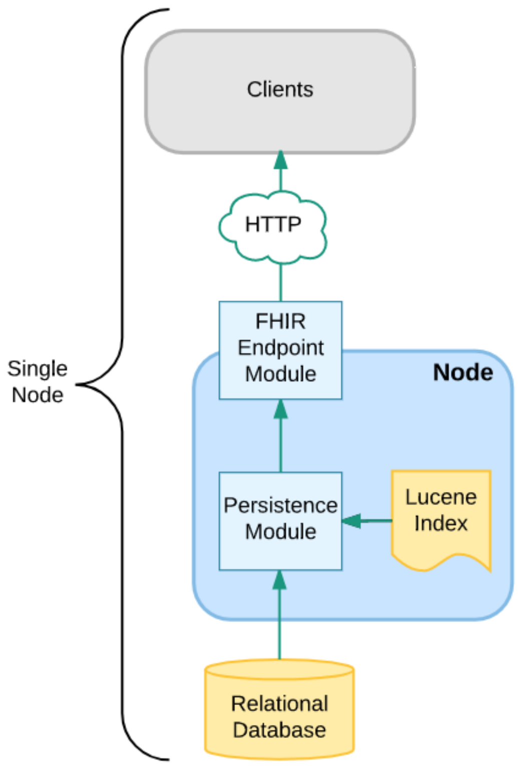

Figure 9 – Smile CDR Architecture Diagram (Simpatico Intelligent Systems, 2018, sec. 4.1.4)

2.6.4 Iguana 6

Iguana 6 is an HL7 integration engine, which is composed by a set of configurable channels that listen for incoming requests, transmitting those requests to a destination, with the possibility to contain a filter between the source and the destination to translate the message's content, using Lua scripting language in Iguana's development environment (iNTERFACEWARE, 2017).

2.6.4.1 Security

Iguana 6 uses a default user and role mechanism for authentication, with a possible alternative to use an external authentication method using a URL that supplies that feature. It also provides logging features that holds auditing information for message events and user sessions, and includes sample channels with authentication, authorization and encryption examples.

2.6.4.2 Ease of implementation

Iguana 6 enables the addition of endpoint channels to the server, that contain a set of scripts which build the channel's logic. In terms of extensibility with FHIR®, new script resources can be developed in the integration tool, along with the necessary datatypes and resource profiles from FHIR's current specification. For every new resource, it should exist an associated mapping script between the FHIR object (JSON or XML) and the existing database, which also requires a script to define the connection.

2.6.4.3 Supportability

Iguana provides channels that can be imported with basic functionalities implemented at first hand. Regarding FHIR, Iguana provides channels with FHIR implementations which can be imported, namely a FHIR Server and a FHIR Client. The FHIR server channel, already provides an example with a Patient resource, the corresponding mappings from the provided Iguana database to JSON or XML and vice-versa, and four HTTP method handlers (GET, POST, PUT and DELETE). Although it provides a starting point for a FHIR server, the scripts' specifications need to be manually added and consulted to guarantee that it corresponds to the official specification for the standardized resource in the current FHIR release.

2.6.4.4 Overview

There are some benefits that can be acquired with this framework, but it also comes with some problems. The fact that Iguana can provide the developer with an interface to configure multiple channels with HTTP endpoints to listen for incoming requests, offers an easy and effective way to establish a business logic, since these channels can also communicate with each other, providing the means to distribute responsibilities and functionalities to each channel (Figure 10). However, to establish a FHIR server, the REST services shall respect the standard's specification, which means that each resource required for the services would need to be standardized as well.

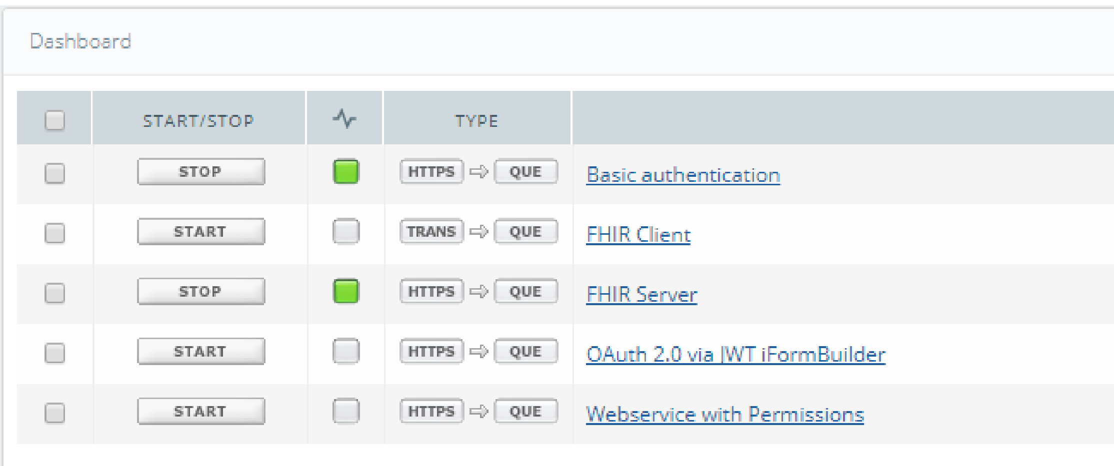

Figure 10 – Iguana channels (Retrieved from Iguana's local server dashboard)

Considering the way how Iguana is built, it's necessary to possess scripts that define the JSON and XML profile of the resources (Figure 11), according to what is defined in the official FHIR resources' specification.

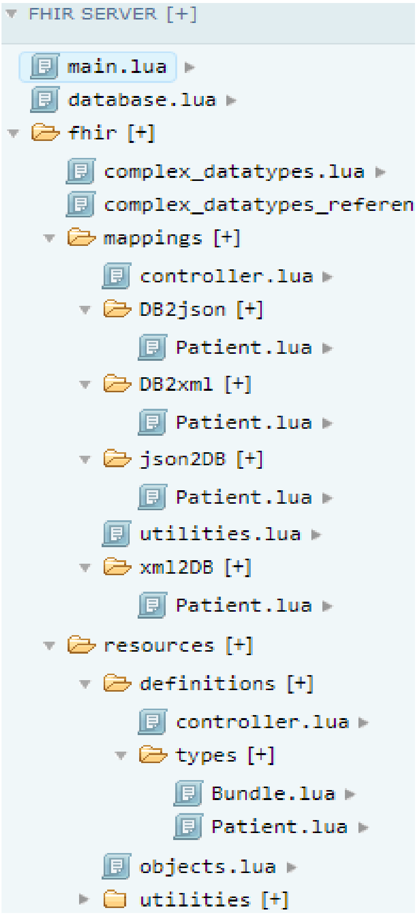

Figure 11 – Iguana FHIR Server channel example (Retrieved from Iguana Webservices Repository)

It'll be required extra development to keep the resources and mappings updated, since Iguana doesn't follow directly FHIR's version releases. Regarding security, the authentication and authorization measures provided by Iguana's repository channels are useful as a first approach of security integration in the server for later adaption, depending on the demanding requirements.

2.6.5 Frameworks Comparison

Table 3 presents a comparison of the most relevant capabilities from the analyzed frameworks. Only HAPI FHIR and Smile CDR include released FHIR specifications, which is a major aspect to consider in a framework, since FHIR is evolving fast and constantly updating their specifications, requiring future updates in the implementation. In terms of REST services, every framework has the capability to define web services which is also a relevant characteristic since it represents the core feature of FHIR®. Similarly, each framework possesses features and approaches to provide, at least, basic security measures for HTTP requests. The last major aspect to consider is the capability of the framework to be applied for a whole business process.

Table 3 – Frameworks comparison

	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
Includes FHIR specifications	×	✓	✓	×
Ease to define REST web services	✓	✓	✓	✓
Provides multiple security features	✓	✓	✓	✓
Can be used to define a whole business workflow	✓	✓	×	✓

Since JAX-RS and HAPI FHIR are frameworks that can be easily integrated in a project developed in Java, their features can be applied in multiple steps of a complex business workflow. The same can be said for Iguana 6, which, thanks to its features, provides the means to define a business workflow, through channel communications, that can be executed through external requests to defined HTTP endpoints. Oppositely, Smile CDR can't provide this at such extent, since its core feature is to provide interoperability for FHIR requests to store or retrieve standardized resources. Therefore, the framework can be useful to handle FHIR requests during a business workflow process, but it can't extend those requests for further processing.

3 Project Context

In this chapter, the problem addressed in this work is described in more detail, as well as the stakeholders, the requirements and the business value analysis.

3.1 Problem

The current section explains with detail the problem involved in this project, considering the proposed questions from the chapter Analyzing a Design Using the Elements of Thought, relative to the book The Thinker's Guide to Engineering Reasoning (Paul et al., 2007), regarding the eight fundamental elements: Engineering Purpose, Question at Hand, Point of View, Assumptions, Engineering Information, Concepts, Inferences and Implications. The meticulous description of the problem will help discuss specific aspects that affect the solution's analysis, design and development, obtaining in this way a global picture of the considerations to follow during the several project's phases. The respective questions can be found in Annex A.

3.1.1 Engineering Purpose

The following points focus on the first fundamental element, in specific to the design's purpose, the market opportunities, mission requirements and the main involved customers.

3.1.1.1 Design's purpose

The main purpose is to develop an interoperability infrastructure based on the latest HL7 standard, FHIR®, to serve as a HIE system, capable of sharing coherent clinical data between an ALERT® customer and other external systems. These systems, that shall adopt FHIR®, can be other customers who also use ALERT®, other HIE systems that belong to non-ALERT customers, and Personal Health Record (PHR) applications for citizens.

3.1.1.2 Market opportunities and mission requirements

The arrival of FHIR® brings new opportunities and value for the clients. The application of this standard expects to improve access to medical information thanks to its enhancement on granularity when it comes to data exchange. Compared to the previous standards in the market, FHIR® brings an easier implementation following a RESTful approach, which also opens the chance to create lightweight mobile health applications, make use of cloud communications and perform operations in a more flexible way (HL7, 2017f).

The market opportunities and mission requirements are defined by the company's vision and strategy, through information gathered from multiple sources, such as, the state of the standards developed by HL7, the IHE profiles and discussions, the HIMSS conferences, the customer feedbacks, the potential market, legal and industry requirements, the healthcare technological trends, among others.

3.1.1.3 Target customers

This project aims for the customers that want to, or, already use ALERT® products to perform clinical tasks and share information with other health organizations, for example, hospitals and private clinics, as well as patients that want to consult their personal information and medical history. It is expected that this project continues to guarantee the interoperability between ALERT® products and the rest of the existing products in the market, including the systems that already adopt FHIR® to share clinical information.

3.1.2 Question at Hand

The following topics focus on the second fundamental element, which refers to the customer's requirements and value, the design's requirement and the importance of time-to-market.

3.1.2.1 Suitable product for the customer's requirements

With the current standards, the exchange of health information is focused on a document paradigm. Full medical reports are shared with information about patient's conditions, doctor's prescriptions, patient treatments and so on. These documents have certain advantages regarding wholeness and stewardship, but the granularity of the information is coarse, and the documents are accessed as single units. FHIR® presents information as resources with a fine granularity, making it easy to compile and retrieve the relevant information from among multiple sources, for it to be efficiently accessed by the customer.

3.1.2.2 Value for the customer

In a complex business area such as healthcare, the decisions bring along a great responsibility and can seriously impact the patient's health if not analyzed correctly. Physicians, nurses and other health personnel need to pay special attention to the prescribed medication and clinical procedures, since the patient’s might have some health conditions that can negatively respond to this medication or these procedures.

Therefore, for the customer, value is gained from the capability to collect relevant and actionable information that can help gain knowledge about the patient’s health, assisting on the specific tasks that better suits his current state and, consequently, improving decision-making. Furthermore, studies have shown (Vahdat et al., 2014) that when patients are more involved in the decision-making process, they are more likely to actually follow the instructions given by medical personnel. This, coupled with an increased concern from citizens with the state of their own health, means there is value in applications that provide patients with access to this information (which FHIR®, with its lightweight nature, is well suited to providing). The innovations that FHIR® brings, related to the access of information with fine granularity, also open the possibility to select a portion of desired data from health documents, which in turn brings advantages for the clients’ data protection and promotes a careful selection of which information is to be retrieved and exchanged between institutions. This topic is further detailed in section 3.4.

3.1.2.3 Design requirements

For this project, a new design is required to implement the FHIR ® specifications with appropriate services, to provide a REST gateway for some of the existing tasks that already adopt previous standards and to add new services suitable for the new standard.

3.1.2.4 Existing designs

It is expected that the developed solution can be integrated into the existing one, which already contains other services regarding previous HL7 standards, to extend it with the services developed for FHIR®, according to the project’s needs.

3.1.2.5 Importance of time-to-market

FHIR® was first released as a “Draft Standard for Trial Use” (DSTU) in 2014 and is currently in its third release as a “Standard for Trail Use” (STU), being the latest standard in the market, but still in an early stage of development. Companies started to develop some implementations that adopt this standard, but some of its content will continue to suffer modifications and improvements until it reaches a stable maturity level in future releases (HL7, 2017j). Thus, although it brings major advantages and value for organizations, the fact that it’s still in trial version means that not all health organizations will desire to adopt FHIR® right away, and the ones that does, will not be able to share information based on FHIR® with organizations that do not use this standard. This results in a lack of interest from some organizations to join the market that is currently adopting FHIR®, due to its reduced range and adhesion. Nonetheless, the early adoption of these innovations aligns with ALERT’s image, and therefore, it carries a great commercial importance for the company.

3.1.3 Point of View

The following topics focus on the third fundamental element, which refers to a point of view for the required design and other external relevant points of view that might influence that design.

3.1.3.1 Point of view for the solution's design

The infrastructure for development main purpose is to enable interoperability between the multiple systems involved, by adopting FHIR®. These communications directly involve many security issues which could compromise the whole infrastructure if an external entity intervened to retrieve or modify the information being exchanged. Considering the referred points, the overall design of the solution should include three main components:

1. Web Services;
2. Security;
3. Client.

The first one is necessary to contain all the web services that can support FHIR® requirements to successfully execute required use cases, the second one should address security measures regarding the transactions of data to be performed by these services, providing secure and reliable requests, and finally, the third one should focus on a client that can be able to request FHIR® operations, with security provisions.

3.1.3.2 Other relevant points of view

The point of view from the following parts should be considered:

1. HL7;
2. ALERT customers;
3. IHE;
4. Regulators;
5. Marketing/Sales.

Being HL7 the organization who is responsible for developing FHIR®, they provide the necessary resources, related to the multiple healthcare concepts, and have a special interest in monitoring the implementations to improve their specifications.

The ALERT customers desire to use an efficient product and configured according to their organizational needs.

IHE also has a great interest in certifying companies for correctly implementing the new IHE profiles that involve FHIR®, since their major goal is to keep improving these profiles to fulfil the healthcare business needs and create new ones for application in real-world clinical scenarios.

Some markets empower regulators to enforce specific rules on healthcare IT systems, which are often related to national legislation. To act within these markets, the products must be able to respect these limitations through required configurations.

Due to the cutting-edge nature of IHE profiles within healthcare interoperability's domain, marketing/sales have a vested interest in the product's solution due to the opportunities it affords, and even though IHE profiles clearly define the use cases to be implemented, they are uniquely positioned to evaluate the prioritization on using these profiles.

3.1.4 Assumptions

The following topics focus on the fourth fundamental element, regarding the assumptions made for multiple factors, such as the environment, the involved risks, the market, the technologies evolution, among others.

3.1.4.1 Environmental and operating conditions assumptions

It is assumed that the end users will have access to devices with a network connection, and that the health organizations will have the infrastructure to deploy the solution.

3.1.4.2 Risks acceptable to date

The information retrieved, relative to a patient, needs to be accurate since the provision of erroneous data can bring serious risks for the patient's health. Wrong information may lead to wrong decisions, for example, if the information retrieved doesn't indicate a certain allergy that the patient might possess, the physician could prescribe a medication that might contain a substance to which the patient is allergic to. Therefore, operations for exchange of medical data need to be correct, guaranteeing that the medical history is associated to the correct patients.

The current draft nature of FHIR® brings with it the risk of sweeping changes that negate the work developed during this stage. Nonetheless, previous experience with other HL7 standards and with FHIR® itself has shown that the changes tend to be more localized, which coupled with the identification of the maturity levels of the various components of the standard as opposed to a single level (with some components already being at a stable level) enables the mitigation of this risk.

Regarding market risks, there is a possibility that, even when FHIR® reaches a stable maturity level, it fails to gain traction among the various healthcare institutions. However, considering the reception that this standard has had and the possibilities it unlocks, this risk has been considered acceptable.

3.1.4.3 Market and economic environment assumed

Some markets, due to recent economic conditions, cut a lot of costs in healthcare expenditure, funneling Healthcare IT vendors to the markets that are investing in this domain. This, coupled with influencing politic plays that further reduces the list of potential clients, leads to a highly competitive environment among Healthcare IT companies.

3.1.4.4 Safety assumptions

It's assumed that if an end user logs in with the correct credentials, he actually is the legitimate holder of those credentials, although, some mitigations are to be implemented regarding token lifetimes and other OAuth security considerations (Lodderstedt et al., 2013).

It is also assumed that the servers in the healthcare institutions' physical location are secure.

3.1.4.5 Maturity level assumed for emerging technologies

Currently, FHIR® contains some resources that already possess a Level 5 maturity level, according to the criteria from FHIR Maturity Model (FMM) previously referred in section 2.3.2, while the great majority of them is still at Level 3 or below, which means that they're still being improved and can suffer major adjustments.

Either way, it is expected that the standard will continue to grow in a fast and efficient way. HL7 is continuously improving the standard, and reaching different levels of maturity for its content, which is also assisted thanks to the analysis and monitoring of the FHIR® implementations and usability in the market.

3.1.4.6 Consequences due to assumptions' discard

A change in market assumptions could alter the development strategy for this product, which leads to a redefinition of priorities. A change in the operational or security assumptions, however, could bring more far-reaching changes, as the sensitive nature of healthcare information would mandate a new analysis of the design and how it could provide certain base assurances regarding informational security.

3.1.4.7 Criteria for an optimum solution

Since most of the operations require service requests based on standards, and considering that recent standards are continuously improving, the most important aspect that an optimum solution must follow is the application of good design patterns that guarantee extensibility and serviceability, reducing the dependencies between each component that composes the solution. Plus, it must be highly configurable since the requirements change from client to client, for example, in terms of safety and criteria of correlation to identify patients between institutions. Lastly, considering the time critical nature of healthcare, performance also plays an important role in determining the suitability of a solution.

3.1.4.8 Assumptions for the availability of materials

It is assumed that the institutions that desire to use ALERT® software, can provide the necessary hardware for deployment.

3.1.4.9 Workforce skills assumed

To implement the desired services, it is assumed that the developer possesses knowledge mainly in Java and Structure Query Language (SQL). The correct application of the services also entails some understanding about health standards that are typically applied for IHE profiles.

Regarding ALERT's clients, it should be assumed that they do not possess any knowledge regarding ALERT® products' workflows, and, hence, require some initial guidance and practice with some of those functionalities, which are to be provided through several training sections. It is, however, assumed that they possess a high-level domain knowledge, allowing that guidance to be focused on the workflows' specificities.

3.1.5 Engineering Information

The following topics focus on the fifth fundamental element, which involves the information and experimental results required for the project.

3.1.5.1 Source of supporting information

The main source of information is the documentation supplied by HL7 regarding FHIR®. It contains a detailed explanation of FHIR® specifications and resources that are to be used for data exchange services, as well as the differences it contains in comparison to the previous standards.

Another important source is the documentation of IHE profiles that already apply FHIR®. The implementations will be based on the detailed information and guidance provided by these profiles, which will correlate with the functional requirements and the healthcare needs. The documentation about IHE profiles might contain references for other standards that should be used along with FHIR®, which is additional information that'll be required. Additionally, relevant books, such as *Principles of Health Interoperability HL7 and SNOMED* (Benson, 2012), and other articles are also relevant sources that provide more information about the standards used for health interoperability.

Lastly, documentation relative to the existing architecture of the system will also be useful as supporting information, to analyze possible integration approaches for the infrastructure based on FHIR® in the current system.

3.1.5.2 Proposed experiments

To ensure a viable execution of the infrastructure, some tests of the implemented services shall be conducted, such as performance and acceptance tests, to verify the accuracy of the results, the correct application of security measures and the overall performance.

Possible experiments that might prove useful are the usage of test tools to parse, analyze and validate HL7 messages, along with the execution of IHE's acceptance tests, which may contain some limitations since FHIR® is recent and still in trial. The conducted experiments are described with more detail in chapter 6.

3.1.5.3 Legacy solutions and problems for study

The solutions applied for the previous standards should be studied. Since the architecture for the services and functionalities implemented in previous solutions are fairly similar to the desired solution, the structure and implementation approaches should be taken into consideration. It also serves as a guide for the analysis of previous problems and how they were solved, or for the enrichment of the solution's overall performance.

Another major aspect that can aid the solution's construction is the study of framework features that shall assist on the implementation for FHIR services and overall specification.

3.1.5.4 Available information sufficiency

It's considered that the existing information is more than enough to apply the new standard, plus, the documentation is being updated progressively to provide better solutions and improved content for the existing resources. Conversely, the updates on the standard's version might require the analysis of new information and other design approaches to meet the requirements.

3.1.5.5 Shop floor's insights and experiences

The application of FHIR® or IHE profiles in real situations, might shed light on edge-case limitations that emerge. However, due to interoperability's nature, these limitations would then be sent to IHE or HL7 as appropriate, to, if approved, update the standard before changing the solution.

3.1.6 Concepts

The following topics focus on the sixth fundamental element, regarding the major concepts involved in the project, in specific the ones from the technologies and the problem.

3.1.6.1 Concepts applicable for the problem

Healthcare interoperability is the main concept applicable throughout the whole project, which is the system's capability to exchange information with others without the need for either one of them to interpret the information exchanged, associating it to its own terms (HIMSS, 2013).

Therefore, the analysis, design and implementation of the standard's specifications involve the assurance of interoperability, which enable the exchange of accurate information.

3.1.6.2 Available technologies

It'll be appropriate to use programming languages that better suit, not only the development of the implementation needs for FHIR® services, but also the integration with the current interoperability solution, which is already developed with specific programming languages. Moreover, technologies that follow RESTful approaches are the most indicated to be applied for this project, since FHIR® adopts a RESTful approach with resources to represent the specific healthcare entities, unlike the previous HL7 standards and solutions.

3.1.6.3 Emerging technologies

At this moment, only new resources or specifications from FHIR® that might appear in the future can be considered for following upgrades in the system, and possibly, other technologies that might integrate FHIR® at a high maturity level.

3.1.7 Inferences

The following topics focus on the seventh fundamental element, regarding candidate solutions and the final solution's practicability and affordability.

3.1.7.1 Viable candidate solutions

Each one of the candidate solutions need to mandatorily adopt FHIR® and be easily integrated into the current interoperability infrastructure. The candidate solutions must contain a client that can perform requests for FHIR® services. Therefore, to perform these operations, the solutions must contain a service layer with the necessary REST services to correctly answer to the incoming requests, considering that these requests must perform secure communications to avoid security leaks and corruption of sensitive information.

3.1.7.2 Rejected candidate solutions

The other candidate solutions were rejected due to the FHIR® specifications, which demands the use of REST services to take advantage of its standardized resources. Additionally, the design of the infrastructure needs to respect ALERT's requirements for the desired solution.

3.1.7.3 Solution's practicability and affordability

The solution will be practicable and affordable since it will be integrated in ALERT's current interoperability solution and deployed for future clients that desire to exchange data with other institutions following FHIR®.

3.1.8 Implications

The following topics focus on the eighth fundamental element, to analyze the major implications involved with the supporting data, the technology, the market, future design decisions, product failures, and social reaction.

3.1.8.1 Major implications of gathered data

Regarding FHIR® specifications, these are still suitable to change, which means that the information about the standard needs to be consulted periodically to keep the system and services updated, according to the standard's criteria.

3.1.8.2 Technology's market implications

This technology favors a fine level of granularity on the exchanged information, which lends itself to a greater degree of automated integration of health information among the intervening healthcare providers. Furthermore, the lightweight RESTful architecture opens the market for a slew of mobile-based opportunities, which coupled with the granularity of the information could lead to an influx of mobile apps for both patients and medical personnel.

3.1.8.3 Technology's implications due to delayed maturity

Since previous standards exist, and are already adopted at a worldwide level, if the new standard fails to mature its content, then the previous standards will still be applied to perform some of the major interoperability operations. As for the technologies on which FHIR® depends, these (like REST) are already mature, so there is no risk on that regard.

3.1.8.4 Importance of after-market sustainability

It is important to sustain the interoperability system and monitor its behavior frequently. The product will always try to meet the client's requirements, this way it is important, especially in the healthcare business market, to keep track of the client's necessities, health standards and other specifications that might affect the business, such as health legislations.

High level of supportability and reliability are also key factors to meet the clients' needs, since system downtime can bring huge consequences. Although the exchange of information represents the least critical factor of risk during downtime, compared to the remaining functionalities, which keep track of the patients' management of medication and treatments in the respective institution, it still occupies a great role in the system's overall operation.

3.1.8.5 Future design evolution and upgrade

The system should be capable of extending its services for future upgrades and innovations that FHIR® can bring, therefore, the design of the infrastructure should consider these extensibility factors, with the evolution of FHIR® and the IHE profiles marking the natural path for evolving the design.

3.1.8.6 Implications of product failure

The implications of both software and hardware failure should be considered. Regarding the software, the system is prepared for having multiple instances running, to guarantee high availability in case one of the servers fail. Typically, the total number of instances running is two, but this quantity can vary from client to client, depending on their requirements. Hardware failure must be resolved by simply exchanging the machine components that miscarried.

Product failure carries extremely severe consequences in health business areas, since it can disturb whole hospitals' operations, affecting the patients that are currently hospitalized. Although the institutions are the ones responsible for providing the required hardware, ALERT takes special attention in monitoring the systems that integrate ALERT® software, to quickly respond to system failures and help the organizations during the recovery phase.

3.1.8.7 Consequences of design features' changes

New versions of FHIR® specifications for REST resources might affect the design of the solution, regarding the implemented services and the data structure for transportation, including the libraries/tools based on FHIR® that might also affect the design.

3.1.8.8 Insensitive design features to other changes

The FHIR® resources specification and the database structure are insensitive to any other changes of design in the system.

3.1.8.9 By-products' potential benefits

By-products could offer any kind of functionalities that would benefit from lightweight access to fine granularity patient's health information, most likely in the form of healthcare mobile apps (for example, for monitoring medication or allergies).

3.1.8.10 Social reaction and change management issues

Considering the nature of the solution as the implementation of an international standard, these issues end up being addressed only indirectly, by keeping up with updates from HL7 and IHE (who do deal with these issues directly by updating their standards and profiles when needed).

3.2 Stakeholders

Considering ALERT® product and the scenarios where the solutions are used, physicians, nurses, administrative clerks and other healthcare technicians typically use ALERT® PFH to manage the major tasks in a hospital environment, such as patient's room allocation, consult appointments, clinical prescription, diagnoses and so on. The same can be applied for smaller institutions such as private clinics and primary care centers, where the involved stakeholders perform similar tasks using products such as ALERT® PRIMARY CARE or ALERT® PRIVATE PRACTICE, specific to the institution's requirements and specialties. Patients can also track their medical history, using applications such as MyALERT®, which can provide information retrieved from the encounters with physicians and other information that was shared by them. Another stakeholder that is not directly related to the healthcare institutions' operations, but also represent a major role for the system, is the team from ALERT Network Operations Center (NOC) which is in charge for monitoring ALERT® products' operation to support and apply preventive measures in case of a problem occurrence, by taking advantage of the information provided by the products, for instance, the software technical issues and the product's availability.

Independently of the system used in the market, the tasks performed by some of these stakeholders usually involve the creation of data that can be shared and visualized throughout institutions and systems. This being said, it can be identified the following stakeholders involved:

- Physician;
- Nurse;
- Patient;
- Administrative Clerk;
- Support personnel.

3.3 Requirements

Considering the identified stakeholders, the applications they use, and the IHE profiles previously described in section 0, it's important to think and analyze the requirements for the HIE infrastructure that will apply FHIR® specifications, to fully visualize the technical features for each functionalities and quality aspects that might affect the whole solution. This analysis is made using FURPS+, the classification system designated to retrieve functional and nonfunctional requirements (Grady, 1992).

3.3.1 Functionality

In first place, it was identified the product's main functions, which are related to the HIE infrastructure features with FHIR®. These were defined based on the IHE profiles of interest that involve FHIR®, and the functionalities that ALERT customers currently use with previous HL7 standards, which also follow existing IHE profiles.

This being said, it was identified the following functionalities:

- Exchange health documents (IHE's MHD profile);
- Correlate patient's identifiers (IHE's PIXm profile);
- Exchange patient's demographic information (IHE's PDQm profile);
- Audit activities (IHE's ATNA profile);
- Exchange granular information (IHE's mXDE profile).

The first function (exchange health documents) role is to manage document transactions made by document consumers, which can be medical devices, patient applications (Personal Health Records), or another type of system. With this function, HIE will be capable of receiving requests that might involve submission, search and retrieval of health documents that are, or will be, stored in a document repository.

The second function (correlate patient identifiers) provides the means to link a patient's identity from a different domain, for instance a hospital, with the patient's identity from a requesting application, such as a mobile device, using known information about the patient as its identifier. HIE will manage these transactions by receiving the incoming cross-reference requests, finding and linking the patient's data from both systems. After cross-referencing the patient's data, the requesting application will be able to demand health-related information about the patient using the identifier provided by the external domain.

The third function (exchange patient demographic information) is simply necessary to retrieve patients' demographic data, such as its name, date of birth, address, gender, among other information.

The fourth function (audit activities) is mandatory to track inappropriate behavior, and to detect unusual activities performed in the system involving sensitive and protected health information. The implementation of this functionality allows HIE to record audit events to verify the actions' legitimacy, according to the domain's security policy.

The fifth function (exchange granular information) enables HIE to get specific data elements from shared documents, without the need to retrieve the whole document. Usually, to perform some operations, it is required to consult a patient's health history, which can be made through clinical documents. However, these documents can contain other information that is not relevant to assist on the current operation and considering that the retrieval of these documents is a heavy operation, the possibility to retrieve only a portion of information can improve the system's performance.

According to what was described previously, Figure 12 presents a tree diagram which sums up all the defined HIE's functions, including each function's branches with different possible actions, that can be performed by the product depending on the incoming requested services.

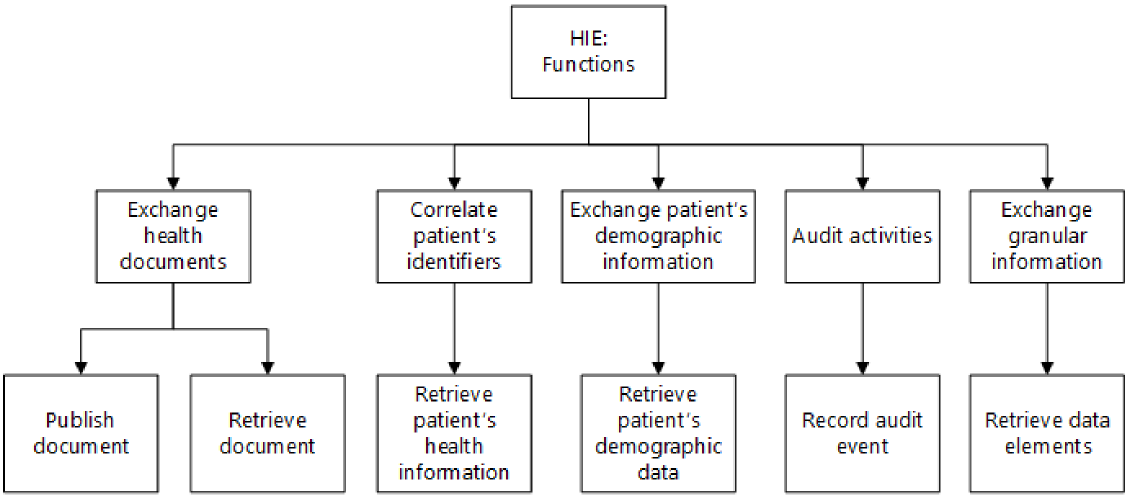

Figure 12 – Tree Diagram HIE

3.3.2 Usability

The infrastructure mainly possesses web services for other software products, therefore, in terms of usability, it won't affect neither the system involved, nor the user interfaces that involve interaction.

3.3.3 Reliability

The system should be prepared to have a minimum of two or more instances running, to guarantee high availability, which can vary depending on the clients' needs. It should also guarantee that the information in transaction is not corrupted and maintains its integrity throughout the exchange process, to present correct data for the end user.

3.3.4 Performance

The operations performed for the previous standards were sometimes slow and heavy due to the size of health documents being transferred, or the complex search of patient information. Usually, the response time for operations regarding auditing are less than one second, while some complex operations such as the retrieval of document sets can go up to thirty seconds.

With FHIR® it is expected to obtain an equal or better performance regarding these operations, so, it is acceptable a response time not superior to the existing services response time.

Regarding memory consumption, the limit of physic memory will depend on each client, since the health documents are stored in their establishment. Even so, the current separation between metadata and document files offers an efficient method for memory management.

3.3.5 Supportability

The system should be (i) highly configurable in terms of access, transactions, security measures, timeout configurations and criteria for patient's data correlation between institutions, (ii) testable, containing a set of tests to evaluate and guarantee the correct functioning of FHIR® services, (iii) easily extensible, to provide the means to add new functionalities and services without affecting or modifying the existing ones, (iv) and easily installable with the assistance of portable technologies.

Additionally, the system shall be integrated in the existing ALERT HIE solution and must be able to send information to ALERT's NOC, for monitoring and assistance purposes, to quickly act over any problem identified in the system. ALERT's system for HIE also possesses its own functionalities for monitoring, regarding logs, memory, tests and others, which should also be used to monitor the new features.

3.3.6 Design Constraints

The solution's design is partially constraint due to the standards' own design regarding the resources that need to follow certain rules and structures, which, for example, might affect the database's model design. Moreover, part of the architectural design might be influenced by the ALERT HIE's components and the IHE profiles, which define a set of recommendations and approaches to implement certain use cases (previously defined in section 0) using a set of standards, affecting the design approach for some of the FHIR features.

3.3.7 Implementation Requirements

It is required to develop the new features in Java, respecting the design from FHIR® specifications which adopt a RESTful paradigm, and IHE's requirements for the profiles of interest in this project.

3.3.8 Interface Requirements

The system must contain a generic interface based on FHIR®, considering initially a low frequency of requests due to the standard's early maturity.

3.3.9 Physical Requirements

The client’s establishments provide the necessary hardware to deploy the product, which requires WebLogic, Oracle and Java.

3.3.10 Defined Requirements

Table 4 sums up the necessary requirements for implementation, considering all the aspects mentioned along this section and the identified stakeholders.

