“On the Development of Data Analytics Solutions in a Small Company: A Case Study”

By

Alex Bruno Paranhos da Silva

B.Sc. Dissertation

SALVADOR

2017

Federal University of Bahia
Computer Science Department
Undergraduate in Computer Science

Alex Bruno Paranhos da Silva

"On the Development of Data Analytics Solutions in a Small Company: A Case Study"

A B.Sc. Dissertation presented to the Computer Science Department of Federal University of Bahia in partial fulfillment of the requirements for the degree of Bachelor in Computer Science.

Advisor: *Eduardo Santana de Almeida*

SALVADOR
2017

To my beloved grandfather, Amadeu Ferreira da Silva (in memoriam).

Acknowledgements

Gostaria de agradecer a todos que de alguma forma me ajudaram a chegar ao final desta etapa e me deram força para enfrentá-la. À minha família a paciência e força, e nunca ter deixado de acreditar em mim. Minhas avós, Urânia e Aurenilza, tias, tios, primas e primos, família que se escreve com carinho.

Ao meu pai todo amor e carinho, toda barra segurada nos momentos mais difíceis, os conselhos, os ensinamentos e cada noite/madrugada adentro que passou fascinado com o meu trabalho. Também a Ana toda ajuda desde sempre, são eternas as dívidas que tenho com você. Muito obrigado por me acolher tão bem.

Agradeço à minha mãe toda a sua dedicação, o amor, a compreensão, cada palavra de conforto nas horas de desânimo, o seu colo acolhedor nos momentos de cansaço e todas as vezes em que me pegou pelo braço, me fez levantar e continuar caminhando. E a Nazaré o apoio, a torcida e o carinho.

À maya, que só na grafia se vê minúscula, pois é o que há de mais maiúscula na minha vida. Agradeço o carinho, o amor, o cuidado e toda nossa Poesia. Obrigado por me aturar esse tempo todo, por me ouvir e, mesmo sem me entender, me ajudar; e os ensinamentos de todo dia. Só você sabe o que passamos para alcançar tudo isso — e falo na primeira pessoa do plural porque você foi a metade (muitas vezes inteira) dessas pernas que correram pra chegar onde chegamos. Muito obrigado por não desistir de mim.

A Cláudio e Chris agradeço o acolhimento, o carinho e a força. Sei que estavam na arquibancada e a torcida de vocês foi necessária.

Aos meus amig@s que, longe ou perto, me ajudaram da sua melhor forma. Aos meus SACAnas (e os desgarrados perdidos pelo mundo), amig@s da UFBa que me acompanharam nessa jornada (pelas horas de estudos, pelas resenhas e pela cerveja gelada no chuleta), aos colegas do LES, ao pessoal de outros cursos, aos da infância pro resto da vida — Magui, Miojo e Paulinha.

Agradeço também a todos os professor@s que passaram por mim e, de uma forma ou de outra, me deixaram algo de lição.

Finalmente agradeço ao meu orientador, Eduardo Santana de Almeida. Muito obrigado por sempre ter acreditado no meu potencial e nunca ter desistido do nosso trabalho juntos. Sempre deixei clara minha admiração pelo profissional que é. Obrigado pelo conhecimento, por toda ajuda nestes anos, por acreditar em mim e me ajudar a enfrentar mais um desafio.

Isso de querer
ser exatamente aquilo
que a gente é
ainda vai
nos levar além

—PAULO LEMINSKI (Incenso fosse música)

Resumo

Conceitua-se por *Business Intelligence and Analytics* (BIA) a exploração e análise de dados, cujas informações e intuições extraídas são utilizadas para embasar a tomada de decisões. *Managers* tem utilizado técnicas e ferramentas de mineração e análise de dados para dar suporte à tomada de decisões, em diversas áreas. De modo semelhante, um número cada vez mais crescente de organizações governamentais tem se valido de sistemas de BIA para assegurar uma maior transparência nos dados do governo, sua contabilidade e também para suscitar mais possibilidades de participação pública. Todavia, há poucas referências de estudos de caso na literatura que sirvam de aporte teórico para o embasamento de pesquisas nessa área e que ajudem a melhor compreender o desenvolvimento de sistemas para BIA. Neste contexto, o presente trabalho teve como objetivo o entendimento do desenvolvimento de uma solução de análise de dados, realizada por uma empresa de pequeno porte, a serviço de uma organização governamental. Para a realização da presente pesquisa, portanto, propôs-se um estudo de caso onde foram investigados os seguintes aspectos: métodos e procedimentos de Engenharia de Software, princípios do manifesto "Engenharia de Software Indutiva" e Requisitos Não Funcionais. O estudo de caso foi realizado por meio da análise de entrevistas que foram conduzidas com a equipe de desenvolvedores e da análise de artefatos do sistema, utilizados para triangulação de dados. Os resultados alcançados através do estudo demonstram quais as práticas de Engenharia de Software foram utilizadas, bem como quais os princípios aplicados em sua realização e, ainda, quais foram os requisitos implementados no desenvolvimento.

Palavras-chave: análise de dados; inteligência de negócios; engenharia de software indutiva; requisitos não funcionais; engenharia de software.

Abstract

Business Intelligence and Analytics (BIA) is stated as the data exploration and analysis to extract important information and insights, in order to aid on decision making. Managers have used techniques and tools of data mining and analytics to support the decision making accross many fields. Furthermore, an increasingly number of governmental organizations have used BIA to aid in data transparency, accountability and public participation. Nevertheless, there is a lack of case studies in the literature that serve as theoretical contribution to research in this area which can help to better understand the development of BIA tools. In this context, the main goal of this work is to understand the development of an analytics solution in a small company. A case study was proposed focusing on the following aspects: the methods and procedures of Software Engineering, the Inductive Software Engineering (ISE) manifesto principles and Non-Functional Requirements (NFR). This study was conducted by the analysis of the interviews conducted with the development team and the analysis of the system's artifacts used to produce triangulation. The findings highlight which practices of Software Engineering were used, as well which principles were applied and which NFRs were implemented.

Keywords: data analytics; business intelligence; inductive software engineering; non-functional requirements; software engineering.

Table of Contents

List of Figures xi

List of Tables xii

List of Acronyms xiii

1 Introduction 1

1.1 Motivation . . . 2

1.2 Problem Statement . . . 3

1.2.1 Research Questions . . . 3

1.3 Overview of the Proposed Solution . . . 4

1.3.1 Context . . . 4

1.3.2 Outline of the Proposal . . . 5

1.4 Out of Scope . . . 5

1.5 Statement of the Contributions . . . 5

1.6 Dissertation Structure . . . 6

2 An Overview on Analytics and Non-Functional Requirements 7

2.1 Analytics . . . 8

2.1.1 Introduction . . . 8

2.1.2 Business Inteligence and Analytics (BIA) . . . 9

2.1.3 The Benefits of Analytics . . . 9

2.1.4 Software Analytics (SA) . . . 12

2.1.5 Inductive Software Engineering (ISE) Manifesto . . . 12

Users Before Algorithms . . . 12

Plan for Scale . . . 12

Early Feedback . . . 14

Be Open-Minded . . . 14

Do Smart Learning . . . 14

Live With the Data You Have . . . 14

Broad Skill Set, Big Toolkit . . . 15

2.2 Non-Functional Requirements . . . 15

2.2.1 Types of Non-Functional Requirements (NFR) . . . 16

2.3 Chapter Summary . . . 18

3 The Case Study Design **20**
3.1 Introduction 20
3.2 Case and Subject Selection 21
3.3 Procedures and Roles 25
3.4 Data Collection 25
3.5 Analysis Procedures 27
3.6 Validity Procedures 27
3.7 Legal and Ethical Issues 28
3.8 Chapter Summary 29

4 Case Study Results **30**
4.1 Participants' Profile 30
4.2 Software Engineering (SE) Process and Methods 31
4.3 ISE Manifesto Principles 33
4.3.1 Users Before Algorithms 33
4.3.2 Plan for Scale 34
4.3.3 Early Feedback 34
4.3.4 Be Open-minded 34
4.3.5 Do Smart Learning 35
4.3.6 Live With The Data You Have 35
4.3.7 Broad Skill Set, Big Toolkit 35
4.4 Non-Functional Requirements 36
4.4.1 Performance 36
4.4.2 Security 36
4.4.3 Availability 37
4.4.4 Usability 37
4.4.5 Scalability 37
4.4.6 Reliability 39
4.4.7 Interoperability 39
4.4.8 Adaptability 39
4.5 Conclusion 39
4.6 Chapter Summary 40

5 Conclusions **42**
5.1 Research Contribution 43
5.2 Future Work 44

References **45**

Appendix **50**

A Available Resources **51**

B Interview Questions **52**

B.1 Basic Questions 52
B.2 Inductive Software Engineering Principles 52
B.2.1 Users Before Algorithms 53
B.2.2 Plan for Scale 53
B.2.3 Early Feedback 53
B.2.4 Be Open-minded 53
B.2.5 Avoid Bad Learning 54
B.2.6 Live with the data you have 54
B.2.7 Broad skill set, big toolkit 54
B.3 Non-Functional Requirements 54
B.3.1 Performance 55
B.3.2 Security 55
B.3.3 Availability 55
B.3.4 Usability 55
B.3.5 Scalability 56
B.3.6 Reliability 56
B.3.7 Interoperability 56
B.3.8 Adaptability 56

C Template of Consent Letter **57**

List of Figures

2.1 BIA framework (Wixom and Watson, 2007) 10
2.2 Spectrum of BIA benefits (Wixom and Watson, 2007) 11
2.3 Steps of the knowledge discovery in database process (Fayyad *et al.*, 1996). 13
2.4 Type of Systems and Relevant NFRs (Mairiza *et al.*, 2010) 19

3.1 Holistic Single-case. 21
3.2 Observatório home screen. 23
3.3 Some Observatório's indicators. 24
3.4 General principles for interview sessions: (a) funnel, (b) pyramid, and (c) timeglass. 25

4.1 Notebook sheets. 32
4.2 White board used by the team. 33
4.3 Mockup from Observatório's screen. 38

List of Tables

2.1 Terminology for using and analyzing data (Davenport, 2014) 8
2.2 Key questions addressed by analytics (Davenport *et al.*, 2010) 10
2.3 NFRs selected. 16

3.1 Observatório statistics. 22

4.1 Case Study Participants. 31
4.2 The principles adopted in the project. 40
4.3 The Non-Functional Requirements adopted in the project. 40

List of Acronyms

BIA	Business Inteligence and Analytics
D1	Developer Number 1
D2	Developer Number 2
F1	Former Developer
FAPESB	Bahia Research Foundation
ISE	Inductive Software Engineering
IT	Information Technology
NFR	Non-Functional Requirements
QA	Quality Attributes
SA	Software Analytics
S1	Statistician and Stakeholder
SE	Software Engineering

1 Introduction

Quando a gente anda sempre em frente,
não pode mesmo ir longe . . .
—ANTOINE DE SAINT-EXUPÉRY (O Pequeno Príncipe)

Nowadays, we are living at a data-driven world. Thanks to the development of social medias, the Web 2.0, sensors and mobile, a large amout of data is being produced in recent years. Some terms related to data mining and analytics have emerged recently, such as Big Data, Business Inteligence and Analytics (BIA), data analytics and Software Analytics (SA). Managers and stakeholders are using data and analysis in order to discover new information and extract insight from their business. According to Davenport and Patil (2012), data scientist is "the sexiest job of the 21st century."

However, there is nothing new about the idea of analyzing business data to obtain important insights to aid on decisions. According to Davenport (2014), this activity has been performed since 1950s and had different terminologies through the time. Even though, each terminology has specific meaning, the base definition is to use data analysis and exploration to help decision makers gather insights and important information to aid on better decisions (Buse and Zimmermann, 2012).

The use of data analytics is bringing important effects on business across many fields on both academy and industry (Chen *et al.*, 2012). From social media, e-commerce and software development, to sports, governments and health organizations (Chen, 2009; Meredith and O'Donnell, 2010; Chau and Xu, 2012; Menzies and Zimmermann, 2013). A successful introduction of data analysis and exploration provides a significant opportunity for a business manager to improve its competitive position or a project manager to increase the quality of their software product, for instance.

Menzies *et al.* (2011) defined a manifesto called Inductive Software Engineering (ISE) listing a set of principles that characterize industrial data analytics. These principles are not intended to be a formal maturity model, but good practices that the authors judge to consider in industrial data analytics development.