Table 4 - Requirements

Identifier	Description	Actor
REQ1	Correlate patient’s identifiers	Client Application
REQ2	Exchange patient’s demographic information	Client Application
REQ3	Exchange health documents	Client Application
REQ4	Exchange granular information	Client Application
REQ5	Audit activities	Server Application
REQ6	Secure access to services	-

3.4 Value Analysis

In this section it is discussed the Value Analysis (VA) for the product, by following a process composed with a variety of techniques that aid on the identification of the main functions that truly bring value to the customer (Rich and Holweg, 2000).

The VA process is composed by a total of five stages, as represented in Figure 13: Orientation, Functional Analysis (which also includes Functional Identification), Creative Alternatives, Analysis and Evaluation, and Implementation. Considering the project’s problem, (1) the Orientation phase discusses at first hand the product to be studied during the VA process, (2) the Functional Analysis phase focuses on identifying the product’s functionalities that bring value to the customer and analyzing these functionalities by level of importance, (3) the Creative Alternatives phase identifies the most suitable frameworks to be applied for the features development, (4) the Analysis and Evaluation phase compares these alternatives based on a set of criteria to conclude which one is the most adequate for application, and finally, (5) the Implementation phase briefly describes the product’s implementation based on the reached conclusions throughout this process. Throughout these phases, a set of techniques are applied to support the decisions and conclusions regarding the product’s characteristics that bring value for the customer.

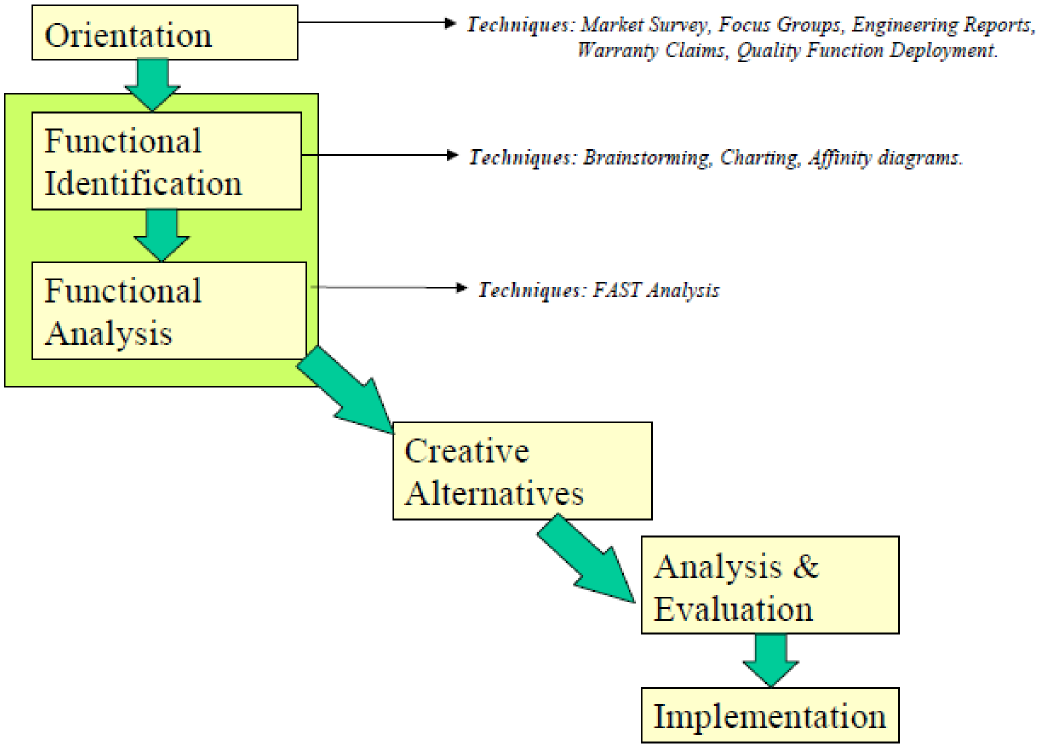

Figure 13 – Value Analysis Process (Rich and Holweg, 2000, p. 12)

3.4.1 Orientation

For the first stage of the VA process, it was applied the Fuzzy Front End (FFE) model which is the first stage of the innovation process (Koen et al., 2004, p. 6), shown in Figure 14.

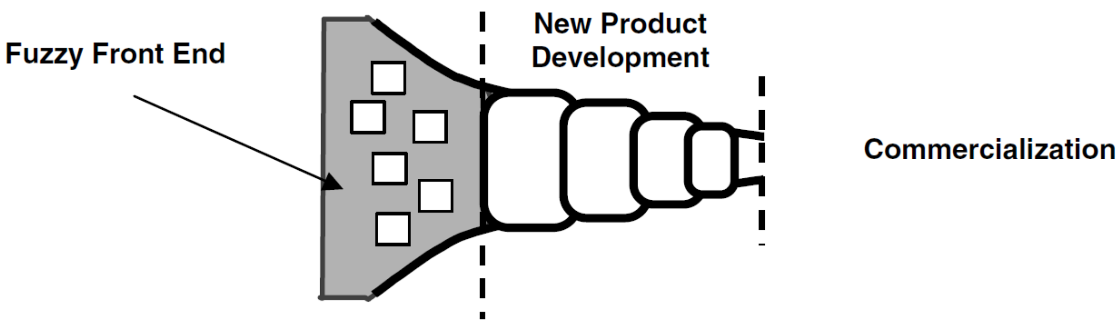

Figure 14 – Innovation Process (Koen et al., 2004, p. 6)

In order to describe the Fuzzy Front End activities, it was applied the New Concept Development (NCD) model, shown in Figure 15, which provides a useful terminology composed by an engine, five front end elements and influencing factors (Koen et al., 2004, p. 8).

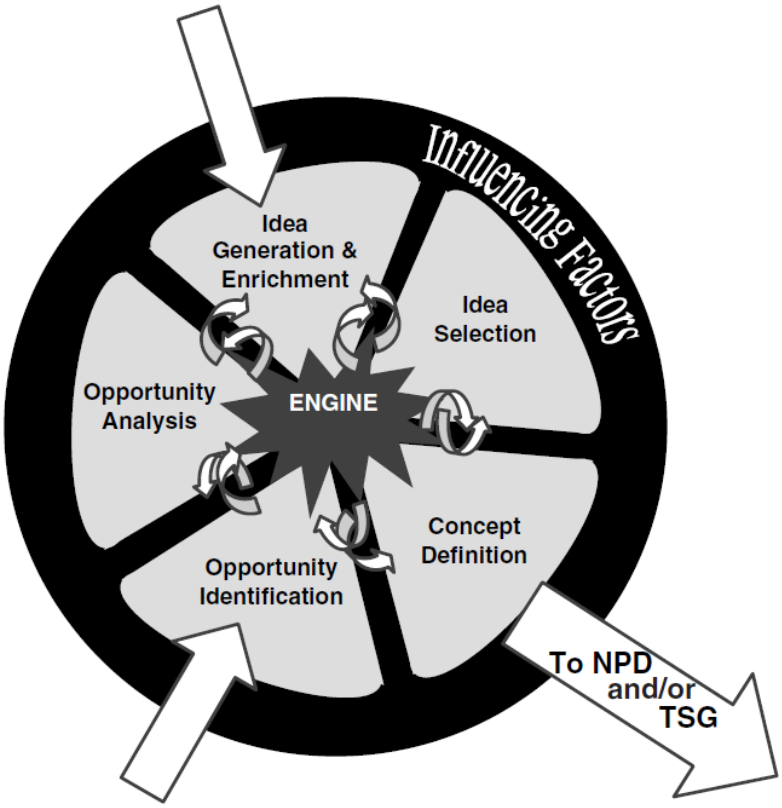

Figure 15 – New Concept Development Model (Koen et al., 2004, p. 8)

3.4.1.1 Influencing Factors

Regarding the current project, the major influencing factors are (1) ALERT's organizational capabilities, (2) their customers and competitors, (3) the external influences that affect the healthcare business area and (4) the technologies in the market that can impact the healthcare IT systems.

As a healthcare IT company, ALERT continuously gathers information from health conferences, news, organizations for healthcare standards' development, and other health related sources, to deal accordingly with upcoming changes, problems or opportunities that can bring a great impact for the healthcare society. The reliability of the company depends on its capacity to adapt to the healthcare market to fulfill their clients' needs and offer products that respect the healthcare organizations' standards.

Other major influencing factors are ALERT's clients, since the company's growth depends on their clients' adhesion and satisfaction. Currently the company possesses numerous clients at an international level, such as hospitals, private clinics, and other health institutions. Besides worrying about their clients, ALERT also worries about their competition. In the health business, owning an IT system that can undertake all the tasks for the correct operation of a health institution is essential, since it facilitates and promotes the professionals daily work, whom are responsible for their patient's health and well-being. However, as was referred in section 3.1, since not all markets are willing to invest in these systems due to economic reasons and having other companies that also desire to sell their IT systems, the market environment between the healthcare IT companies is highly competitive.

Regarding the external influences, modifications in the legislation for the health business, or adjustments to health standards can impact ALERT's products and consequently affect their clients. These factors are the major reason why ALERT keeps a high monitoring of external events, since the changes need to be dealt with as soon as possible.

Finally, the technologies are also another significant influencing factor for the company. The arrival of new technologies or the modification of existing ones that are used in ALERT's system, can lead the company to establish new design decisions for their product since it might directly affect the product functionalities.

3.4.1.2 Engine

Concerning the engine, ALERT seeks to create products of quality and excellence, considering its business area deals with serious situations which are directly related to life itself. Since the beginning, ALERT established a fixed purpose, which is to respect life in all its forms and prolong it by improving the healthcare society quality, with products that can assist health professionals with their daily tasks (ALERT Life Sciences Computing, 2017f). ALERT is organized in multiple teams which work together to guarantee that the product is correctly developed and improved to best satisfy the clients' requirements, along with the team leaders that establish the goals according to ALERT's main purpose.

3.4.1.3 Opportunity Identification

ALERT already adopts several health standards used in the worldwide market, previously described in section 2.2, such as, ICD-10 and ICD-9 for the definition of existing diseases, HL7 for the exchange of clinical information, LOINC® for the provision of an universal code system, among others (ALERT Life Sciences Computing, 2017g). As mentioned before, ALERT takes special attention to the market trends and technologies that might affect their products and clients. This allowed the company to recognize the appearance of HL7 latest standard, FHIR®, which was the crux for the company to identify a new opportunity for improving their product, ALERT® HIE.

3.4.1.4 Opportunity Analysis

As previously mentioned in the importance of time-to-market (section 3.1.2.5), HL7 is still updating FHIR® to reach higher levels of maturity, even so, some of its content progressively improved throughout the years and reached a stable maturity level for the implementation of its services. The company's adoption of this standard would significantly improve, not only its presence in the healthcare IT market as a company that offers interoperability products with the latest standards, but also their products' capabilities thanks to the new advantages and improvements offered by FHIR®.

To further study this opportunity, it should be analyzed the value that the product with the new standard brings for the customer. Table 5 shows a list of value based drivers related to the perceived value for the customer (Woodall, 2003), which, according to Zeithaml (1988, p. 14), "is the consumer's overall assessment of the utility of a product based on perceptions of what is received and what is given".

Table 5 - Benefits and Sacrifices (Woodall, 2003)

BENEFITS		SACRIFICES
Attributes	**Outcomes**	
Perceived quality	Functional benefits	Price
Product quality	Utility	Market price
Quality	Use function	Monetary costs
Service quality	Aesthetic function	Financial
Technical quality	Operational benefits	Costs
Functional quality	Economy	Costs of use
Performance quality	Logistical benefits	Perceived costs
Service performance	Product benefits	Search costs
Service	Strategic benefits	Acquisition costs
Service support	Financial benefits	Opportunity costs
Special service aspects	Results for the customer	Delivery and installation costs
Additional services	Social benefits	Costs of repair
Core solution	Security	Training and maintenance costs
Customisation	Convenience	Non-monetary costs
Reliability	Enjoyment	Non-financial costs
Product characteristics	Appreciation from users	Relationship costs
Product attributes	Knowledge, humour	Psychological costs
Features	Self-expression	Time
Performance	Personal benefits	Human energy
	Association with social groups	Effort
	Affective arousal	

These values were taken into account, to identify a set of benefits that the product will bring to the customer, and the related sacrifices that it implies, which are defined in Table 6, considering the product, the service, and the relationship with its supplier.

Regarding the product value, the major benefits for the customer will focus on the product's quality and customization. The adoption of FHIR® will give the product new and enriched services and an expected improved performance, which in turn, will increase the product's overall quality for a better customer experience.

Table 6 – Perceived Value

	Product	Service	Relationship
Benefit	• Services quality improvement; • Better experience for the customer; • Performance improvement; • Custom features' configuration; • Security features.	• Full time support for emergencies; • Assistance on product deployment and configuration.	• Mature and trustworthy; • Continuous supervision and assistance.
Sacrifice	• Acquisition costs.	• Maintenance costs.	• Setting up the new software specifications and configurations.

In addition, due to the diverse healthcare organizations' natures, requirements, procedures and restrictions in terms of privacy, the product's configurations will always be adapted to fulfil the customer's needs and requests. In terms of sacrifices, the initial acquisition of the product comes with a monetary cost, which needs to be paid by the customer for the product to be deployed and functional in the facility.

Concerning the service value, an important factor for healthcare organizations is to possess a highly available system with the minimum downtime possible, and due to this, ALERT takes special attention in offering a full-time support for their clients, to quickly respond to any kind of urgency, and keep the software updated. Additionally, during the deployment process, ALERT also aids during the products' software configurations in the client's environment for a better transition phase. The continuous maintenance support from ALERT also comes with a monetary cost.

Finally, for the relationship with the organization, one of the major benefits that bring value for the customer is the organization's maturity and continuous supervision of their client's products, which establishes a trustworthy relationship. As sacrifices, it'll be required some time and effort from the customer to set up the product and the necessary configurations in their systems.

3.4.1.5 Idea

Therefore, to preserve ALERT® HIE interoperability capabilities, ALERT decided to adopt FHIR® to keep up with the market needs and clients that want to share information inside the organization, or with other external systems. Additionally, the services offered by the standard open new doors to simpler implementations for the distribution and access of clinical information, which in turn improves the solutions' user-friendliness for the customers.

3.4.1.6 Concept Definition

Considering all the aspects previously mentioned and the maturity limitations of some of the standard's content, the concept of the project can be defined as the development of an interoperability infrastructure that adopts FHIR® and its new services to exchange clinical information, while maintaining the existing functionalities already implemented in ALERT® HIE with previous HL7 standards.

3.4.1.7 Quality Function Deployment

Before advancing to the Functional Analysis phase, it will be applied the Quality Function Deployment (QFD) technique (Warwick Manufacturing Group, 2007) to identify the quality requirements with more relevance to the client, in conjunction with the products' technical features. Figure 16 presents the House of Quality which demonstrates the relationship values between the customer requirements and the technical characteristics.

It was identified the following customer quality requirements ("Whats"):

- Good performance;
- Reliable information;
- Respects the institution's criteria;
- Safe exchange of information;
- Actions monitoring.

Along with the following technical characteristics ("Hows"):

- Authentication;
- Authorization;
- Transaction time;
- System configurations;
- Data integrity;
- Operation logs;
- FHIR standard.

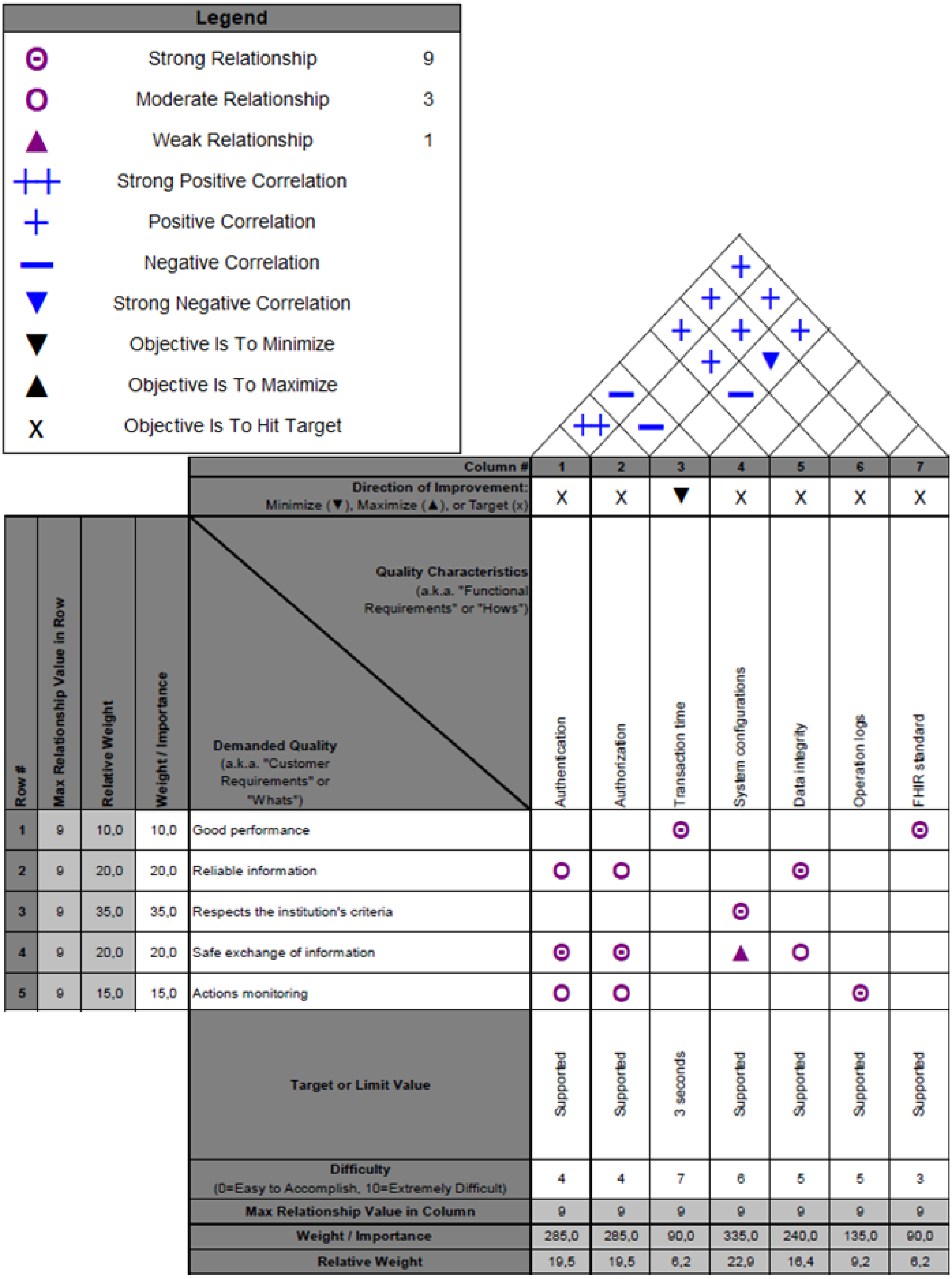

Figure 16 – House of Quality

Explaining with more detail the house of quality, and the evaluation made according to Figure 16's legend, in terms of performance the most important technical aspects to consider in the product's design are the transaction time and the FHIR standard, which have a great impact in the product's services performance. Regarding the reliability of information exchanged, authentication, authorization and data integrity measures represent a great role for this requirement, to guarantee that sensitive data is not modified during the transaction, reaching its destination with credible information. System configurations are required to satisfy the institution’s criteria, which can influence some of the product’s features implementation. The safety of information exchange can also be partially affected by system configurations, since it might require new security considerations, along with authentication, authorization and data integrity measures that will considerably improve this requirement, by restricting the access to information exchange operations. Finally, for actions monitoring, operation logs will be the main feature to address this requirement, by taking advantage of the authentication and authorization features, to identify the authors that performed the recorded operations.

Regarding the relationship between the product’s technical characteristics, the authentication has a strong relationship with the authorization, since the combination of both features promote the system’s overall security. Additionally, this feature also improves the security and promotes the implementation of features such as data integrity, operation logs and FHIR standard. The same applies for authentication which also improves the security of these features. The total transaction time however, might grow with the addition of security features, since more verifications and security measures will be applied before performing the actual requested operation. Nonetheless, it is expected that FHIR’s specifications improve the performance of the system, reducing the transaction time.

3.4.2 Functional Analysis

The next step in the VA process is the identification and analysis of the major product functions that bring value for the customer. The products’ functionalities were already identified and described in section 3.3.1, due to the fact that these are required features that need to be in conformance with the standard’s restrictions and some IHE profiles, described in section 0, that define the development approaches. Even so, the identified functions are in accordance with the value identified for the customer and the quality aspects for the product.

Next, to identify the functions’ priority and importance, the following step will pass by ranking each function, adopting a pairwise comparison, which is presented in Figure 17. This method ranks the functions using three scores of importance, namely minor (1 point), medium (2 points) and major (3 points), which scores will then be added based on the points received from each comparison. The score defined for each function will define its level of importance compared to the others (Rich and Holweg, 2000).

Pairwise Comparison: HIE

Functions:	**A**	Exchange health documents
	B	Correlate patient's identifiers
	C	Exchange patient's demographic information
	D	Audit activities
	E	Exchange granular information

A vs B	**B: 2**	**Score:**	
A vs C	**C: 2**	A:	3 points
A vs D	**D: 3**	B:	6 points
A vs E	**A: 3**	C:	5 points
B vs C	**B: 1**	D:	10 points
B vs D	**D: 2**	E:	0 points
B vs E	**B: 3**		
C vs D	**D: 2**		
C vs E	**C: 3**		
D vs E	**D: 3**		

Key to Points:	**1**	Little more important / better
	2	Significantly more important / better
	3	A lot more important

Figure 17 – Pairwise Comparison HIE

As shown in Figure 17, the scores indicate that the most important function, from all the ones identified for the product, is the *audit activities* with a total of 10 points. The following functions with minor levels of importance were the *correlate patient's identifiers* function (6 points), the *exchange patient's demographic information* function (5 points), the *exchange health documents* function (3 points), and finally, the least important function, *exchange granular information* (0 points).

3.4.3 Creative Alternatives

The next step in the VA process is to select possible alternatives for implementation of the established functions. Considering the nature of the technologies involved with the standard in study, it was selected the following frameworks that presented suitable capabilities for the product's development:

- JAX-RS;
- HAPI FHIR;
- Smile CDR;
- Iguana 6.

Each framework contains a set of features that would considerably help with the task to develop the identified functionalities, considering the value for the customer and his quality requirements. The technical aspects for these frameworks are described with more detail in section 2.6.

3.4.4 Analysis and Evaluation

For this next stage, it was applied the Analytic Hierarchy Process (AHP) method to evaluate the different alternatives of implementation previously described, which is a process composed by multiple phases that provide the means to compare the various qualitative and quantitative alternatives' criteria, using a hierarchical division for the problem (Saaty, 2008). The criteria used to compare the alternatives was based on framework's capabilities that would assist on reaching the quality requirements with value for the customer, namely:

- Security;
- Ease of implementation;
- Supportability.

Security is an important aspect to consider since the frameworks can already provide implemented features to promote the services' overall security, ease of implementation focuses on the frameworks' capability to be applied for complex business processes involving the execution of RESTful services based on the standard's specification, and finally, supportability is related to the framework's range of support for FHIR®.

3.4.4.1 AHP Evaluation

Using the AHP method, it will be proved which framework alternative is the most appropriate for use based on the respective criteria and previous analysis. First, to apply the AHP method it should be constructed the decision hierarchical tree to define the problem, the criteria and the alternatives. Figure 18 shows the decision tree, where the problem involved is the choice for the framework to be used to implement the required functions for the HIE infrastructure, the criteria is based on the security that the framework can offer, the ease of implementation for the necessary services and the supportability for multiple aspects that involve the standard's application, and finally the selected framework alternatives.

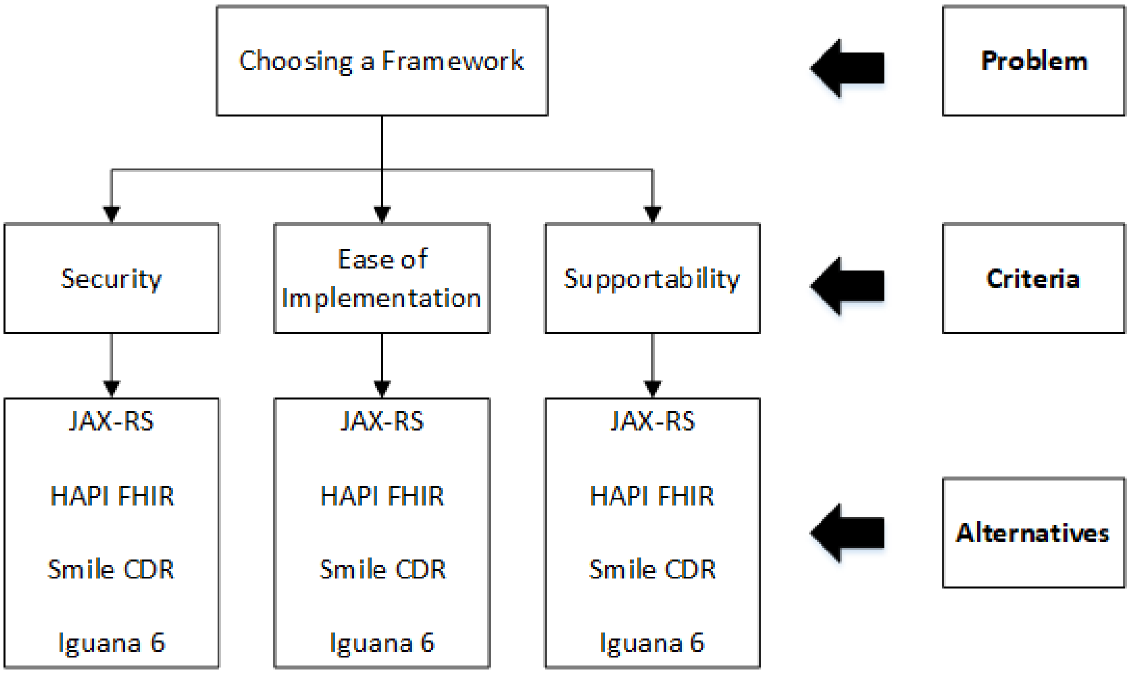

Figure 18 – Decision Hierarchical Tree

The next step involves the comparison between the defined criteria using Saaty's fundamental scale of absolute numbers (Saaty, 2008, p. 86) presented in Figure 19.

Intensity of Importance	*Definition*	*Explanation*
1	Equal Importance	Two activities contribute equally to the objective
2	Weak or slight	
3	Moderate importance	Experience and judgement slightly favour one activity over another
4	Moderate plus	
5	Strong importance	Experience and judgement strongly favour one activity over another
6	Strong plus	
7	Very strong or demonstrated importance	An activity is favoured very strongly over another; its dominance demonstrated in practice
8	Very, very strong	
9	Extreme importance	The evidence favouring one activity over another is of the highest possible order of affirmation
Reciprocals of above	If activity *i* has one of the above non-zero numbers assigned to it when compared with activity *j*, then *j* has the reciprocal value when compared with *i*	A reasonable assumption
1.1–1.9	If the activities are very close	May be difficult to assign the best value but when compared with other contrasting activities the size of the small numbers would not be too noticeable, yet they can still indicate the relative importance of the activities.

Figure 19 – Saaty's Fundamental Scale of Absolute Numbers (Saaty, 2008, p. 86)

Table 7 presents the defined scale between the criteria. It was considered that *security* overall has a greater importance over *ease of implementation* (Intensity of importance equal to 3) and *supportability* (Intensity of importance equal to 2), although the difference of importance with *supportability* is lower since both *security* and *supportability* will carry more value for the product features development.

Table 7 – Criteria Comparison

	Security	**Ease of Implementation**	**Supportability**
Security	1	3	2
Ease of Implementation	$1/3$	1	$1/2$
Supportability	$1/2$	2	1

From this, the next stage passes through the normalization of Table 7, presented in Table 8 and Table 9, including the calculation of the priorities (Table 10), which will define the priorities vector to be used in the following steps.

Table 8 – Criteria First Step of Normalization

Comparison Matrix for Second Level Criteria			
	Security	Ease of Implementation	Supportability
Security	1	3	2
Ease of Implementation	$1/3$	1	$1/2$
Supportability	$1/2$	2	1
SUM	$11/6$	6	$7/2$

Table 9 – Criteria Second Step of Normalization

Normalized Matrix for Second Level Criteria			
	Security	Ease of Implementation	Supportability
Security	$6/11$	$1/2$	$4/7$
Ease of Implementation	$2/11$	$1/6$	$1/7$
Supportability	$3/11$	$1/3$	$2/7$

Table 10 – Criteria Priorities Calculation

Normalized Matrix for Second Level Criteria				
	Security	Ease of Implementation	Supportability	Priorities
Security	$6/11$	$1/2$	$4/7$	0.54
Ease of Implementation	$2/11$	$1/6$	$1/7$	0.16
Supportability	$3/11$	$1/3$	$2/7$	0.30

The results obtained for the priorities indicate that *security* is the criteria with the highest level of importance (54%), followed by *supportability* (30%) and *ease of implementation* at last (16%).

After obtaining the priorities vector, the next phase passes through the calculation of the Consistency Ratio (CR). The formula to calculate CR is $CR = CI / RI$, where the Consistency Index (CI) is calculated with the formula $CI = (\lambda_{max} - n)/(n-1)$. The Random Index (RI) value corresponds to one of the values presented in Table 11, defined by the Oak Ridge National Laboratory, depending on the criteria's total number, which in this case is a total of 3. Before calculating the CI it's necessary to determine the value of λ_{max}, using the formula $Ax = \lambda_{max}\, x$, considering x the Priorities Vector ($x = \begin{bmatrix} 0.54 \\ 0.16 \\ 0.30 \end{bmatrix}$):

1. $\begin{bmatrix} 1 & 3 & 2 \\ 0.33 & 1 & 0.5 \\ 0.5 & 2 & 1 \end{bmatrix} \times \begin{bmatrix} 0.54 \\ 0.16 \\ 0.30 \end{bmatrix} \cong \lambda_{max} \begin{bmatrix} 0.54 \\ 0.16 \\ 0.30 \end{bmatrix}$
2. $\begin{bmatrix} 1.62 \\ 0.49 \\ 0.89 \end{bmatrix} \cong \lambda_{max} \begin{bmatrix} 0.54 \\ 0.16 \\ 0.30 \end{bmatrix}$
3. $\lambda_{max} = 3.0097$

Table 11 - RI values defined by the Oak Ridge National Laboratory (Nicola, 2017)

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0.00	0.00	0.58	0.90	1.12	1.24	1.32	1.41	1.45	1.49	1.51	1.48	1.56	1.57	1.59

With this, the value of CI can be determined:

- $CI = (3.0097 - 3)/(3 - 1) = 0.00485$

Having the value from CI, we can finally calculate CR's value:

- $CR = 0.00485/0.58 = 0.008$

If the CR value is superior to 0.1, it's considered that the results do not present consistent values, since the judgements were too close for the comfort of randomness.

- Since $0.008 < 0.1$, we can conclude that the priorities' values are consistent.

The following phase in the AHP method is to calculate the priorities vector for each criterion, considering each one of the alternatives and following the same steps made from Table 7 to Table 10. First it will be calculated the priorities vector for the security criterion.

Table 12 – Security Criterion's Alternatives Comparison

Security	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	1	$1/2$	$1/3$	1
HAPI FHIR	2	1	$1/2$	2
Smile CDR	3	2	1	3
Iguana 6	1	$1/2$	$1/3$	1

Table 13 – Security Criterion's Alternatives First Step of Normalization

Security	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	1	$1/2$	$1/3$	1
HAPI FHIR	2	1	$1/2$	2
Smile CDR	3	2	1	3
Iguana 6	1	$1/2$	$1/3$	1
SUM	7	4	$13/6$	7

Table 14 – Security Criterion's Alternatives Second Step of Normalization

Security	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	$1/7$	$1/8$	$2/13$	$1/7$
HAPI FHIR	$2/7$	$1/4$	$3/13$	$2/7$
Smile CDR	$3/7$	$1/2$	$6/13$	$3/7$
Iguana 6	$1/7$	$1/8$	$2/13$	$1/7$

Table 15 – Security Criterion's Alternatives Priorities Calculation

Security	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6	Priorities
JAX-RS	$1/7$	$1/8$	$2/13$	$1/7$	0.14
HAPI FHIR	$2/7$	$1/4$	$3/13$	$2/7$	0.26
Smile CDR	$3/7$	$1/2$	$6/13$	$3/7$	0.46
Iguana 6	$1/7$	$1/8$	$2/13$	$1/7$	0.14

For the security criterion, it was obtained the following priorities vector: $\begin{bmatrix} 0.14 \\ 0.26 \\ 0.46 \\ 0.14 \end{bmatrix}$.

The following calculations regard the ease of implementation criterion.

Table 16 – Ease of Implementation Criterion's Alternatives Comparison

Ease of Implementation	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	1	$^{1}/_{4}$	5	4
HAPI FHIR	4	1	7	6
Smile CDR	$^{1}/_{5}$	$^{1}/_{7}$	1	$^{1}/_{3}$
Iguana 6	$^{1}/_{4}$	$^{1}/_{6}$	3	1

Table 17 – Ease of Implementation Criterion's Alternatives First Step of Normalization

Ease of Implementation	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	1	$^{1}/_{4}$	5	4
HAPI FHIR	4	1	7	6
Smile CDR	$^{1}/_{5}$	$^{1}/_{7}$	1	$^{1}/_{3}$
Iguana 6	$^{1}/_{4}$	$^{1}/_{6}$	3	1
SUM	$^{109}/_{20}$	$^{131}/_{84}$	16	$^{34}/_{3}$

Table 18 – Ease of Implementation Criterion's Alternatives Second Step of Normalization

Ease of Implementation	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	$^{20}/_{109}$	$^{21}/_{131}$	$^{5}/_{16}$	$^{6}/_{17}$
HAPI FHIR	$^{80}/_{109}$	$^{84}/_{131}$	$^{7}/_{16}$	$^{9}/_{17}$
Smile CDR	$^{4}/_{109}$	$^{12}/_{131}$	$^{1}/_{16}$	$^{1}/_{34}$
Iguana 6	$^{5}/_{104}$	$^{14}/_{131}$	$^{3}/_{16}$	$^{3}/_{34}$

Table 19 – Ease of Implementation Criterion's Alternatives Priorities Calculation

Ease of Implementation	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6	Priorities
JAX-RS	$^{20}/_{109}$	$^{21}/_{131}$	$^{5}/_{16}$	$^{6}/_{17}$	0.25
HAPI FHIR	$^{80}/_{109}$	$^{84}/_{131}$	$^{7}/_{16}$	$^{9}/_{17}$	0.59
Smile CDR	$^{4}/_{109}$	$^{12}/_{131}$	$^{1}/_{16}$	$^{1}/_{34}$	0.05
Iguana 6	$^{5}/_{104}$	$^{14}/_{131}$	$^{3}/_{16}$	$^{3}/_{34}$	0.11

For the ease of implementation criterion, it was obtained the following priorities vector: $\begin{bmatrix} 0.25 \\ 0.59 \\ 0.05 \\ 0.11 \end{bmatrix}$.

And for last, the supportability criterion calculations.

Table 20 - Supportability Criterion’s Alternatives Comparison

Supportability	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	1	$1/5$	$1/4$	2
HAPI FHIR	5	1	2	6
Smile CDR	4	$1/2$	1	5
Iguana 6	$1/2$	$1/6$	$1/5$	1

Table 21 - Supportability Criterion’s Alternatives First Step of Normalization

Supportability	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	1	$1/5$	$1/4$	2
HAPI FHIR	5	1	2	6
Smile CDR	4	$1/2$	1	5
Iguana 6	$1/2$	$1/6$	$1/5$	1
SUM	$21/2$	$28/15$	$69/20$	14

Table 22 - Supportability Criterion’s Alternatives Second Step of Normalization

Supportability	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6
JAX-RS	$2/21$	$3/28$	$5/69$	$1/7$
HAPI FHIR	$10/21$	$15/28$	$40/69$	$3/7$
Smile CDR	$8/21$	$15/56$	$20/69$	$5/14$
Iguana 6	$1/21$	$5/56$	$4/69$	$1/14$

Table 23 - Supportability Criterion’s Alternatives Priorities Calculation

Supportability	JAX-RS	HAPI FHIR	Smile CDR	Iguana 6	Priorities
JAX-RS	$2/21$	$3/28$	$5/69$	$1/7$	0.10
HAPI FHIR	$10/21$	$15/28$	$40/69$	$3/7$	0.51
Smile CDR	$8/21$	$15/56$	$20/69$	$5/14$	0.32
Iguana 6	$1/21$	$5/56$	$4/69$	$1/14$	0.07

For the supportability criterion, it was obtained the following priorities vector: $\begin{bmatrix} 0.10 \\ 0.51 \\ 0.32 \\ 0.07 \end{bmatrix}$.

The final step passes through the calculation of the most adequate framework based on the criteria weight and the priorities vectors that indicate the frameworks' level of importance for each criterion, which values are presented in Figure 20.

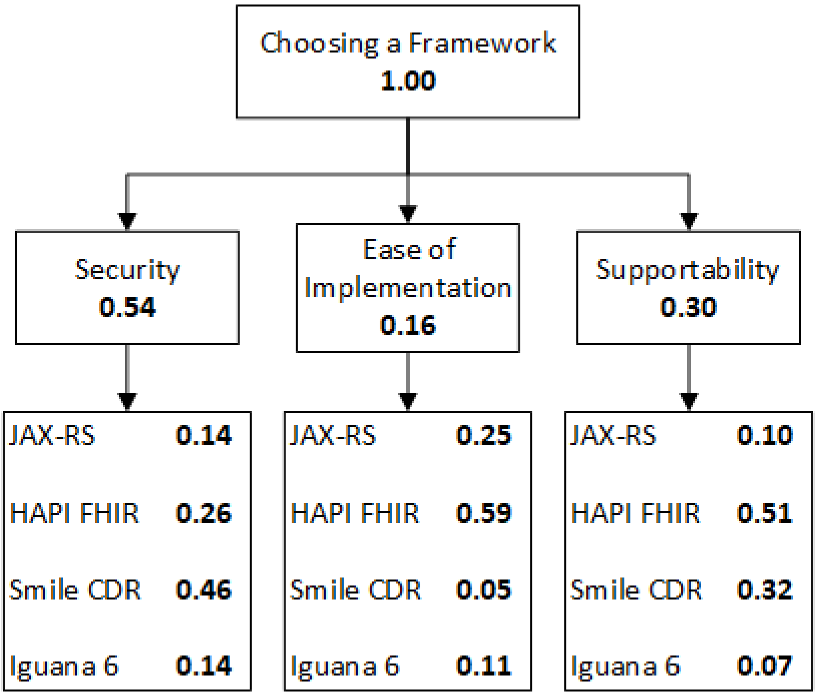

Figure 20 – Frameworks and criteria weights

Next, it'll be obtained the composite priority for the alternatives:

$$\begin{bmatrix} 0.14 & 0.25 & 0.10 \\ 0.26 & 0.59 & 0.51 \\ 0.46 & 0.05 & 0.32 \\ 0.14 & 0.11 & 0.07 \end{bmatrix} \begin{bmatrix} 0.54 \\ 0.16 \\ 0.30 \end{bmatrix} = \begin{bmatrix} 0.15 \\ \mathbf{0.39} \\ 0.35 \\ 0.11 \end{bmatrix}$$

Observing the results, we can conclude that HAPI FHIR is the most adequate choice to be used as a framework during implementation, since it obtained the highest value (0.39) compared to the remaining frameworks (JAX-RS with 0.15, Smile CDR with 0.35 and Iguana 6 with 0.11).

3.4.5 Implementation

The last stage of the process is the implementation of the product, which, for this project in concrete, is the implementation of the identified functionalities with FHIR®, using the selected framework based on the previous criteria specified, and the results acquired from the AHP method.

4 Analysis and Design

Before starting modeling a solution for development, it’s necessary to analyze with detail the IHE profiles, which indicate the involved actors, transactions and FHIR resources that should be used for each use case. Additionally, this chapter describes additional aspects related to the solution’s architecture, features, and business workflow.

4.1 IHE Profiles’ Technical Specification

Although the current infrastructure already possesses a set of features based on IHE profiles that allow the sharing of relevant information to be consulted by several stakeholders, these are still required to exist with FHIR® to extend the product’s interoperability capabilities with other systems. Therefore, throughout this section, the required IHE profiles’ technical specification based on FHIR® are described.

4.1.1 PIXm

The PIXm profile is related to the correlation of patient’s identifiers. The implementation approach defined by this profile is based on existing IHE profiles (PIX and PIXV3 profiles), and uses the FHIR resources presented in Table 24.

According to IHE’s technical framework (IHE ITI Technical Committee, 2017e, p. 50), in order to correctly adopt and use this profile, the involved domains must agree on the following terms:

- Policies to cross-reference patient identities across the domains;
- Processes to administer policies;
- Administration authority to manage the processes and policies.

Table 24 – PIXm FHIR resources (IHE ITI Technical Committee, 2017a)

FHIR resource	FMM Level
Patient	5
Parameters	5
OperationOutcome	5

Figure 21 shows the main involved PIXm transactions (Mobile Patient Identifier Cross-reference Query [ITI-83]) and actors (PIXm Patient Identifier Cross-reference Manager and PIXm Patient Identifier Cross-reference Consumer), along with PIX and PIXV3 profiles, which are represented in gray. Since both PIXm and PIX profiles follow similar requirements, the new PIXm's [ITI-83] transaction serves as a FHIR® alternative to the previous existing PIX's [ITI-9] transaction, which is based on another HL7 standard.

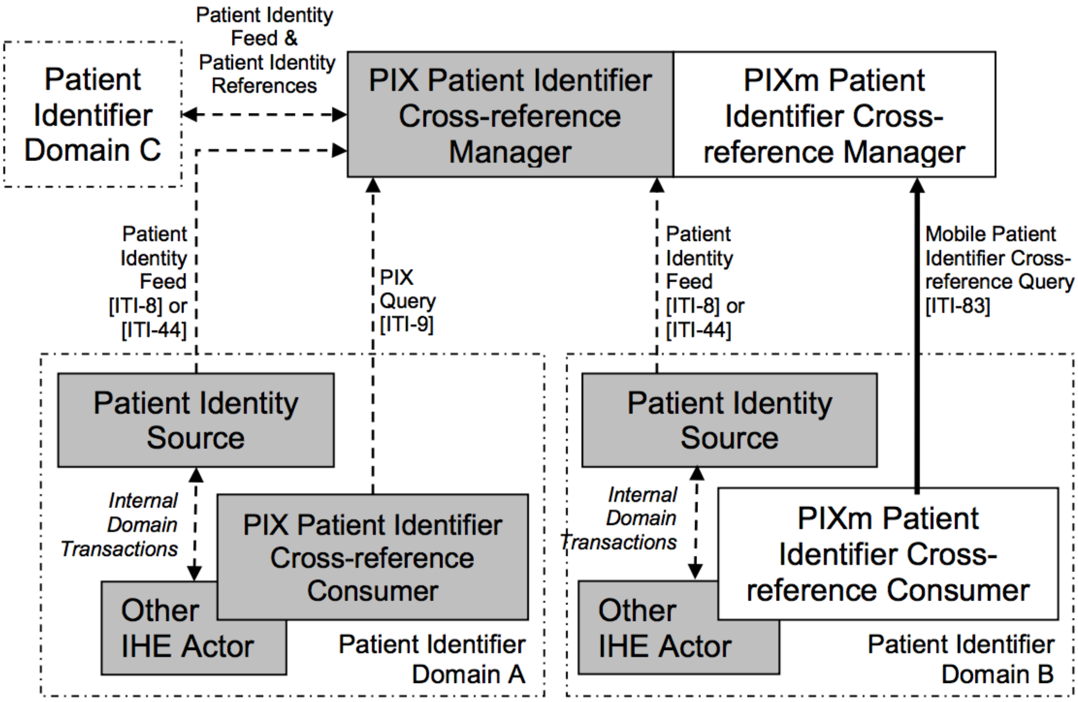

Figure 21 – PIXm overview (IHE ITI Technical Committee, 2017a, p. 13)

For this profile, the PIXm Consumer requests a list of possible patient identifiers to the PIXm Manager, using FHIR's *$ihe-pix* operation for a Patient resource type, including in the request a patient identifier along with the assigning authority. The PIXm Manager cross-matches the requested patient identifier with existing ones from other domains, returning a list of corresponding identifiers to the PIXm Consumer, using a Parameters resource in case of success. If the PIXm Manager can't find or identify the requested fields, it shall return an OperationOutcome resource indicating the error, as presented in Example 9 (IHE ITI Technical Committee, 2017a, vol. 2).

Example 9 – PIXm OperationOutcome response


```
{
    "resourceType": "OperationOutcome",
    "issue": [
        {
            "severity": "error",
            "code": "not-found",
            "diagnostics": "sourceIdentifier Patient Identifier not found"
        }
    ]
}
```


4.1.2 PDQm

The PDQm profile focuses on the management of the patient's demographic information. The implementation approach defined by this profile is based on the IHE's PDQ profile and uses the FHIR resources presented in Table 25.

Table 25 – PDQm FHIR resources (IHE ITI Technical Committee, 2017b)

FHIR resource	FMM Level
Patient	5
OperationOutcome	5
Bundle	5

Figure 22 shows PDQm main actors, which are the Patient Demographics Supplier and the Patient Demographics Consumer, and the involved transaction, the Mobile Patient Demographics Query.

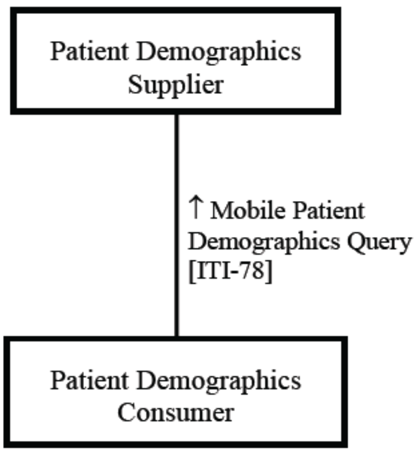

Figure 22 – PDQm overview (IHE ITI Technical Committee, 2017b, p. 9)

Each actor represents a distinct system with distinct tasks, as a client-server environment, where the Patient Demographics Consumer acts as a client to perform the request, and the Patient Demographics Supplier acts as a server to process the respective [ITI-78] transaction.

For this profile the Patient Demographics Consumer requests to the Patient Demographics Supplier, a list of patients with demographic information using a query with search criteria that a Patient resource must satisfy. After receiving the request, the Patient Demographics Supplier matches the supplied criterion with patients from other domains, returning a Bundle resource with a set of Patient resources, filled with their demographic information, to the Patient Demographics Consumer (cf. Example 10).

Example 10 – Response example for PDQm [ITI-78] transaction


```
{
    "resourceType": "Bundle",
    "id": "...",
    "type": "searchset",
    "total": 1,
    "link": [
        {
            "relation": "self",
            "url": ".../fhir/Patient?given=...&family=..."
        }
    ],
    "entry": [
        {
            "fullUrl": ".../fhir/Patient/p1",
            "resource": {
                "resourceType": "Patient",
                "id": "p1",
                "name": [
                    {
                        ...
                    }
                ],
                "gender": "male",
                ...
            }
        }
    ]
}
```

If no patients are found based on the supplied criteria, the Patient Demographics Supplier shall return a Bundle with an empty set. In case one of the domains is not recognized, or the Supplier isn't capable of responding in the required response format (XML or JSON), the request shall return an OperationOutcome resource, indicating the respective error (IHE ITI Technical Committee, 2017b).

Alternatively, the Consumer could request only the demographic information from one single patient. In this case, the Supplier shall search for the patient demographic information using the identifier provided by the Consumer, returning either a Patient resource in case of success, or an OperationOutcome in case it fails to find the demographic record (IHE ITI Technical Committee, 2017b).

4.1.3 MHD

The MHD profile is associated with the exchange of health documents. The implementation approach defined by this profile uses the FHIR resources presented in Table 26.

Table 26 – MHD FHIR resources (IHE ITI Technical Committee, 2017c)

FHIR resource	FMM Level
Bundle	5
DocumentManifest	2
DocumentReference	3
List	1
OperationOutcome	5
Binary	5

Figure 23 shows MHD actors and the transactions involved between them, namely, a Provide Document Bundle [ITI-65] transaction between a Document Source (client system) and a Document Recipient (server system), and the transactions Find Document Manifests [ITI-66], Find Document References [ITI-67] and Retrieve Document [ITI-68], between a Document Consumer (client system) and a Document Responder (server system).

For this profile, to publish documents, the Document Source performs a FHIR *transaction* operation, which request contains a Bundle resource with a DocumentManifest resource, at least one DocumentReference resource, none or more List resources, and none or more Binary resources. Even though the specification includes all the FHIR resources mentioned in Table 26, some of those resources still have a low maturity level, for example the List resource, which suggests that this profile might suffer major modifications in future FHIR updates.

The Document Recipient receives the request, persists the document(s) and returns a Bundle with a set of processing results, one for each entry requested (IHE ITI Technical Committee, 2017c).

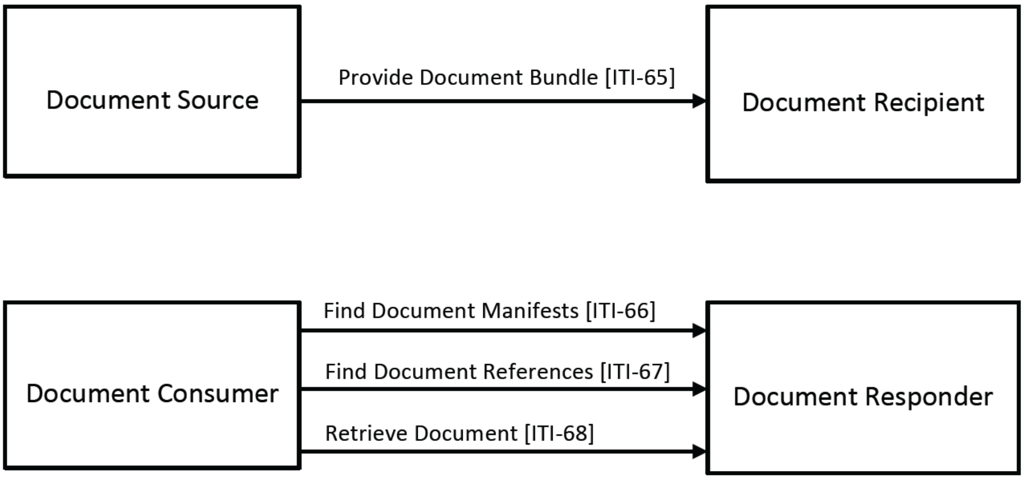

Figure 23 – MHD overview (IHE ITI Technical Committee, 2017c, p. 13)

The other transactions involve finding and retrieving documents metadata and the documents themselves. Regarding the transaction to retrieve documents, the Document Consumer can request to the Document Responder a list of DocumentManifest resources based on a query with some search criteria. After finding the document manifests, the Document Responder returns a Bundle resource which contains a set of DocumentManifest resources. Each DocumentManifest represents a collection of documents that are related together to provide information about a specific subject, providing references to target either DocumentReference resources or Media resources. The Document Consumer can also request the references for single documents to the Document Responder, which is performed using a query with search parameters to a DocumentReference resource. Like the previous transaction, the Document Responder returns a Bundle resource with a set of DocumentReference resources that matched the search criteria, including OperationOutcome resources in case of errors' occurrence. The last transaction is used to retrieve the document itself, where the Document Consumer requests it to the Document Responder, using the document reference URL retrieved in the previous transaction (IHE ITI Technical Committee, 2017c).

4.1.4 ATNA

The Audit Trail and Node Authentication profile describes approaches for auditing activities which are already implemented and being used by other profiles in the existing product's modules. In contrast to the other profiles, ATNA does not require FHIR® resources and services, but its audit trail features are still required to be integrated in the profiles' business workflow (Record Audit Event [ITI-20] transaction), since every transaction needs to be recorded in the existing audit record repository (ALERT® ARR). Although a specification to retrieve audit events with FHIR services exists, considering this project's requirements, the FHIR services are not required to perform the desired audit trail feature.

Figure 24 shows ATNA profile main actors and transactions. ATNA is composed by a Secure Application, a Secure Node, an Audit Record Repository, and an Audit Record Forwarder.

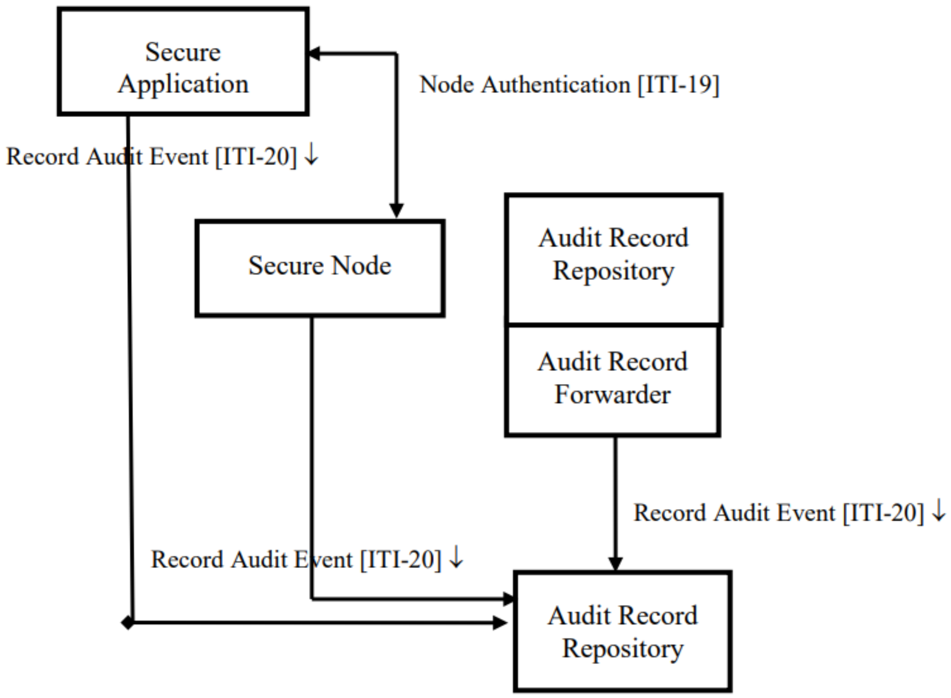

Figure 24 – ATNA overview (IHE ITI Technical Committee, 2017e, p. 73)

The Secure Node actor represents the element that provides security services for the whole system and performs Node Authentication transactions for network connections to authorize access for Secure Application actors, storing its activity in the Audit Record Repository through Record Audit Event transactions. Unlike the Secure Node, the Secure Application actor only involves the actors he is grouped with, providing similar security services and audit transactions regarding the application's features. The Audit Record Repository actor stores all the incoming audit events with message formats specified by IHE, providing local security and user access controls. Lastly, the Audit Record Forwarder actor major responsibility is to forward audit messages, received by the Audit Record Repository related to it, to other existing Audit Record Repositories based on a forwarding configuration.

4.1.5 mXDE

Figure 25 sums up the overall levels of granularity involved with a mXDE profile scenario, representing the services executed by the middle infrastructure which can retrieve coarse-grained information (CDA documents) or fine-grained information (data elements).

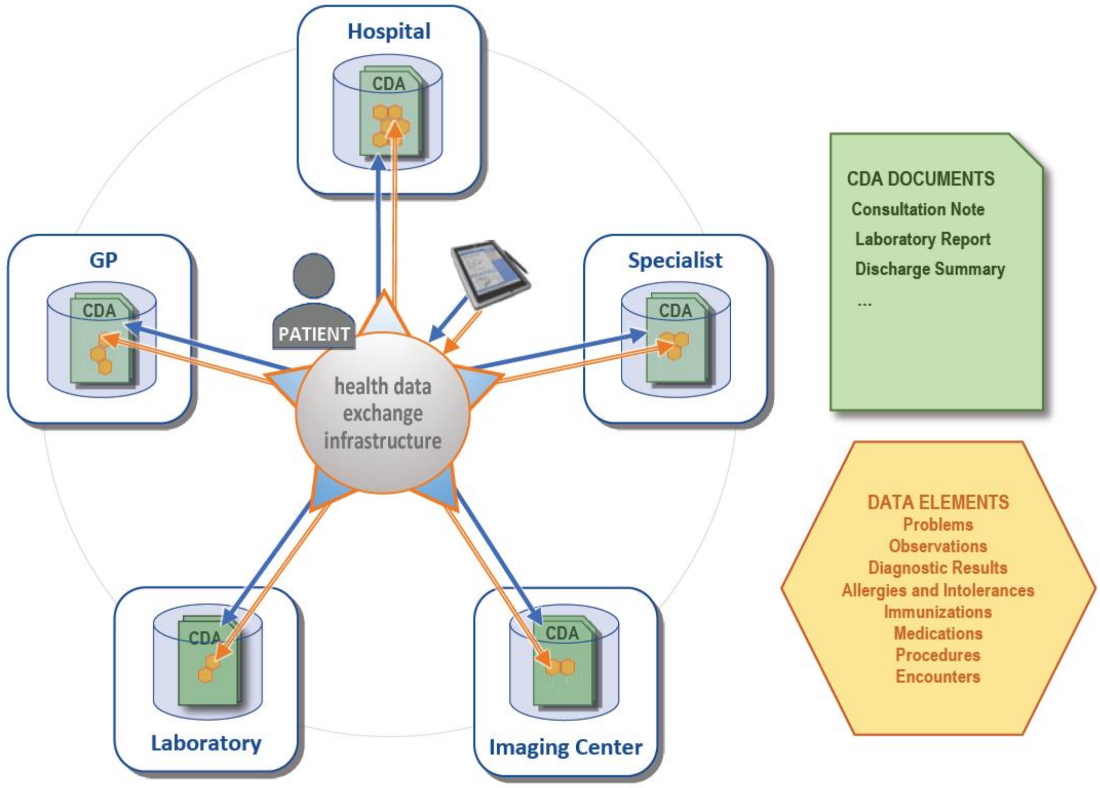

Figure 25 – mXDE levels of granularity (IHE ITI Technical Committee, 2017d, p. 13)

Figure 26 shows the involved actors (Data Element Provenance Consumer and Data Element Extractor) that are grouped with other IHE profile actors to perform the required transactions that access documents' information.

In this case, the Data Element Provenance Consumer uses IHE's Query for Existing Data for Mobile (QEDm) profile transaction (Mobile Query Existing Data [PCC-44]) to request a query for data elements to the Data Element Extractor, which extracts those data elements from the respective documents, stored in Document Repositories, assembling the provenance information for return. After receiving the provenance information, the Data Element Provenance Consumer can request the Document Repositories to retrieve the documents referenced by the provenance, using the transactions from the XDS Document Consumer or the MHD Document Consumer.

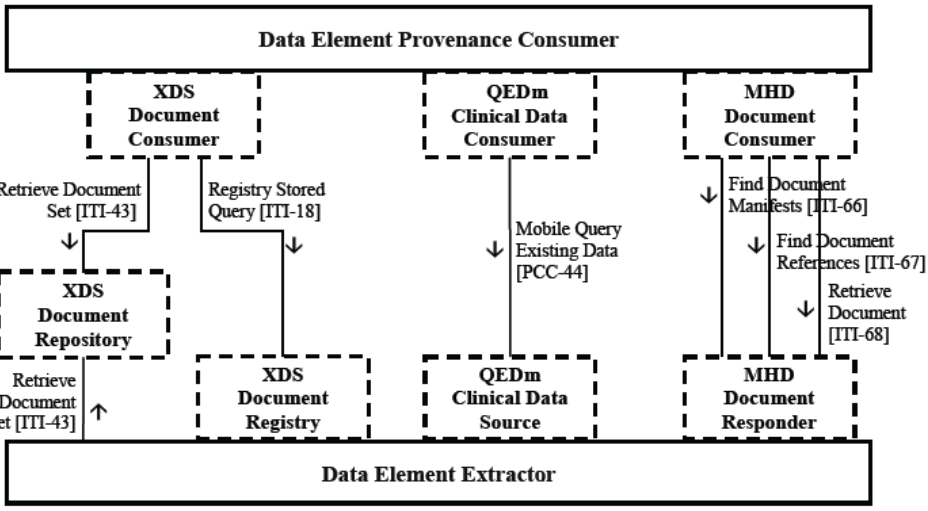

Figure 26 – mXDE overview (IHE ITI Technical Committee, 2017d, p. 16)

4.1.6 IUA

The IUA profile offers an approach to verify the users' authorization to perform certain actions, providing a system with access control to RESTful services and enabling security for certain FHIR® resources. Figure 27 shows the main involved actors, which are the Authorization Client, the Resource Server and the Authorization Server, and the respective transactions between these, namely the Incorporate Authorization Token [ITI-72] and the Get Authorization Token [ITI-71].

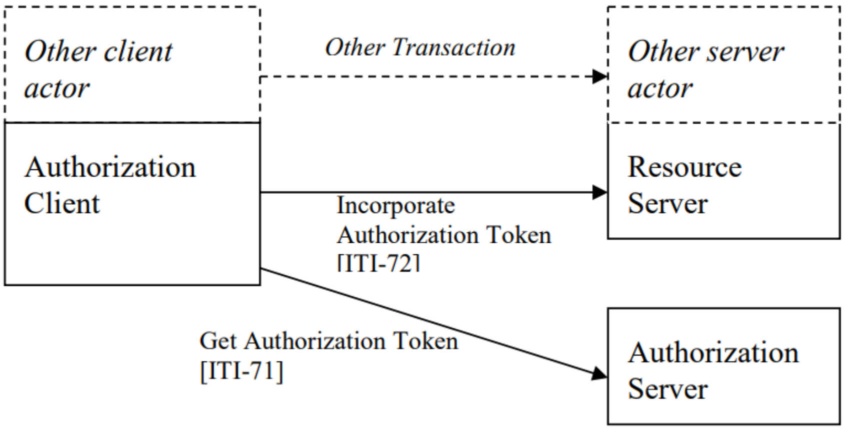

Figure 27 – IUA overview (IHE ITI Technical Committee, 2015, p. 10)

In this profile, the AuthorizationClient requests to the AuthorizationServer permission to execute an operation. After that, the AuthorizationServer returns an authorization token, which can be a JSON Web Token (JWT) (Jones et al., 2015), a Security Assertion Markup Language (SAML) token (Lockhart et al., 2008), or an OAuth Bearer token (Jones and Hardt, 2012), which can then be used by the client to verify its authority to the ResourceServer, enabling the client to perform the resource request.

4.2 Domain Model

Figure 28 presents a domain model (for clarity, attributes are omitted) encompassing all major concepts and their relationships present in this project.

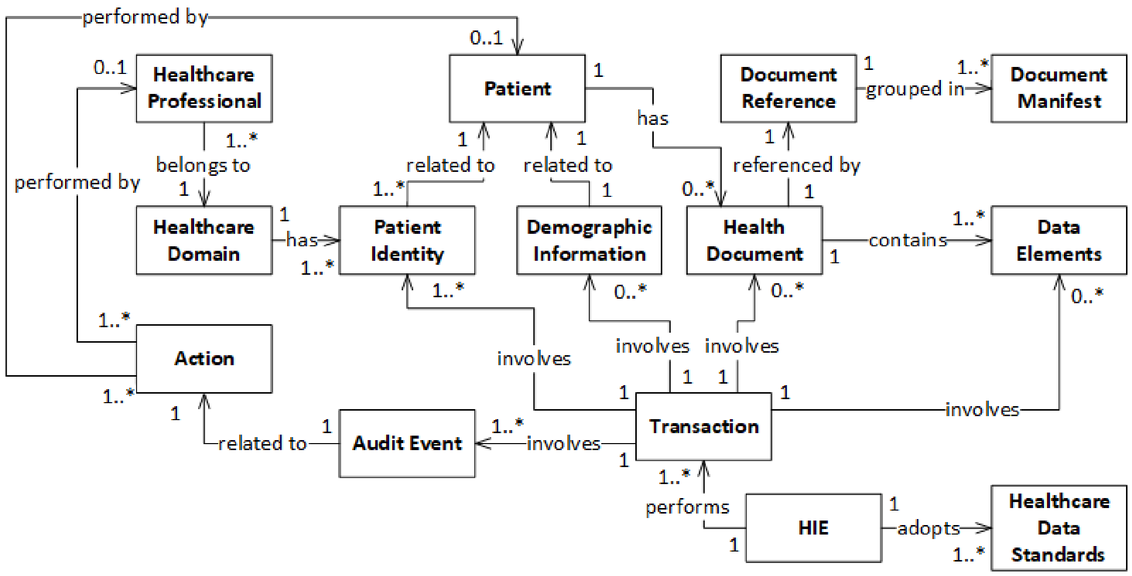

Figure 28 – Domain Model

Based on the previous requirements, it can be acknowledged that multiple healthcare professionals, such as nurses and physicians, belong to a healthcare domain, where each domain has a set of patient identities which are related to the patients that frequented that domain. Multiple patient identities can be related to a single patient, since several departments or institutions can have patient identifiers in their systems, related to the same person. Each health document is identified by a single document reference, which can be grouped in a document manifest. Since several health documents, from multiple areas of expertise, can be related to one patient, usually these are clutched to a document manifest, providing a set of health-related information about a single patient. However, the existence of at least one health document for each existing patient is not obligatory. To retrieve and use all of this information, HIE performs a set of transactions which respect several healthcare data standards, to manage the patients' identifiers, their demographic information, the health documents related to them, and even a portion of data that is contained in these documents, to aid on specific tasks. The HIE transactions are always related to at least one patient identity, but some may not require the use of patients' demographic information, nor access to health documents and their content, requiring only the patient identifier. Moreover, these transactions also involve audit events, which provide information about the actions performed in the client's system, either by patients or healthcare professionals.

4.3 HIE Architecture Design

This section describes the new module added to the ALERT® HIE product and the major dependencies created with other existing modules for the required IHE profiles.

4.3.1 ALERT® HIE

A new module, which was named ALERT® FHIR®, was added to the ALERT® HIE current solution (previously described in section 2.5.2) to perform the required IHE profiles' transactions while using some core features and implementations offered by HAPI FHIR.

Figure 29 shows the ALERT® modules with the new ALERT® FHIR® module, represented in bold, which provides the HAPI FHIR framework through FHIR Core, and a FHIR Restful API for the consumption of FHIR services. This module also consumes services provided by ALERT® XDS for the documents, ALERT® PIXPDQ for the patients, and HIE-Security for security features, which required new logic to be added for the existing ALERT modules, since it is necessary for the new services' business workflow, considering the IHE profiles' specification. Additionally, ALERT® PIXPDQ, ALERT® XDS, and HIE-Security, also consume ALERT® FHIR® to use some features provided by the HAPI FHIR framework.

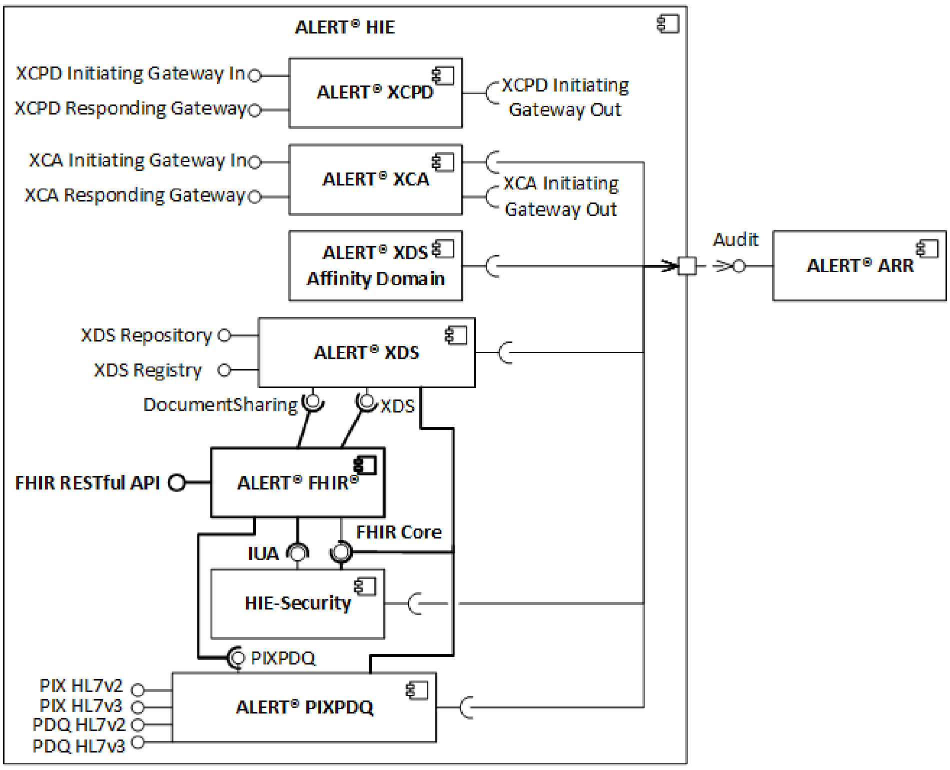

Figure 29 - HIE high level architecture diagram with new module

The ALERT® XCPD, ALERT® XCA and ALERT® XDS Affinity Domain modules did not contribute in any way to the development of the new required profiles, since they lack features or business logic that would contribute in executing the new transactions, and for that reason, they are not further discussed.

The decision to add this new module was taken based on the following points:

- **Expose FHIR services:** to define the FHIR services, HAPI FHIR capabilities are required for the development of a FHIR servlet that exposes endpoints for the required IHE profiles' transactions;

- **Expose FHIR features for other modules to use:** to make the most of HAPI FHIR's features, the framework should be isolated from the remaining modules to make their access from existing and future components easier;

- **Consume existing services:** the new FHIR services should be capable of using existing ALERT services if they are suitable for the required IHE's profile features.