An increase in data's volume and speed are leading to faster analytics (Larson and Chang, 2016). Davenport (2014) stated that Big Data changed the traditional BIA as data had to be processed as quickly as possible. In this context, the quality attributes are very important to a successful BIA project.

Moreover, Robertson and Robertson (2012) defined Non-Functional Requirements (NFR)s as a quality that the software must have. NFR are not essential to place the final software to work, although they are related to the applications of quality aspects. However, they are often neglected at development process (Mairiza *et al.*, 2010). This negligence leads research community to improve the understanding about the NFRs and motivate in-depth studies about it.

The focus of this dissertation is to provide an understanding about the development of analytics solutions concerning ISE principles and NFRs. In order to achieve this goal we performed a case study research in a small company, to comprehend the practices used on the development of data analytics applications.

This chapter contextualizes the focus of this dissertation. Section 1.1 presents the motivation of this work; Section 1.2 discusses the definition of the problem and the research questions to answer; Section 1.3 outlines the proposed solution; Section 1.4 shows the topics out of scope; Section 1.5 presents the contributions of this work and, finally; Section 1.6 defines the structure of this dissertation.

1.1 Motivation

Data analysis and exploration is bringing important effects on both academy and industry nowadays. Davenport *et al.* (2010) asserted that analytics can improve performance on business. Moreover, many work discussed how BIA improves organizational performance as in productivity (Lee and Park, 2005; Borrajo *et al.*, 2010; Demirkan and Delen, 2013), revenue (Counihan *et al.*, 2002; Ko and Osei-Bryson, 2004; Abbasi *et al.*, 2012) and also customer and employee satisfaction (Chae, 2014; Watson *et al.*, 2006; Park *et al.*, 2012).

A review of empirical studies in BIA was conducted involving work published from January, 2000 to August, 2015 (Trieu, 2017). It collected 106 papers containing keywords such as "big data", "business intelligence", "data mining", "business analytics" and "data warehousing". They observed that the main focus of research is how BIA can improve

organizational performance. Another aspect that received attention in the literature is BIA assets, composed of "BI technology, human resources and application portfolios." They stated BI technology as "BI tools designed in a way that fits an organization's task and data strategy combined with hardware infrastructure." Although there are many work in literature about BIA tools such as (Kou *et al.*, 2003; Wang *et al.*, 2013; Speier and Morris, 2003), they focus on technologies or algorithms. Gibson and Arnott (2010) stated that there is a lack of case studies about the development of BIA solutions. In addition, to the best of our knowledge, we could not find any paper concerning Software Engineering (SE) processes and NFRs.

In some domain, Non-Functional Requirements (NFR) are commonly referred as the quality attributes of the system. Sommerville (2010) defines NFRs as "requirements that are not directly concerned with the specific services delivered by the system to its users." Robertson and Robertson (2012) go further and state that non-functional attribute is a quality, or property, that the system must have, such as "an appearance, or a speed or accuracy property."

NFR are not essential to place the final product to work, however, they are concerned to their quality aspects and are very important to a successful system project (Chung *et al.*, 2012). Morever, many software failures are related to poor requirements analysis (Mairiza *et al.*, 2009). Mairiza *et al.* (2010) asserted that NFRs are "often neglected, poor understood and not considered adequately in software development." This negligence leads research community to improve the understanding about the NFRs and motivate in-depth studies about it. Furthermore, we could not find any work related to Non-Functional Requirements (NFR) in analytics.

1.2 Problem Statement

Motivated by the scenario presented in Section 1.1, the goal of the work described in this dissertation can be stated as: "*What are the practices used to develop analytics solutions in a small company concerning Software Engineering (SE) practices, Inductive Software Engineering (ISE) principles and Non-Functional Requirements (NFR)?*"

1.2.1 Research Questions

In order to evaluate these aspects, we defined the following research questions:

RQ1. Are the Inductive Software Engineering (ISE) principles followed by the development team?

Rationale: The goal is to verify if the ISE principles emerged through the development process. These principles listed by Menzies *et al.* (2011) are not a formal model, but valuable good practices.

RQ2. What are the non-functional requirements used in the development of data analytics?

Rationale: This question intends to identify the Quality Attributes (QA)s covered by the development team.

> **RQ2.1. How these non-functional requirements are specified, designed, implemented and tested?**
>
> Rationale: If there is a positive response to **RQ2**, this question investigates how the NFR was specified, designed, implemented and tested.

1.3 Overview of the Proposed Solution

Runeson *et al.* (2012) stated that a case study is an exploratory research that provides a deep understanding of the phenomena under investigation. Therefore, in order to accomplish the goal of this work, we proposed a case study to explore the development of an analytic solution in a small company. This case study concerns two main topics: Inductive Software Engineering (ISE) principles and Non-Functional Requirements (NFR). Moreover, we analyzed the Software Engineering (SE) methods applied at the development. The reminder of this section presents the context where this work was developed and the outline of the proposed solution.

1.3.1 Context

This dissertation describes a case study conducted in a small company that provides Information Technology (IT) solutions to Bahia Research Foundation (FAPESB). FAPESB is a government organization that funds scientific research in the state of Bahia, Brazil, since 2001. The organization has the objective to stimulate and support scientific and technologic activities in order to develop the economy and life quality of the people in the state of Bahia.

FAPESB's informatics sector is composed of a software development team and an infrastructure department. The development team is a small group of developers, with experience in web systems, and one designer with experience in human-computer interaction. On the other hand, they did not had experience with data analytics before. The team had already developed a number of systems to FAPESB such as:

- Roberto Santos Plataform[1], that is in the first version;
- Sistema de Bolsas[2], already in production;
- Observatório[3], already in production.

Based on the context of this work, only the last two of them are related to data mining and analysis, and only the Observatório is investigated in depth in this dissertation.

1.3.2 Outline of the Proposal

This work investigates a development process of a data analysis system concerning the general SE practices, the ISE principles and its NFRs. In order to address it, we proposed a case study to be conducted in a small company with real world data-oriented systems. This research is a project with resources constraints and time limits, thereby is necessary a protocol to detail all the planning. The research was designed in agreement with Runeson *et al.* (2012) guidelines, and will present the elements considered valuable in the design of a case study, such as *case and subject selection*, *procedures and roles*, *data collection*, *analysis procedures*, *validity procedures* and *legal and ethical issues*.

1.4 Out of Scope

- **Infrastructure.** Although BI hardware artefacts is a topic that has not much attention by researchers (Trieu, 2017), and FAPESB has an infrastructure department, this aspect will not be investigated; and
- **BI Impacts.** This work does not analyze the impacts that the system can provide for the organization.

1.5 Statement of the Contributions

As a result of the work presented in this dissertation, the following contributions can be highlighted:

- **A study on NFR into analytics solutions.** Which can provide the research and industry communities a view of the attributes used in data analytics solutions;

[1] http://www.robertosantos.fapesb.ba.gov.br/
[2] http://bolsas.fapesb.ba.gov.br/
[3] http://observatorio.fapesb.ba.gov.br/

- **A case study replication.** In order to compare the efficiency of the practices and methods choosed or to expand the work; and
- **A technical reporting** that the company can analyze and extract insights to improve its products.

1.6 Dissertation Structure

The remainder of this dissertation is organized as follows:

- Chapter 2 reviews the essential concepts used through this dissertation: Analytics, its various definitions and benefits, and the definition and types of Non-Functional Requirements (NFR).
- Chapter 3 describes the case study design proposed. It discusses the case selected, data collection, procedures and roles, threats to validity and ethical and legal issues.
- Chapter 4 presents the findings of the case study. Discussing the SE practices used, the principles and NFRs founded.
- Chapter 5 provides the concluding remarks and presents the contributions of this work, limitations and future work.

2

An Overview on Analytics and Non-Functional Requirements

A good decision is based on knowledge and not on numbers.

—PLATO

The use of analytics and business intelligence is bringing important effects on both academic and business communities (Chen *et al.*, 2012). Davenport *et al.* (2010) stated that the use of data and analysis can improve performance in business, by providing managers and decision makers valuable insights to make better decisions. Gibson and Arnott (2010) asserted that there is a need of research about BIA applications development. Moreover, the increase of the volume, variety, and velocity of data has affected in the use of information in business, resulting in new tendencies of fast analytics (Larson and Chang, 2016). Because of that, there is an increasingly need of quality BIA tools.

NFRs are commonly referred as the software's quality aspects (Robertson and Robertson, 2012). Mairiza *et al.* (2010) stated the importance of NFRs to a successful software project. However, the quality attributes are often neglected in software development.

Based on the context of this work, this chapter discusses the understanding of two important topics for this dissertation: Analytics and Non-Functional Requirements (NFR). Thus, Section 2.1 discusses Analytics, its various definitions, characteristics and bene-

Terminology	Time	Specific meaning
Decision support	1970-1985	Use of data analysis to support decision making
Executive support	1980-1990	Focus on data analysis for decisions by senior executives
Online analytical processing (OLAP)	1990-2000	Software for analyzing multi-dimensional data tables
Business Intelligence	1989-2005	Tools to support data-driven decisions, with emphasis on reporting
Analytics	2005-2010	Focus on statistical and mathematical analysis for decisions
Big data	2010-present	Focus on very large, unstructured, fast-moving data

Table 2.1 Terminology for using and analyzing data (Davenport, 2014)

fits.Section 2.2 explains the definitions of NFR, its categories and quality attributes.

2.1 Analytics

2.1.1 Introduction

We are living the era of data analytics. Davenport and Patil (2012) stated that data scientist would be the "the sexiest job of the 21st century." Nevertheless, there is nothing new about the idea of analyzing business data to extract valuable information (Davenport, 2014). This activity has been performed since 1950s and has been named as decision support, executive support, online analytic processing, business intelligence, analytics, and now, big data. Table 2.1 shows the terminology and its specific meaning used through the time. However, analytics has been bringing important effects on businesses across many fields nowadays such as finance, retail, sports and government organizations (Davenport *et al.*, 2010; Chen *et al.*, 2012). Moreover, Buse and Zimmermann (2012) stated that analytics refer to the use of data analysis and exploration to help decision makers extract important information and insights. Chen *et al.* (2012) defined BIA as a set of techniques, technologies, systems, practices, methodologies, and applications that

investigate business data to help managers to understand its business and make decisions. Kinoshita and Mizuno (2017) defined big data as a concept of data that cannot be treated by traditional technologies, because of its large volume, high velocity and variety form. However, Davenport (2014) tried to deconstruct the term big data and according to him, the name is a problematic one from several reasons, such as the term *big* relating to only one characteristic of the new forms of data, which is not the most important aspect for many organizations. However, he affirmed that the concept is revolutionary and allows a transformation on almost every business. Buse and Zimmermann (2012) defined Software Analytics (SA) as analytics on software data to aid the process of software development.

Despite this list of vary terminologies and its definition found in the literature, we consider that all of these terms have the same base meaning: the use of data analysis to extract information to help decisions, of course, respecting the specifications from each one.

2.1.2 Business Inteligence and Analytics (BIA)

Commonly, Business Inteligence and Analytics (BIA) refers to technologies, systems, practices, methodologies and applications that analyze data in order to help managers to understand its business and market (Chen *et al.*, 2012; Lim *et al.*, 2013). Wixom and Watson (2007) present a business intelligence framework including two primary activities: getting data in, i.e., carry data from several systems to an integrated database, and getting data out, business users accesing the information extracted from the database and applications in form of reporting, queries and predictive analysis (see Figure 2.1).

Lim *et al.* (2013) credit the advance of the Web 2.0, social networks, *e-commerce* and blogs to lead what they called Business Intelligence 2.0, that is centered on web-based unstructured data. Moreover, the use of Web 2.0 by the government is creating the opportunity for BIA research in *e-government* systems, turning its process more transparent and participatory (Chen, 2009; Chen *et al.*, 2012). Morabito (2015) also observed an increase use of the Internet by the governments to aid in "transparency, accountability, and public participation activities." Yet this leads to new government challenges such as data ownership, data quality, privacy, civil liberties and equality.

2.1.3 The Benefits of Analytics

Trieu (2017) conducted a literature review on empirical studies in Business Intelligence. The review covered BI research published from January 2000 to August 2015, collecting

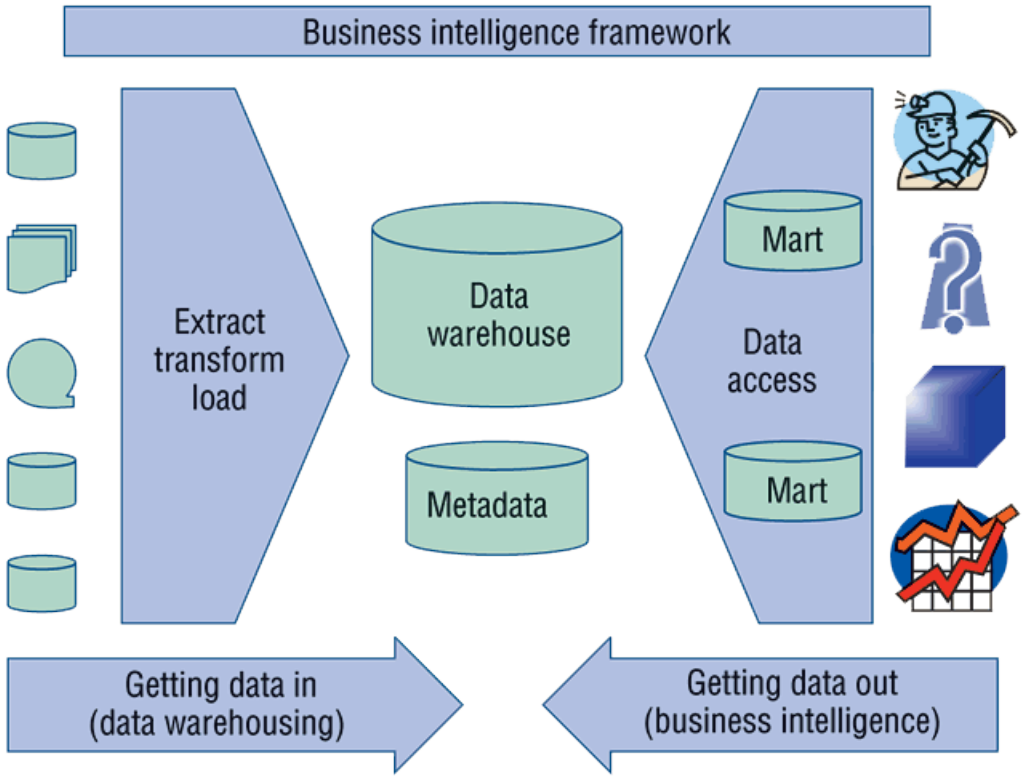


Figure 2.1 BIA framework (Wixom and Watson, 2007)

	Past	Present	Future
Information	What happened? (Reporting)	What is happening now? (Alerts)	What will happen? (Extrapolation)
Insight	How and why did it happen? (Modelling, experimental design)	What's the next best action? (Recommendation)	What's the best/worst that can happen? (Prediction, optimization, simulation)

Table 2.2 Key questions addressed by analytics (Davenport *et al.*, 2010)

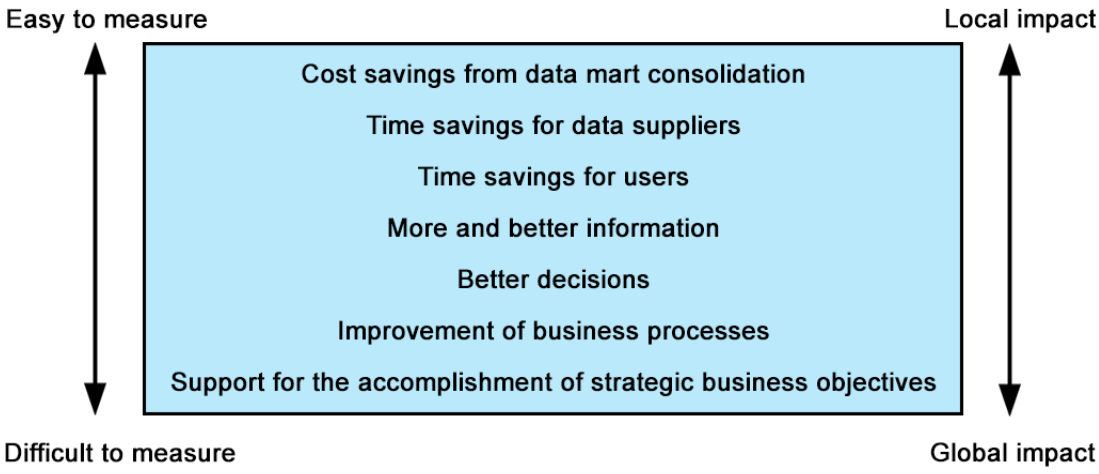


Figure 2.2 Spectrum of BIA benefits (Wixom and Watson, 2007)

106 papers containing the following keywords: "business intelligence", "business analytics", "big data", "data mining" or "data warehousing." They observed that the main focus on BI research is what they called BI Impacts. There are several papers in the literature confirming that analytics can bring real improvements to business efficiencies transforming business processes (Trkman *et al.*, 2010; Park *et al.*, 2012), developing organizational intelligence (Counihan *et al.*, 2002; Lau *et al.*, 2012) and developing new products and services (Deng and Chi, 2012; Chae, 2014). Furthermore, BI tools is another topic that has received attention in the literature. Although, Gibson and Arnott (2010) stated that there is a lack of research about the development of BIA tools.

Davenport *et al.* (2010) enumerate some benefits of being analytical in business such as: understand the dynamic of business; know if interventions is bringing desired results; support investments in IT and information to extract more insights and business value; improve efficiency using predictive models to move quickly; manage risks; detect patterns to anticipate changes; and improve the decision process analyzing the previous decisions taked. Wixom and Watson (2007) complement saying that "as business users mature to performing analysis and prediction, the level of benefits become more global in scope and difficult to quantify." Figure 2.2 shows this spectrum of benefits that BIA bring to businesses.

Furthermore, Davenport *et al.* (2010) stated some questions that analytics can answer. Table 2.2 shows those questions in two dimensions: time frame, if the questions reffer to past, present or future; and Innovation, if the goal is to know information or gain new insights.

2.1.4 Software Analytics (SA)

Along with this list of terminologies, there is another one to refer the use of data analysis in software development. Buse and Zimmermann (2012) defined Software Analytics (SA) as analytics on software data to aid the process of software development, giving software engineers and project managers information to ground their decisions and improve software projects. Menzies *et al.* (2011) developed a manifesto called Inductive Software Engineering (ISE) that enumerates a set of principles to characterize the difference between academic and industrial data mining.

2.1.5 Inductive Software Engineering (ISE) Manifesto

After several analyzes on data mining work, Menzies *et al.* (2011) organized a formalized set of principles to build analytics systems in a fashionable manner. These principles are not intended to be a formal maturity model, but good practices that the authors judged to be valueable. Next, we define the seven principles: Users Before Algorithms; Plan for Scale; Early Feedback; Be Open-Minded; Do Smart Learning; Live With the Data You Have; and Broad Skill Set, Big Toolkit.

Users Before Algorithms

Menzies and Zimmermann (2013) stated that users viewpoint is vital for the project. Understand the users goals early in the project enables to focus on crucial issues. Also, analytics solutions are really useful in industry when it comes to real-world data analysis. Furthermore, in Menzies *et al.* (2011) there are some heuristics that aid to evaluate a successful meeting with users. For instance, the users interrupt the meeting to debate the results presented (it shows that either your results are comming from valuable issues or they are understanding what you are explaining); the users keep bringing data sources to be explored; “the users bring senior management to the meetings”; and the users desire to know more about certain analysis after the meeting.

Plan for Scale

Figure 2.3 shows the definition of data mining applied to real-world context (Fayyad *et al.*, 1996). Although, Menzies *et al.* (2011) observed some issues with the proposal, they extracted many insightful aspects, such as that the data mining is only a small portion of the whole process. Moreover, they noted the cyclic steps of the process: finding one

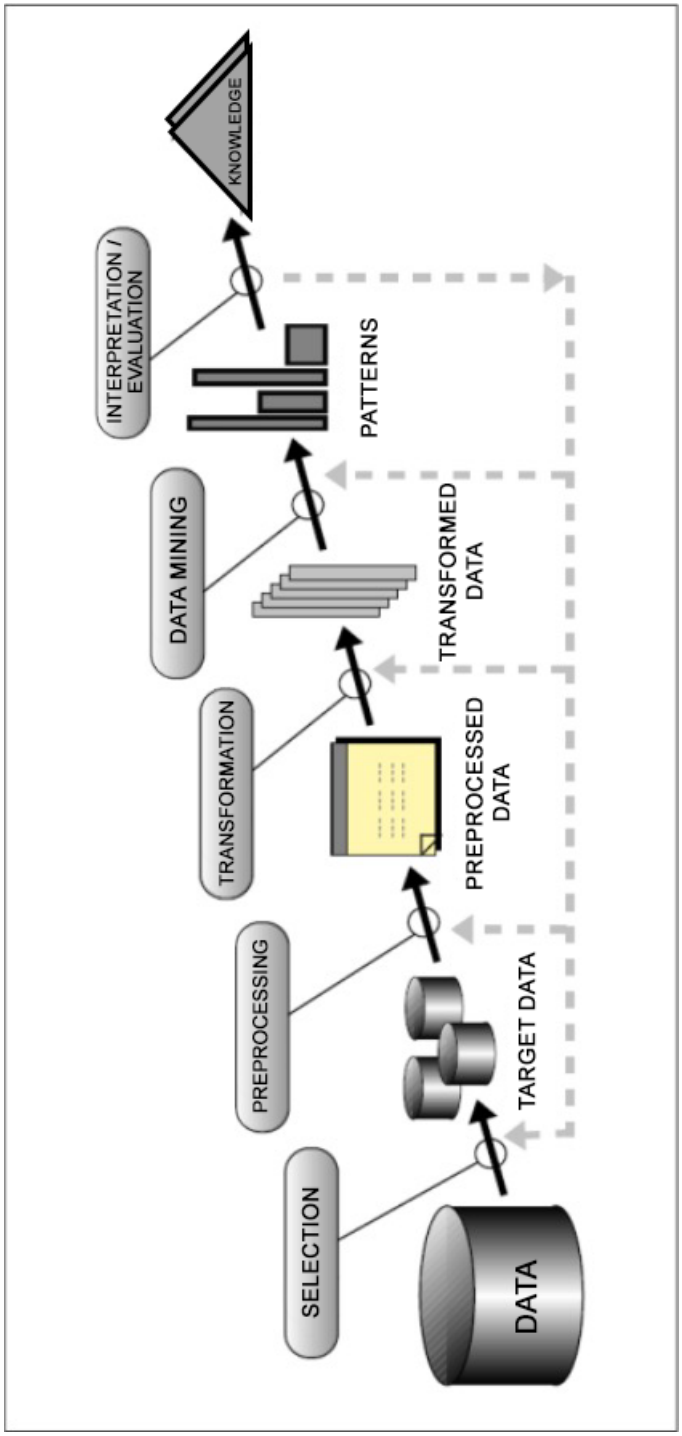


Figure 2.3 Steps of the knowledge discovery in database process (Fayyad *et al.*, 1996).

pattern induct to new questions that aid to refine the data mining goals, leading to another cycle of the process; trying diverse methods to produce feedback that allows users refine and mature the objectives of the project; and trying different methods to discover how to find patterns in data. Finnaly, Menzies *et al.* (2011) stated that this repetitive process implies that in industrial data analytics solutions, the data mining methods are repeated "to answer an extra user question, make some enhancement and/or bug fix to the method, or to deploy it to a different set of users."

Early Feedback

In Menzies *et al.* (2011) experiences, it is complicated for users to explain what they really want from their data. Thereby, getting feedback from users, as early as possible, allows the changes to be done as soon as possible. Moreover, when the results start to show, the requirements quickly mature. Zhang and Xie (2012) suggest the early use of prototypes in order to start a feedback loop with stakeholders.

Be Open-Minded

Following the principle before, having an early and constant feedback allows initial goals to change. Thus, it is important to be open-minded. Menzies and Zimmermann (2013) stated that is not wise perform an analysis with fixed approaches and hypotheses, and try to explore additional ideas to solve a particular issue hard to be solved.

Do Smart Learning

Menzies *et al.* (2011) advise check and validate the results. Since mining methods can uncover misleading data, it is required to apply validation methods to minimize the errors. Furthermore, they recommend "to avoid conclusions based on a very small percent of the data." Zhang and Xie (2012) also affirm the need of the results being evaluated according to real tasks that the users carry out.