HAPI FHIR provides an interesting number of features that can be used to define the services' endpoints and major core configurations, required to create a FHIR server. To isolate these features from the remaining product, the ALERT® FHIR® module was created with two sub-components that will be further described in the following section, which provide the means for external modules to access FHIR core features, without creating unnecessary dependencies.

This design promotes FHIR services' extensibility for new future profiles, the implementation of which should be in compliance with the Open Closed Principle (OCP) defined by Robert C. Martin (2000, p. 4), and establishes a separation of concerns between FHIR core features, and the profiles' business model and logic. Furthermore, since the standard is still evolving and might suffer additional changes in its specification, the centralization of these features will help minimize the impact of future standard modifications over the existing modules.

4.3.2 ALERT® FHIR®

As referred in the previous section, two sub-components were added to ALERT® FHIR®, which can be visualized in Figure 30, namely HIE-FHIR-Core and HIE-FHIR.

As can be seen in the previous diagram, each component has a set of packages that were created to separate the required features for FHIR services:

- **HIE-FHIR-Core:** this component handles the HAPI-FHIR framework specification and provides core features such as utility classes, FHIR paging configurations, transaction client services, interceptors, and FHIR error handlers and diagnostics, which are used for REST services with FHIR resources and business workflows;

- **HIE-FHIR:** this component defines the servlet and the respective FHIR endpoints, using Resource Providers or Plain Providers which are implemented using the HAPI-FHIR framework to define the services' parameters according to IHE's profile specifications.

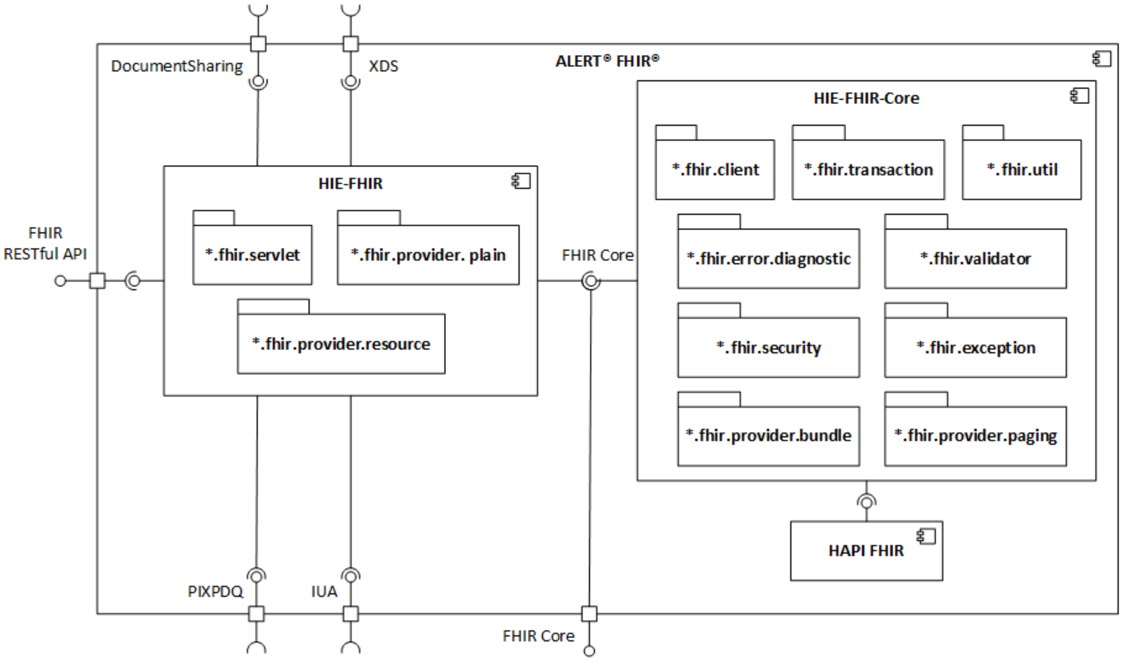

Figure 30 – ALERT® FHIR® component diagram

Overall the HIE-FHIR component was created to use HAPI FHIR's Resource or Plain Providers in order to define the services' endpoints. The Resource Providers define an endpoint with a FHIR REST resource in the URL before the parameters, while the Plain Providers simply define the base URL without FHIR REST resources, expecting a set of parameters depending on the respective methods. The servlet was also created using HAPI FHIR, to define the major configurations for the FHIR server, such as the version of the standard that is being used, and the interceptors required to manipulate incoming requests with additional behavior, among other features.

The HIE-FHIR-Core mainly provides utility classes and other features that will stay constant throughout the implemented services, for example, error diagnostics that shall follow IHE's criteria, and paging and bundle features, which define the format and amount of FHIR resources that are returned within a Bundle FHIR resource (the FHIR resource used to return a list of other FHIR resources).

4.3.3 ALERT® PIXPDQ

The ALERT® PIXPDQ module (Figure 31), to which was added new business logic regarding the PIXm and PDQm profiles, is one of the ALERT® modules consumed by ALERT® FHIR® to use implemented features regarding the PIX and PDQ profiles.

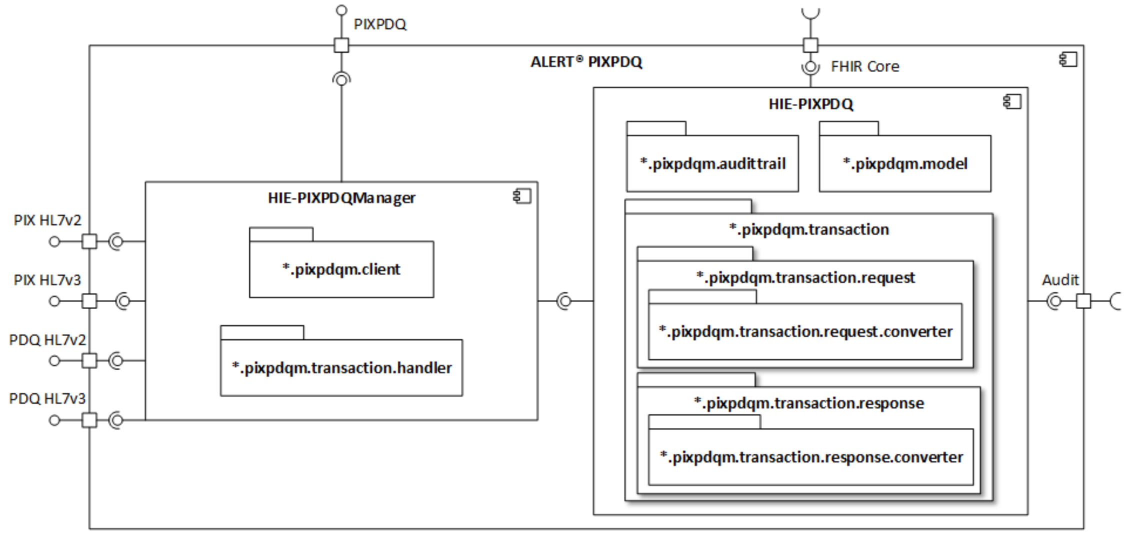

Figure 31 – ALERT® PIXPDQ component diagram

By observing the previous diagram, two of ALERT's sub-components are identified:

- **HIE-PIXPDQManager:** this component contains clients for the PIXm and PDQm profile, that will handle incoming requests from the HIE-FHIR component and initiate the business workflow for the existing PIX and PDQ services, and transaction handlers that execute database queries for those services;

- **HIE-PIXPDQ:** this component contains the new main model classes used to perform the service, including a set of FHIR converters and audit message builders which are used during the business workflow execution.

The package logic adopted follows the product's current architecture, since most of these packages hold a set of classes required to perform the integration profiles workflow. The components that represent the product "frameworks" usually contain packages that involve the service's transactions defined by IHE, along with request and response converters required to convert the incoming HTTP requests content into the respective transaction model. In this case, the HIE-PIXPDQ component contains that logic, while the HIE-PIXPDQManager uses the other component's logic as a framework to manage and process the integration profile's services. The workflow and the classes used in these packages are described with detail in section 4.4 and chapter 5.

4.3.4 ALERT® XDS

The ALERT® FHIR® module consumes the ALERT® XDS module (Figure 32) to perform MHD services while using existing XDS services.

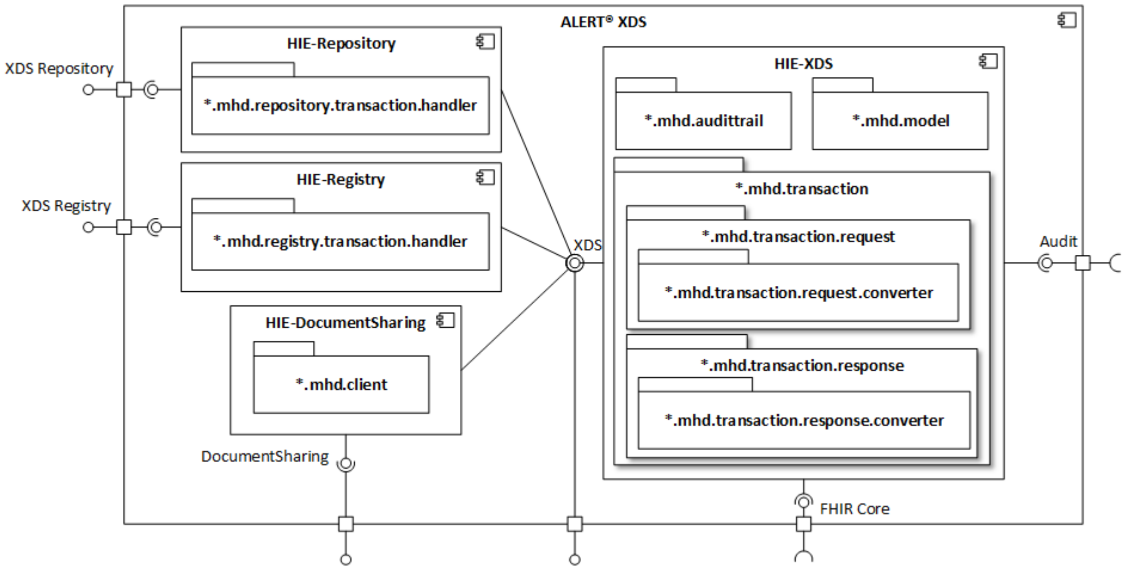

Figure 32 - ALERT® XDS component diagram

There are three main ALERT's sub-components involved in the MHD profile within this module, namely:

- **HIE-Registry:** this component contains the logic to perform queries to the database, regarding the documents' metadata;
- **HIE-DocumentSharing:** this component contains the clients that will handle incoming requests from the HIE-FHIR component and initiate the business workflow for the existing XDS services;
- **HIE-XDS:** this component contains the transactions' logic, request and response converters, and the audit message builder required to perform MHD services;

As it can be seen in the previous diagram, the package logic is similar to the one from the ALERT® PIXPDQ module, with small variations regarding the transaction handler package, and the client package. Due to the XDS profile requirements, the HIE-DocumentSharing component holds client logic for the existing registry and repository services, and because of this, the new MHD transaction clients were also added to this component. Regarding the transaction handler, since the MHD required transactions that involve document metadata and the actual documents, the transaction handlers' packages were added to the HIE-Registry component (for document metadata transactions) and the HIE-Repository component (for actual documents transactions).

4.3.5 HIE-Security

HIE-Security (Figure 33) is the last component consumed by the ALERT® FHIR® module. This component contains the security features for the incoming requests to the server, and therefore, contains the FHIR security logic in regard to the IUA profile requirements.

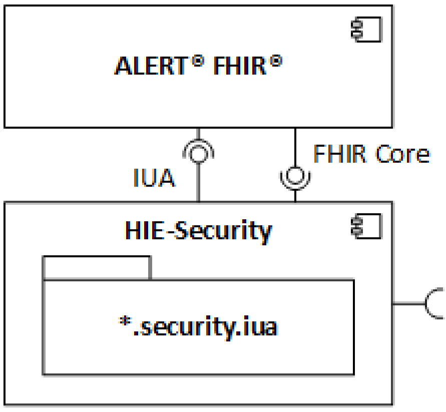

Figure 33 – HIE-Security component diagram

Since the IUA profile only defines criteria to handle an authorization token, it was acceptable to group the major features in a single package from this component, while consuming the ALERT® FHIR® module's HAPI FHIR core features.

4.4 IHE Profiles Design

This section describes the integration profiles' business workflow, the correlations between each other, through the use of Enterprise Integration Patterns (EIP) diagrams, defined by Hohpe and Woolf (2004), to explain the major integration tasks, and the respective Java classes that were defined to implement the profiles.

4.4.1 Conceptual Design

To implement each transaction workflow and requirements, a conceptual design was defined to illustrate and describe the major elements involved, from the client's initial request, to the server's final response, which is represented in Figure 34.

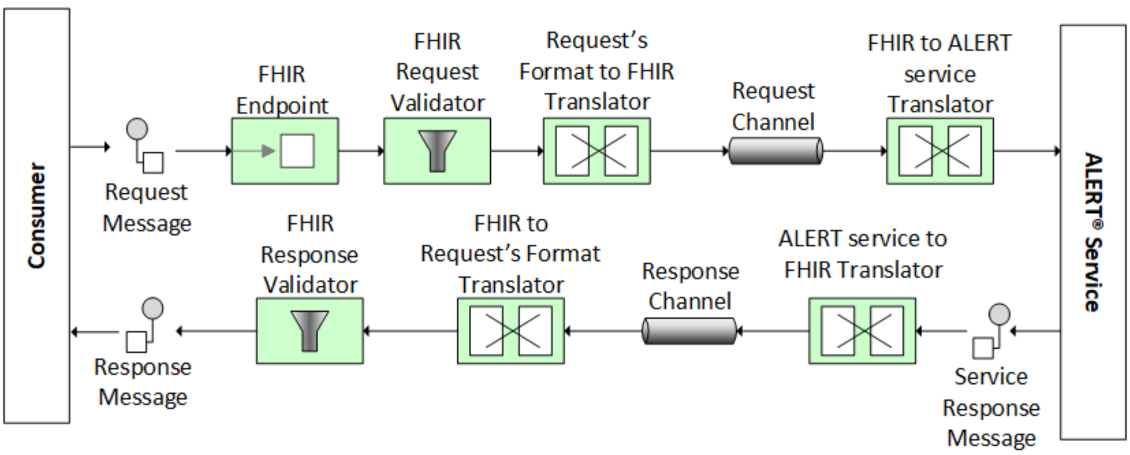

Figure 34 – Conceptual EIP design

For each required FHIR transaction, a single endpoint is exposed in the server side, to answer other system's requests. After reaching the endpoint, these transaction's requests initiate a process, composed by authentication (not represented in Figure 34), FHIR validations, and transformations that convert the request's initial format (JSON or XML) to the server's equivalent HAPI FHIR's Java object. The endpoint definition, the FHIR validations and the transformations between the request format and the FHIR Java object are always performed by HAPI FHIR's server features in the ALERT® FHIR® module. At this point of the process, and to take advantage of existing implementations, this Java object will suffer further transformations, for it to reach a format that can be used by an existing ALERT service, capable of handling this request. The service's response shall be exposed to further transformations and validations in the opposite direction, which is, subsequently, returned to the requesting system.

4.4.1.1 Pros and Cons

This approach presents some good and bad points. One major advantage is the reutilization of existing services, since it does not require the reimplementation and repetition of the current business logic that was previously discussed and implemented for the ALERT services. On the other hand, to use these, additional processing is necessary due to the transformations and verifications that must be considered while converting the FHIR objects to the corresponding ALERT services' objects. Although of little significance, this approach will undoubtedly increase the processing time that a request would take without all these transformations but will also reduce the amount of repeating code and features that were already in the product. Moreover, it also guarantees that, if the ALERT services change in the future due to new business rules, the new module will not be affected by those changes, since it only consumes the respective services.

4.4.1.2 Defined classes

To support the selected IHE profiles with the proposed design, a set of classes are defined to be used in the existing and new components, namely:

- **Resource or Plain Provider:** these classes receive incoming requests for a specific endpoint that can be related to a FHIR REST resource, and use a specific client to consume the parameters received and process the required service;

- **Client:** the client classes are responsible for creating the transactions and defining its properties, initiating the transaction process to use the ALERT® services;

- **Transaction:** transaction classes define a transaction that contains a request and a response for each respective service, following ALERT's design;

- **Request:** a request class contains the required and optional parameters for the services, which are in accordance to the respective Resource/Plain Provider incoming parameters, following ALERT's design;

- **Response:** a response class contains the required and optional fields to answer a specific service request, following ALERT's design;

- **Request Converter:** these classes are responsible for converting FHIR requests to other existing requests defined in the ALERT® HIE solution, which are used to perform existing ALERT services, regarding other IHE profiles;

- **Response Converter:** response converters are responsible for converting ALERT® transaction responses to the required FHIR responses, returning a message with a specific format and structure according to FHIR®;

- **Transaction Handler:** transaction handlers are related to a single transaction and perform the required operations to the database, based on the transaction's request, returning in the end a response for its transaction.

The Resource and Plain Providers are created with HAPI FHIR features and represent the FHIR Endpoints previously described in the EIP conceptual design (Figure 34), while the Request Converter and Response Converter represent the EIP Translator from FHIR to the ALERT service, and the EIP Translator from the ALERT service to FHIR, respectively. The remaining classes are used to assist in the whole process, where the client, the services of which are executed by the Resource and Plain Providers, creates the Transaction with its respective Request and Response objects, forwarding the process to the ALERT service workflow. The Transaction Handlers are used during the workflow, to perform the respective call to the database and, if necessary, some validations before and after the database query execution, to respect some specification requirements from the new IHE profiles.

These sets of classes are created not only to follow and adapt the implementation of HIE's current architecture, but also to respect the Single Responsibility Principle (SRP) pattern, which, according to Martin (2002), states that a class should only have one single reason to change. This approach also promotes Low Coupling and High Cohesion principles, to reduce the impact of change, improve maintainability, and to evenly distribute the functionalities among the involved objects (Larman, 2004).

4.4.2 PIXm and PDQm EIP

Figure 35 presents an EIP diagram for the PIXm and PDQm profiles. Since both profiles use the same FHIR resource (Patient), and have similar implementation requirements, their integration services are accessed through a single endpoint via HTTP GET requests, which is responsible for delegating the requested service to the correct profile.

After receiving a message request in PIXPDQm endpoint, the server starts by validating the incoming message to verify if the request format and parameters are supported by the server for the Patient resource. If the request is not valid, an error response message is sent back to the client, otherwise, the message will then be converted to HAPI FHIR objects. Next, and based on the message content, the server directs the request to either PIXm or PDQm profile workflow, where the message is converted to an equivalent object used for the PIX or PDQ profile transactions. The server performs the patient search with the existing features in ALERT® PIXPDQ module, returning a list of found identifiers (for PIXm), or a list of found patients (for PDQm), converting them to the respective FHIR resource (Parameters for PIXm, and Patient for PDQm).

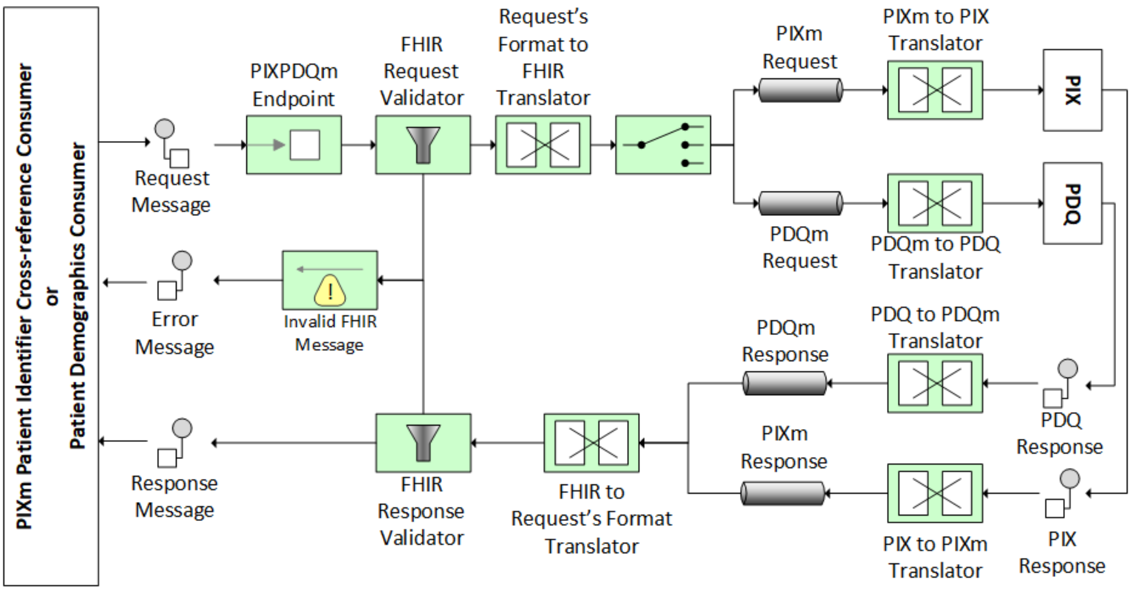

Figure 35 - PIXPDQm EIP Diagram

Next, HAPI FHIR framework converts that resource to the request's message format (JSON or XML), validating the response message with FHIR resource schemas, to verify if it complies with the standard, and sending it back to the client.

4.4.3 MHD EIP

The MHD profile is composed by mainly two business workflows, one for document registry and the other for the search and retrieval of documents.

4.4.3.1 Register Documents

Figure 36 describes the MHD profile workflow through an EIP diagram, regarding the registry of health documents. The defined endpoint for this service corresponds to the server's base URL and is accessed through a HTTP POST request since it is a FHIR transaction operation that contains a set of operations in the message body related to other FHIR resources, namely, DocumentManifest, DocumentReference, Binary and List.

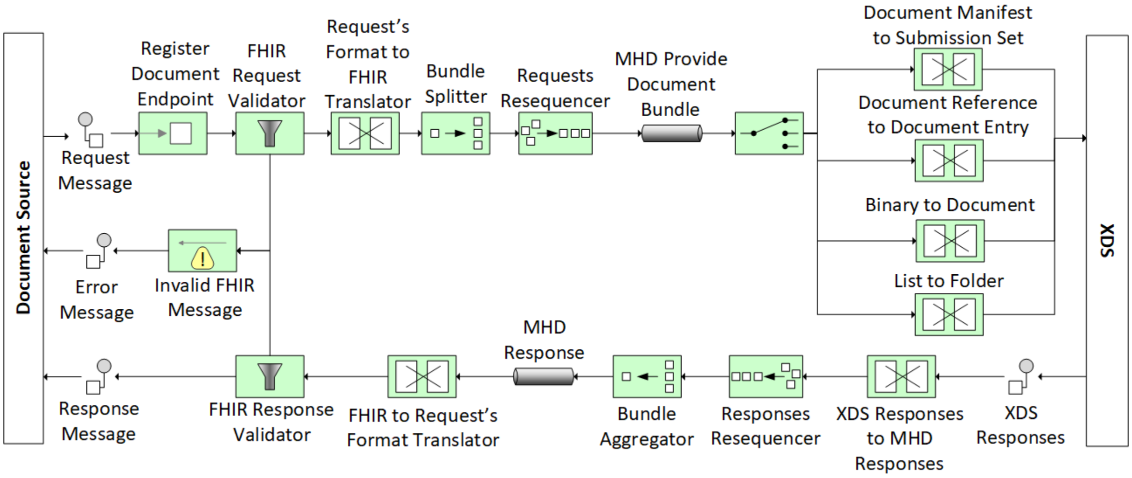

Figure 36 – MHD Register Documents EIP Diagram

After receiving an incoming request message, the server starts by validating its content and structure against FHIR's Bundle schema. If the validation fails, an error response is immediately returned to the client, if not, the transaction proceeds by converting the request to HAPI FHIR Java objects. Since this is a complex operation it is required to perform the requested operations in a specific order, depending on the HTTP verb, starting in the first place with the DELETE interactions, followed in second by the POST interactions, in third place with the PUT interactions, and finally in fourth place with the GET interactions. Although this is a requirement specified by the FHIR standard, the IHE specification only defines POST operations for this profile, which does not invalidate this required. Moreover, the server is responsible for executing these tasks in the required order, but the client is not obligated to send a request with a set of ordered operations.

Therefore, to respect the standard's requirements, after converting the request to a FHIR resource, the server is then responsible for splitting the Bundle (Bundle Splitter) and ordering the inner operations for later processing (Requests Resequencer). Next, the server sequentially performs each operation, by converting the respective request resource to the associated Java object related to the XDS profile, which is already implemented in the product. After converting the resources, the server processes the request through the existing XDS implementation, which creates the required document metadata and save its binary information in HIE's database. The database responses are then converted back to the expected MHD responses, which in this case is the operations' result, and grouped together in a Bundle resource (Bundle Aggregator). Since the operations' responses need to be returned in the same order as the first request, the server orders these responses before converting them to a Bundle (Responses Resequencer), finishing the transaction by converting the ordered Bundle resource to the request's initial format (XML or JSON), and validating the response against a FHIR response Bundle schema, which can either return an error or a successful message depending on the validation result.

4.4.3.2 Search Documents

Figure 37 describes the search operations for the MHD profile. Although the profile defines three search operations for multiple FHIR resources (DocumentManifest, DocumentReference, and Binary), each have a similar business workflow performed through HTTP GET requests to the respective resource endpoints, and therefore, only a single diagram is presented to describe these three operations.

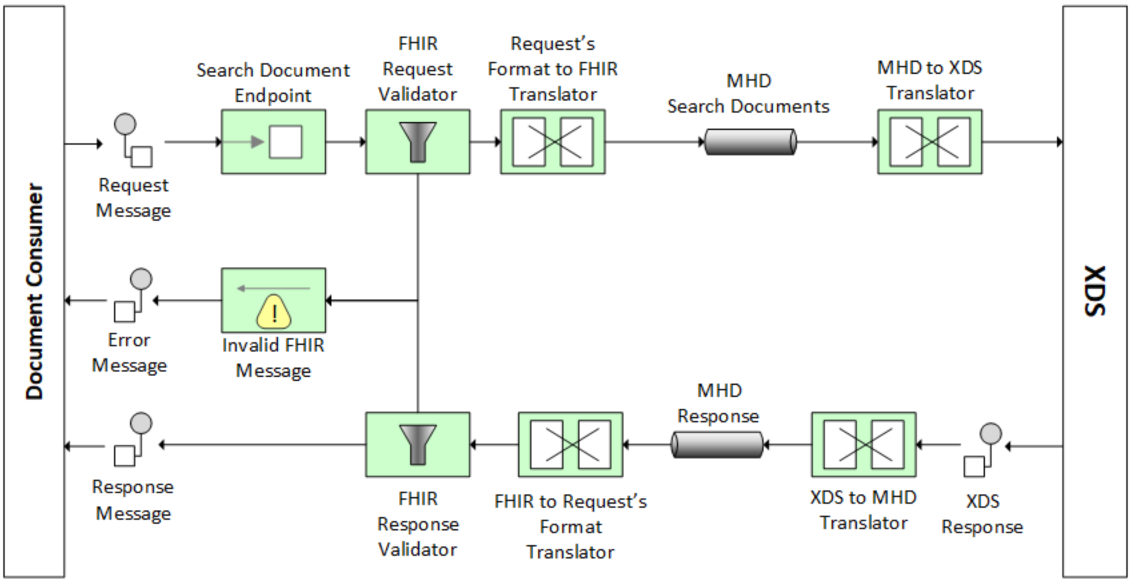

Figure 37 – MHD Search Documents EIP Diagram

After receiving an incoming request, the server validates the request format and parameters, returning an error message if the requested service for that resource is not found by the server. If the validation is successful, the server translates the request format to HAPI FHIR objects, which are then converted to an equivalent object used for XDS profile transactions. The server proceeds with the search using the existing features in ALERT® XDS, converting the found elements to FHIR resources. The server converts the FHIR response to the request's initial format (XML or JSON), validating that response against FHIR respective schema and returning it to the client.

4.4.4 QEDm EIP

Figure 38 shows the QEDm business workflow which is executed through the mXDE profile, with HTTP GET requests to the respective endpoints. For this profile, nine FHIR resources are used to retrieve fine-grained information from health documents and to define the FHIR endpoint (represented as QEDm Resource Endpoint in Figure 38), namely: Allergy Intolerance, Condition, Diagnostic Report, Encounter, Immunization, Medication Request, Medication Statement, Observation and Procedure. To simplify the representation, the diagram will only contain a single endpoint that represents one of these nine FHIR resources.

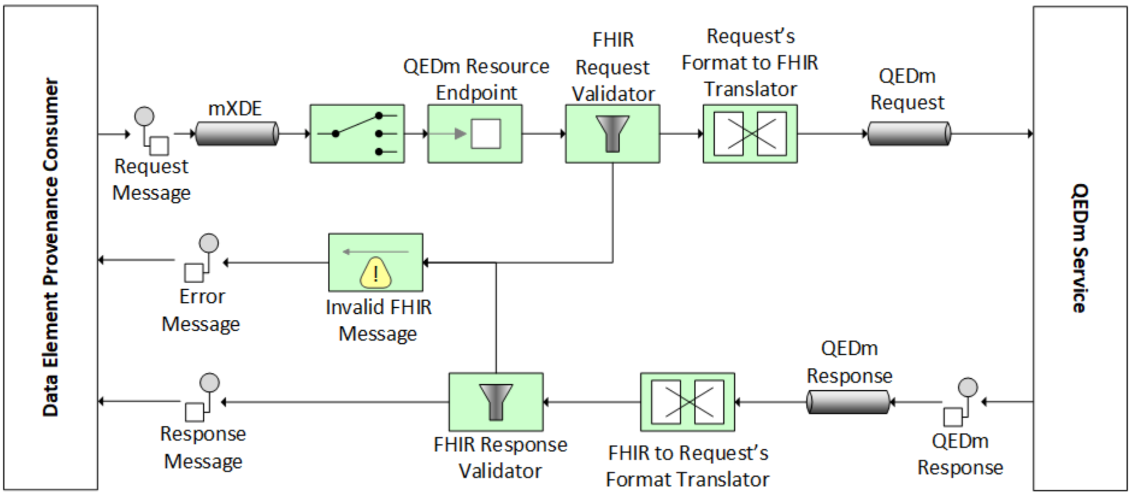

Figure 38 – QEDm EIP Diagram

After receiving a request in the mXDE profile, the QEDm profile processes the message through one of the respective endpoints, depending on the FHIR resource being used, which is also validated against the necessary FHIR request content and structure verifications. After processing the transaction and finding the desired information with QEDm services, the server converts the FHIR resource to the request's initial format (XML or JSON), validating the response and returning an error or successful message.

4.4.5 mXDE EIP

Figure 39 and Figure 40 describe mXDE profile workflow, which includes business workflows from other profiles, namely QEDm and MHD, and is invoked with HTTP GET requests. Although this profile uses multiple services from the other profiles, their consecutive execution is not obligatory for each mXDE event. Either way, the following business workflow example presents a more complex scenario where multiple services from multiple profiles are used together for a specific end.

A scenario will be considered where a client desired to retrieve partial information from a document stored in a repository, and after receiving that information, the client additionally requested the whole document for further reading. In the first place, the client sends a request to retrieve fine-grained data from a document which is directed to a specific endpoint from the QEDm profile depending on the data required (Figure 39). After finding the data, the server returns the response with the data content along with its provenance, which refers to the document location.

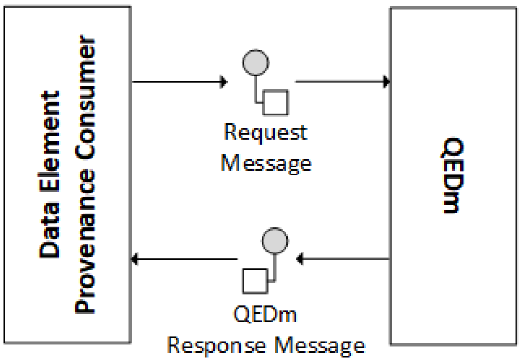

Figure 39 – mXDE Retrieve Partial Information EIP Diagram

After that, the provenance content is filtered and used to send a request to search and retrieve the document through the MHD profile, returning the respective response to the client (Figure 40).

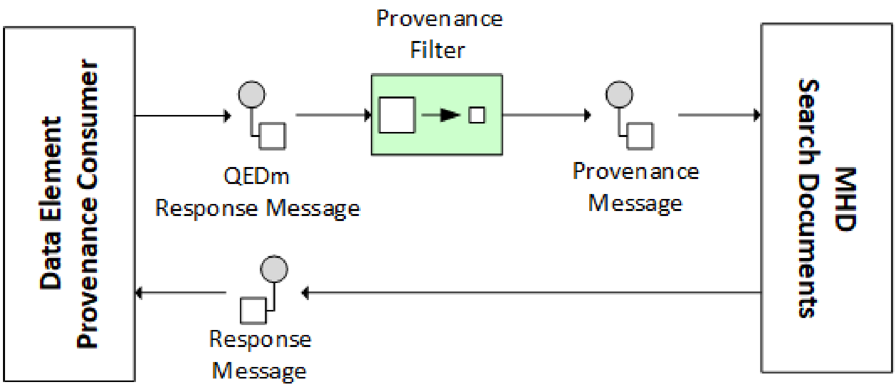

Figure 40 – mXDE Retrieve Document EIP Diagram

The mXDE profile specification suggested the possibility to optionally use the XDS profile to perform the document search, but since the MHD profile already adopts the XDS features, the document searches will always be executed through the MHD profile.

4.4.6 IUA EIP

The IUA profile is composed of two workflows, one to retrieve the authorization token, and the other to incorporate the authorization token in FHIR request messages. Although this profile was applied to the product, both transactions are mostly directed to the client, which consequently require the server to be capable of verifying and approving the authorization token.

4.4.6.1 Retrieve Token

Figure 41 presents the workflow regarding the token retrieval for the IUA profile, through an HTTP GET request.

In first place the client makes a request to the authorization endpoint, which authenticates the user that is making the request and verifies if the authenticated user is authorized to request a token. If the user has enough permissions, the server returns a generated token to be used for FHIR resource requests. It's relevant to refer that this transaction involves the client and the authorization server but does not involve any required implementation to the FHIR server, which is the one currently being implemented.