Live With the Data You Have

Another principle listed by Menzies *et al.* (2011), suggest to mining with the data that you have on hands, and not the data that you wish. Moreover, before learning the data set is recommended remove spurious data that can be found.

Broad Skill Set, Big Toolkit

Menzies *et al.* (2011) affirm that industry analytics solutions may try a number of methods to produce insightful results to the users. Hence, it is recommended the use of multiple technologies to achieve this level of results. Finnaly, it is desirable to make use of tools with an active community to support.

Along with the principles from the ISE Manifesto, Zhang and Xie (2012) also enumerated certain lessons learned from SA experiences, besides the ones already stated before, such as: "the target problems of software analytics should be what practitioners care about;" and "domain knowledge is important for correctly understanding the data under analysis." For the last one, they present three steps to follow:

1. Data interpretation, the need to understand basic domain-specific terminologies concepts and principles;

2. Data selection, the need to understand the relation among the data and the problem to be solved to select the suitable dataset to perform the analysis; and

3. Data filtering, the need to review and filter the data to avoid misleadingness. To perform it, it is necessary to understand the errors and limitations of the data.

2.2 Non-Functional Requirements

Although Non-Functional Requirements (NFR) are not well defined in the literature (Siegmund *et al.*, 2012), many work refer to them as Quality Attributes (QA) (Chung and Prado Leite, 2009; Robertson and Robertson, 2012; Glinz, 2007).

According to Sommerville (2010), non-functional requirements "are requirements that are not directly concerned with the specific services delivered by the system to its users." They are not essential to place the final software to work, although they are related to their quality aspects and are very important to a successful software project (Chung *et al.*, 2012). NFRs are related to how or in what circumstances the software achieves its goals, and do not define what a component will execute. They concern to the whole system architecture, and may not affect individual parts of the application. Robertson and Robertson (2012) define a NFR as "a property, or quality, that the product must have, such as an appearance, or a speed or accuracy property." Some examples of those properties are reliability, adaptability, securiy, performance, and so on.

	Web System	Information System	Execution Qualities
Performance	✓	✓	✓
Security	✓	✓	✓
Availability	-	✓	✓
Usability	-	✓	✓
Scalability	✓	-	✓
Reliability	-	✓	✓
Interoperability	✓	✓	✓
Adaptability	-	-	✓
Privacy	✓	✓	-

Table 2.3 NFRs selected.

The concept of NFR is not so clear in SE community (Mairiza *et al.*, 2010). Glinz (2005) discusses about to rethink the notion of NFR and define a new classification. Furthermore, Mairiza *et al.* (2010) stated that NFRs are "often neglected, poorly understood and not considered adequately in software development." They are not either elicited or documented at the same time and in details as the Functional Requirements. However, is broadly known as very important to software development.

2.2.1 Types of Non-Functional Requirements (NFR)

There are different types of NFRs stated in literature. Mairiza *et al.* (2010) organized them as: quality attributes, such as maintainability, performance, and reliability; development constraints, e.g., timing, cost, and development personnel; external interfaces requirements, for instance, user interface, look & feel, and system interfacing; business rules, such as production life span; and others, for example, cultural, political and environmental. Furthermore, they observed that the QAs performance, reliability, usability, security and maintainability are the most considered NFRs. They also identified five types of systems with their relevants NFRs showed in Figure 2.4.

Mari and Eila (2003) divided the Non-Functional Requirements (NFR) into two categories, execution and evolution QA. The execution QAs are the ones that can be observed at system's runtime. The execution qualities are: Performance; Security; Availability; Usability; Scalability; Reliability; Interoperability; and Adaptability. On the other hand, evolution QAs are related to static structures of the system. The evolution qualities are: Maintainability, Flexibility, Modifiability, Extensibility, Portability, Reusability, Integrability and Testability.

In this work, we decided to explore the execution qualities listed by Mari and Eila

(2003) and the relevant requirements of both lists, Web Systems and Information Systems, presented in Mairiza *et al.* (2010). The criteria used to determine the attributes to be investigated was to select the requirements that appears at least in two of the three lists. Table 2.3 shows the three sets of NFRs and the selected ones. Although the Privacy attribute appears in two of the lists, we decided do not include in the final set because it is not an Execution requirement and it can be related to the Security QA. Moreover, we decided to include the Adaptability requirement, even it appearing only in one set, as long as the case of our study is an external application and it can be useful to explore this quality attribute.

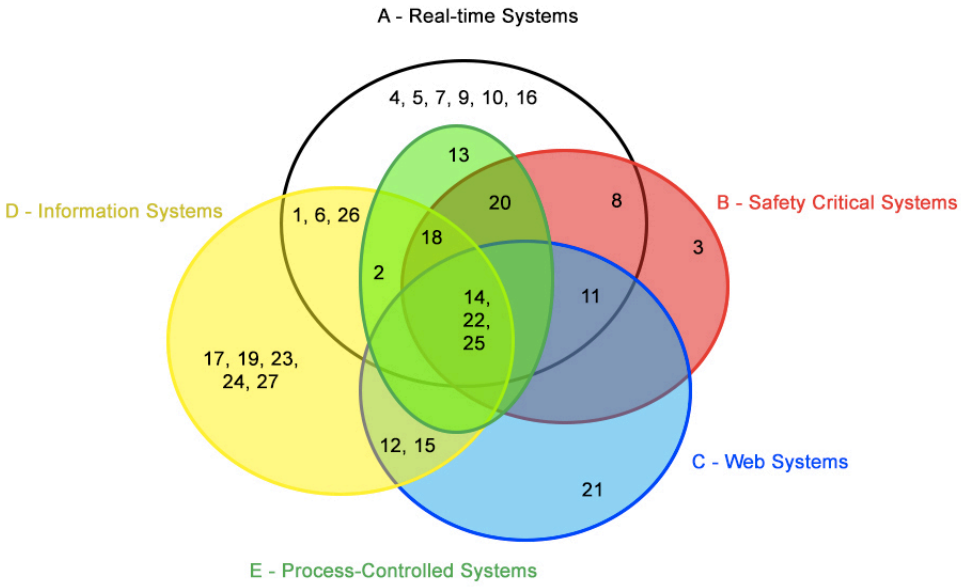
Next, we present the definitions of the eight Non-Functional Requirements (NFR) selected for this work.

- **Performance**: The requirement that specifies the capacity of the software to responds to events in some range of time.
- **Security**: Security requirements specify the system's capability of resist unauthorized access to the application, and denial of service, while providing its services to users.
- **Availability**: This requirement determines the period of time that the systems will be up and running.
- **Usability**: This QA is related to users actions. It specifies the interactions with the system and the effort needed to learn and operate the application. Also concerns the system's efficiency, error avoidance and error handling.
- **Scalability**: Scalability defines how easy is to modify the system to fit a problem or evolve it.
- **Reliability**: This requirement is related to the capability of the system to operate over the time without failure and maintains the performance when used under stated conditions for a range of time.
- **Interoperability**: Interoperability is the ability of the system to communicate with other parts of the own system or external applications.
- **Adaptability**: Adaptability concerns to the ability of the system to adapt itself through different environments or users.

2.3 Chapter Summary

This chapter presented the literature review of the two important topics for this work: Analytics and Non-Functional Requirements. It included fundamental aspects of Analytics such as Business Inteligence and Analytics (BIA) and Software Analytics (SA), its various terminologies and definitions and the importance and benefits of the use of analysis on businesses across many fields. Moreover we discussed about Inductive Software Engineering (ISE) Manifesto and its principles. We also presented the definition of Non-Functional Requirements (NFR), the diverses types of requirements and the relevants NFR for five types of systems. Finally, we defined the eight NFRs selected to conduct this work.

In the next chapter, the elements of the case study design will be presented.



Legend:

1. Accuracy	10. Installability	19. Reusability
2. Availability	11. Integrity	20. Safety
3. Communicativeness	12. Interoperability	21. Scalability
4. Compatibility	13. Maintainability	22. Security
5. Completeness	14. Performance	23. Standardizability
6 Confidentiality	15 Privacy	24. Traceability
7. Conformance	16. Portability	25. Usability
8. Dependability	17. Provability	26. Verifiability
9. Extensibility	18. Reliability	27. Viability

Figure 2.4 Type of Systems and Relevant NFRs (Mairiza *et al.*, 2010)

3

The Case Study Design

Criar é não se adequar à vida como ela é,
Nem tampouco se grudar às lembranças pretéritas
Que não sobrenadam mais.
Nem ancorar à beira-cais estagnado,
Nem malhar a batida bigorna à beira-mágoa.

—WALY SALOMÃO (Sargaços)

3.1 Introduction

The goal of this study is to understand the application of the Inductive Software Engineering (ISE) principles, the Software Engineering (SE) practices and the implementation of Quality Attributes (QA) used during the development of analytics solutions. Trieu (2017) asserted that BIA tools is a dominant research subject in the literature. Although, there is a lack of case studies related to the development of analytics applications (Gibson and Arnott, 2010).

This case study was a project with resources constraints and time limits, thereby was necessary a protocol to detail the planning. In this chapter, we present the elements of the case study design. In agreement with Runeson *et al.* (2012), these elements are the ones to be considered in the design of a case study.

The reminder of this chapter is organized as follows: Section 3.2 presents the case and subject selected; Section 3.3 discusses the procedures followed in this work, the researchers and their roles; Section 3.4 describes the methods to collect the data; Section 3.5 presents the procedures to analyze the data collected; Section 3.6 discusses about

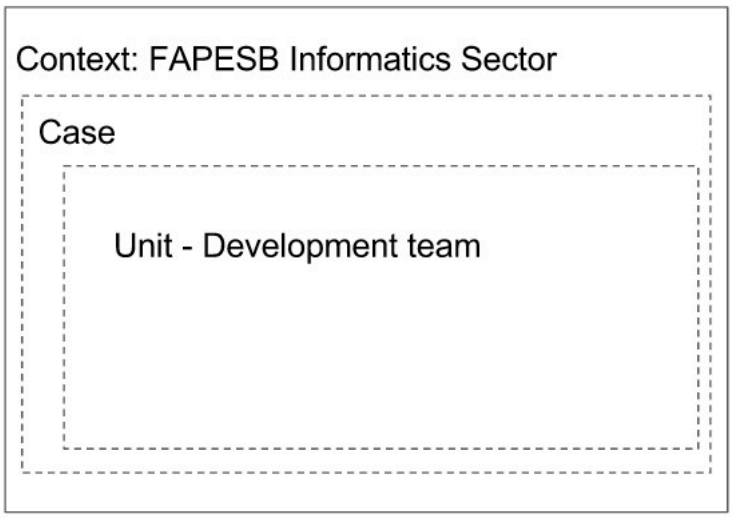


Figure 3.1 Holistic Single-case.

the procedures to validate the case study; Section 3.7 presents the legal and ethical issues; and Section 3.8 summarizes the chapter.

3.2 Case and Subject Selection

According to Runeson *et al.* (2012), in Software Engineering (SE), the case of the study can be anything that is a current SE event in real life. Since the investigation was on a small company, the study was designed as holistic single-case one, as showed in Figure 3.1.

The case selected was the Bahia Research Foundation (FAPESB). FAPESB is a government organization that funds scientific research in the state of Bahia, Brazil. The organization has the objective to stimulate and support scientific and technologic activities in order to develop the economy and life quality of the people in the state of Bahia.

Moreover, the FAPESB informatics sector are composed by a software development team and an infrastructure department. The development team is a small group of developers, with experience in both front-end and back-end development, and one designer with knowledge in human-computer interaction. Furthermore, the chief of the infrastructure department is an expert in IT security.

FAPESB team was chosen to be subject of the case study because:

- The evident rising of analytics in governmental institutions to aid in transparency (Morabito, 2015);

Language	Files	Blank lines	Comment lines	Lines of Code
CSS	4	1192	55	12638
JavaScript	9	166	37	11131
Java	70	914	370	2723
HTML	2	37	8	203
Bourne Shell	1	30	65	138
DOS Batch	1	32	0	113
Maven	1	15	0	97
XML	1	0	0	5
Total	89	2386	535	27048

Table 3.1 Observatório statistics.

- Both organization's personnel and physical locations are accessible for the researchers;
- The size of the team is appropriate so the researchers can take up all the interviewing and analyzing process feasibly in the schedule proposed; and
- The team develops real world data-oriented systems, which can bring valuable insights.