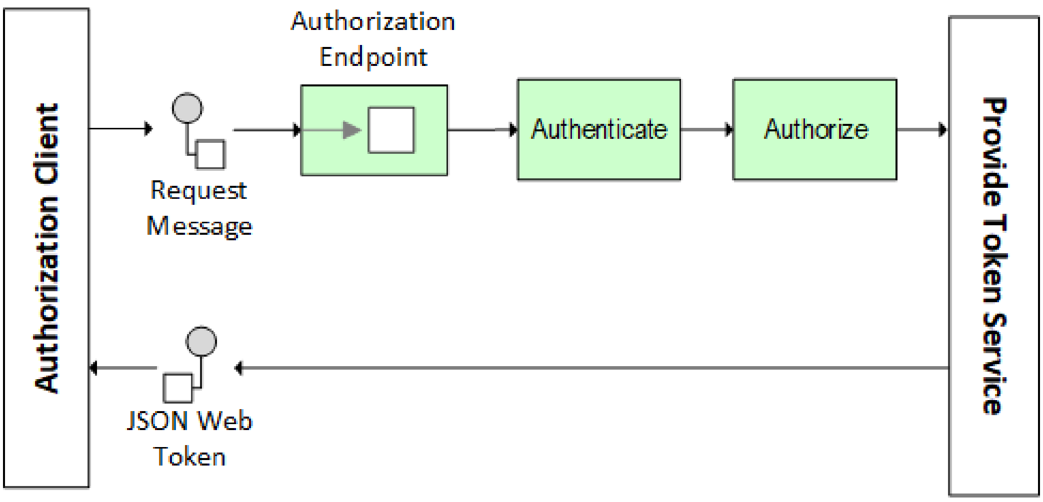

Figure 41 – IUA Retrieve Token EIP Diagram

4.4.6.2 Incorporate Token

Figure 42 describes the workflow regarding the token incorporation in requests directed to a FHIR resource endpoint. Each request made for a specific endpoint in the FHIR server must first be authorized with a JSON Web Token, which is applied for all the IHE profiles described previously, considering that IUA's focus is to deliver security measures for all server requests.

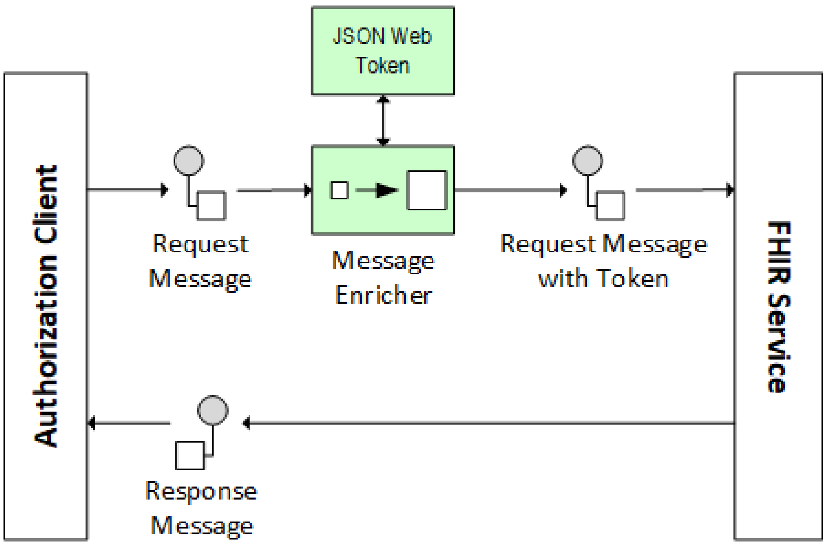

Figure 42 – IUA Incorporate Token EIP Diagram

For this scenario, the client simply incorporates the JSON Web Token, retrieved from the authorization endpoint, in the request message, sending it to the server and verifying if a token exists in the requests. The server verifies if a token exists in the incoming request, performing the necessary verifications and authorization measures to execute the desired service. For the profile's implementation in the product, the required portion of this transaction to be developed in the server is about the token's approval and request's security validations.

5 Development

This chapter describes in detail the implementation of the new components, the business logic, and other relevant aspects added to ALERT® HIE to implement the stipulated IHE profiles, using UML class diagrams, while considering the integration process and EIP diagrams previously described.

5.1 PIXm

For IHE's PIXm profile, the main modules involved are ALERT® FHIR®, composed by the components HIE-FHIR and HIE-FHIR-Core, and ALERT® PIXPDQ, composed by the components HIE-PIXPDQManager and HIE-PIXPDQ.

Considering the PIXm and PDQm EIP business workflow, the first step is to provide a PIXPDQm Endpoint to receive incoming requests. To do this, a HAPI FHIR Resource Provider was added to HIE-FHIR component named PatientResourceProvider, which, according to the profile specification, had to contain a FHIR operation named "$ihe-pix" with one required parameter and an optional parameter. Thanks to HAPI-FHIR framework the definition of this method was straightforward with the "@Operation" tag, since it defines the FHIR operation name, and the "@OperationParam" tag to define the required parameters for an operation service (Example 11). Usually IHE specification requires some parameters with specific data types that correspond to FHIR's data types, which in this case are "token" and "uri". HAPI-FHIR also provides these data types as Java objects that are named "TokenParam" and "UriParam" respectively.

Example 11 – PIXm service definition

```
@Operation(name="$ihe-pix", idempotent=true)
public Parameters pixmRequest(
        @OperationParam(name="...") TokenParam ...,
        @OperationParam(name="...") UriParam ...,
        HttpServletRequest ...) { ... }
```

The FHIR validations and conversions from a JSON or XML message to these required data types are all performed by the HAPI FHIR classes, since the PatientResourceProvider implements HAPI FHIR's IResourceProvider and is defined in the server's configurations (Figure 43). The PatientResourceProvider, and any other Resource Provider required, are logically grouped in the HIE-FHIR component's "*.fhir.resource.provider" package.

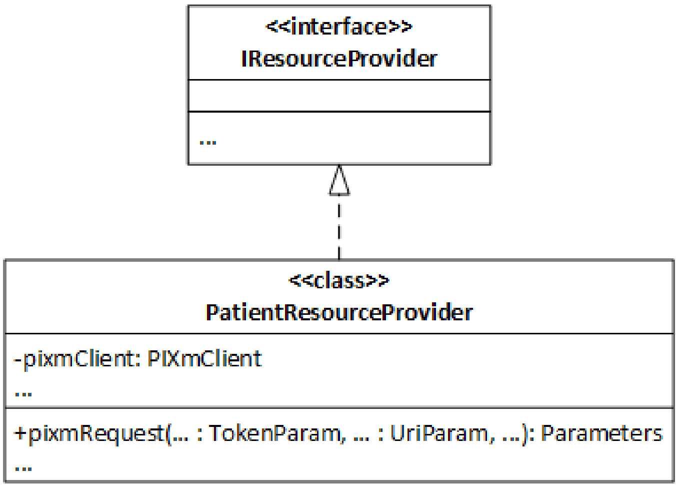

Figure 43 – Patient Resource Provider class diagram

Next, to consume the PIX service from ALERT® PIXPDQ, the PatientResourceProvider uses a client named PIXmTransactionClient, which is defined in the "*.pixpdqm.client" package from the HIE-PIXPDQManager component, to create a PIXmQueryTransaction and set a group of required properties, such as the request's URL and headers to perform further actions regarding the transaction's audit event. The actual creation of FHIR transactions is performed by a class named BaseFhirTransactionClient, located in the HIE-FHIR-Core component's "*.fhir.client" package, which shall be used by every FHIR client to perform a required service (Figure 44).

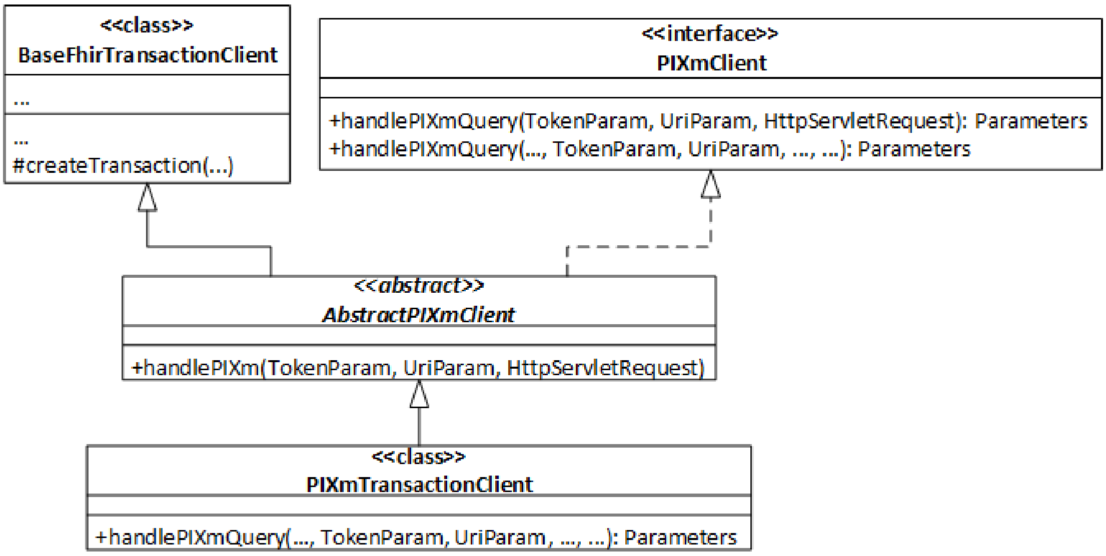

Figure 44 – PIXm Client class diagram

Regarding the transaction, the PIXmQueryTransaction is composed by a PIXmQueryRequest and a PIXmQueryResponse, and adopts ALERT's model, which is required to execute ALERT services (Figure 45). The transaction and its respective request and response are located in three of the HIE-PIXPDQ component's packages, namely in the "*.pixpdqm.transaction", "*.pixpdqm.transaction.request" and "*.pixpdqm.transaction.response" packages.

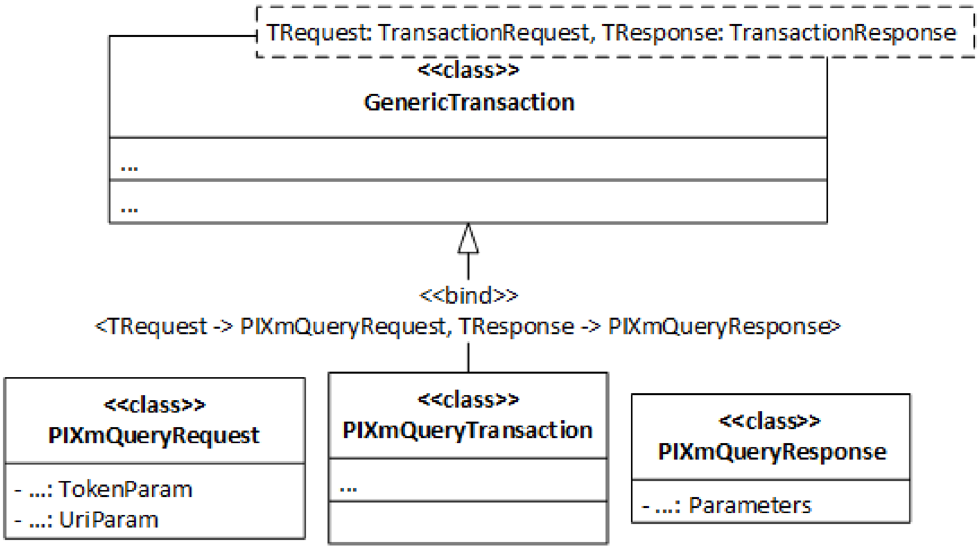

Figure 45 – PIXm Transaction class diagram

The next step of the process relates to the conversion from PIXm model to PIX model, which is performed by using a converter named PIXmQueryRequestToPIXQueryRequestConverter presented in Figure 46. This converter was created in the HIE-PIXPDQ component's "*.pixpdqm.transaction.request.converter" package, to transform the PIXm transaction's request (PIXmQueryRequest) to the corresponding PIX transaction's request, named PatientIdentifiersQueryRequest.

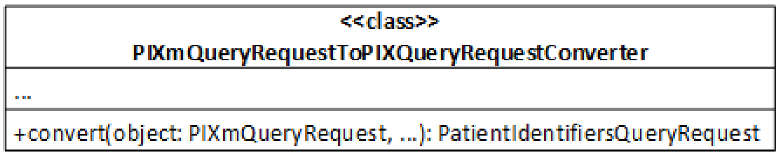

Figure 46 – PIXm Request Converter class diagram

The following step in the process is to execute the PIX service, which, in order to perform additional validations, uses a Transaction Handler named PIXmQueryTransactionHandler (Figure 47), defined in the HIE-PIXPDQManager component's "*.pixpdqm.transaction.handler" package.

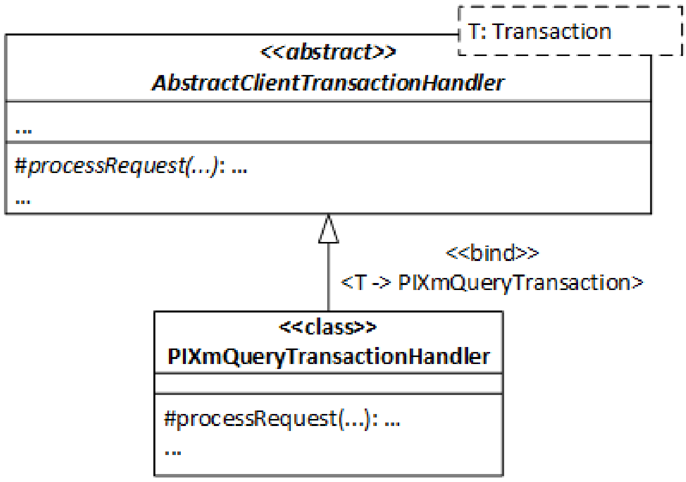

Figure 47 – PIXm Transaction Handler class diagram

The PIXmQueryTransactionHandler uses the PatientIdentifiersQueryRequest to perform the existing PIX query operation in the product, requesting the data to the respective database. If the query was successful and the response from the database returned without any errors, the handler will then return a PatientIdentifiersQueryResponse, which will be converted to a PIXmResponse by the response converter PIXQueryResponseToPIXmQueryResponseConverter (Figure 48).

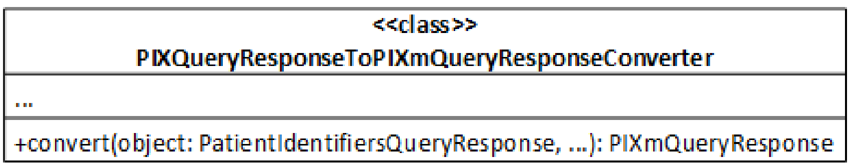

Figure 48 – PIXm Response Converter class diagram

If the response returns a set of search errors, which can be related to required identifiers that were not found during the query, the handler will process the errors and create an OperationOutcome resource which is an obligatory FHIR resource to be returned in any error case situation. Depending on the error, the OperationOutcome fields (severity, type and diagnostic) vary to be in conformance with IHE's PIXm specification regarding error treatment, returning the resource with a specific HTTP status code. The OperationOutcome resource is solely created by the FhirExceptionHandler (Figure 49), which was defined in the HIE-FHIR-Core component's "*.fhir.exception" package for this unique purpose.

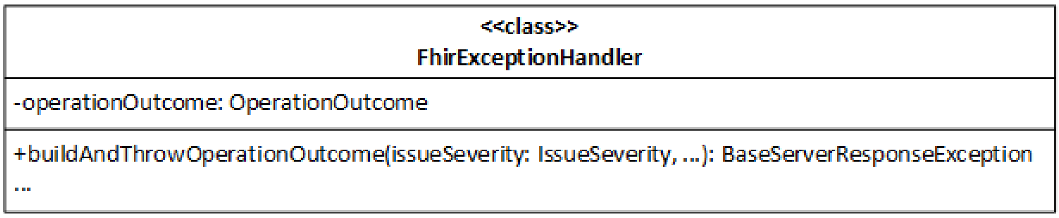

Figure 49 – FHIR Exception Handler class diagram

After converting the transaction's response to the PIXm model, a set of existing observers in the product are notified to perform the audit event which is stored in an audit repository, indicating that the transaction was executed. To successfully take advantage of the HIE's audit observers, each transaction requires an audit trail message builder. Therefore, the message builder PIXmQueryTransactionAuditTrailMessageBuilder was created to construct the audit message in conformance with IHE specification and send it to the audit repository, which is further described in section 5.4, regarding the ATNA profile.

Finally, after the required HAPI FHIR's transformations and validations are concluded, the PatientResourceProvider will return the required FHIR REST resource for the service, which can be either a Parameters or an OperationOutcome resource, concluding the service transaction.

5.2 PDQm

For PDQm profile, the implementation followed the same approach as PIXm since both use the same FHIR resource and the same components. PIX and PDQ business logic are currently grouped in the same components (HIE-PIXPDQ and HIE-PIXPDQManager), which led to an architectural decision to use the same components for PIXm and PDQm. The transactions, requests, responses, converters, handlers and clients for both profiles are grouped in the same packages to simplify the implementation and to follow the same business process and architecture.

To perform PDQm profile's transaction, a new service was added to the PatientResourceProvider which is a FHIR search request represented in Example 12 with the definition of the service method for a Patient resource, along with the expected optional parameters for the transaction.

Example 12 – PDQm service definition

```
@Search()
public IBundleProvider pdqmRequest(@OptionalParam(name=...) ..., ...) {...}
```

As can be seen in the example above, the method receives a set of "@OptionalParam" which is a type of parameter defined by HAPI-FHIR that can only be used for "@Search" operations, along with the "@RequiredParam" and others, which are not used for this service due to the profile's specification and requirements.

Regarding the transaction processing, most of the required classes were added to the packages indicated in section 4.3.3, with a few differences. Unlike PIXm, the received parameters in the PatientResourceProvider were all grouped in a single class named "PDQmParams", located in the HIE-PIXPDQ component's "*.pixpdqm.model" package, since this profile required many parameters for the patient demographics query. Additionally, for this profile a Bundle resource is expected for return, which must contain a set of Patient resources with the demographics information retrieved from the database, and according to the specification, the bundle paging feature is required for an incremental response processing. FHIR paging provides the means to return a bundle with a large amount of results through a set of partitions, by using hypermedia controls to request further results from the first request. Example 13 shows an example of FHIR paging with hypermedia controls for the service's Bundle response.

Example 13 – Partial JSON Bundle response with paging

```
{
    "resourceType": "Bundle",
    "id": "...",
    "type": "searchset",
    "total": 246,
    "link": [
        {
            "relation": "self",
            "url": "http://.../fhir/Patient?..."
        },
        {
            "relation": "next",
            "url": "http://.../fhir?_getpages=...&_getpagesoffset=150
                    &_count=150&_pretty=true&_bundletype=searchset"
        }
    ],
    ...
}
```

To implement the bundle paging feature, a BundlePagingProvider was created to define the FHIR server paging configurations. This provider extends HAPI-FHIR's BasePagingProvider which also implements an IPagingProvider interface to define the paging default size and maximum size, to implement the behavior to store the result list from the first request and to retrieve that same stored list. Figure 50 describes these relationships through the class diagram, which were added in the HIE-FHIR-Core component's "*.fhir.provider.paging" package, since this feature shall be used for every FHIR service that requires paging behavior.

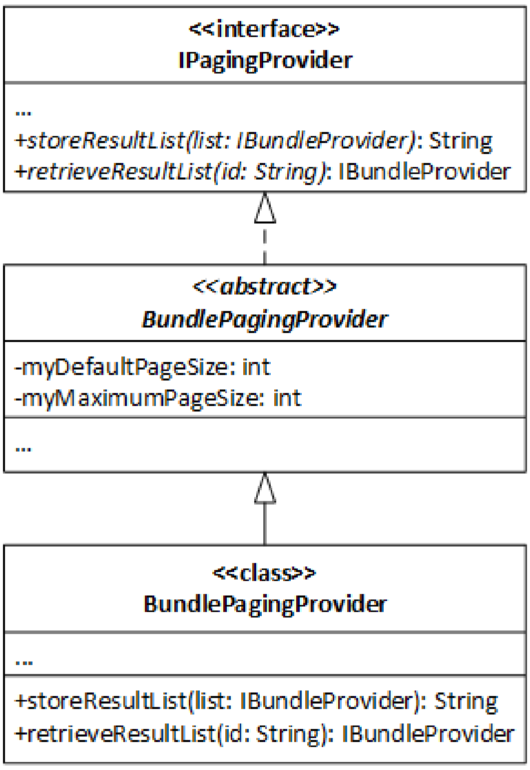

Figure 50 – Bundle Paging Provider class diagram

The provider only delivers the means to access bundle lists that are to be used for paging purposes, but some additional features are still required to be able to return a response bundle with paging URL links to execute them. To return a bundle with paging behavior an object that implements HAPI-FHIR's IBundleProvider is necessary, which contains the required methods to execute the hypermedia links. This way, a PatientBundleProvider was created in the HIE-FHIR-Core component's "*.fhir.provider.bundle" package, which extends an abstract class named ResourceBundleProvider that implements IBundleProvider (Figure 51).

The abstract class was created to define a base behavior to retrieve the resources through the URL links' fields (_getpageoffset and _count) and to provide an easier extension for additional FHIR services to adopt this paging behavior for bundle responses, or to provide their own. Each class that extends the abstract class must implement one abstract method that defines the page size for that service (preferredPageSize()), which is customizable for each existing service.

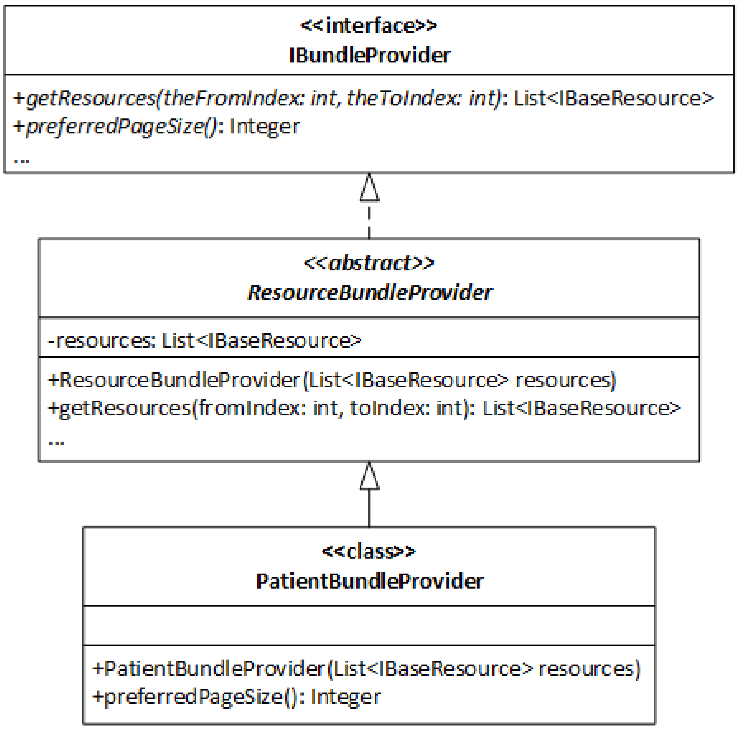

Figure 51 – Patient Bundle Provider class diagram

5.3 MHD

This section describes in detail the implementation approaches for MHD transactions through UML class diagrams to explain the implementation involved in document transactions, as well as the conjunction between the MHD profile and the XDS profile, which defines the product's existing document services.

5.3.1 Find Document Manifests

The implementation approach for the Find Document Manifests transaction was like PIXm and PDQm architectural design since each FHIR transaction uses similar objects to perform the service. For the FHIR endpoint, the DocumentManifestResourceProvider class was created in the HIE-FHIR component's "*.fhir.provider.resource" package (Figure 52), being also a search operation, like the PDQm search request in PatientResourceProvider.

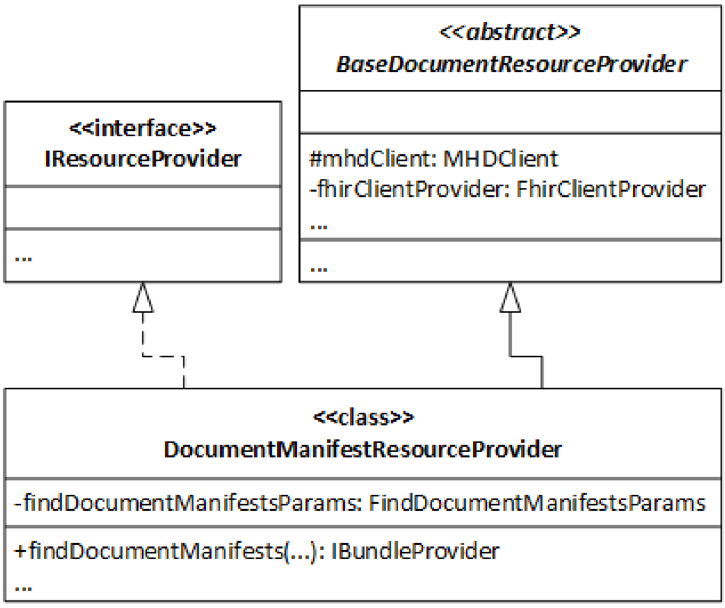

Figure 52 – Document Manifest Resource Provider class diagram

The DocumentManifestResourceProvider extends a BaseDocumentResourceProvider since the MHD Resource Providers will all use the same client to perform different requests. This provider also requires the execution of other implemented FHIR services, in specific a service to retrieve a Patient REST resource using an id parameter. According to IHE's MHD specification, the service request requires a minimum of two parameters to successfully perform the FHIR operation, namely, a patient identifier and a status code. The patient identifier parameter can be either the actual value of the identifier (Example 14) or a reference to a Patient resource (Example 15), which in the last case requires an additional FHIR service to request the Patient reference's data and retrieve the associated identifier.

Example 14 – DocumentManifest HTTP GET request with patient identifier

```
http://.../fhir/DocumentManifest?patient.identifier=urn:oid:1.1.1|22&status=current
```

Example 15 – DocumentManifest HTTP GET request with patient reference

```
http://.../fhir/DocumentManifest?patient=Patient/123&status=current
```

To call FHIR services from internal requests, a FhirClientProvider was created in the HIE-FHIR-Core component's "*.fhir.client" package to initialize new FHIR clients that implement HAPI-FHIR's IBasicClient interface. This way, these clients will be able to request defined services in existing FHIR Resource Providers, by defining in the class interface the service's exact name, type of request annotation, and parameters. To implement the service that retrieves a Patient resource, a PatientFhirClient interface, that extends the IBasicClient interface, was created in the same package as the FhirClientProvider with a method named "getPatientById", which is a defined "@Read" service from PatientResourceProvider that holds an id parameter (Example 16).

Example 16 – PatientResourceProvider read service

```
@Read()
public Patient getPatientById(@IdParam IdType id) { ... }
```

When a Patient resource reference is provided as a parameter, the DocumentManifestResourceProvider requests the FhirClientProvider to create the client PatientFhirClient, using that client to perform the FHIR request (Figure 53).

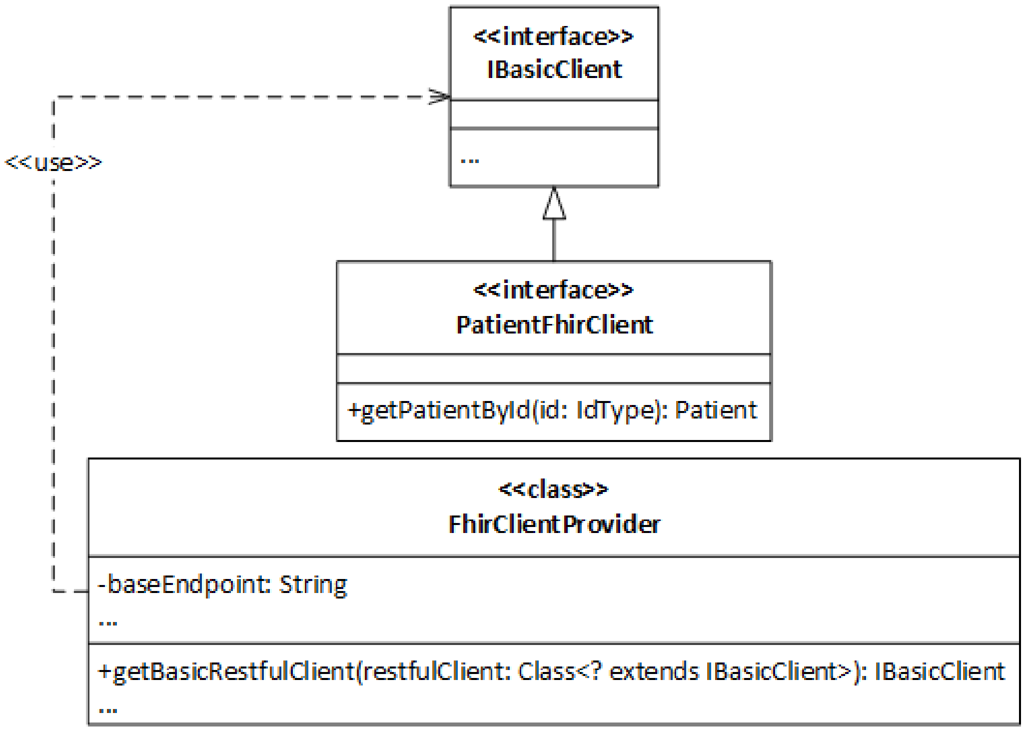

Figure 53 – HIE-FHIR-Core client provider

After performing the initial HAPI FHIR validations and transformations, and, if required, performing the additional FHIR services, the DocumentManifestResourceProvider uses the MHDTransactionClient class, the main purpose of which is to provide the services for MHD's related operations. The client was added to the HIE-DocumentSharing component's "*.mhd.client" package, since this component holds the services and clients for document related operations (Figure 54).

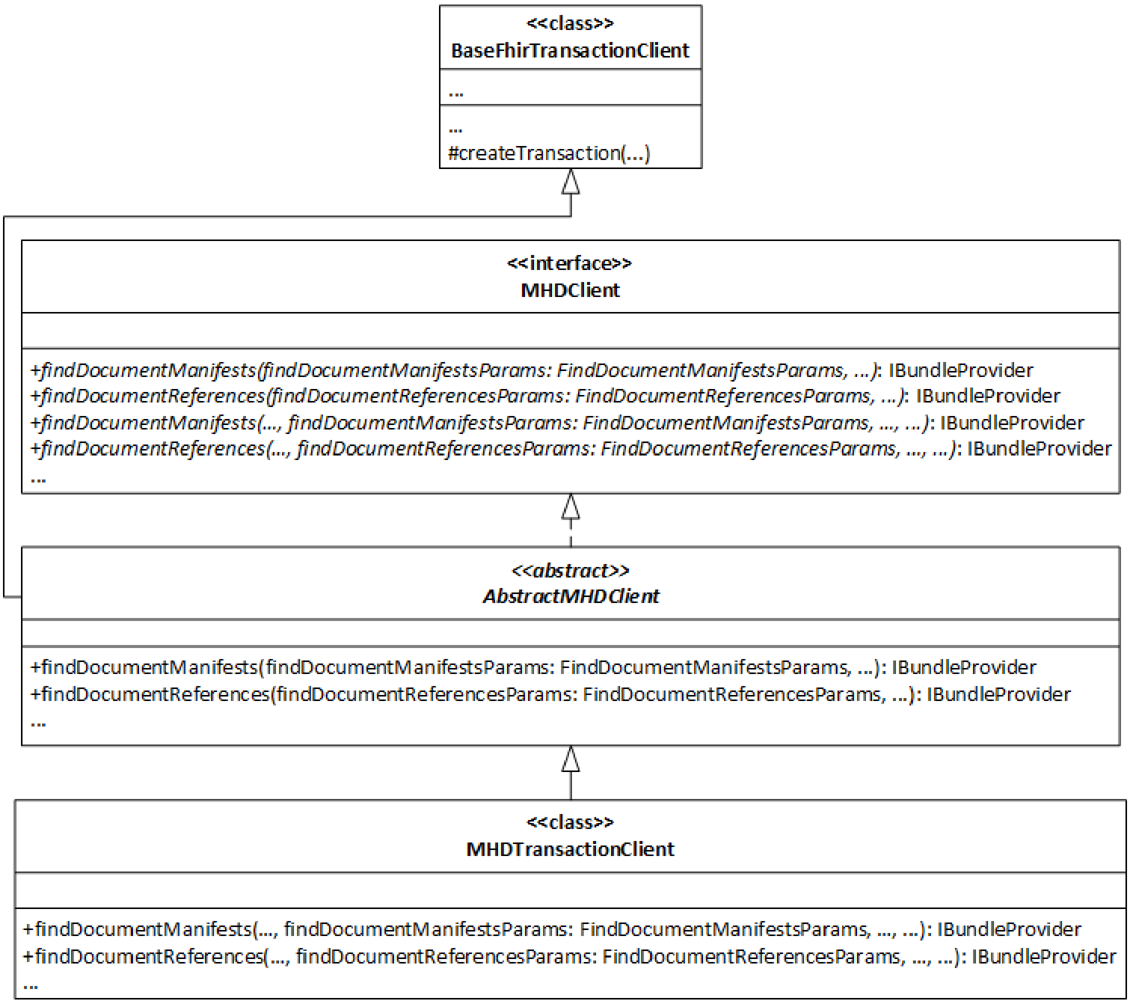

Figure 54 – HIE-DocumentSharing client classes

For the following operations in the process, regarding the transaction's conversions and the ALERT's XDS service execution, the MHD profile uses an almost identical set of classes as the PIXm profile, which were also logically grouped in the same group of packages according to MHD's package structure presented in section 4.3.4.

The MHDTransactionClient uses a FindDocumentManifestsTransaction, which is composed by a FindDocumentManifestsRequest and a FindDocumentManifestsResponse. Following the same logic as the PDQm request class, the FindDocumentManifestsRequest contains a FindDocumentManifestsParams object (located in the HIE-XDS component's "*.mhd.model" package) with all the required fields to perform the service, while the FindDocumentManifestsResponse contains a DocumentManifestsBundleProvider object (located HIE-FHIR-Core component's "*.fhir.provider.bundle" package) which will return a Bundle resource with a set of found DocumentManifest resources (Figure 55).