The system produced by FAPESB software development team under study is called Observatório[1] (Observatory). It is a web system that shows the FAPESB data in a visual form, as showed in Figure 3.2. The platform can generate indicators to help users to extract relevant information about the institution. It also supports diverse types of charts to show relations among the data over the years. The main goal is to aid in transparency with the organization's investments. Moreover, Figure 3.3 shows a screen with some of the indicators.

All the figures of the system were edited in order to translate the terms in Portuguese. However, only the text was modified, preserving all the original data and style.

The first phase of the Observatório was developed in six months from requirements gathering to implementation and deployment. The following technologies were used: Java 8, Angular, Bootstrap, HTML, CSS and SQL. According to Cloc tool [2], the Observatório system consists of almost 28k lines of code, excluding comments and blank lines. Table 3.1 shows the code statistics for the project.

[1] http://observatorio.fapesb.ba.gov.br/
[2] https://github.com/AlDanial/cloc

Figure 3.2 Observatório home screen.

Figure 3.3 Some Observatório's indicators.

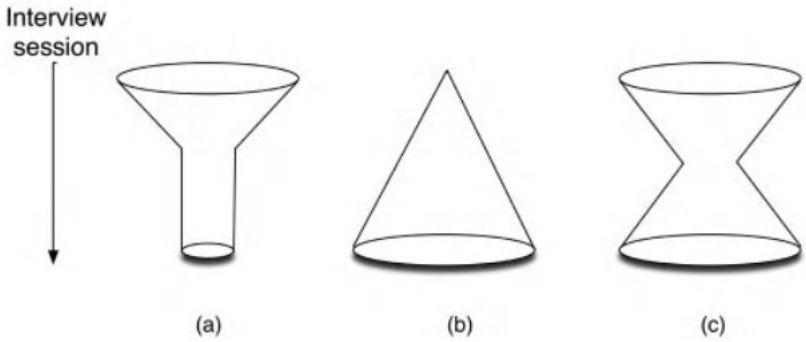


Figure 3.4 General principles for interview sessions: (a) funnel, (b) pyramid, and (c) timeglass.

3.3 Procedures and Roles

The research team was composed by the author of this work, Alex Bruno Paranhos da Silva, and the master student Daniel Amador dos Santos. Although both have previous experience with Software Engineering (SE) in academy and industry, they did not have previous experience with data mining or software analytics.

The researchers conducted the interviews with the subjects alternatively along the sessions. While one performed the interview, the other one assisted the session with possible forgotten issues or raising additional questions.

After the interview sessions, the researchers transcripted all the audio sessions, except the pilot interview. Each researcher conducted the transcription of two audios. Then they shared with each other to review and fix possible misunderstandings.

Next, both researchers started to analyze the data collected. After each one analyzed the data by himself, we discussed in group the findings and came to the conclusion of the results.

3.4 Data Collection

In this case study, we selected interviews as a data collection instrument. According to Runeson *et al.* (2012), it is the most frequently used type of data collections in case studies in Software Engineering (SE), either for primary data collection or as validation of other data.

We interviewed two developers from FAPESB's team: a chief project manager, refered hereafter as Developer Number 1 (D1), and the other a recently-hired programmer analyst, from here as Developer Number 2 (D2). A former developer, called here for Former

Developer (F1), the main responsible for conceiving the Observatório, was interviewed as well. Finally, we conducted an interview with a supporting statistician that also works as a project stakeholder, named here as Statistician and Stakeholder (S1), demanding and validating requirements.

The researchers conducted interviews with the specified team members. Each member that participated in the projects answered questions regarding the principles of the ISE Manifesto and non-functional requirements. While in the questions about the principles specific terminology was omitted, e.g., the exactly name of the principle. However, in the quality attributes questions, we described in the begining the actual non-functional requirements.

For the interview, a semi-structured session was developed. We used a question list defined by us. Whenever we felt that some theme could be more explored, we kept on asking additional questions to clarify the interviewees' ideas and descriptions. The questions were designed using a *funnel* structure, starting from more general topics to more specific questions, as presented in Figure 3.4 (a). The questions were divided into three sections. In the first one, we asked about the participant's background and its description about both systems. The second section we investigated the ISE principles. And the last one, covered the non-functional attributes. Appendix B presents the interview questions.

For each issue to be investigated, we made questions that asked if the principle is used. If so, the question goes deeper on how and if there is something formalized or endorsed by some stakeholder to use that. Moreover, for the quality attributes, we wanted to investigate: if each requirement is present on the systems and how they were specified, designed, implemented and tested. On the other hand, when performing the interview with the Statistician and Stakeholder (S1), we performed the questions from a different point of view. First, we asked, as a stakeholder, if the requirements had been addressed. Next, for data discovery related questions, we were interested in data validation and feedback. Nevertheless, all the questions were made regardless the fact that the S1 could not answer about software programming subjects.

In addition, we also gathered artifacts developed by the team. The participants gently provided us notes about requirements sessions and graphic user interface wireframes used. Figure 4.3 shows one of the wireframes used by the team. The following assets could be appreciated by the reasearchers *in loco*: database schemas, database migration scripts, source code and Java's projects structure. We were allowed to take pictures from their working place whiteboard and some notes attached to the wall.

3.5 Analysis Procedures

Both researchers conducted and recorded, using an audio recorder, all the interviews. The sessions lasted approximately 3.5 hours, plus 1.2 hours of the pilot interview. After listening to the audios sessions, the following activities were defined:

1. Analyze the participants' profile and experiences;
2. Identify the overall software engineering process;
3. Observe if the Inductive Software Engineering (ISE) principles were being applied and how; and
4. Investigate which non-functional requirements were beign used and how.

Regarding the ISE principles, we had a set of questions that tracked each one to the correspondent principle. We checked the participants' answers against each other to assess if they agree or disagree about the affirmations regarding the principles. When they agreed, we could conclude that some fact has a high likelihood to be true. When there was non conformity between the answers, it is likely that for the item asked there is not a standardized approach to deal with it, since the members do not have a common understanding about the referred practice. In some other cases, the answers could be complementary, it is when the participants present different views about an agreeing issue. The artifacts were used to complement this triangulation.

Next, we performed the analysis on the NFR issues. Initially, we analyzed the interviewees' answers in order to find the existing QA in the systems. Sometimes, the participants could provide different views about the presence or not of a specific requirement. For those which we could attest they were in fact present, we investigated how they were specified, designed, implemented and tested. Whenever applicable, we analyzed the artifacts to confirm if the attribute was present or not.

Finally, we collected informations regarding SE processes and methods through the answers about the development process and the observation of their work place.

3.6 Validity Procedures

According to Shull *et al.* (2007), the study often reveals a couple of limitations. Therefore, if they can be spotted early enough, these difficulties can be defeated or mitigated in the case study design.

Before the interviews, we performed a pilot interview as an approach to mitigate threats to construct validity. Runeson *et al.* (2012) stated that pilot interviews are used to certify if the questions are understandable and if it is possible to extract useful answers. After the pilot, we adapted some questions to better understanding. For example, the question about information discovery scalability was understood by the pilot interviewee as software scalability, which was not what we would to investigate at that moment.

Moreover, in terms of minimizing possible internal validity threats, we advised the participants to not comment the interview with their co-workers before all the sessions were concluded. We did that to avoid that interviewed actors to influence the opinions on those subjects yet to participate. We have also inserted some "ice-breaker questions" into the interviews, to relieve inhibitions and tensions between the participants and interviewers. We were afraid that the participants could feel uncomfortable with the whole interview procedures, making them shy when describing the processes used in the company.

Finally, the sample size is a threat to internal validity because it was very small. Even though the Observatório deals with real world data, the size of the system is small in comparison with large cases such as, Twitter and Microsoft (Lin and Ryaboy, 2013; Czerwonka *et al.*, 2013).

3.7 Legal and Ethical Issues

Runeson *et al.* (2012) stated that it is fundamental to case study researchers to pay attention to legal and ethical issues in the case study design, development and reporting of their research, because frequently case studies take account of confidential information. He suggests to be clear at the beginning of how this kind of information will be handled.

In order to protect the interviewees, we developed a consent letter stating that we would preserve the confidentiality of the answers and any kind of personal information. Moreover, any identifying characteristics will be excluded of the report. During the actual sessions, we reaffirmed our commitment to those terms when the participants felt some answer could compromise them. The template of the consent letter is presented in Appendix C.

3.8 Chapter Summary

In this chapter we described the elements of the case study design. Firstly, we presented the subject selected to be studied and the reasons that why we choosed it. Then, we defined the procedures and roles of the resesearch team. Next, we detailed how and which data will be collected. After that, the analysis procedures were stated. Finally, we discussed the threats to validity and legal and ethical issues.

Next chapter presents the results of the case study.

4

Case Study Results

Solitude, récif, étoile
A n'importe ce qui valut
Le blanc souci de notre toile.
—MALLARMÉ (Salut)

Initially, we planned to select both systems: the Observatório and Sistema de Bolsas. Nonetheless, during the interview sessions, we identified a bias from the subjects to talk more about the Observatório than the other one. After an internal meeting, we decided to focus on the Observatório, which had real data analytics characteristics, and to discard the Sistema de Bolsas, because the first one has real business intelligence and analytics characteristics and is robuster than the second one.

Based on the context of this work, this chapter presents the results and concerns about the case study. Section 4.1 presents an overview of the participants of the case study. Section 4.2 discusses the Software Engineering methods and process applied at the system development. Section 4.3 describes the findings related to ISE principles. Section 4.4 presents the results related to Non-Functional Requirements. Finally, Section 4.5 presents the final remarks about the results.

4.1 Participants' Profile

At the beginning of the interviews, personal questions were performed in order to describe their professional background. There are three developers with experience in full software development. However, the most experienced works with software development for five years. All three developers already worked with both back-end and front-end

Code Name	Occupation	SE Experience	Analytics Experience
D1	Chief project manager and developer	5 years	None
D2	Developer	3 years	None
F1	Former developer	4 years	None
S1	Statistician and stakeholder	None	In academy

Table 4.1 Case Study Participants.

programming. Nonetheless, none of them worked with nor data analytics or business intelligence before. On the other hand, the statistician, chief of the scholarships department at FAPESB, has knowledge about analytics, but no skills with programming. Moreover, there are two more stakeholders that were mentioned by the participants. Table 4.1 summarizes the participants' profile.

4.2 Software Engineering (SE) Process and Methods

According to the members interview, they use an agile-like development process, i.e., they use the philosophy and inspiration from agile methods, but use a custom methodology. According to D1, there are daily meetings, as well as weekly meetings, only with the developers. In those meetings, the members discuss the projects and the technical issues related to them, such as software architecture. Figure 4.2 shows the Scrum-like white board that the team use, on which they post sticky notes, each one related to a task. The board is divided in several categories representing a status. The notes are moved around depending on the project condition.

It was interesting to note that the developers clearly consider to be negative to have too much documentation. D2 expressed her dislike towards the use of extensive documents; D1 states that a well-structured source code and a good architecture have more value than documentation.

Moreover, the idea of avoiding documentation was true even for requirements elicitation phase. Each member used his/her own notes to register their impressions about what needs to be done and implemented, as identified from the notebook sheets provided by the team members in Figure 4.1. A final requirement file is not produced at all, at the end of the elicitation. It seems that making those notes intelligible to other people are not a concern; they work more as a team artifact.

Finally, the subjects agreed that F1 had freedom and autonomy to decide most of the technical aspects of the software. F1 states that while participated in the team meetings, his work flow was set by himself only. Based on those testimonials, we could

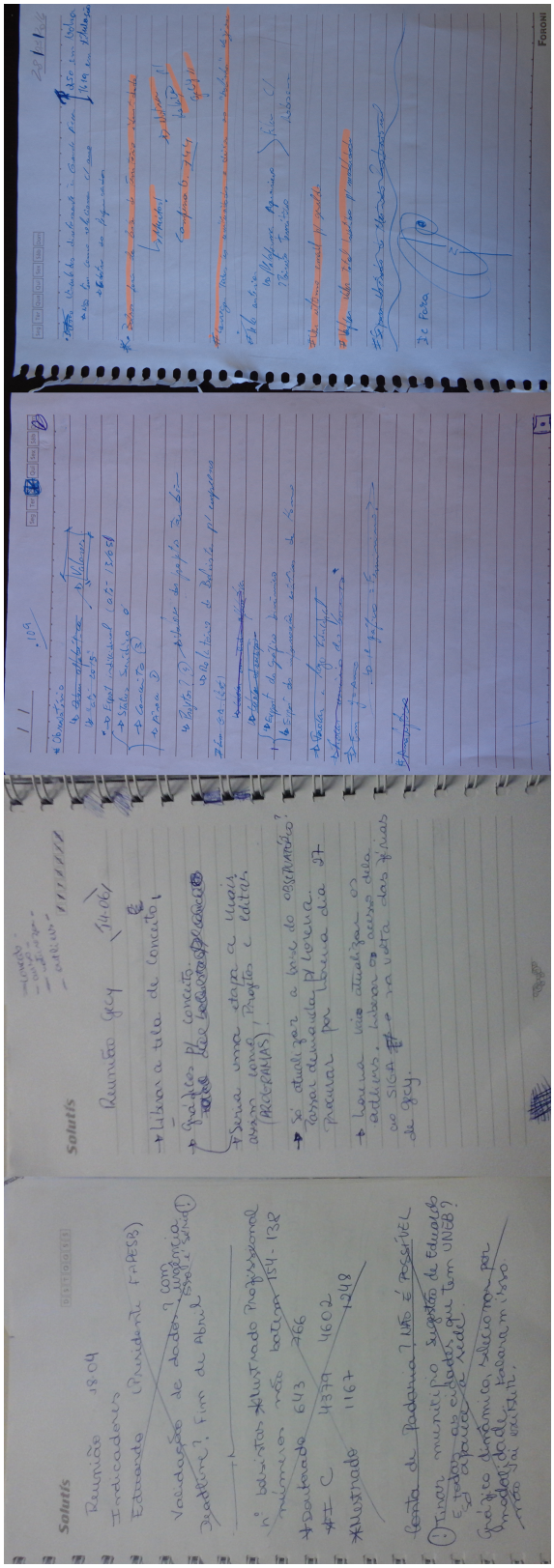


Figure 4.1 Notebook sheets.

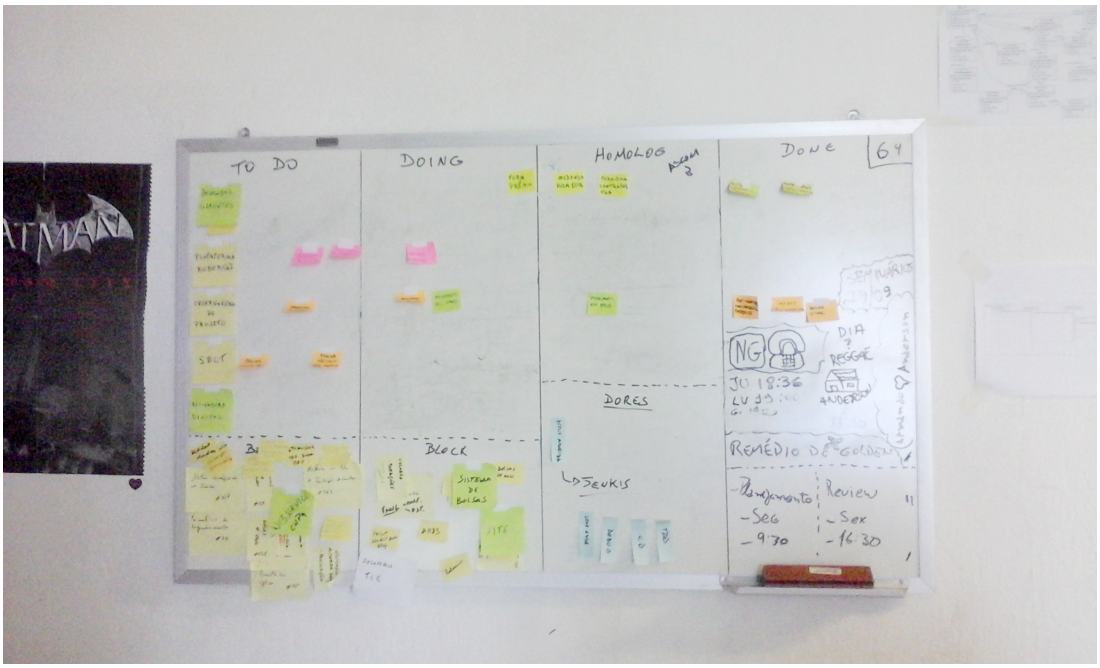


Figure 4.2 White board used by the team.

conclude that the development was centered on this developer in terms of design and implementation.

4.3 ISE Manifesto Principles

4.3.1 Users Before Algorithms

Regarding the principle "users before algorithms," all developers were very emphatic that the user needs were the ultimate motivation when implementing filters and algorithms for new information discovery. According to D1, the team discussed how present the data in a certain manner, e.g., removing worthless information. D1 attested that the development team was always thinking what is the relevance of the information and even how that might be visualized to the end user, in terms of usefulness and elegance. Furthermore, S1 said that whenever the stakeholders felt some new filter or algorithm was not fulfilling their needs, they had the freedom to communicate the team about it and request them to adapt the algorithm somehow.

4.3.2 Plan for Scale

In the principle "plan for scale," D2 mentioned how well designed the software architecture was made by F1. According D2, in order to search for new information, it is possible to use the existing implementation through an interface which the developer can set an additional parameter. D1 reported that for the Observatório finds new data relations is straightfoward (due to the ease to produce new graphics). However, sometimes they had to find information in legacy databases that were not well conceived. When it happened, performing that task became quite difficult. F1 described the search algorithm as "dumb," because it returned the search exactly how the developer describe it, but the normalized database might help in making discovery easier. None of the interviewees could mention a formalized process that the former developer followed to implement this scalability in information discovery.

In terms of helping information discovery, F1 stated that by building a well-structured database, a successive mining process became facilitated. We could verify that observing the new database schema and modelling diagram, that is simpler than the oldest, and the simplicity of the database migration script, what already deals with inconsistencies.

4.3.3 Early Feedback

Regarding "early feedback," according to developers, the current ones and the former, the access to the stakeholders is quite direct. They have access to S1's room and phone easily. The validation of the information can happen instantly in case she is not busy. Therefore, S1 confirmed that the developers asked for data validation even when some discovery was not finished. We could infer that there was none bureaucracy when the team needs to validate an information discovery with a stakeholder. According to F1's reading, it it would be unfeasible to receive feedback only in the latter development stages. As D1 affirmed, although the president has limited presence availability, he is often virtually available through mobile phone messenger applications.

4.3.4 Be Open-minded

About the "be open-minded" principle, D2 said that a filter's algorithm is only modified when it is not working properly. F1 confirmed that, and from his point of view, there was no time available for doing so. He goes even further when he says that information out of scope defined with the stakeholders were find spontaneously. In S1's point of view, the developer had a proactivity to go through different paths when discovering information,

but hardly ever they have to refactor a search method from the scratch once they have started to implement it. Although, D1 described that sometimes they find inconsistent data (since the systems receive their data directly from the users, manually, and they might provide wrong data), which force the development team to approach its processing and interpretation in a different way. However, D1 stated that it is something that emerges spontaneously. There is nothing formalized about that. F1 also described this situation in a similar way. Thus, we considered that the principle was not adopted by the development team.

4.3.5 Do Smart Learning

Regarding to "do smart learning" principle, strangely, we had conflicting answers. F1 said that there was not statistical validation at all. However, S1 affirmed that was performed statistical analysis to verify the correctness of the answers shown by the Observatório. Moreover, the current developers confirmed that S1 and a additional person with a position in FAPESB were responsible for data validation, so the information discovered had a solid meaning. Even there were conflicted answers, we could concluded that statistic validation was performed. Because the positive answers out number the negative and there are specific stakeholders with statistical experience able to perform it.

4.3.6 Live With The Data You Have

We considered the principle "live with the data you have" as applicable, since the developers had to make the most of the data at hand. All the participants agreed that was very difficult to work with the original database. F1 stated that they found several cases of misleading data and wrong relation among data. Moreover, D1 declared some procedures to deal with this possibles failures such as, the migration to a new database and the removal of spurious data manually. The researchers could observe that exploring the database schema and the migration script.

4.3.7 Broad Skill Set, Big Toolkit

About "broad skill set, big toolkit," when asked if there was any specific tools for information extraction, D1 said that any retrieval is made manually. They have written their own scripts to do that. Moreover, D1 understands that using a proper tool would made that task more effective. We asked both current developers if they could make future predictions using the existing systems and methods. They stated that all the information

available is about the past years; the system does not have any data about the ongoing year. Furthermore, S1 clarified that with the information that the system has, at the time this research was done, it is not able to generate valid predictions. She also said that the system needs to be fed with more data in order to generate suitable analyses so they can be statistically trustworthy. F1 stated that there was neither a specific tool for discovering information nor there was any effort to bring new insights about data that was already mined.

4.4 Non-Functional Requirements

4.4.1 Performance

Regarding system performance, the current developers said that it was not a customer's requirement at the first moment. On the other hand, F1 mentioned that the customer required that the system performance had to be satisfactory. Besides the stakeholders' request, the researchers could observe an inherent motivation from the team to make the system runs with acceptable performance. For all the interviewees, the most important decision was to create a new database. According to the subjects, the general one (the database used for the whole organization) had several inconsistencies that made its usage difficult. Moreover, the team adopted a number of actions to increase the performance level, e.g., F1 listed the relocation of the system process to the client side, the choice of techonologies with an excellent performance and the design of their architecture. Moreover, D1 declared the use of flat design was a choice to increase the system performance. However, the web-servers performance concerns responsabilities are delegated to the infrastructure department. The employees monitor the number of access and also run some tests such as stress test.

4.4.2 Security

According to F1, security is one of the least critical requirements, because Observatório is a read-only system and all its data is public. Moreover, the application does not store personal data such as email or name. Thus, the risk to happen a data leak is unfeasible. According to F1, it is not possible to modify the data through the system, because there are not insert commands. The customer also did not request any security attribute. Furthermore, the chief of the infrastructure department, that is an expert in security, runs a few tests to ensure that the system will not be affected by SQL injections or DDoS

attacks.

4.4.3 Availability

As stated before, the development team is not in charge of the infrasctructure. Thereby, availability was not a concern to the team. Although it is essential for the customer that the system is always available since it is open to external access. From the user's point of view, the system availability is an attribute achieved, S1 stated that never happened that the system was offline or crashed. Nevertheless, no participant could answer clearly what was done to maintain the system availability. The team did not perform tests to ensure the availability.

4.4.4 Usability

As the Observatório is an open system with external access, usability was a concern for the development team. Although, the usability was not critical for the customer, as D1 declared, the customer cares about the functionality. Nevertheless, there is a designer specialist in human-computer interaction in the team that developed the whole system interface. They made use of prototyping to validate the concepts of the screens. Figure 4.3 shows a wire-frame of the home screen. Moreover, the team also did not execute usability tests.

The Figure 4.3 was edited in order to translate the terms in Portuguese. However, only the text was modified, preserving all the structure.

4.4.5 Scalability

D1 affirmed that the Observatório was planned to be scalable. The customers required it because new features could be added to the system, such as new charts or indicators. Furthermore, they are implementing new modules in the Observatório with new data set. F1 stated that the main point for the scalability is the architecture. Thus, the development team deals with scalability at source code level making use of a MVC-based architecture, with a well separated structure of the Model, View and Controller packages. Also, they added two more packages of Services and another one to only generates the JavaScript code. Finally, we verified the use of Java reflection on source code to generate the methods to produce new charts in runtime.