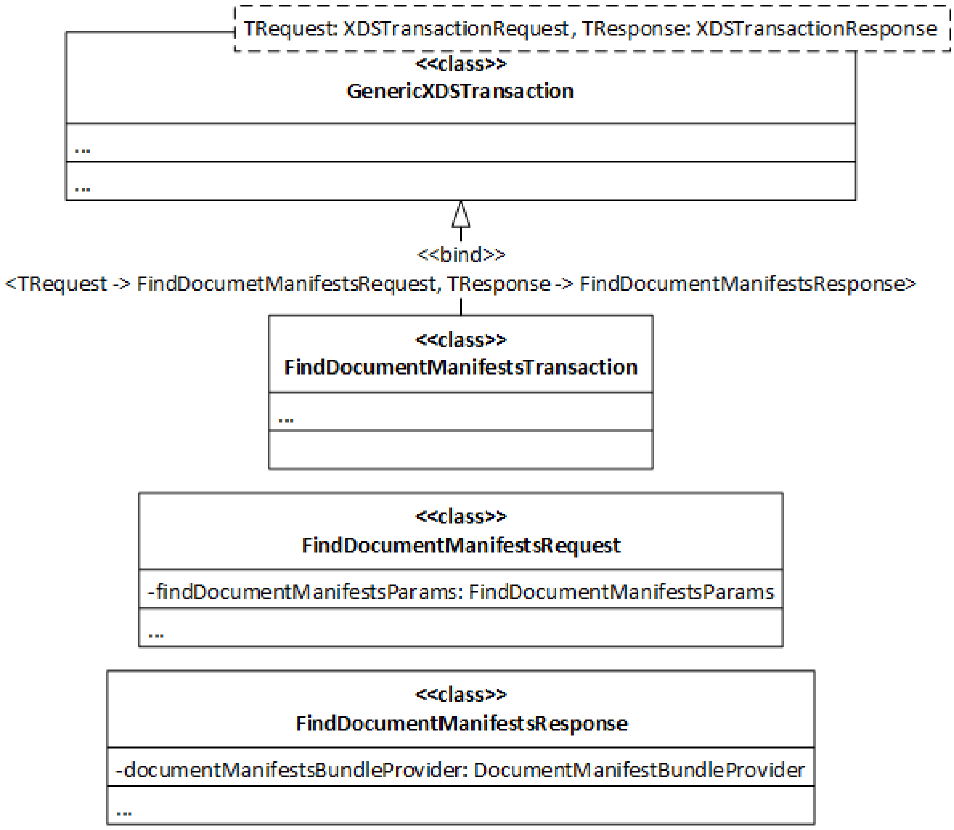

Figure 55 – Find Document Manifests Transaction class diagram

Considering that the FindDocumentManifestsTransaction is similar to the XDS RegistryStoredQueryTransaction, the succeeding operations in the process, regarding the request's conversion, the use of a transaction handler to perform the database operation, and the response's conversion, are performed by the following classes, which are also presented in Figure 56:

- **FindDocumentManifestsRequestToRegistryStoredQueryRequestConverter:** the transaction request's converter, the main responsibility of which is to convert the request parameters to the respective request parameters from the XDS's RegistryStoredQueryRequest;

- **FindDocumentManifestsTransactionHandler:** the MHD transaction handler used to validate and execute the database query for the XDS's Registry Stored Query transaction, which was added to the HIE-Registry component's "*.mhd.transaction.handler" package since this component contains all the major business classes required for the XDS transaction;

- **RegistryStoredQueryResponseToFindDocumentManifestsResponseConverter:** the transaction response's converter, the main responsibility of which is to convert the RegistryStoredQueryResponse to the MHD's FindDocumentManifestsResponse.

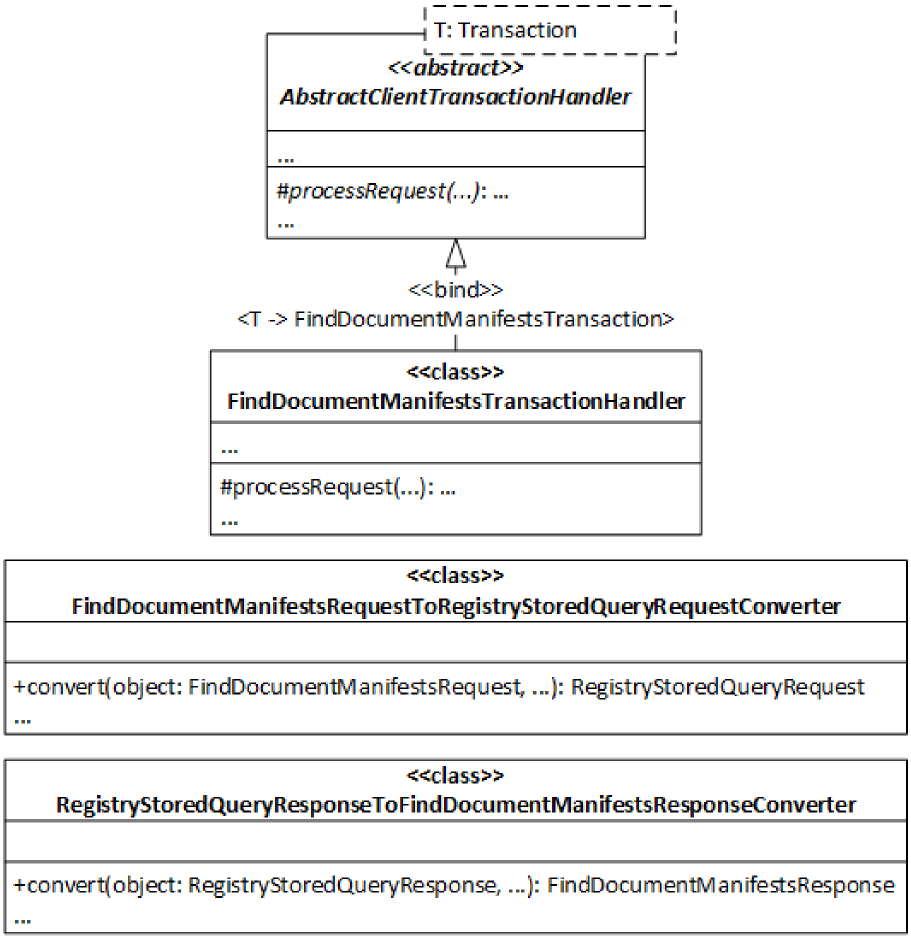

Figure 56 – Find Document Manifests Response Converter class diagram

The final tasks follow the normal approach like the PDQm process, with HAPI FHIR response validations and conversions to the request's format, with no additional classes required.

5.3.2 Find Document References

The Find Document Reference transaction is identical to the Find Document Manifests, with small variations regarding the conversions, and the request process implementation for the transaction handler. Just like the previous one, a set of corresponding classes for this transaction were created in the respective packages, since both Find Document Manifests and Find Document References transactions are based in XDS's Registry Stored Query.

5.4 ATNA

As referred in section 4.1.4, the ATNA profile is already implemented in the product to be used in conjunction with existing profiles, and, even though FHIR resources are not used, audit events are still required for FHIR profiles according to IHE's specification. Currently, audit events are registered via observers that are notified when a transaction is executed in ALERT® HIE and require a class for each transaction that can construct an audit message.

To adopt this feature for the remaining FHIR transactions, an audit trail message builder was created in each one of the components that contain FHIR transactions. For PIXm and PDQm profiles, the message builders were defined in HIE-PIXPDQ component's "*.pixpdqm.audittrail" package for both PIXmQueryTransaction and PDQmQueryTransaction, respectively (Figure 57). Each message builder extends an abstract class that implements the audit trail message creation, since the audit message is similar for both profiles, with small variations in the content values, while each message builder provides their patient ids for the audit message based on their transaction.

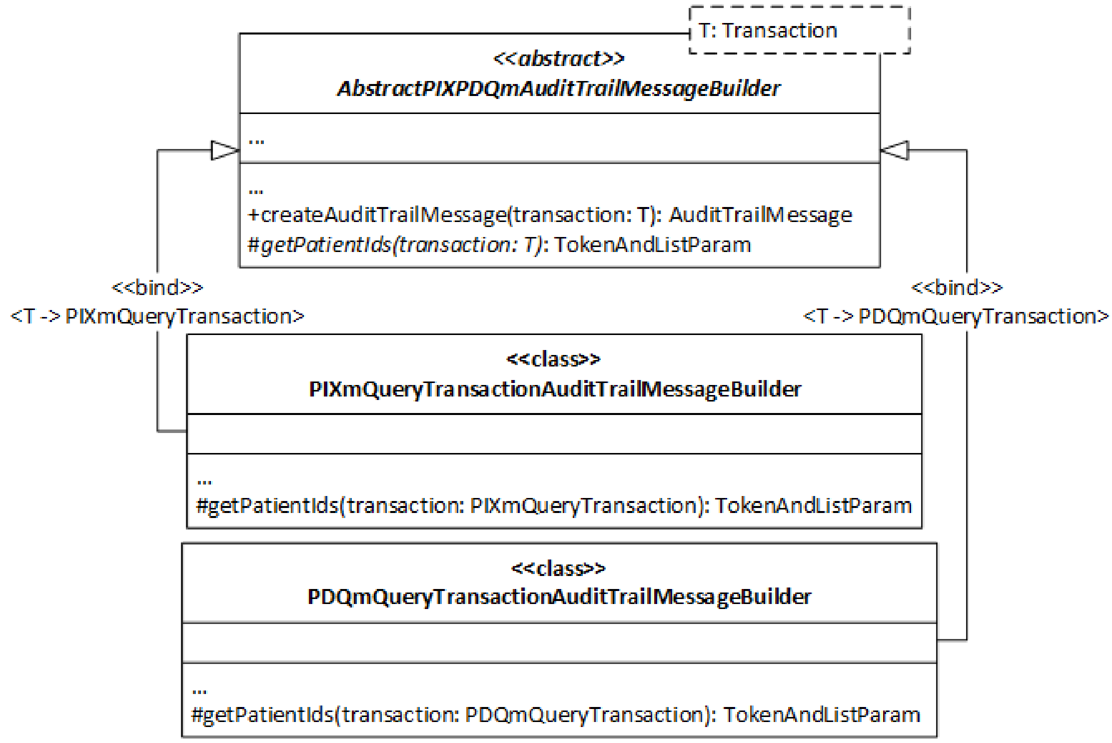

Figure 57 – Audit Trail for HIE-PIXPDQ component

Regarding the MHD profile, the message builders for the transactions FindDocumentManifestsTransaction and FindDocumentReferencesTransaction were defined in the HIE-XDS component's "*.mhd.audittrail" package (Figure 58). Like PIXm and PDQm, the MHD profile also has a message builder for each transaction, where each extends an abstract class that contains the implementation for the audit message creation. Additionally, each message builder provides the request parameters used for their transaction to retrieve additional information for the audit trail event.

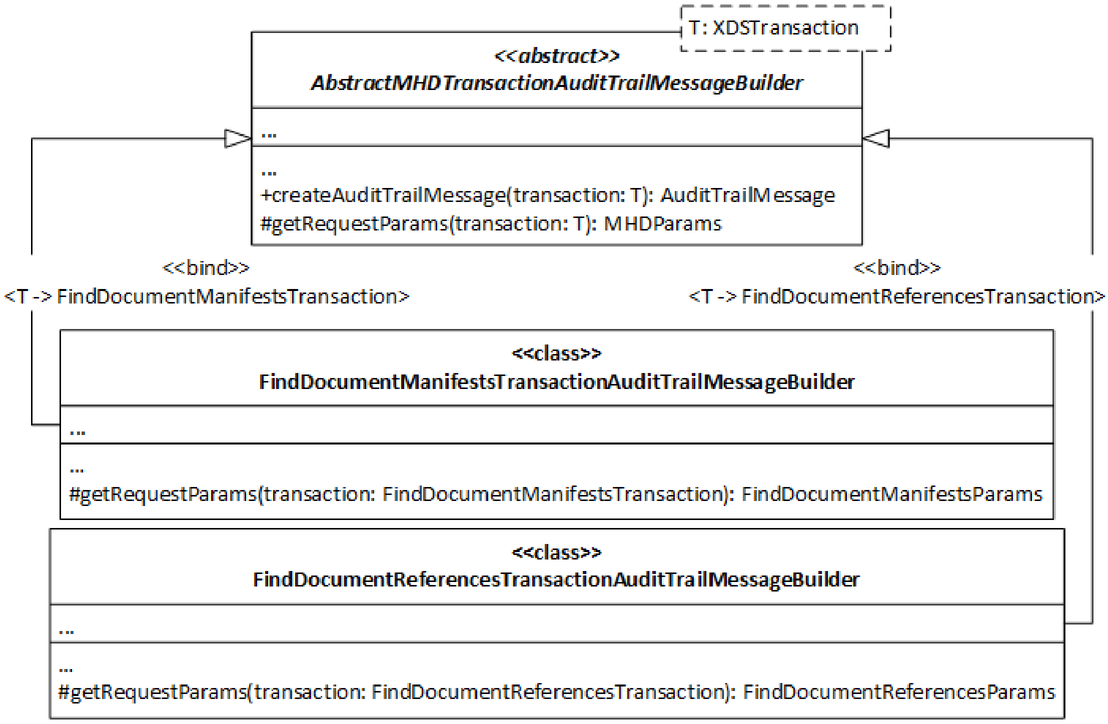

Figure 58 – Audit Trail for HIE-XDS component

5.5 IUA

A set of additional security features were required for the IUA profile implementation, which mainly involved the adoption of JSON Web Token as an authorization method for FHIR request services. This profile involved two transactions, one respective to the web token's retrieval and the other respective to its use in a FHIR request. Since the focus is for HIE to support FHIR and new IHE profiles, only the aspects involved with the server-side requests were taken into consideration, since the transaction to retrieve the web token involves client-side requirements which are related to other products.

To validate every FHIR request to the server, the most logical approach was to use an interceptor feature provided by HAPI-FHIR. An interceptor is useful for logging aspects, or for security measures to be applied before the execution of a request, which was used in this case to validate incoming requests and to verify the token's legitimacy.

First, the IuaServerInterceptor was created in the HIE-Security component's "*.security.iua", and extends HAPI-FHIR's InterceptorAdapter, enabling the server to intercept incoming FHIR requests, passing them through IuaServerInterceptor first, which in turn will validate the request before continuing with the actual service (Figure 59). The interceptor verifies if the request header "Authorization" exists, which is a mandatory header for every request, and if its content follows the Bearer authentication scheme (Authorization: Bearer <token>).

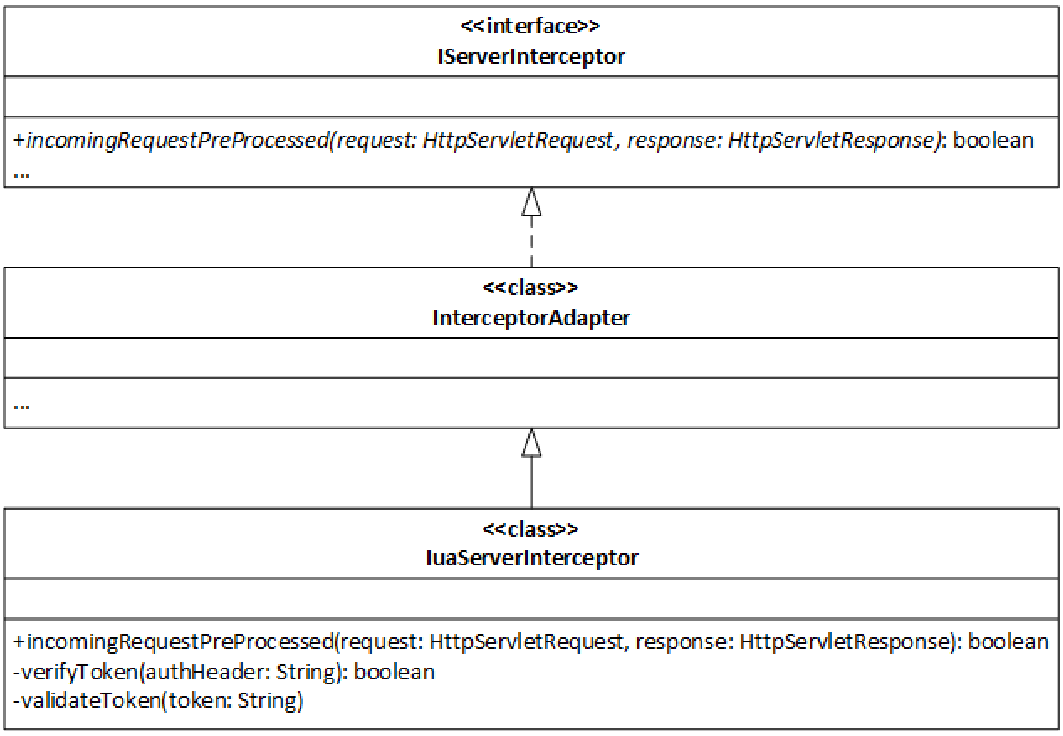

Figure 59 – Iua Server Interceptor

As mentioned in the profile’s specification, there still isn’t an official authorization code to be used for the authorization header. Until there is, the code to be used by IHE will be IHE-JWT, for example: “Authorization: IHE-JWT fJHGjkagb1[…]88hsgThGwsj”. The interceptor proceeds with further validations to verify the token’s signature, validate the claims and check the user’s permissions.

Another point that required attention was the validation of internal client requests in HIE. Usually IHE’s transactions, such as the ones from MHD, require additional FHIR services to retrieve a Patient resource using references contained in a DocumentReference resource. This involves another request to the server endpoint for the respective resource which will then trigger the IuaServerInterceptor, consequently requiring the same token from the first request to be used in the following requests. To solve this problem a new interceptor was created which, instead of intercepting incoming server requests, will intercept internal outgoing client requests, to incorporate the token in the request. Figure 60 represents a class diagram with the required classes to intercept and incorporate the token in these requests, created in HIE-FHIR-Core component’s “*.fhir.security” package.

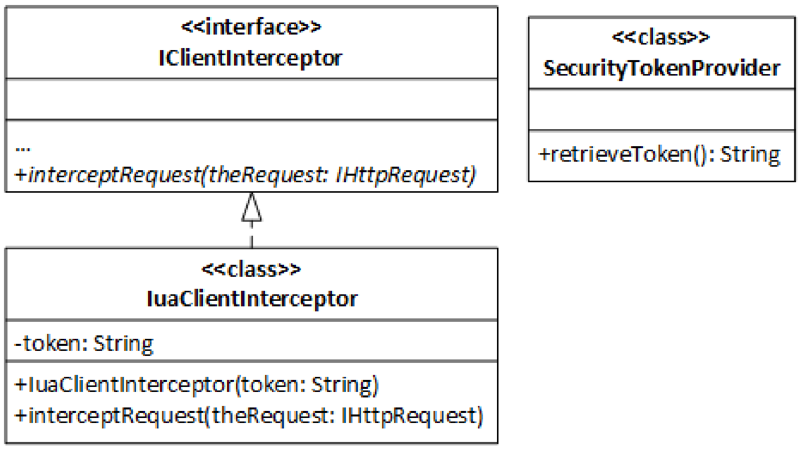

Figure 60 – Iua Client Interceptor

The SecurityTokenProvider is used by the client to retrieve the stored token and send it to a new IuaClientInterceptor instance, which in turn is registered as an interceptor for that client before proceeding with the request. After capturing the request, the interceptor adds an authorization header to it with the respective token, proceeding with the normal FHIR service execution, which passes through the IuaServerInterceptor to perform the necessary validations.

5.6 Summary

To sum up, throughout this chapter a set of classes are described for each one of the implemented profiles' transactions. The creation of these classes is in accordance to the previous analysis and design described throughout chapter 4. Section 4.3 explained the architectural design approach to add new logic regarding the standard's specification to the existing architecture, while section 4.3 explained the business workflow designed for each profile. These sections provided an explanation of the major decisions taken to set up the product with a suitable logic to implement the transactions according to the integration profiles' specification and the defined business workflows. Each one of the classes described in this chapter were added to fulfill a single purpose required for the process stages, and to distribute responsibilities throughout the solution's modules, considering their package logic and the IHE profiles related to them.

6 Experiments

In this chapter the experiments' goals, the setup planned, and the procedures established for each experiment are described, along with a description for those experiments' results and analysis.

6.1 Goals

The performed experiments take into consideration some of the requirements defined in section 3.3:

- The FHIR® services are expected to obtain an equal or better performance regarding the existing services;
- The implementation should respect FHIR® specification and IHE's implementation requirements.

Considering these project requirements and the developed FHIR features, a suite of experiments serves as the means to obtain some evidences allowing conclusions regarding the hypothesis that the HL7 FHIR standard implemented services possess an equal or better performance compared to the previous existing services based on other HL7 standards, while respecting IHE's criteria.

To check the defined hypothesis, the following set of experiment goals were defined:

- **Services' Conformance experiments:** since the developed features should respect IHE's criteria, a set of experiments need to be performed to validate if the new services respect IHE's profile specifications;

- **Response Time experiments (Performance):** to analyze the services performance, a set of response time experiments were conducted and compared to the equivalent existing services that use a previous HL7 standard, under the same environment conditions.
- **Response Size experiments (Performance):** to complement the performance experiments, response size experiments were also conducted to analyze if the response size influences the services' response time.

6.2 Setup

For each of the defined goals, a test environment was defined considering the experiments' requirements and existing resources to perform them.

6.2.1 Services' Conformance

For the services' conformance, a few existing IHE tools were chosen to fulfill this objective. Since IHE is responsible for certifying the profiles implementation, the organization provides free tools that can be used by the developers to test their integration profiles' services, regarding their requests and responses (IHE, 2018).

6.2.1.1 Test Environment

The IHE's Gazelle PatientManager tool (Figure 61) was chosen to test the PIXm and PDQm requests, since it was specifically designed for these two profiles.

The Gazelle PatientManager is a web tool that acts as a client-server simulator for the PIXm and PDQm services, where the client is provided by the web tool itself, while the server can be chosen by the user from a list of available online supported servers (IHE International, 2018a). Since it's not possible to integrate the developed server in the Gazelle PatientManager tool, the online IHE FHIR Server will be used to simulate and validate service requests (in specific the requests' URL), by using the developed services' parameters and values for the experiments.

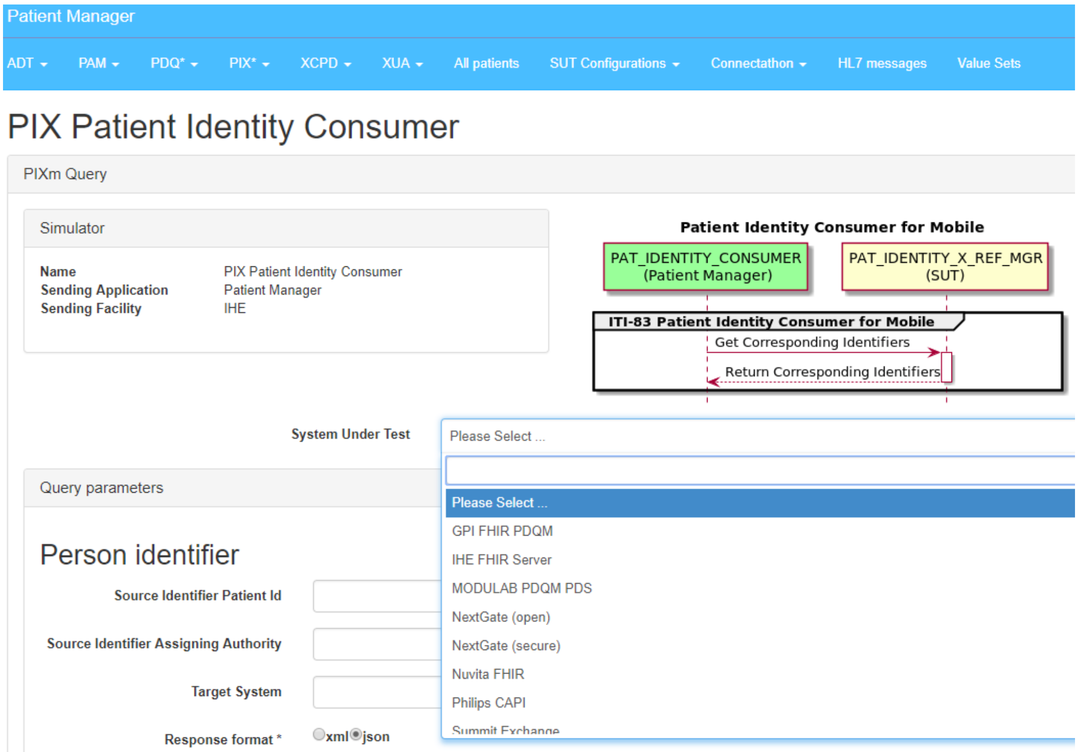

Figure 61 – Gazelle PatientManager tool interface (IHE International, 2018b)

The IHE's External Validation Service Front-end (EVSClient) tool was chosen to test the PIXm, PDQm and MHD's responses (Figure 62).

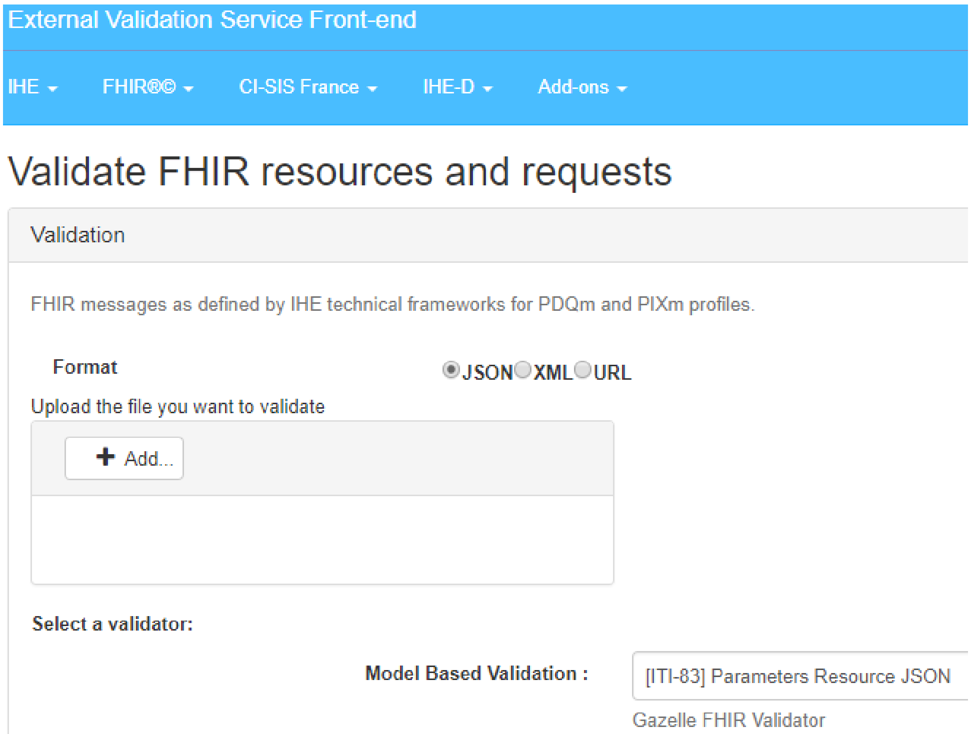

Figure 62 – EVSClient tool interface (IHE International, 2018c)

The EVSClient is a web tool that validates external files (provided by the person who's using the tool) against IHE's criteria (IHE International, 2018d). By observing the file structure and content, which may be a JSON or XML file, the tool can verify if the file structure respects, not only FHIR's resource criteria, but also IHE's profile criteria regarding mandatory FHIR resources, fields and values that shall be returned as the service's response.

6.2.1.2 Available Data

Some of the conformance experiments could not be performed due to the lack of existing IHE tests. There are no available tests, neither for the IUA profile, nor for the QEDm profile, and regarding the MHD transactions, only tests for the services' responses are available. The only suitable tool to test the MHD profile is the EVSClient, and from the available options for test, there are no existing choices that can validate the MHD transactions' requests.

6.2.2 Performance

For the response time experiments, a set of performance tests were developed to retrieve the services total time of execution, which covers the period between the moment where the client performs the request, and the moment where the server returns a response. A similar approach was applied for the response size experiments, but with different parameters to obtain responses with variable sizes.

6.2.2.1 Test Environment

To test the developed services against the existing ones' performance, a set of environment conditions were defined for the experiments. Both services, regarding FHIR® and the HL7 v3, shall use the same parameters and values to perform the profiles' requests, and both shall be executed in the same development server. The involved services shall cover all the workflow process tasks, such as security and audit features, according to their implementations.

6.2.2.2 Available Data

Since the requests are performed to the same development server, and since the developed services use functionalities that were already implemented in the server, the experiments will retrieve the same data available in the respective database.

6.3 Service's Conformance Experiments

This section describes the conducted experiments for the service's conformance, describing the test procedures, presenting the test results and analyzing the results' values.

6.3.1 Procedures

For each IHE profile developed, a set of test requests were defined to be validated according to IHE's metrics and evaluation methods.

6.3.1.1 Test Requests

For the PIXm ITI-83 transaction (Mobile Patient Identifier Cross-reference Query) and PDQm ITI-78 transaction (Mobile Patient Demographics Query), two conformance tests were performed for each:

- **Test the service request will all the parameters:** To verify the services' request conformance, the Gazelle PatientManager was used to perform a HTTP GET request with all the supported parameters for the developed transactions (a PIXm request is presented in Example 17);

- **Validate the service response with a server's response:** For the services' response, the EVSClient was used to validate a response file from the developed server (Example 18) against IHE's criteria.

Example 17 – PIXm request applied for IHE validation

```
.../fhir/Patient/$ihe-pix?
sourceIdentifier=urn:oid:1.3.6.1.4.1.33233.2.2.5.4.12|71710460050002
&targetSystem=urn:oid:1.3.6.1.4.1.33233.3.1.1.1&_format=json
```

Example 18 – PIXm response applied for IHE validation


```
{
    "resourceType": "Parameters",
    "parameter": [
        {
            "name": "targetIdentifier",
            "valueIdentifier": {
                "use": "official",
                "system": "urn:oid:1.3.6.1.4.1.33233.3.1.1.1",
                "value": "81058"
            }
        }
    ]
}
```


For the MHD transactions ITI-66 (Find Document Manifests) and ITI-67 (Find Document References), one conformance test was performed for each:

- **Validate the service response with a server's response:** similar to the PIXm and PDQm transactions, the MHD services' response files were validated using the EVSClient.

6.3.1.2 Metrics

In this case, the IHE's criteria has been adopted as metrics, which are in accordance to the implementation requirements described in the profile's technical specification. Unfortunately, the specific metrics of evaluation are not stated by IHE in these tests, either way, the test results provide the user with information regarding some of the profile's criteria and identified issues.

6.3.1.3 Metrics Appraisal

The metrics' appraisal means are exclusively defined by the IHE tools, where the validations are performed based on a set of assertions that are linked to entities. These assertions are covered in test rules and models, and can also be visualized in IHE's Assertion Manager (Figure 63) which provides information for a profile criterion, namely the predicate, the document's section and page where it can be found, the prescription level, and the status. The profile's document can also be consulted in the same page which is also presented along with the assertion information (full image can be found in Annex B).

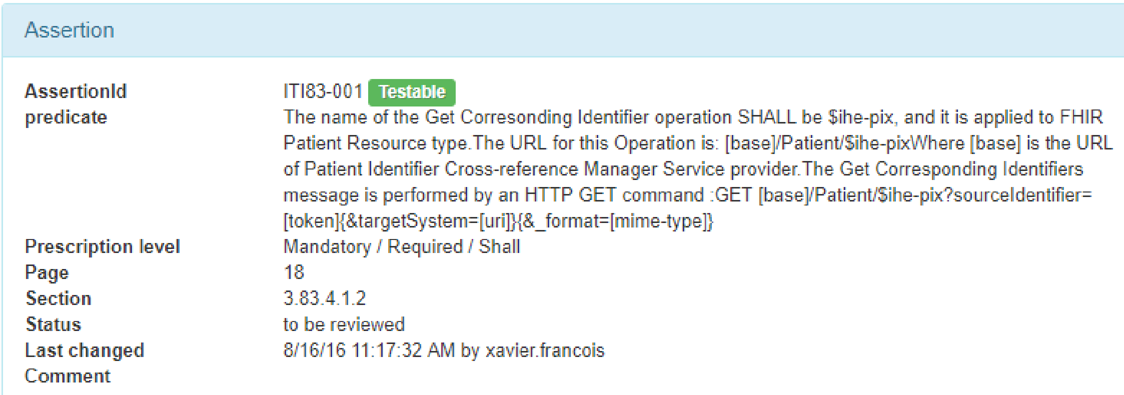

Figure 63 – IHE's Assertion Manager (IHE International, 2018e)

6.3.2 Test Results

The figures presented in this section illustrate the tools' results for each one of the respective profiles.

6.3.2.1 PIXm

Figure 64 presents Gazelle PatientManager's test results for the PIXm request, displaying the URL used to perform the request and the summary information regarding the validation results.

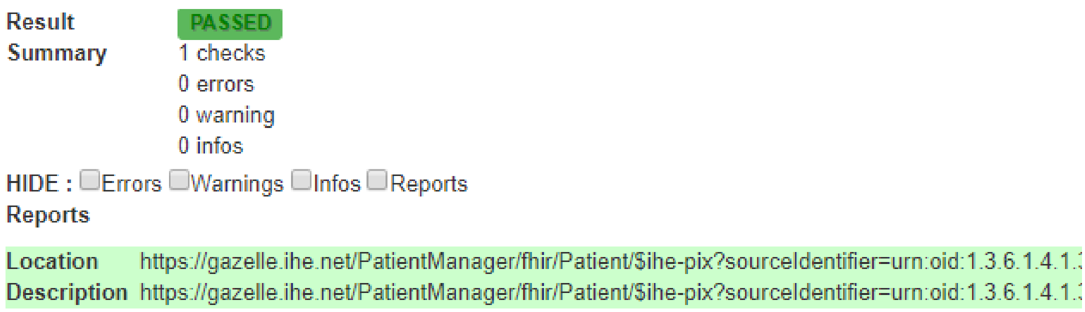

Figure 64 – Gazelle PatientManager results for PIXm request (Full image in Annex B)

To validate the PIXm response, the response JSON file (previously presented in section 6.3.1.1) from the developed FHIR service was imported to the EVSClient tool, the results of which can be visualized in Figure 65.

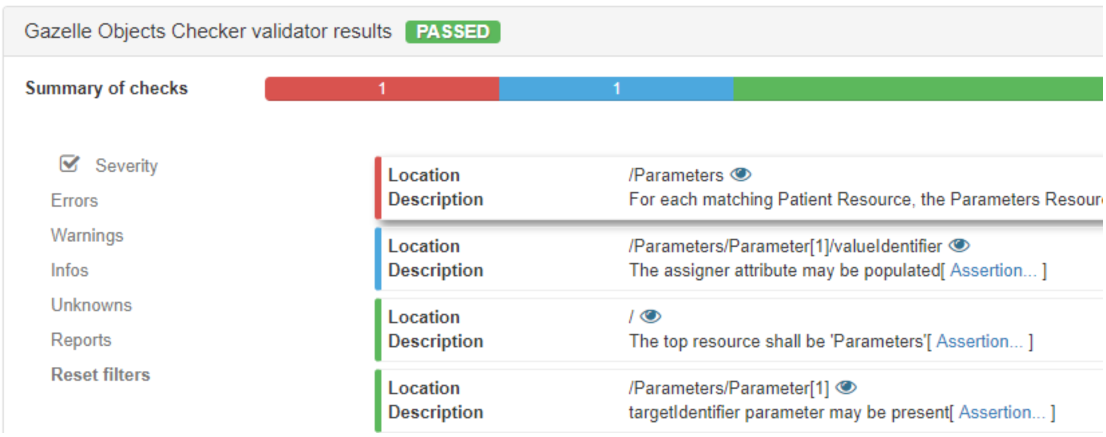

Figure 65 – EVSClient results for PIXm response (Full image in Annex B)

6.3.2.2 PDQm

Once again, the Gazelle PatientManager tool was used to test the request for the PDQm profile, the results of which are presented in Figure 66. The test was performed with all the provided fields, which are also used for PDQm requests in the developed server.