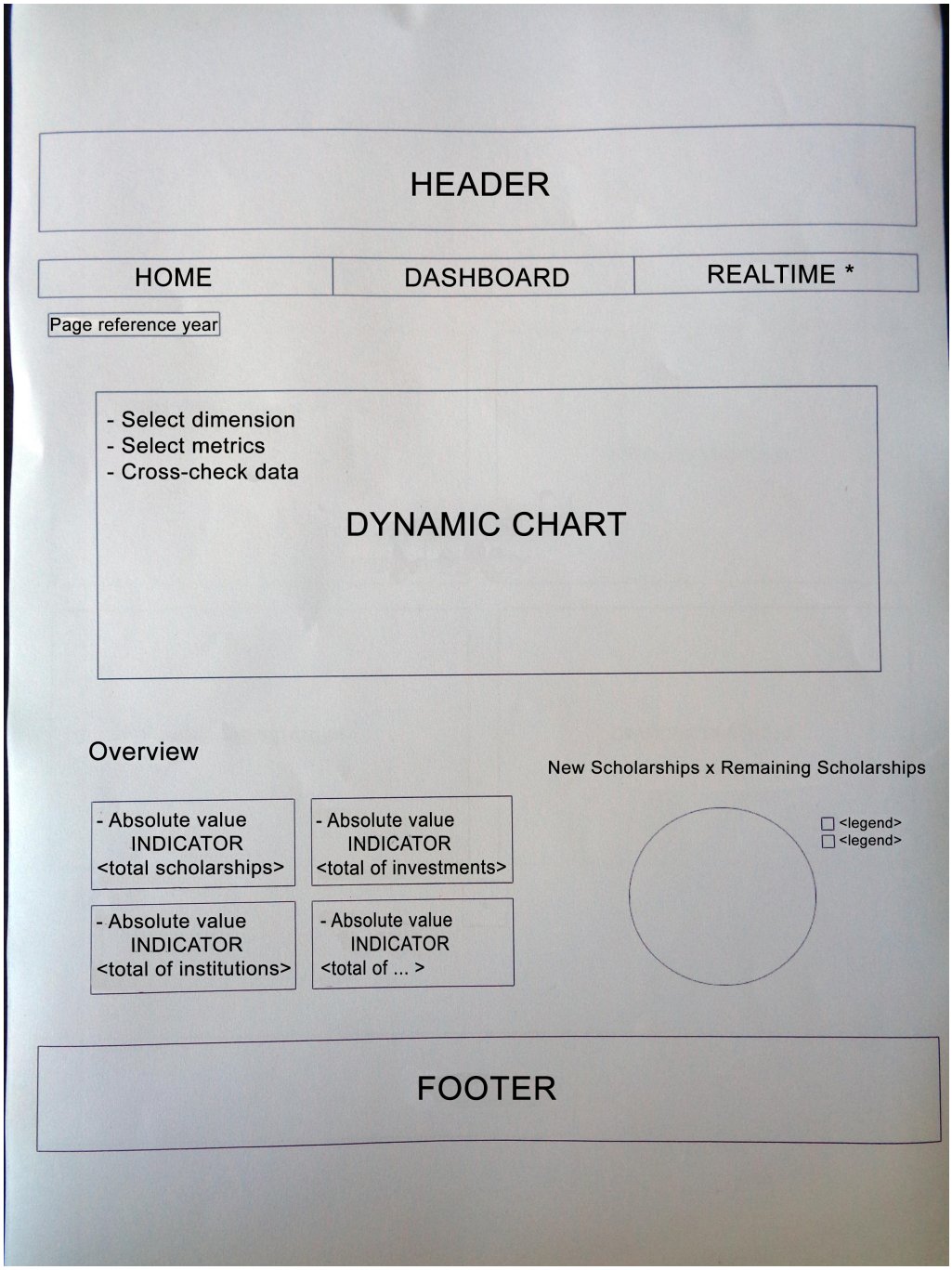


Figure 4.3 Mockup from Observatório's screen.

4.4.6 Reliability

The team did not address reliability issues. F1 joked about the fact when he states that the action for that is simple "do things right.," such as the use of good programming practices. They experienced a few minor problems such as incompatibility with some browser, however, nothing critical related to the system reliability. The team did not have a specific plan to handle with this. They fix the bugs whenever the developers perceive that something is not working as it should.

4.4.7 Interoperability

Regarding to quality attribute "interoperability," all the interviewees agreed that is not a concern. The system does not communicate with any other application. As stated by D1, the Observatório uses its own database and do not uses any API or external services. Although, there is a data migration from an external database which occurs mannualy through a SQL script.

4.4.8 Adaptability

Finally, the adaptability was a requirement from the customer. They said that the system should execute on several devices such as desktops, tablets, and smartphones, besides the most used browsers, since it is available to the external public. In addition, the team used the responsive layout approach to achieve the adaptability attribute. On the other hand, the team did not execute tests formally or use tools to verify the level of adaptability. The researchers tested the system on several devices to verify the system adaptability.

4.5 Conclusion

We observed the adoption of 4 from the 7 principles stated in the literature (Table 4.2): Users Before Algorithm; Early Feedback; Do Smart Learning; and Live With the Data You Have. Moreover, we could conclude that the principle Plan for Scale was partially applied, because the scalability observed was about the system and the generation of new charts and indicators, not for information discovery.

As showed in Table 4.3, we found 4 from the 8 QA: Performance, due to the actions adopted to improve system performance; Usability, as a result of the use of prototyping and the expertise in human-computer interaction of a team member; Scalability, as the

Principles	Adoption
Users Before Algorithms	✓
Plan for Scale	Partially
Early Feedback	✓
Be Open-Minded	-
Do Smart Learning	✓
Live With the Data You Have	✓
Broad Skill Set, Big Toolkit	-

Table 4.2 The principles adopted in the project.

Non-Functional Requirements	Adoption
Performance	✓
Security	-
Availability	-
Usability	✓
Scalability	✓
Reliability	-
Interoperability	-
Adaptability	✓

Table 4.3 The Non-Functional Requirements adopted in the project.

architecture modelling and the new database; and Adaptability, due to the tests performed by the researchers.

Larson and Chang (2016) asserted that the waterfall methodology and traditional software development process are not good practices to accomplish BIA solutions. In accordance with that, we observed some characteristics related to Software Engineering (SE) processes as follows: the use of a non-formalized flexible development process, leaving a developer to work freely in a project, a self-managed teams culture, and a preference to avoid extensive documentation.

4.6 Chapter Summary

This chapter presented the main findings of the case study. Initially, we presented the data collected and how the analysis was conducted. Next, we started the analysis of the results. We presented the participant's profile in order to show the interviewees background and experiences; the findings about SE processes and methods; the results concerning the ISE principles; and the outcomes related to NFRs. Finally, we presented the final remarks about the results.

Next chapter presents the concluding remarks and future work of this dissertation.

5
Conclusions

eu sou como eu sou
vidente
e vivo tranquilamente
todas as horas do fim.

—TORQUATO NETO (cogito)

Business Inteligence and Analytics (BIA) has proven to be an important activity to improve performance on both industry and academy (Davenport *et al.*, 2010; Chen *et al.*, 2012). Managers are adopting data analysis and exploration to extract information to aid on decision making in many areas, such as in sports, finance, social media, software development and so on (Chen, 2009; Meredith and O'Donnell, 2010; Chau and Xu, 2012; Menzies and Zimmermann, 2013). Furthermore the use of web by governments is leading to BIA researches into *e-government* applications (Chen *et al.*, 2012). Governments are using the web to aid in transparency, accountability and public participation activities (Morabito, 2015). Morevover, many work discussed the significant improvements that BIA brings to organizational performance (Counihan *et al.*, 2002; Lee and Park, 2005; Chae, 2014). Nevertheless, Gibson and Arnott (2010) stated that there is a lack of research on BIA tools development.

Although the concept of Non-Functional Requirements (NFR) is not clear enough in literature, they are acknowledged as very important to software development (Mairiza *et al.*, 2010). NFRs do not define what the system will do, however, they are related to how or in which circumstances the software achieves its goals. Furthermore, they stated that NFRs are "often neglected, poorly understood and not considered adequately in software development." Moreover, developers do not pay enough attention to them

and do not either elicit or document at the same detail as the Functional Requirements.

Chapter 2 summarized an overview of the basic concepts about analytics, Business Inteligence and Analytics (BIA), Software Analytics (SA) and Non-Functional Requirements (NFR). We also discussed how we defined the attributes to be studied.

In order to understand the development process of analytics solutions concerning the concepts previously stated, this dissertation presented a case study design to be conducted in a small company. We defined the elements of our study as the case selected, its context and the reasonings to be choosed; the procedures and roles performed by the research team; the definition of data collection; the analysis procedures; the threats to validity; and the legal and ethical concerns.

The findings of the case study are: the participants profile and early experiences, the main SE methods and processes used by the development team, the four ISE principles adopted in the project and the NFRs implemented in the project. These results highlight some characteristics related to SE processes such as, the use of a non formalized flexible development methodologie and a preference to avoid extensive documentation, in accordance with Larson and Chang (2016). The adoption of the following ISE principles: *users before algorithms*, *early feedback*, *do smart learning* and *live with the data you have*. Moreover, the following NFRs were implemented in the project: *performance*, *usability*, *scalability* and *adaptability*.

5.1 Research Contribution

The main contributions of this research are described as follows:

- **An investigation of a particular BIA tool,** that can contribute with the research community to explore more cases on BIA tools development.
- **A study on NFR into analytics solutions.** This can provide to both academy and industry communities a view of which types of QA are relevant to analytics solutions. Moreover, investigate the Non-Functional Requirements (NFR) in order to provide a better understand of them;
- **A case study to future replications.** In the future, researchers can use this work as baseline to either compare other analytics solution developments or expand it; and
- **A technical reporting for the company.** The company can use this report to extract important information in order to improve its product and development processes.

5.2 Future Work

We are aware that this dissertation covers a simple case scenario and there are others aspects to be investigated. However, we think that this work is a good start point to stimulate the community in order to produce more research in this area. Thus, some important topics can be investigated as future work:

- **Impacts on organization.** An exploratory study must be conducted in order to analyze if the application brings real benefits for the organization;
- **Case study replication.** It will be interesting conduct a replication study in other contexts in order to reinforce or refute the findings.
- **Further evaluation at the same company.** It can be interesting to investigate the development of other modules of the system with the same company to investigate the principles.

References

Abbasi, A., Albrecht, C., Vance, A., and Hansen, J. (2012). Metafraud: A meta-learning framework for detecting financial fraud. *MIS Q.*, **36**(4), 1293–1327.

Borrajo, F., Bueno, Y., de Pablo, I., Santos, B., Fernndez, F., Garca, J., and Sagredo, I. (2010). Simba: A simulator for business education and research. *Decision Support Systems*, **48**(3), 498 – 506. New concepts, methodologies and algorithms for business education and research in the 21st century.

Buse, R. P. L. and Zimmermann, T. (2012). Information needs for software development analytics. In *Proceedings of the 34th International Conference on Software Engineering*, ICSE '12, pages 987–996, Piscataway, NJ, USA. IEEE Press.

Chae, B. K. (2014). A complexity theory approach to it-enabled services (iess) and service innovation: Business analytics as an illustration of {IES}. *Decision Support Systems*, **57**, 1 – 10.

Chau, M. and Xu, J. (2012). Business intelligence in blogs: Understanding consumer interactions and communities. *MIS Q.*, **36**(4), 1189–1216.

Chen, H. (2009). Ai, e-government, and politics 2.0. *IEEE Intelligent Systems*, **24**(5), 64–86.

Chen, H., Chiang, R. H. L., and Storey, V. C. (2012). Business intelligence and analytics: From big data to big impact. *MIS Q.*, **36**(4), 1165–1188.

Chung, L. and Prado Leite, J. C. (2009). Conceptual modeling: Foundations and applications. chapter On Non-Functional Requirements in Software Engineering, pages 363–379. Springer-Verlag, Berlin, Heidelberg.

Chung, L., Nixon, B. A., Yu, E., and Mylopoulos, J. (2012). *Non-functional requirements in software engineering*, volume 5. Springer Science & Business Media.

Counihan, A., Finnegan, P., and Sammon, D. (2002). Towards a framework for evaluating investments in data warehousing. *Information Systems Journal*, **12**(4), 321–338.

Czerwonka, J., Nagappan, N., Schulte, W., and Murphy, B. (2013). Codemine: Building a software development data analytics platform at microsoft. *IEEE Software*, **30**(4), 64–71.

Davenport, T., Harris, J., and Morison, R. (2010). *Analytics at Work: Smarter Decisions, Better Results*. Harvard Business Publishing. Harvard Business Press.

Davenport, T. H. (2014). *Big Data at Work: Dispelling the Myths, Uncovering the Opportunities*. Harvard Business Review Press, USA.

Davenport, T. H. and Patil, D. J. (2012). Data Scientist: The Sexiest Job Of the 21st Century. *Harvard Business Review*, **90**(10), 70–76.

Demirkan, H. and Delen, D. (2013). Leveraging the capabilities of service-oriented decision support systems: Putting analytics and big data in cloud. *Decision Support Systems*, **55**(1), 412 – 421.

Deng, X. and Chi, L. (2012). Understanding postadoptive behaviors in information systems use: A longitudinal analysis of system use problems in the business intelligence context. *Journal of Management Information Systems*, **29**(3), 291–326.