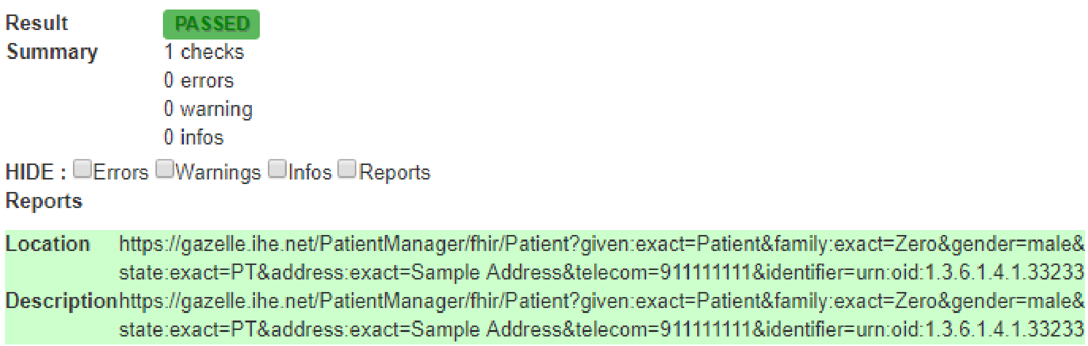

Figure 66 – Gazelle PatientManager results for PDQm request (Full image in Annex B)

Using EVSClient, a PDQm response from the server was validated to check its conformance with the profile's criteria (Figure 67), the content of which is a Bundle resource with a set of Patient resources.

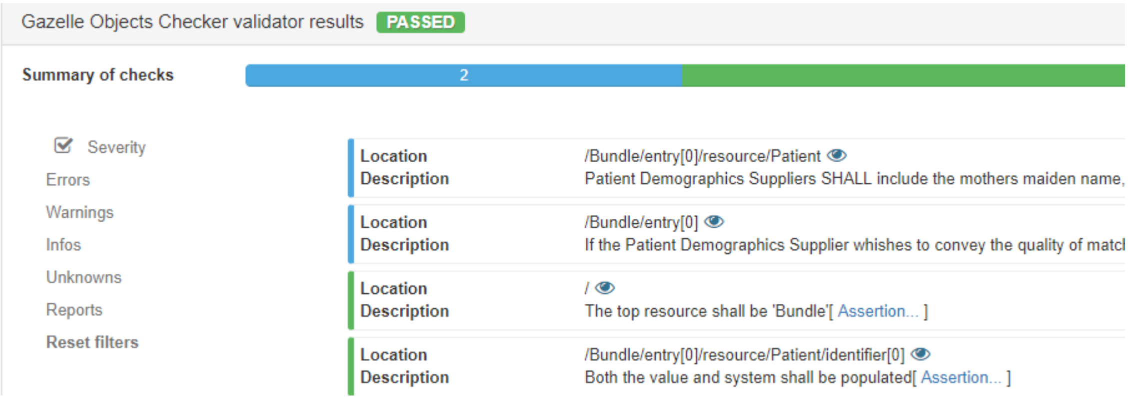

Figure 67 – EVSClient results for PDQm response (Full image in Annex B)

6.3.2.3 MHD

Figure 68 shows the EVSClient test results for the response validation regarding the ITI-66 transaction (Find Document Manifests), which, unfortunately, does not provide any additional info regarding the criteria, indicating only that the validation was a success.

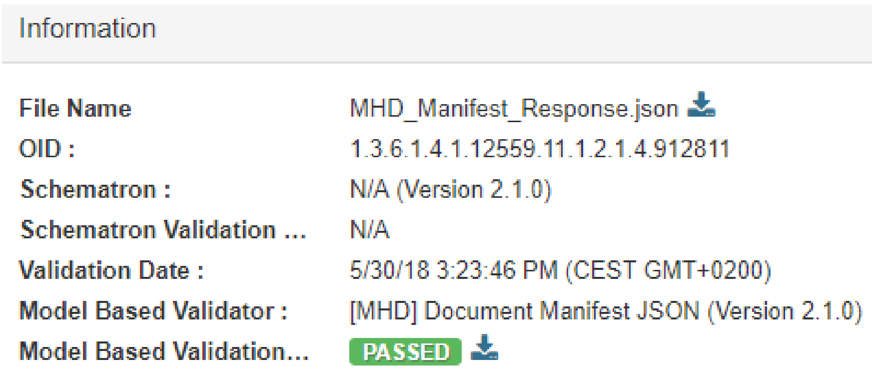

Figure 68 – EVSClient result for MHD's ITI-66 response

Like the previous test, the ITI-67 transaction (Find Document References) response also passed in the EVSClient validation but did not present any additional information (Figure 69).

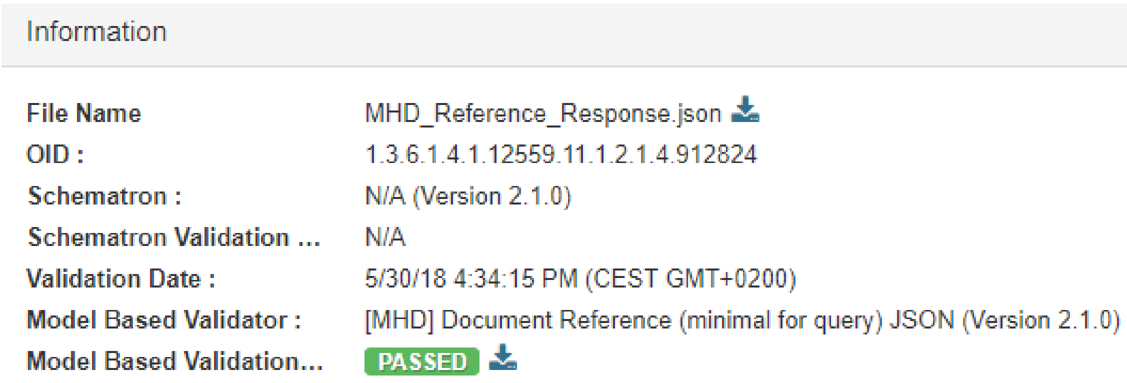

Figure 69 – EVSClient result for MHD's ITI-67 response

It's also relevant to refer that these validations passed through a series of iterations to correct the previous errors, although the majority of those errors were related to some field values that did not respect the required nomenclature from other standards (LOINC and SNOMED CT). Since the results were obtained from a development environment, some of the values regarding the required fields were not according to some of the standards official codes, which caused some validation errors. Figure 70 shows an example of these errors for the ITI-67 transaction (Find Document References) response.

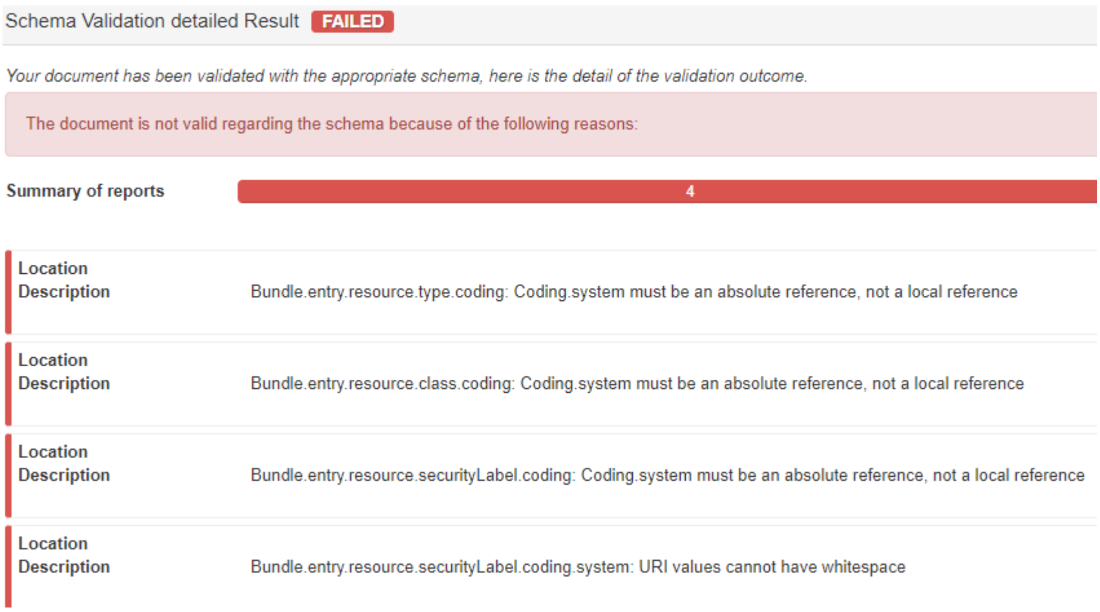

Figure 70 – IHE's Validation Errors for MHD's ITI-67 Response (Full image in Annex B)

6.3.3 Results Analysis

In this section, each one of the profiles' conformance test results is analyzed and discussed, considering the tools' information.

6.3.3.1 PIXm

For the request results, and according to the Gazelle PatientManager, the request URL format and content passed in IHE's server test. This URL was also tested in the developed FHIR server with the same parameters and values, which successfully responded to the request.

For the response results, the EVSClient validated the response file with success, presenting one error, one info, and five success checks. The error states that for each matching Patient resource, a parameter with name "targetId" shall be included in the Parameters response, but this caused a bit of controversy since this response does not match a Patient resource. According to what is stated in the PIXm technical specification "The Patient Identifier Cross-reference Manager returns Patient Identifiers and can optionally also return Patient Resource References that are associated with the identifier provided by the Patient Identifier Cross-reference Consumer …" (IHE ITI Technical Committee, 2017a, p. 19). Furthermore, it also states that "For each matching identifier, the Parameters Resource shall include one parameter element with name="targetIdentifier". For each matching Patient Resource, the Parameters resource shall include one parameter element with name="targetId"." (IHE ITI Technical Committee, 2017a, p. 21).

Considering that an element "targetIdentifier" was returned in the server's response and successfully validated in the test, and the specification indicates that the return of a Patient Resource Reference (which shall include a "targetId" element) is optional, it can be identified a criteria incoherency regarding the identified error. Either way, the remaining assertions and validations passed successfully, which overall states that the server's response file is in accordance to the PIXm specification.

6.3.3.2 PDQm

Just like the PIXm profile, for the PDQm profile, the URL and the parameters' values used for the test were also used in the developed FHIR server, which successfully returned a response. Since all of the parameters for this request are optional, the request was performed with all of the available test parameters, using values that provided some responses in the FHIR server, presenting a successful result in the Gazelle PatientManager tool.

For the PDQm response, the EVSClient tool presented two info checks and five success checks. The two info checks provide some criteria that shall be respected in certain situations, which in this case are the inclusion of the mother's maiden name if known, and the inclusion of a match score attribute if desired. Although the test approved the current server's response, the mother's maiden name field was not used in the response, because of some factors that caused doubts during the development phase. Since this field is directly related to a Pediatric Demographic Option defined in the PDQm technical specification, which can be or not supported by the server, it was not clear enough if, even not supporting this option, the server was obligated to return this field if known. Due to this, this field shall be discussed in further developments.

6.3.3.3 MHD

Regarding both MHD responses, after passing through some corrections, both passed successfully in the EVSClient validations. The results presented by the tests did not provide any additional information though, comparing to the PIXm and PDQm responses which indicated the error, info and success checks. But, as explained in section 6.3.2.3, these tests passed through a series of iterations to correct some of the presented error checks, which for example, indicated that URI values cannot have whitespaces or that none of the codes provided for the security labels were values from the FHIR's security label value set.

All of these errors occurred due to some values that were not according to the standard's requirements, since they came from a development environment database, where some of the stored data might not be fully in conformance with the standard's required values. Nonetheless, these values were corrected according to the standard, to validate if the file's structure respected IHE's specification.

6.3.3.4 Overall Observations

Some of the conformance tests specified a few issues regarding the response files, whereas no issues were pointed by the tools for the requests, while using all the available parameters.

Although some of the pointed issues created some doubts while considering the respective profiles' specification, these issues shall be reviewed and studied with more detail in further analysis to guarantee that the design and the implemented features are in total accordance with the profiles' requirements. Nonetheless, the IHE tools approved with success the response files, based on the developed server's responses, which indicates that the implemented services are following a correct approach regarding the responses format and content.

6.4 Response Time Experiments

This section describes the procedures, the test results, and the results analysis for the response time experiments, to evaluate the services' performance.

6.4.1 Procedures

A set of procedures were defined for PIXm and PDQm test requests, along with the metrics and respective appraisal.

6.4.1.1 Test Requests

The performance requests will be compared between FHIR PIXm and HL7 v3 PIX, and between FHIR PDQm and HL7 v3 PDQ services. These services were chosen for the performance requests since the PIX profile is usually requested with great frequency, while the PDQ involves a complex search query regarding the patient's demographic information, returning a great amount of data. Each service was consecutively executed 20 times using the same query parameters for the respective pair of profiles.

6.4.1.2 Metrics

For the metrics, the mean and the standard deviation values, regarding the service's time of execution in seconds, were considered to establish the necessary conclusions and evaluations regarding the services' response time in the product.

6.4.1.3 Metrics Appraisal

To conclude whether the FHIR® services have a better response time or not, compared to the remaining services, the average time of response will be used to verify if it was lower than the previous services' average time of response, along with the standard deviation to evaluate the values' dispersion. By obtaining a lower average response time value for FHIR® services, while considering the standard deviation, it can be concluded that the implemented FHIR services have indeed a better overall response time compared to the existing services.

6.4.2 Test Results

The following sections present the profiles' time of execution results (in seconds) by iteration, along with the mean, variance and standard deviation values.

6.4.2.1 PIXm

Table 27 presents PIXm and PIX service times for each iteration, and their mean, variance, and standard deviation values.

The mean obtained from the PIX profile requests was approximately 1.107 seconds, and, by using the standard deviation value to determine the mean's regular range, the following values can be obtained:

- $Lower\ value$: $1.1073 - 0.0443419 = 1.0629581\ sec$
- $Upper\ value$: $1.1073 + 0.0443419 = 1.1516419\ sec$

With the previous calculations it can be confirmed that the PIX casual values are between, approximately, 1.063 and 1.152 seconds, which corresponds to 65% of the registered times.

For the PIXm profile the mean corresponded to approximately 0.103 seconds, and the regular limit values, which are within one standard deviation of the mean, were obtained through the following calculations:

- $Lower\ value$: $0.1026 - 0.0639284 = 0.0386716\ sec$

- $Upper\ value$: $0.1026\ +\ 0.0639284 =\ 0.1665284\ sec$

Based on the previous calculations, the casual values obtained for the PIXm profile are between, approximately, 0.039 and 0.167 seconds, which corresponds to 90% of the registered times.

Table 27 – PIXm vs PIX response times

	Service Times (sec)	
Iteration	**FHIR PIXm**	**HL7 v3 PIX**
IT 1	0.052	1.179
IT 2	0.117	1.099
IT 3	0.076	1.116
IT 4	0.121	1.055
IT 5	0.072	1.056
IT 6	0.068	1.114
IT 7	0.068	1.166
IT 8	0.092	1.065
IT 9	0.109	1.107
IT 10	0.119	1.138
IT 11	0.076	1.115
IT 12	0.329	1.073
IT 13	0.068	1.129
IT 14	0.224	1.057
IT 15	0.051	1.073
IT 16	0.084	1.150
IT 17	0.068	1.070
IT 18	0.091	1.046
IT 19	0.059	1.134
IT 20	0.108	1.204
Mean (μ)	0.1026	1.1073
Variance (σ^2)	0.0040868	0.0019662
Standard Deviation (σ)	0.0639284	0.0443419

6.4.2.2 PDQm

Table 28 presents PDQm and PDQ service times for each iteration, and their mean, variance, and standard deviation values.

The mean obtained from the PDQ profile requests was approximately 8.814 seconds, and, with the help of the standard deviation value to determine which values are within the mean's regular range, the following values can be obtained:

- $Lower\ value$: $8.8137 - 0.1003973\ =\ 8.7133027\ sec$

- $Upper\ value$: $8.8137\ +\ 0.1003973\ =\ 8.9140973\ sec$

Table 28 – PDQm vs PDQ response times

	Service Times (sec)	
Iteration	**FHIR PDQm**	**HL7 v3 PDQ**
IT 1	9.138	8.892
IT 2	7.907	8.765
IT 3	7.885	8.763
IT 4	7.933	8.723
IT 5	7.802	8.831
IT 6	7.831	8.986
IT 7	7.863	8.842
IT 8	7.889	8.817
IT 9	8.047	8.869
IT 10	7.919	8.676
IT 11	7.932	9.048
IT 12	7.969	8.872
IT 13	8.878	8.811
IT 14	7.883	8.776
IT 15	7.895	8.644
IT 16	7.835	8.961
IT 17	7.882	8.770
IT 18	7.912	8.719
IT 19	7.959	8.779
IT 20	8.013	8.644
Mean (μ)	8.0186	8.8137
Variance (σ^2)	0.1137019	0.0100796
Standard Deviation (σ)	0.3371972	0.1003973

With the previous calculations it can be confirmed that the PDQ casual values are between, approximately, 8.713 and 8.914 seconds, which corresponds to 70% of the registered times.

For the PDQm profile the mean corresponded to approximately 8.019 seconds, and the regular limit values, which are within one standard deviation of the mean, were obtained through the following calculations:

- $Lower\ value: 8.0186 - 0.3371972 = 7.6814028\ sec$
- $Upper\ value: 8.0186 + 0.3371972 = 8.3557972\ sec$

Based on the previous calculations, the casual values obtained for the PDQm service are between, approximately, 7.681 and 8.356 seconds, which corresponds to 90% of the registered times.

6.4.3 Results Analysis

In this section, each one of the profiles' response time results is analyzed and compared with each other, to verify if the FHIR services performance improved over the existing services.

6.4.3.1 PIXm

In Figure 71 the obtained values for both PIX and PIXm services can be analyzed, where the upper line corresponds to HL7 v3 PIX result times and the lower line corresponds to the FHIR PIXm result times. Overall, the FHIR results presented a faster response time than the ones from HL7 v3 where, comparing both mean times, the difference is approximately 1.005 seconds.

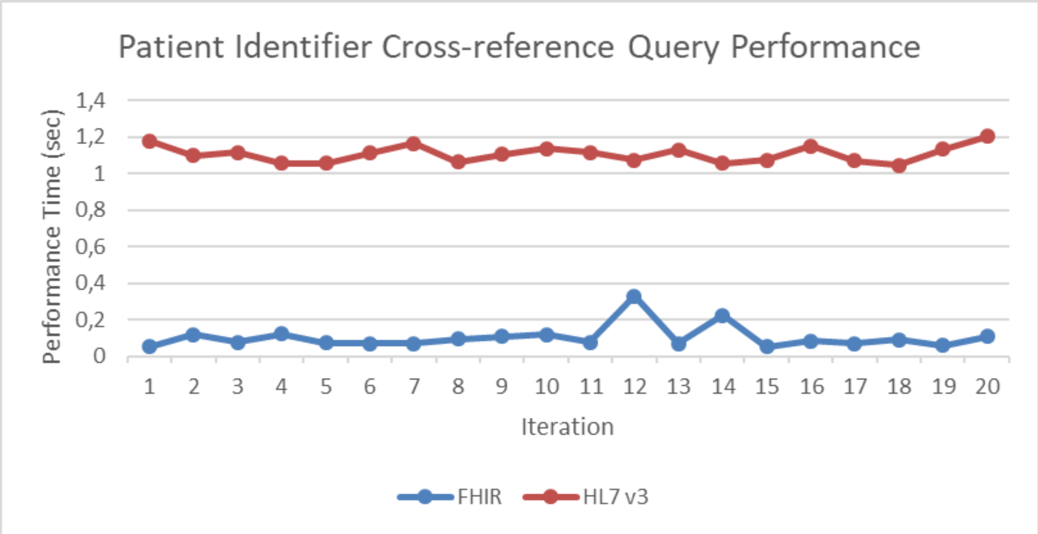

Figure 71 – PIX and PIXm Performance Line Chart

Figure 72 is like the previous but presents the results without the extreme values that went over the standard deviation limits, representing the results that maintained similar values within the range from the respective mean.

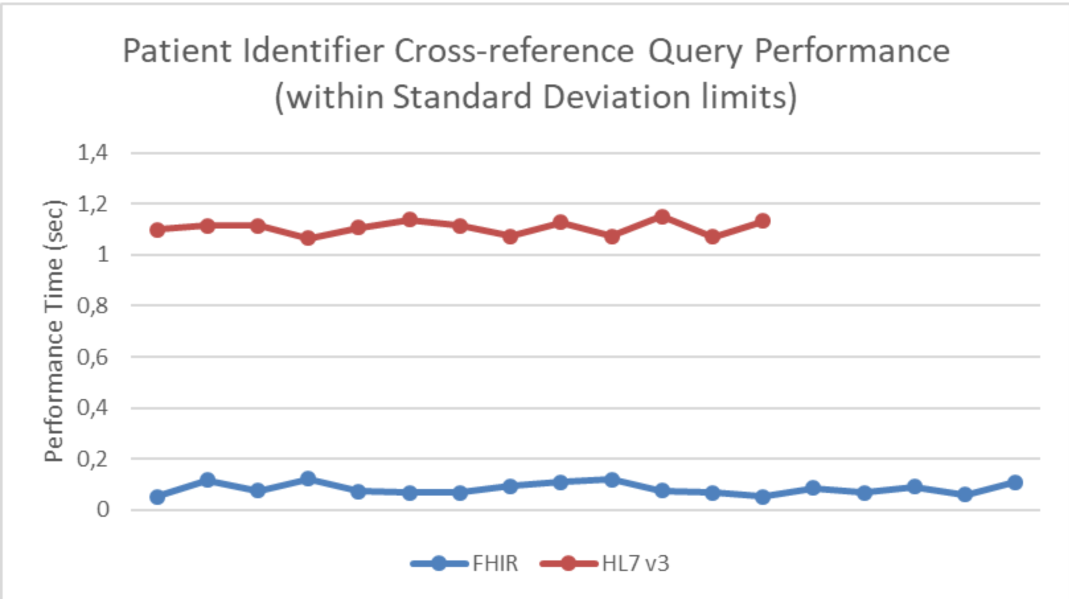

Figure 72 – PIX and PIXm Performance (within Standard Deviation limits)

It can also be observed that the FHIR service presented a bigger amount of stable values (90%) within the standard deviation, compared to the HL7 v3 ones (65%).

6.4.3.2 PDQm

In Figure 73 the obtained values for both PDQ and PDQm services can be analyzed, where the upper line corresponds to HL7 v3 PDQ result times and the lower line corresponds to the FHIR PIXm result times. Overall, the FHIR results presented a faster response time than the ones from HL7 v3 where, comparing both mean times, the difference is approximately 0.795 seconds, which is similar to the PIXm service results.

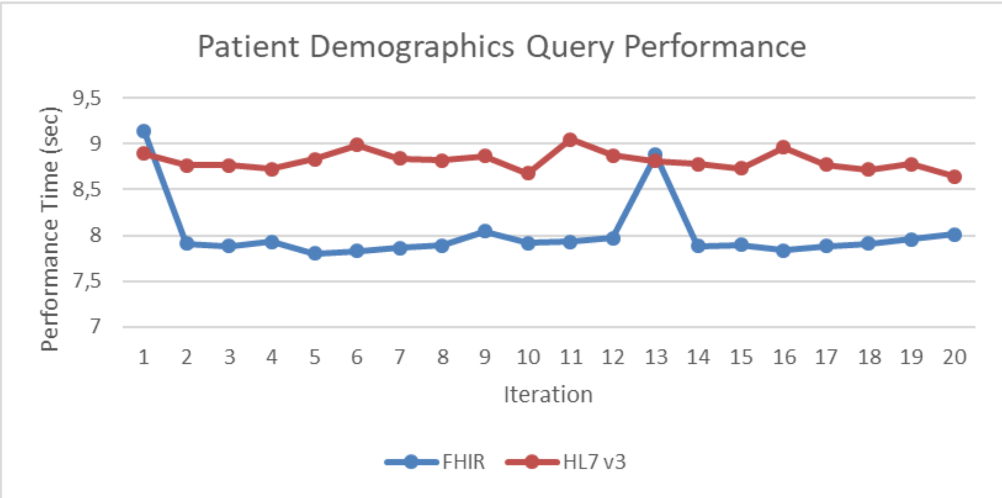

Figure 73 – PDQ and PDQm Performance Line Chart

Figure 74 is like the previous but presents the results without the extreme values that went over the standard deviation limits, representing the results that maintained similar values within the range from the respective mean.

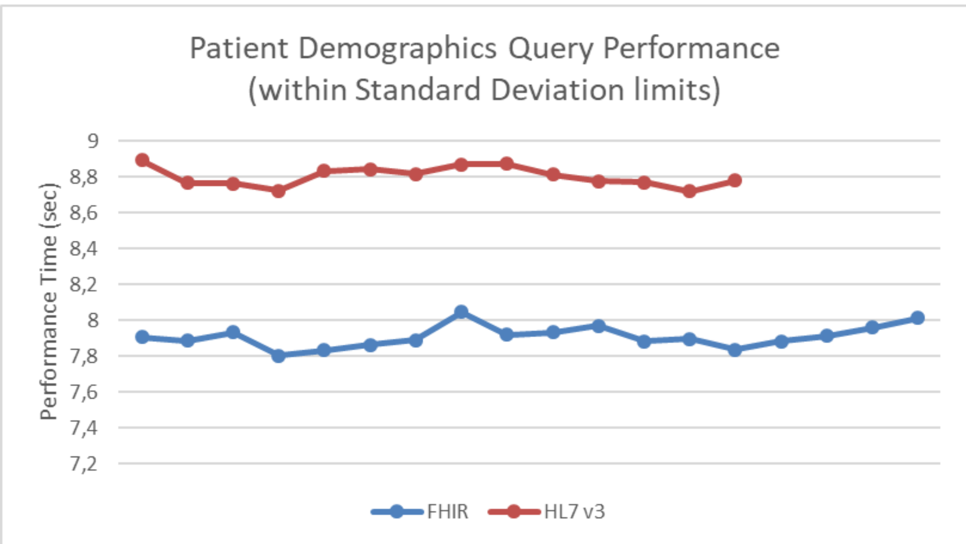

Figure 74 – PDQ and PDQm Performance (within Standard Deviation limits)

It can also be observed that the FHIR service presented a bigger amount of stable values (90%) within the standard deviation, compared to the HL7 v3 ones (70%).

6.4.3.3 Overall Observations

Considering the previous results, in terms of response time, the FHIR services presented a better performance regarding the HL7 v3 services. There can be many reasons for these results, namely FHIR's lightweight nature, or the profile's design and implementation.

With the implementation of FHIR services, that follow mainly a RESTful architecture, it is possible that most of the transformations involved from the services request to the Java objects, might involve lighter operations than the ones defined for the previous HL7 standards. Additionally, the security token decryption might also be one of the main factors that are improving the performance.

Regarding the ALERT services, a SAML token is used as the security measure for all the respective requests. A SAML token is XML-based, while the security token used for all the FHIR requests, which is the JSON Web Token, is JSON-based. The token decryption and validation might also have some performance variations regarding one type of token over the other, considering that the JWT token's size is considerably smaller than the SAML token's size.

These and other possible factors, might be the major reasons why the FHIR services presented a better performance over the existing ones, which mostly presented an improvement of one second for the services' response time.

6.5 Response Size Experiments

This section describes the procedures, the test results, and the results analysis for the response size experiments, to evaluate how much the services' response size influence the services' performance, in both FHIR services and HL7 v3 services.

6.5.1 Procedures

To conduct some experiments regarding the response size, a set of procedures were established to obtain some results and to evaluate the experiments.

6.5.1.1 Test Requests

The response times will be compared between PDQm FHIR service and PDQ HL7 v3 service, with three different response size results. 10 request iterations were performed to obtain some results, which were performed for requests that returned a response with 12, 57, and 150 elements.

6.5.1.2 Metrics

For the metrics, the mean regarding the differences of time between the FHIR service and the HL7 v3 service for each response size, will be used to retrieve some conclusions regarding the impact that the response size can have over the services performance.

6.5.1.3 Metrics Gauging

To verify the level of performance impact from one service to the other, the time of response (in seconds) difference between the two profiles will be calculated for each iteration. The time difference for the 10 iterations were also used to obtain the mean for the three response time experiments. By observing the mean difference for multiple response sizes, some conclusions shall be retrieved to verify if these size variations do have impact in the performance from one service to the other.

6.5.2 Test Results

The following tables present the results acquired for each size response, which in this case varies based on the number of FHIR Patient elements present in the Bundle resource.

6.5.2.1 Service times for 12 response elements

Table 29 shows the FHIR PDQm and HL7 v3 PDQ response times with 12 elements returned, along with the time differences and their mean, which obtained approximately 1.10 seconds.

Table 29 – Service Times for 12 elements

	Service Times (sec)		
Iteration	**FHIR PDQm**	**HL7 v3 PDQ**	**Time Difference**
IT 1	5.326	6.494	1.168
IT 2	5.415	6.406	0.991
IT 3	5.411	6.413	1.002
IT 4	5.327	6.566	1.239
IT 5	5.205	6.410	1.205
IT 6	5.327	6.416	1.089
IT 7	5.353	6.407	1.054
IT 8	5.304	6.313	1.009
IT 9	5.398	6.433	1.035
IT 10	5.311	6.480	1.169
Time Difference Mean (µ)			**1.0961**

6.5.2.2 Service times for 57 response elements

Table 30 shows the FHIR PDQm and HL7 v3 PDQ response times with 57 elements returned, along with the time differences and their mean, which obtained approximately 1.10 seconds.

Table 30 – Service Times for 57 elements

Service Times (sec)			
Iteration	**FHIR PDQm**	**HL7 v3 PDQ**	**Time Difference**
IT 1	7.885	8.757	0.872
IT 2	7.989	8.749	0.76
IT 3	7.822	8.783	0.961
IT 4	7.882	8.987	1.105
IT 5	7.831	8.896	1.065
IT 6	7.778	9.261	1.483
IT 7	7.932	9.026	1.094
IT 8	7.664	8.927	1.263
IT 9	7.675	8.999	1.324
IT 10	7.737	8.781	1.044
Time Difference Mean (μ)			**1.0971**

6.5.2.3 Service times for 150 response elements

Table 31 shows the FHIR PDQm and HL7 v3 PDQ response times with 150 elements returned, along with the time differences and their mean, which obtained approximately 1.05 seconds.

Table 31 – Service Times for 150 elements

Service Times (sec)			
Iteration	**FHIR PDQm**	**HL7 v3 PDQ**	**Time Difference**
IT 1	0.546	1.767	1.221
IT 2	0.694	1.624	0.93
IT 3	0.902	1.742	0.84
IT 4	0.533	1.720	1.187
IT 5	0.722	1.683	0.961
IT 6	0.531	1.702	1.171
IT 7	0.893	1.743	0.85
IT 8	0.534	1.645	1.286
IT 9	0.745	1.704	0.959
IT 10	0.548	1.661	1.113
Time Difference Mean (μ)			**1.0518**

6.5.3 Results Analysis

By observing the previous results, for the three case scenarios, the mean difference between the FHIR services' response times and the HL7 v3 services' response times, is approximately equal to one second. Although the message response sizes varied, not many differences were observed between these two services. In terms of performance, and by considering the previous results, the variations in response size from one integration profile to the other didn't reveal a great impact regarding the services' transaction processing.

7 Conclusion

This chapter describes the conclusions based on the features' fulfillment according to the defined objectives and the experimental results.

7.1 Overview

Concerning the initial analysis, a set of candidate frameworks were analyzed based on their three major factors:

- The capability to provide security features for REST services, regarding authentication and authorization measures;
- The ease of implementation that the framework can provide to implement REST services based on the FHIR specification and implementation requirements;
- The supportability that the framework can provide regarding the FHIR specification.

Based on the analysis and calculations performed, the selected framework was HAPI FHIR, which proved its effectiveness by delivering a set of features that promoted the server's security and provided a fast and efficient implementation for the required IHE profiles' services. Another relevant aspect was its support for the current FHIR version, and the evidence of continuous improvements for the future FHIR updates, which tremendously reduced the development time and the need to implement basic features and logic that were specifically required for the standard's data types, operations, REST resources, among other requirements.

Regarding the design phase, a new module was added to the existing ALERT® solution, which mainly used the HAPI FHIR framework features, with the purpose to centralize all of the FHIR requirements for the services' resource endpoints, server configurations, utility aspects, and other features involved with the standard's request and response structure. This approach enabled the separation of FHIR features from the other existing modules, which are mainly concerned with IHE profiles' specification. It also contributed to add new logic and transactions for the new IHE profiles in the ALERT existing modules, while using the FHIR module to complement the transactions' business workflow with the standard's specification, for example during the requests and responses conversions stages between the FHIR model and the ALERT services' model.

The workflow design for the new IHE profiles followed a similar approach with slight differences for each of the profiles. The transactions' workflow begins by receiving an incoming message to a FHIR endpoint, previously authorized by the server, which will then pass through some standard validations and posterior required model conversions, to use the existing ALERT services that already contained the necessary business logic for the IHE profiles. The ALERT services' response passes through a similar process on its way back, through the required converters and FHIR validators, which will then be returned to the requesting client. Even though the profiles were not entirely developed, the overall workflow design was considered appropriate for the transactions' implementation and provides the possibility to easily add new business logic thanks to the reduced dependencies between the involved components.

Regarding the IHE validations, all the implemented FHIR transaction requests and responses passed the validations successfully for the existing tests, which indicates that these requests follow IHE's structure and schema criteria. Nonetheless, IHE is still developing new test features for the remaining transactions and some of the implemented transactions still require improvements in the business process and data fields' treatment, in order to provide a stable service. This means that, just like the standard, the evaluation methods are still improving and will be used in the near future to validate the transactions with the profiles' current version.

Considering the performance experiments, the test results suggest that the FHIR services do improve the product's overall performance. Although it didn't present a noteworthy discrepancy for the human eye regarding the response times, it's still notable that, for several requests performed to a service which involves great processing, the service response time presented better results and small improvements considering the design and the new implemented FHIR features.

7.2 Features' Fulfillment

A fulfillment criteria was established from 0% to 100%, which is the same for the transactions Mobile Patient Identifier Cross-Reference Query [ITI-83] (PIXm profile), Mobile Patient Demographics Query [ITI-78] (PDQm profile), Provide Document Bundle [ITI-65] (MHD profile), Find Document Manifests [ITI-66] (MHD profile), Find Document References [ITI-67] (MHD profile), Retrieve Document [ITI-68] (MHD profile), and Mobile Query Existing Data [PCC-44] (mXDE and QEDm profiles), where each percentage allocated varied based on the developed features and IHE's criteria conformance:

- **Zero percent:** it's considered that the transaction's feature was not developed at all;

- **Twenty-five percent:** it's considered that the transaction feature was implemented using the suitable FHIR resources but did not respect the defined IHE's requirements for the corresponding profile;

- **Fifty percent:** it's considered that the transaction was implemented while respecting only IHE's requirements for the requests and responses' format, which are related to the messages' FHIR structure and content, but without respecting a portion of IHE's criteria to handle certain data fields for specific workflow situations;

- **Seventy-five percent:** it's considered that the transaction was implemented and respected most of IHE's specific criteria, along with FHIR's unique operations that can modify some of the transaction's desired behavior;

- **One hundred percent:** it's considered that the transaction was fully implemented and followed all of IHE's specification criteria.