Fayyad, U. M., Piatetsky-Shapiro, G., and Smyth, P. (1996). Advances in knowledge discovery and data mining. chapter From Data Mining to Knowledge Discovery: An Overview, pages 1–34. American Association for Artificial Intelligence, Menlo Park, CA, USA.

Gibson, M. and Arnott, D. (2010). A participatory case study of business intelligence systems development. In *Proceedings of the 2010 Conference on Bridging the Socio-technical Gap in Decision Support Systems: Challenges for the Next Decade*, pages 199–210, Amsterdam, The Netherlands, The Netherlands. IOS Press.

Glinz, M. (2005). Rethinking the notion of non-functional requirements. In *in Proceedings of the Third World Congress for Software Quality (3WCSQ'05*, pages 55–64.

Glinz, M. (2007). On non-functional requirements. In *15th IEEE International Requirements Engineering Conference (RE 2007)*, pages 21–26.

Kinoshita, E. and Mizuno, T. (2017). *What Is Big Data*, pages 91–101. Springer International Publishing, Cham.

Ko, M. and Osei-Bryson, K.-M. (2004). Using regression splines to assess the impact of information technology investments on productivity in the health care industry. *Information Systems Journal*, **14**(1), 43–63.

Kou, G., Liu, X., Peng, Y., Shi, Y., Wise, M., and Xu, W. (2003). Multiple criteria linear programming approach to data mining: Models, algorithm designs and software development. *Optimization Methods and Software*, **18**(4), 453–473.

Larson, D. and Chang, V. (2016). A review and future direction of agile, business intelligence, analytics and data science. *Int. J. Inf. Manag.*, **36**(5), 700–710.

Lau, R. Y., Liao, S. S., Wong, K.-F., and Chiu, D. K. (2012). Web 2.0 environmental scanning and adaptive decision support for business mergers and acquisitions. *MIS Quarterly*, **36**(4), 1239–1268.

Lee, J. H. and Park, S. C. (2005). Intelligent profitable customers segmentation system based on business intelligence tools. *Expert Systems with Applications*, **29**(1), 145 – 152.

Lim, E.-P., Chen, H., and Chen, G. (2013). Business intelligence and analytics: Research directions. *ACM Trans. Manage. Inf. Syst.*, **3**(4), 17:1–17:10.

Lin, J. and Ryaboy, D. (2013). Scaling big data mining infrastructure: The twitter experience. *SIGKDD Explor. Newsl.*, **14**(2), 6–19.

Mairiza, D., Zowghi, D., and Nurmuliani, N. (2009). Managing conflicts among non-functional requirements. In *Australian Workshop on Requirements Engineering*. University of Technology, Sydney.

Mairiza, D., Zowghi, D., and Nurmuliani, N. (2010). An investigation into the notion of non-functional requirements. In *Proceedings of the 2010 ACM Symposium on Applied Computing*, SAC '10, pages 311–317, New York, NY, USA. ACM.

Mari, M. and Eila, N. (2003). The impact of maintainability on component-based software systems. In *Euromicro Conference, 2003. Proceedings. 29th*, pages 25–32. IEEE.

Menzies, T. and Zimmermann, T. (2013). Software analytics: So what? *IEEE Softw.*, **30**(4), 31–37.

Menzies, T., Bird, C., Zimmermann, T., Schulte, W., and Kocaganeli, E. (2011). The inductive software engineering manifesto: Principles for industrial data mining. In *Proceedings of the International Workshop on Machine Learning Technologies in Software Engineering*, MALETS '11, pages 19–26, New York, NY, USA. ACM.

Meredith, R. and O'Donnell, P. (2010). A functional model of social media and its application to business intelligence. In *Proceedings of the 2010 Conference on Bridging the Socio-technical Gap in Decision Support Systems: Challenges for the Next Decade*, pages 129–140, Amsterdam, The Netherlands, The Netherlands. IOS Press.

Morabito, V. (2015). *Big data and analytics: Strategic and organizational impacts*. Springer International Publishing.

Park, S.-H., Huh, S.-Y., Oh, W., and Han, S. P. (2012). A social network-based inference model for validating customer profile data. *MIS Q.*, **36**(4), 1217–1237.

Robertson, S. and Robertson, J. (2012). *Mastering the Requirements Process: Getting Requirements Right*. Addison-Wesley Professional, 3rd edition.

Runeson, P., Host, M., Rainer, A., and Regnell, B. (2012). *Case Study Research in Software Engineering: Guidelines and Examples*. Wiley Publishing, 1st edition.

Shull, F., Singer, J., and Sjøberg, D. I. (2007). *Guide to Advanced Empirical Software Engineering*. Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Siegmund, N., Rosenmüller, M., Kuhlemann, M., Kästner, C., Apel, S., and Saake, G. (2012). Spl conqueror: Toward optimization of non-functional properties in software product lines. *Software Quality Journal*, **20**(3-4), 487–517.

Sommerville, I. (2010). *Software Engineering*. Addison-Wesley Publishing Company, USA, 9th edition.

Speier, C. and Morris, M. G. (2003). The influence of query interface design on decision-making performance. *MIS Q.*, **27**(3), 397–423.

Trieu, V.-H. (2017). Getting value from business intelligence systems. *Decis. Support Syst.*, **93**(C), 111–124.

Trkman, P., McCormack, K., de Oliveira, M. P. V., and Ladeira, M. B. (2010). The impact of business analytics on supply chain performance. *Decision Support Systems*, **49**(3), 318 – 327.

Wang, H., Wei, Q., and Chen, G. (2013). From clicking to consideration: A business intelligence approach to estimating consumers' consideration probabilities. *Decision Support Systems*, **56**, 397 – 405.

Watson, H. J., Wixom, B. H., Hoffer, J. A., Anderson-Lehman, R., and Reynolds, A. M. (2006). Real-time business intelligence: Best practices at continental airlines. *Information Systems Management*, **23**(1), 7–18.

Wixom, B. H. and Watson, H. J. (2007). The current state of business intelligence. *Computer*, **40**, 96–99.

Zhang, D. and Xie, T. (2012). Software analytics in practice: Mini tutorial. In *Proceedings of the 34th International Conference on Software Engineering*, ICSE '12, pages 997–997, Piscataway, NJ, USA. IEEE Press.

Appendix

A

Available Resources

In order to make this work public and facilitate a replication, a repository is available containing all the artifacts produced for the planning and analyzing of this case study. The repository can be accessible at http://github.com/alexbrunops/mate27casestudy.

B

Interview Questions

B.1 Basic Questions

Duration estimate: $\sim$ 15 minutes

Q.1.1 What is your name?

Q.1.2 What is your job at FAPESB?

Q.1.3 How long do you work at FAPESB?

Q.1.4 Do you like working here?

Q.1.5 Which software development process do you like most?

Q.1.6 Is this your first experience with data analytics? If not, which was?

Q.1.7 How long have you been working on the projects?

Q.1.8 Can you describe both projects to us?

Q.1.9 Which software engineering practices do you use?

Q.1.10 Which tools did you use for data extraction?

B.2 Inductive Software Engineering Principles

Duration estimate: $\sim$ 30 minutes

B.2.1 Users Before Algorithms

Q.2.1.1 When you implement some algorithm or some feature, do you think about the benefits that they bring to the users? How?

Q.2.1.2 Is there some plan or action for that? Is there any kind of software engineering process to help on that?

Q.2.1.3 Have you developed some prototype? How was the prototype development process?

Q.2.1.4 How is the relationship with the clients? How close is the development team to the clients?

Q.2.1.5 How were the meetings with the clients?

B.2.2 Plan for Scale

Q.2.2.1 When you implement some algorithm or method to discover a certain information, this same algorithm can be used to perform another discovery?

Q.2.2.2 What practices do you adopt to ensure that?

Q.2.2.3 Is there some software engineering process or tools that help you to achieve that (for instance, design patterns, specific programming language feature, agile methods)?

B.2.3 Early Feedback

Q.2.3.1 When you implement a filter or discover some new information, how early is this information verified with users?

Q.2.3.2 Is there any practice to capture client's feedback in a systematic way?

Q.2.3.3 How was the requirements elicitation to extract the information from the databases?

Q.2.3.4 Is this connected to the feedback from the system?

B.2.4 Be Open-minded

Q.2.4.1 Is there some effort to discover new strategic informations from the data that you have?

Q.2.4.2 When you go to investigate the database, do you do that with a fixed objective in mind or do you do predisposed to find new relations (new informations)?

Q.2.4.3 Is there any plan for that or is it intuitive?

B.2.5 Avoid Bad Learning

Q.2.5.1 Is there any validation about the information discovered?

Q.2.5.2 Is there some statistical methods (validation?) in the process?

Q.2.5.3 Is this in the software development process?

Q.2.5.4 Through which practices?

B.2.6 Live with the data you have

Q.2.6.1 Imagine that the client required a feature that depends of certain type of data that you do not have at the moment, but it will come someday. In a strict deadline scenario, do you prefer wait for the new data or do you prefer try to solve the problem with the data do you already have? Why?

Q.2.6.2 Is there something in the software development process that allow you to think that way? If yes, how?

B.2.7 Broad skill set, big toolkit

Q.2.7.1 If there is an efficient way to extract the information, is there some effort to change the methods and algorithms or keep the way that it is?

Q.2.7.2 Is there some systematic way?

Q.2.7.3 How is this process?

Q.2.7.4 When do you identify the need?

B.3 Non-Functional Requirements

Duration estimate: $\sim$ 30 minutes

B.3.1 Performance

Q.3.1.1 How do you deal with the system performance issues?

Q.3.1.2 Which specific tests do you apply to ensure the performance?

Q.3.1.3 Is the requirement defined by the clients?

Q.3.1.4 How is it specified, designed, implemented and tested?

B.3.2 Security

Q.3.2.1 How do you deal with the system security issues?

Q.3.2.2 Is there some access profile or login feature?

Q.3.2.3 How this security issues influenced the software development process?

Q.3.2.4 How is it specified, designed, implemented and tested?

B.3.3 Availability

Q.3.3.1 Are there concerns about the availability from the client?

Q.3.3.2 Is this a task that the development team are in charge or is delegated to another department?

Q.3.3.3 How is it specified, designed, implemented and tested?

B.3.4 Usability

Q.3.4.1 Is there any concern about usability?

Q.3.4.2 Is there someone able to design the layout?

Q.3.4.3 How is the usability evaluated?

Q.3.4.4 In which development process step do you deal with usability concerns?

Q.3.4.5 How is it specified, designed, implemented and tested?

B.3.5 Scalability

Q.3.5.1 How do you deal with the system scalability?

Q.3.5.2 Is the process integrated to data discovery?

Q.3.5.3 Is there something at the software architecture that supports the growth of the system?

Q.3.5.4 How is it specified, designed, implemented and tested?

B.3.6 Reliability

Q.3.6.1 Do you have actions to verify if the system can run for a long period without maintenance?

Q.3.6.2 Did you experience any critical failure?

Q.3.6.3 Which actions did you took?

Q.3.6.4 How is it specified, designed, implemented and tested?

B.3.7 Interoperability

Q.3.7.1 Does the system communicate with other application?

Q.3.7.2 How?

Q.3.7.3 How is it specified, designed, implemented and tested?

B.3.8 Adaptability

Q.3.8.1 Did you have any concerns about the system being adaptable to diverse platforms?

Q.3.8.2 Or is it a requirement from the client?

Q.3.8.3 Is the system adaptable to diverse devices such as, smartphones and tablets?

Q.3.8.4 How is it specified, designed, implemented and tested?

C

Template of Consent Letter

Consent Information Letter
"Understanding the development of data analytics solutions"

This interview is conducted by the researchers Alex Bruno Paranhos da Silva and Daniel Amador dos Santos as part of the case study "**Understanding the development of data analytics solutions: A case study**." The investigation is needed as a partial evaluation activity in the masters course "Case Study in Software Engineering," lectured by Professor Dr. Eduardo Santana de Almeida.

You are invited to participate in this interview that aims to understand the development process of data analytics systems implemented in FAPESB. The interviews sessions will be conducted in the form of questions and answers.

We would like to emphasize that:

- Your participation is voluntary;
- You are free to refuse to answer any question;
- You are free to withdraw at any time.

At the end of the study, the interview will be available, fully or partially, as an appendix from the final research report. The additional artifacts will not be publicized. However, your name, or any identifying characteristics, will not be included in the report.

Salvador, _____ _____, 2016.

_____

Interviewee signature