The fulfillment criteria were also defined for ATNA and IUA profiles, which were rated with either 0%, 50% or 100%. Regarding ATNA, and more specifically the Record Audit Event [ITI-20] transaction, the following criteria was established for each percentage:

- **Zero percent:** it's considered that the audit event was not implemented for any profile to record the transaction occurrence;

- **Fifty percent:** it's considered that the audit event was applied for the implemented profiles partially respecting IHE's audit criteria for the audit message format and content;

- **One hundred percent:** it's considered that the audit event was applied for the implemented profiles and respected all of IHE's audit criteria regarding the corresponding audit message format and content.

Regarding the Incorporate Authorization Token [ITI-72] transaction for IHE's IUA profile, the following criteria was established:

- **Zero percent:** it's considered that the transaction was not implemented;

- **Fifty percent:** it's considered that a set of security methods to verify the request's authorization token legitimacy were implemented for all incoming FHIR requests to the server;

- **One hundred percent:** it's considered that the token was fully validated and the FHIR request was authorized based on the token's content.

Table 32 shows the fulfillment percentage for the IHE profiles' transactions, which are in accordance to the criteria defined previously, including the initial requirements fulfillment percentage based on their transactions.

From the chosen profiles for implementation, only the IHE's ITI-65 (MHD profile), ITI-68 (MHD profile) and PCC-44 (QEDm for mXDE profile) transactions were not developed due to the lack of required resources and time for implementation, along with some other aspects that would require further analysis and business decisions to be discussed with other team members involved in ALERT® HIE. Despite the fact that these transactions were not implemented in time, their analysis and design is done, and therefore, the implementation will be linear according to the defined architecture.

For the remaining transactions, only ITI-83 from PIXm was fully implemented respecting IHE's specification criteria. The ITI-78 (PDQm profile), ITI-66 (MHD profile) and ITI-67 (MHD profile) transactions were implemented without entirely following IHE's criteria due to some business rules and specific aspects regarding the product that limited the implementation, for example, database fields that should always contain a value but are not currently mandatory, and new fields defined in FHIR that are missing in the database model and that might affect existing database constraints. This also requires further discussion and decisions with other team members to fully implement these transactions with all IHE's requirements.

Table 32 – Fulfillment criteria for IHE Profiles

Requirement	Requirement Transactions	Transactions fulfillment	Requirement fulfillment
REQ1 - Correlate patient's identifiers (PIXm)	Mobile Patient Identifier Cross-Reference Query [ITI-83]	100%	100%
REQ2 - Exchange patient's demographic information (PDQm)	Mobile Patient Demographics Query [ITI-78]	50%	50%
REQ3 - Exchange health documents (MHD)	Provide Document Bundle [ITI-65]	0%	37.5%
	Find Document Manifests [ITI-66]	75%	
	Find Document References [ITI-67]	75%	
	Retrieve Document [ITI-68]	0%	
REQ4 - Exchange granular information (mXDE)	Mobile Query Existing Data [PCC-44]	0%	0%
REQ5 - Audit activities (ATNA)	Record Audit Event [ITI-20]	50%	50%
REQ6 - Secure access to services (IUA)	Incorporate Authorization Token [ITI-72]	50%	50%

Concerning the ITI-20 transaction for the audit event, the implementation was only partial because, with the adoption of IUA's profile, further information was required for the transaction's audit message, which was related with the data contained in the authorization token concerning the person who performed the request. However, this was not implemented due to some optimization aspects regarding the token's decryption so as to avoid reading its information several times during the request process, which requires additional implementation improvements, and a design review regarding the IUA workflow and possible alternatives to store the token during the transaction process.

Finally, for the IUA transaction (ITI-72), only the authorization measures for each user are missing. Although the token was validated with required user information fields, these fields were not used to perform any authorization due to the lack of discussion in this matter, since it involves registered users or domains. Therefore, it is still necessary to discuss with ALERT's team which group of users or domains shall access the new FHIR services defined in the required IHE's profiles.

Besides the transactions implementation, which were mainly performed in the last stage of the project, for the previous work which was more related to the solution's analysis, study, and design, it can be stated that all the objectives were fulfilled successfully. The following points highlight the most important tasks performed during the project development, which were required for proceeding with the transactions implementation phase:

- **Analyze the FHIR standard:** This task involved the study of the standard's specification, which also provided suitable information to take into consideration during the design phase. By studying the major points regarding the standard's RESTful nature, and its core features, the architectural design of the solution was adapted based on the standard's specification;

- **Analyze suitable frameworks:** This task was one of the most important stages of the project. Considering the standard's specification size and the amount of effort and features that the standard demanded by itself, the search and analysis of suitable frameworks was crucial to guarantee that the solution was built in an efficient way. This analysis was based on a set of criteria related to the value that the framework could bring not only for the existing product, but also for the customer. This task involved a series of meticulous calculations and evaluations to choose the most acceptable framework to implement the required features and assist on the solution's design;

- **Propose an integration in the current ALERT solution:** the architectural design stage was performed taking into consideration the previous analysis tasks. This also required a study of the current ALERT solution to analyze its logic, and to design an architectural solution that could make the most out of the frameworks' features, while respecting the FHIR® specification and without creating worrisome dependencies in the product;

- **Design the business workflow for the new FHIR transactions:** this task was performed right before the transactions' implementation stage. This design took into consideration the architectural design elaborated, and the IHE profiles' specification. Considering that these profiles could be implemented while using previous existing features, the design needed to be adapted to make the most out of the ALERT services that could fulfill the profiles' requirements. This way, this stage involved the construction of a conceptual design, along with more specific business workflows for the required IHE profiles, to cover necessary classes and features for the implementation, that could be integrated in the ALERT solution without affecting its existing features.

7.3 Future Work

Regarding the future work, a set of improvements and tasks were identified to correct some of the current features' limitations regarding the profiles' transactions:

- **Mobile Patient Demographics Query [ITI-78]:** discuss the *mother's maiden name* field for the PDQm profile, to conclude if this field, if known, is obligatory for return in every transaction response. Additionally, the existing database search query and database model shall be reviewed in order to support new FHIR fields, which are required to correctly implement the PDQm transaction query;

- **Provide Document Bundle [ITI-65]:** required implementation for the Provide Document Bundle [ITI-65] transaction, according to the respective design;

- **Find Document Manifests [ITI-66]:** discuss whether an existing or a new database ID field shall represent the DocumentReference resources returned within the Find Document Manifests [ITI-66] transactions' response, in order to retrieve those resources using the respective ID for the Find Document References [ITI-67] transaction;

- **Find Document References [ITI-67]:** certify that an URL value is always provided for the documents' location in the database, and that it is always returned within the Find Document References [ITI-67] transaction response;

- **Retrieve Document [ITI-68]:** required implementation for the Retrieve Document [ITI-68] transaction, according to the respective design;

- **Mobile Query Existing Data [PCC-44]:** required implementation for the Mobile Query Existing Data [PCC-44] transaction, according to the respective design;

- **Record Audit Event [ITI-20]:** add the user information provided in the authorization token, to the audit event messages related to each one of the profiles' transactions;

- **Incorporate Authorization Token [ITI-72]:** provide authentication measures for the IUA authorization token, using the information contained within the request's token.

Besides the previous tasks, additional work is required for some of FHIR's features, which are to be used for the profiles' search queries. One of the features that is not implemented in the previous profiles' transactions, is the *:exact* operation for FHIR parameters of type string, which indicates that a parameter name attached to the *:exact* operation, shall perform a search for database values that are exactly equal to the respective parameter value. Since the profiles were implemented using existing ALERT services, the current queries do not support this feature, and therefore, this FHIR operation shall be discussed with further analysis in future developments.

Finally, further experiments and tests shall be performed, to gather more evidences of proper reasons that might explain the performance improvements and which stage of the workflow brings those enhancements. Since the process involves multiple operations throughout the transactions' processing, such as authentication, validations, and conversions, the tests' reinforcement will assist on identifying which steps of the process evidence bigger differences compared to the previous transactions.

7.4 Final Thoughts

Overall, in my opinion, the work developed in terms of analysis and design was very thorough and extensive, considering that the framework analysis was maybe one of the most important steps in this project, since it promoted a better design approach, and an easier path to implement the required profiles and to add FHIR logic to the product. The implementation was probably one of the most challenging stages in the project. Despite the fact that the addition of the new module's features was pretty straightforward, the following steps regarding the integration in the product's existing modules required extra effort and time. Bearing in mind what was initially planned, the analysis and design stage was completed within the estimated time, while the development stage took a bit longer than expected, due to some implementation difficulties for some of the IHE profiles, mainly the PDQm and MHD transactions. Either way, the implemented features respected the architectural design, and, from my standpoint, with some extra effort and time, the remaining transactions can be fully implemented in accordance to the IHE profiles' criteria while following the current architecture.

Considering all the previous observations and results, it can be concluded that some additional features to improve the current implemented services are still required, as well as additional effort to provide some IHE transactions that were not implemented during the established development period. Regardless, for the current implemented features, and by observing the overall results, it can be stated that the architectural and design decisions, along with the chosen framework, provided good results for the expected requirements. The requests and responses format for the implemented features passed the IHE current tests with success, and it revealed enhancements within the product's performance with FHIR's specification, which suggests that any future developments should follow the current implementation and architectural approach to maintain and increase the results’ success.

References

ALERT Life Sciences Computing, 2017a. ALERT® PAPER FREE HOSPITAL (PFH) | ALERT® ONLINE - EN [WWW Document]. URL http://www.alert-online.com/pfh (accessed 11.16.17).

ALERT Life Sciences Computing, 2017b. MyALERT® - Personal Health Record | ALERT® ONLINE - EN [WWW Document]. URL http://www.alert-online.com/myalert (accessed 11.16.17).

ALERT Life Sciences Computing, 2017c. ALERT® PRIVATE PRACTICE | ALERT® ONLINE - EN [WWW Document]. URL http://www.alert-online.com/private-practice (accessed 11.17.17).

ALERT Life Sciences Computing, 2017d. ALERT® CARE | ALERT® ONLINE - EN [WWW Document]. URL http://www.alert-online.com/care (accessed 11.17.17).

ALERT Life Sciences Computing, 2017e. ALERT® HEALTH INFORMATION EXCHANGE also traded as HEALTH BOX® | ALERT® ONLINE - EN [WWW Document]. URL http://www.alert-online.com/hie (accessed 11.17.17).

ALERT Life Sciences Computing, 2017f. Mission and values | ALERT® ONLINE - EN [WWW Document]. URL http://org-www.alert-online.com/mission-values (accessed 12.5.17).

ALERT Life Sciences Computing, 2017g. Standards, certification and memberships | ALERT® ONLINE - EN [WWW Document]. URL http://org-www.alert-online.com/standards-memberships (accessed 12.5.17).

Benson, T., 2012. Principles of Health Interoperability HL7 and SNOMED, Health Information Technology Standards. Springer London, London. https://doi.org/10.1007/978-1-4471-2801-4

Busse, R., Geissler, A., Quentin, W., Wiley, M., 2011. Diagnosis-Related Groups in Europe.

Carter, J.H., 2008. Electronic health records : a guide for clinicians and administrators, Second.

ed. ACP Press.

Committee on Data Standards for Patient Safety, 2004. Patient Safety: Achieving a New Standard for Care. The National Academies, Washington, D.C.

DICOM Standard, 2017. Overview – DICOM Standard [WWW Document]. URL http://www.dicomstandard.org/about/ (accessed 12.12.17).

Fielding, R.T., 2000. Architectural Styles and the Design of Network-based Software Architectures 180.

Grady, R., 1992. Practical software metrics for project management and process improvement. Prentice Hall.

Groupe Speciale Mobile Association, 2016. Digital HealtHcare interoperability.

Health Level Seven International, 2017a. HL7 Standards - Section 1: Primary Standards [WWW Document]. URL http://www.hl7.org/implement/standards/product_section.cfm?section=1 (accessed 12.13.17).

Health Level Seven International, 2017b. HL7 Standards Product Brief - HL7 Version 2 Product Suite [WWW Document]. URL http://www.hl7.org/implement/standards/product_brief.cfm?product_id=185 (accessed 12.13.17).

Health Level Seven International, 2017c. HL7 Standards Product Brief - HL7 Version 3 Product Suite [WWW Document]. URL https://www.hl7.org/implement/standards/product_brief.cfm?product_id=186 (accessed 12.13.17).

Health Level Seven International, 2017d. HL7 Standards Product Brief - HL7 Version 3: Reference Information Model (RIM) [WWW Document]. URL https://www.hl7.org/implement/standards/product_brief.cfm?product_id=77 (accessed 12.13.17).

Health Level Seven International, 2017e. HL7 Standards Product Brief - CDA® Release 2 [WWW Document]. URL http://www.hl7.org/implement/standards/product_brief.cfm?product_id=7 (accessed 12.13.17).

Health Level Seven International, 2017f. Summary - FHIR v3.0.1 [WWW Document]. URL https://www.hl7.org/fhir/summary.html (accessed 11.21.17).

Health Level Seven International, 2017g. Formats - FHIR v3.0.1 [WWW Document]. URL https://www.hl7.org/fhir/formats.html (accessed 12.18.17).

Health Level Seven International, 2017h. CapabilityStatement - FHIR v3.0.1 [WWW Document]. URL https://www.hl7.org/fhir/capabilitystatement.html (accessed 12.18.17).

Health Level Seven International, 2017i. Resourcelist - FHIR v3.0.1 [WWW Document]. URL http://hl7.org/fhir/resourcelist.html (accessed 2.7.18).

Health Level Seven International, 2017j. Versions - FHIR v3.0.1 [WWW Document]. URL https://www.hl7.org/fhir/versions.html#maturity (accessed 12.21.17).

Health Level Seven International, 2017k. Extensibility - FHIR v3.0.1 [WWW Document]. URL http://hl7.org/fhir/extensibility.html (accessed 12.18.17).

Health Level Seven International, 2017l. Http - FHIR v3.0.1 [WWW Document]. URL http://hl7.org/fhir/http.html (accessed 12.19.17).

Health Level Seven International, 2017m. Messaging - FHIR v3.0.1 [WWW Document]. URL http://hl7.org/fhir/messaging.html (accessed 12.19.17).

Health Level Seven International, 2017n. Documents - FHIR v3.0.1 [WWW Document]. URL http://hl7.org/fhir/documents.html (accessed 12.19.17).

Health Level Seven International, 2017o. Services - FHIR v3.0.1 [WWW Document]. URL http://hl7.org/fhir/services.html (accessed 12.19.17).

Health Level Seven International, 2017p. Foundation-module - FHIR v3.0.1 [WWW Document]. URL http://hl7.org/fhir/foundation-module.html (accessed 12.19.17).

Healthcare Information and Management Systems Society, 2017. Interoperability and Health Information Exchange | HIMSS [WWW Document]. URL http://www.himss.org/library/interoperability-health-information-exchange (accessed 11.13.17).

Healthcare Information and Management Systems Society, 2013. Definition of Interoperability. Himss 2013.

Healthcare Information and Management Systems Society, 2005. Interoperability Definition and Background. Himss.

Hersh, W., Totten, A., Eden, K., Devine, B., Gorman, P., Kassakian, S., Woods, S.S., Daeges, M., Pappas, M., McDonagh, M.S., 2015. Health Information Exchange, Health Information Exchange. https://doi.org/10.23970/AHRQEPCERTA220

Hohpe, G., Woolf, B., 2004. Enterprise Integration Patterns, First. ed.

Humphrey, W.S., 1987. Characterizing the Software Process: A Maturity Framework. Techn. Rept 73–79.

IHE International, 2018a. Patient Manager [WWW Document]. URL https://gazelle.ihe.net/PatientManager/home.seam (accessed 5.29.18).

IHE International, 2018b. PIX Patient Identity Consumer [WWW Document]. URL https://gazelle.ihe.net/PatientManager/hl7fhir/pixm/consumer.seam (accessed 6.21.18).

IHE International, 2018c. Validate FHIR resources and requests [WWW Document]. URL https://gazelle.ihe.net/EVSClient/fhir/validator.seam?standard=FHIR-IHE&extension=IHE (accessed 6.21.18).

IHE International, 2018d. External Validation Service Front-end [WWW Document]. URL https://gazelle.ihe.net/EVSClient/home.seam (accessed 5.29.18).

IHE International, 2018e. Assertion Manager [WWW Document]. URL https://gazelle.ihe.net/AssertionManagerGui/assertions/show.seam?idScheme=ITI83&assertionId=ITI83-001 (accessed 5.29.18).

IHE ITI Technical Committee, 2017a. Patient Identifier Cross-reference for Mobile (PIXm) 23.

IHE ITI Technical Committee, 2017b. Patient Demographics Query for Mobile (PDQm) 31.

IHE ITI Technical Committee, 2017c. Mobile access to Health Documents (MHD) With XDS on FHIR 54.

IHE ITI Technical Committee, 2017d. Mobile Cross-Enterprise Document Data Element Extraction (mXDE) 32.

IHE ITI Technical Committee, 2017e. IHE IT Infrastructure (ITI) Technical Framework 1, 331.

IHE ITI Technical Committee, 2015. Internet User Authorization (IUA) 28.

Integrating the Healthcare Enterprise, 2018. IHE Test Tool Information - IHE Wiki [WWW Document]. URL https://wiki.ihe.net/index.php/IHE_Test_Tool_Information (accessed 5.29.18).

Integrating the Healthcare Enterprise, 2017. Profiles [WWW Document]. URL https://www.ihe.net/Profiles/ (accessed 12.21.17).

Integrating the Healthcare Enterprise, 2016a. IHE.net [WWW Document]. URL https://www.ihe.net/ (accessed 11.13.17).

Integrating the Healthcare Enterprise, 2016b. IHE Connectathon [WWW Document]. URL https://www.ihe.net/connectathon.aspx (accessed 11.21.17).

Integrating the Healthcare Enterprise, 2016c. IHE Process [WWW Document]. URL https://www.ihe.net/IHE_Process/ (accessed 1.30.18).

iNTERFACEWARE, 2017. The Iguana HL7 Integration Engine - iNTERFACEWARE [WWW Document]. URL https://www.interfaceware.com/iguana.html#solution (accessed

1.17.18).

International Business Machines, 2001. Rational Unified Process - Best Practices for Software Development Teams, Rational Software White Paper.

Jones, M.B., Bradley, J., Sakimura, N., 2015. JSON Web Token (JWT) 30.

Jones, M.B., Hardt, D., 2012. The OAuth 2.0 Authorization Framework: Bearer Token Usage 18.

Koen, P.A., Ajamian, G.M., Boyce, S., Clamen, A., Fisher, E., Fountoulakis, S., Johnson, A., Puri, P., Seibert, R., 2004. 1. Fuzzy Front End: Effective Methods, Tools, and Techniques, in: Belliveau, P., Griffin, A., Somermeyer, S. (Eds.), The PDMA ToolBook 1 for New Product Development. John Wiley & Sons, pp. 5–35.

Larman, C., 2004. Applying UML and Patterns: An Introduction to Object-Oriented Analysis and Design and Iterative Development, Analysis. Addison Wesley Professional. https://doi.org/10.1016/j.nec.2006.05.008

Lockhart, H., Campbell, B., Ragouzis, N., Hughes, J., Philpott, R., Maler, E., Madsen, P., Scavo, T., 2008. Security Assertion Markup Language (SAML) V2.0 Technical Overview 51.

Lodderstedt, E.T., McGloin, M., Hunt, P., 2013. RFC 6819 - OAuth 2.0 Threat Model and Security Considerations. https://doi.org/10.17487/RFC6819

Martin, R., 2002. SRP: Single Responsibility Principle, in: Agile Software Development, Principles, Patterns, and Practices. p. 529.

Martin, R., 2000. Design principles and design patterns. Object Mentor 1–34.

Nicola, S., 2017. Análise_Valor_Aula3_30NOV_2017_1hora_AHP.

Oracle, 2017a. Oracle | Integrated Cloud Applications and Platform Services [WWW Document]. URL https://www.oracle.com/index.html (accessed 12.15.17).

Oracle, 2017b. Securing RESTful Web Services [WWW Document]. URL https://docs.oracle.com/cd/E24329_01/web.1211/e24983/secure.htm#RESTF113 (accessed 12.15.17).

Oracle, 2013. The Java EE 6 Tutorial.

Organization for the Advancement of Structured Information Standards, 2017. About Us | OASIS [WWW Document]. URL https://www.oasis-open.org/org (accessed 12.13.17).

Parisot, C., 2016. Interoperability for Mobile applications: New IHE profiles.

Paul, R., Niewoehner, R., Elder, L., Foundation for Critical Thinking., 2007. The thinker's guide to engineering reasoning. Foundation for Critical Thinking.

Personal Connected Health Alliance, 2016. PERSONAL CONNECTED HEALTH ALLIANCE Because Health is Personal.

Regenstrief Institute, 2017. LOINC — The freely available standard for identifying health measurements, observations, and documents. [WWW Document]. URL https://loinc.org/ (accessed 11.14.17).

Rich, N., Holweg, M., 2000. Value Analysis, Value Engineering.

Richardson, L., 2008. The Maturity Heuristic, Crummy.

Ringholm bv, 2011. An explanation of HL7 version 3 in terms of HL7 version 2 [WWW Document]. URL http://www.ringholm.de/docs/01200_en_HL7v3_using_HL7v2_terms.htm (accessed 2.7.18).

Saaty, T.L., 2008. Decision making with the analytic hierarchy process. Int. J. Serv. Sci. 1, 83. https://doi.org/10.1504/IJSSCI.2008.017590

Simpatico Intelligent Systems, 2018. Table of Contents - Smile CDR Documentation [WWW Document]. URL https://smilecdr.com/docs/current/introduction/table_of_contents.html (accessed 1.17.18).

SNOMED International, 2017a. SNOMED International [WWW Document]. URL https://www.snomed.org/ (accessed 12.12.17).

SNOMED International, 2017b. SNOMED International [WWW Document]. URL https://www.snomed.org/snomed-ct (accessed 12.12.17).

The Office of the National Coordinator for Health Information Technology, 2016a. Health IT Dashboard [WWW Document]. HealthIT.gov. URL https://dashboard.healthit.gov/index.php (accessed 11.13.17).

The Office of the National Coordinator for Health Information Technology, 2016b. 2016 Report to Congress on Health IT Progress: Examining the HITECH Era and the Future of Health IT 3001, 32.

University Health Network, 2018a. Server Security - HAPI FHIR [WWW Document]. URL http://hapifhir.io/doc_rest_server_security.html (accessed 1.15.18).

University Health Network, 2018b. HAPI FHIR - The Open Source FHIR API for Java [WWW Document]. URL http://hapifhir.io/ (accessed 1.4.18).

University Health Network, 2018c. RESTful Operations - HAPI FHIR [WWW Document]. URL http://hapifhir.io/doc_rest_operations.html (accessed 1.16.18).

University Health Network, 2018d. Download - HAPI FHIR [WWW Document]. URL http://hapifhir.io/download.html (accessed 1.19.18).

University Health Network, 2017. Welcome to the University Health Network [WWW Document]. URL http://www.uhn.ca/ (accessed 12.22.17).

Vahdat, S., Hamzehgardeshi, L., Hessam, S., Hamzehgardeshi, Z., 2014. Patient Involvement in Health Care Decision Making: A Review. Iran. Red Crescent Med. J. https://doi.org/10.5812/ircmj.12454

Warwick Manufacturing Group, 2007. Product Excellence using Six Sigma: Quality Function Deployment 1–67.

Woodall, T., 2003. Conceptualising 'Value for the Customer'': An Attributional, Structural and Dispositional Analysis.' Acad. Mark. Sci. Rev. 12, 1–42.

World Health Organization, 2017. WHO | International Classification of Diseases [WWW Document]. URL http://www.who.int/classifications/icd/en/ (accessed 12.13.17).

Zeithaml, V.A., 1988. Consumer Perceptions of Price, Quality, and Value: A Means-End Model and Synthesis of Evidence. J. Mark. 52, 2–22. https://doi.org/10.2307/1251446

Annex A

Analyzing a Design Using the Elements of Thought

Analyzing a Design Using the Elements of Thought

Engineering purpose	What is the purpose of this design? What are the market opportunities or mission requirements? Who defines market opportunities/mission requirements? Who is the customer?
Question at hand	What system/product/process will best satisfy the customer's performance, cost, and schedule requirements? How does the customer define "value"? Is a new design or new technology required? Can an existing design be adapted? How important is time-to-market?
Point of view	A design and manufacturing point of view is typically presumed. What other points of view deserve consideration? Stockholders? Component vendors/suppliers? Marketing/sales? Customers? Maintenance/repair/parts? Regulators? Community affairs? Politicians? Environmentalists?
Assumptions	What environmental or operating conditions are assumed? What programmatic, financial, market or technical risks have been considered acceptable to date? What market/economic/competitive environment is assumed? What safety/environmental assumptions are we making? Are these assumptions acceptable? What maturity level or maturation timeline is assumed for emerging technologies? What happens if we change or discard an assumption? What criteria have historically been assumed in defining a "best" or "optimum" solution? What assumptions have been made on the availability of materials? What manufacturing capability was assumed? What workforce skills or attributes have been assumed?

Analyzing a Design Using the Elements of Thought (cont'd)

Engineering information	What is the source of supporting information (handbook, archival literature, experimentation, corporate knowledge, building codes, government regulation)? What information do we lack? How can we get it? Analysis? Simulation? Component testing? Prototypes? What experiments should be conducted? Have we considered all relevant sources? What legacy solutions, shortcomings, or problems should be studied and evaluated? Is the available information sufficient? Do we need more data? What is the best way to collect it? Have analytical or experimental results been confirmed? What insights and experiences can the shop floor provide?
Concepts	What concepts or theories are applicable to this problem? Are there competing models? What emerging theory might provide insight? What available technologies or theories are appropriate? What emerging technologies might soon be applicable?
Inferences	What is the set of viable candidate solutions? Why were other candidate solutions rejected? Is there another way to interpret the information? Is the conclusion practicable and affordable?
Implications	What are some important implications of the data we have gathered? What are the most important market implications of the technology? What are the most important implications of a key technology not maturing on time? How important is after-market sustainability? Is there a path for future design evolution and upgrade? Are there disposal/end-of-service-life issues we need to consider? What are the most important implications of product failure? What design features if changed, profoundly affect other design features? What design features are insensitive to other changes? What potential benefits do by-products offer? Should social reaction and change management issues be addressed?

Annex B

IHE's Test Validation Results

Validation result (Status: PASSED)

External Validation Report

General Informations

Validation Date 2018, 05 29 - 06:02:24

Validation Service Gazelle FHIR Validator (Version : 2.1.0)

Validation Test Status PASSED

Result overview

XML

ModelBased Validation PASSED

XML Validation Report

The document you have validated is supposed to be an XML document. The validator has checked if it is well-formed, results of this validation are gathered in this part.

THE XML DOCUMENT IS NOT WELL-FORMED, FOR THE FOLLOWING REASONS:

Model Based Validation details

«

Result	PASSED
Summary	1 checks 0 errors 0 warning 0 infos

HIDE : ☐Errors ☐Warnings ☐Infos ☐Reports

Reports

Location	https://gazelle.ihe.net/PatientManager/fhir/Patient/$ihe-pix?sourceIdentifier=urn:oid:1.3.6.1.4.1.33233.2.2.5.4.12|71710460050002&targetSystem=urn:oid:1.3.6.1.4.1.33233.3.1.1.1&_format=json
Description	https://gazelle.ihe.net/PatientManager/fhir/Patient/$ihe-pix?sourceIdentifier=urn:oid:1.3.6.1.4.1.33233.2.2.5.4.12|71710460050002&targetSystem=urn:oid:1.3.6.1.4.1.33233.3.1.1.1&_format=json is a valid PIXm URL request [Assertion...]

Assertion

AssertionId	ITI83-001 Testable
predicate	The name of the Get Corresonding Identifier operation SHALL be $ihe-pix, and it is applied to FHIR Patient Resource type.The URL for this Operation is: [base]/Patient/$ihe-pixWhere [base] is the URL of Patient Identifier Cross-reference Manager Service provider.The Get Corresponding Identifiers message is performed by an HTTP GET command :GET [base]/Patient/$ihe-pix?sourceIdentifier=[token]{&targetSystem=[uri]}{&_format=[mime-type]}
Prescription level	Mandatory / Required / Shall
Page	18
Section	3.83.4.1.2
Status	to be reviewed
Last changed	8/16/16 11:17:32 AM by xavier.francois
Comment	

Document name	Provenance	Revision	Action
IHE ITI PIXm Supplement	IHE.net	1.3	

18 de 23 Tamanho automático

The Get Corresponding Identifiers message is conducted by the Patient Identifier Cross-reference Consumer by executing an HTTP GET against the Patient Identifier Cross-reference Manager's `Patient` Resource URL.

The URL for this operation is: `[base]/Patient/$ihe-pix`

Where **[base]** is the URL of Patient Identifier Cross-reference Manager Service provider.

The Get Corresponding Identifiers message is performed by an HTTP GET command shown below:

```
GET [base]/Patient/$ihe-pix?sourceIdentifier=[token]{&targetSystem=[uri]}{&_format=[mime-type]}
```

Well-formedness PASSED

The document you have validated is supposed to be a well-formed document. The validator has checked if it is well-formed, results of this validation are gathered in this section.

The document is well-formed

Schema Validation detailed Result PASSED

Your document has been validated with the appropriate schema, here is the detail of the validation outcome.

The document is valid regarding the schema

Gazelle Objects Checker validator results PASSED

Summary of checks 1 1 5

Severity
Errors
Warnings
Infos
Unknowns
Reports
Reset filters

Location	/Parameters
Description	For each matching Patient Resource, the Parameters Resource shall include one parameter with name="targetId"[Assertion...]
Location	/Parameters/Parameter[1]/valueIdentifier
Description	The assigner attribute may be populated[Assertion...]
Location	/
Description	The top resource shall be 'Parameters'[Assertion...]
Location	/Parameters/Parameter[1]
Description	targetIdentifier parameter may be present[Assertion...]
Location	/Parameters[1]
Description	"targetIdentifier" parameter shall be formatted as a valueIdentifier[Assertion...]
Location	/Parameters/Parameter[1]/valueIdentifier
Description	Both the value and system shall be populated[Assertion...]
Location	/Parameters[$INDEX$]
Description	"targetId" and "targetIdentifier" are the only allowed parameters in PIXm response[Assertion...]

Validation result (Status: PASSED)

External Validation Report

General Informations

Validation Date 2018, 05 30 - 11:34:55

Validation Service Gazelle FHIR Validator (Version : 2.1.0)

Validation Test Status PASSED

Result overview

XML

ModelBased Validation PASSED

XML Validation Report

The document you have validated is supposed to be an XML document. The validator has checked if it is well-formed, results of this validation are gathered in this part.

THE XML DOCUMENT IS NOT WELL-FORMED, FOR THE FOLLOWING REASONS:

Model Based Validation details

«

Result PASSED

Summary 1 checks
0 errors
0 warning
0 infos

HIDE : Errors Warnings Infos Reports

Reports

Location https://gazelle.ihe.net/PatientManager/fhir/Patient?given:exact=Patient&family:exact=Zero&gender=male&birthdate=2018-05-08&address-city:exact=Porto&address-country:exact=PT&address-postalcode:exact=1111-111&address-state:exact=PT&address:exact=Sample Address&telecom=911111111&identifier=urn:oid:1.3.6.1.4.1.33233.3.1.1.1|48086&identifier=urn:oid:1.3.6.1.4.1.33233.3.1.1.1|&_format=json

Description https://gazelle.ihe.net/PatientManager/fhir/Patient?given:exact=Patient&family:exact=Zero&gender=male&birthdate=2018-05-08&address-city:exact=Porto&address-country:exact=PT&address-postalcode:exact=1111-111&address-state:exact=PT&address:exact=Sample Address&telecom=911111111&identifier=urn:oid:1.3.6.1.4.1.33233.3.1.1.1|48086&identifier=urn:oid:1.3.6.1.4.1.33233.3.1.1.1|&_format=json is a valid PDQm URL request [Assertion...]

Well-formedness **PASSED**

The document you have validated is supposed to be a well-formed document. The validator has checked if it is well-formed, results of this validation are gathered in this section.

The document is well-formed

Schema Validation detailed Result **PASSED**

Your document has been validated with the appropriate schema, here is the detail of the validation outcome.

The document is valid regarding the schema

Gazelle Objects Checker validator results **PASSED**

Summary of checks 2 5

Severity
Errors
Warnings
Infos
Unknowns
Reports
Reset filters

Location	/Bundle/entry[0]/resource/Patient
Description	Patient Demographics Suppliers SHALL include the mothers maiden name, if known[Assertion...]
Location	/Bundle/entry[0]
Description	If the Patient Demographics Supplier whishes to convey the quality of match, it shall represent the confidence of a particular match within the bundle as a score attribute[Assertion...]
Location	/
Description	The top resource shall be 'Bundle'[Assertion...]
Location	/Bundle/entry[0]/resource/Patient/identifier[0]
Description	Both the value and system shall be populated[Assertion...]
Location	/Bundle/entry[0]/resource/Patient/identifier[0]
Description	When the assigning authority name is provided, the actor shall also populate the display attribute[Assertion...]
Location	/Bundle/entry[0]/resource/Patient/identifier[0]
Description	The assigner attribute may be populated[Assertion...]
Location	/Bundle/entry[0]/resource
Description	The bundle entries shall be Patient resources[Assertion...]

Schema Validation detailed Result **FAILED**

Your document has been validated with the appropriate schema, here is the detail of the validation outcome.

The document is not valid regarding the schema because of the following reasons:

Summary of reports

4	2	1

Location	
Description	Bundle.entry.resource.type.coding: Coding.system must be an absolute reference, not a local reference
Location	
Description	Bundle.entry.resource.class.coding: Coding.system must be an absolute reference, not a local reference
Location	
Description	Bundle.entry.resource.securityLabel.coding: Coding.system must be an absolute reference, not a local reference
Location	
Description	Bundle.entry.resource.securityLabel.coding.system: URI values cannot have whitespace
Location	
Description	Bundle.entry.resource.securityLabel: None of the codes provided are in the value set http://hl7.org/fhir/ValueSet/security-labels (http://hl7.org/fhir/ValueSet/security-labels, and a code should come from this value set unless it has no sui table code) (codes = Connect-a-thon confidentialityCodes#1.3.6.1.4.1.21367.2006.7.103)
Location	
Description	Bundle.entry.resource.content.attachment.language: ValueSet null not found
Location	
Description	Bundle.entry.resource.type: None of the codes provided are in the value set http://hl7.org/fhir/ValueSet/c80-doc-typecodes (http://hl7.org/fhir/ValueSet/c80-doc-typecodes, and a code is recommended to come from this value set) (code s = LOINC#34108-1)