FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Engenharia Reversa de Padrões de Interação

Clara Sacramento

WORKING VERSION

Mestrado Integrado em Engenharia Informática e Computação

Supervisor: Ana Paiva (PhD)

June 9, 2014

Engenharia Reversa de Padrões de Interação

Clara Sacramento

Mestrado Integrado em Engenharia Informática e Computação

June 9, 2014

This work is financed by the ERDF - European Regional Development Fund through the COMPETE Programme (operational programme for competitiveness) and by National Funds through the FCT - Fundação para a Ciência e a Tecnologia (Portuguese Foundation for Science and Technology) within project FCOMP-01-0124-FEDER-020554.

Abstract

Graphical user interfaces (*GUIs*) are populated with recurring behaviors that vary only slightly. For example, authentication (login / password) is a behavior common to many software applications. However, there are different behaviors between different implementations of this behavior. Sometimes a message appears when the user does not enter the correct data, sometimes, the application software only erases entered data and shows no indication to the user. These recurring behaviors (*UI patterns*) are well identified in the literature.

The goal of this dissertation is to continue the work already done on an existing tool called PARADIGM-RE, a dynamic reverse engineering approach to extract User Interface (UI) Patterns from existent Web applications. As such, we will develop a data analysis module with the goal of improving and substantiate the existing identifying heuristics set, and we will extend the current set of identifiable patterns.

First the theme to be developed during the course of the dissertation is introduced, starting by defining the context and issue at hand and describing the goals of this dissertation. Afterwards we present a literary review on reverse engineering approaches, approaches that infer patterns from Web applications, and data mining algorithms and tools relevant to the problem. Lastly, we will provide an estimated work plan for the project development.

Resumo

As interfaces gráficas estão populadas de comportamentos recorrentes que variam apenas ligeiramente. Por exemplo, a autenticação (*login/password*) é um comportamento comum a muitas aplicações de software. No entanto, há comportamentos diferentes entre diferentes implementações desse padrão. Por vezes, aparece uma mensagem quando o utilizador não introduz os dados correctos, outras vezes, a aplicação de software apenas apaga os dados introduzidos e não apresenta indicação nenhuma ao utilizador. Estes comportamentos recorrentes (*padrões de interface*) estão bem identificados na literatura.

O objetivo deste trabalho é dar continuidade ao trabalho já realizado numa ferramenta existente chamada PARADIGM-RE, uma abordagem de engenharia reversa dinâmica para extrair padrões de interface de aplicações Web existentes. Como tal, vamos desenvolver um módulo de análise de dados, cujo objetivo é melhorar e fundamentar as heurísticas de identificação existentes definidas, e vamos ampliar o atual conjunto de padrões identificáveis.

Primeiro é introduzido o tema a ser desenvolvido durante da dissertação, a começar por definir o contexto e o assunto em questão e descrever os objetivos desta dissertação. Em seguida apresentamos uma revisão literária sobre abordagens de engenharia reversa, abordagens que inferem padrões de aplicações Web, e os algoritmos e ferramentas relevantes para o problema de análise de dados. Por fim, iremos fornecer um plano de trabalho estimado para o desenvolvimento do projeto.

Acknowledgements

"You should be glad that bridge fell down.
I was planning to build thirteen more to that same design"

Isambard Kingdom Brunel

Contents

List of Figures

LIST OF FIGURES

List of Tables

Abbreviations

FEUP	Faculdade de Engenharia da Universidade do Porto *(Faculdade de Engenharia da Universidade do Porto)*
RIA	Rich Internet Applications
API	Application Programming Interface
DSL	Domain Specific Language
AUA	Application Under Analysis
CIO	Concrete Interaction Objects
EFG	Event Flow Graph
GUI	Graphical User Interface
SUT	System Under Testing
TDD	Test-Driven Development
UI	User Interface
HTML	HyperText Markup Language
UML	Unified Modeling Language
REGUI	Reverse Engineering of Graphical User Interface
XML	eXtensible Markup Language

Chapter 1

Introduction

This chapter aims at giving a general overview about the themes addressed by this dissertation. We will address the context in which the dissertation is inserted, as well as the motivation that led to its proposal. Furthermore there will be a brief description of the main objectives of this dissertation, and the methods that will be used to achieve those objectives.

1.1 Context

Web applications are getting more and more important. Due to their stability and security against losing data, there is a growing trend to move applications towards the Web, with the most notorious examples being Google's mail and office software applications. Web applications can now handle tasks that before could only be performed by desktop applications [G+05], like editing images or creating spreadsheet documents.

Despite the relevance that Web applications have in the community, they still suffer from a lack of standards and conventions [CL02], unlike desktop and mobile applications. This means that the same task can be implemented in many different ways, which makes automated testing difficult to accomplish and inhibits reuse of testing code.

GUIs *(Graphical User Interfaces)* of all kinds are populated with recurring behaviors that vary slightly. For example, authentication *(login/password)* is a common behavior in many software applications. However, the implementation of those behaviors may vary significantly. For a login, in some cases an error message may appear when the authentication fails; in others, the software application simply erases the inserted data and doesn't send a message to the user. These behaviors (patterns) are called User Interface (UI) patterns [VWVDVE01] and are recurring solutions that solve common design problems. Due to their widespread use, UI patterns allow users a sense of familiarity and comfort when using applications.

1.2 Motivation and Objectives

This dissertation is part of an investigation project named PBGT *(Pattern-based GUI Testing)* [MPM13]. The goal of this investigation project is to develop a model-based GUI testing tool and approach, usable as an industrial tool. This project has five parts: a DSL *(Domain Specific Language)* named **PARADIGM** to define GUI testing models based on UI patterns; a modeling and testing environment, named **PARADIGM-ME**, made to support the creation of test models; an automatic test case generation tool, named **PARADIGM-TG**, that generates test cases from test models defined in PARADIGM; a test case execution tool, named **PARADIGM-TE**, which executes test cases, analyzes their coverage, and returns detailed execution reports; and finally **PARADIGM-RE**, a Web application reverse engineering tool whose purpose is to extract UI patterns from Web pages without access to their source code, and use the extracted patterns to generate a test model defined in PARADIGM.

The relationship between the different components can be better understood in Figure 1.1. The activities (rounded corner rectangles) with the human figure mean that they are not fully automatic requiring manual intervention. The activities with the cog mean that part (or all) of that activity is automatic. The numbers within the activities define their sequencing.

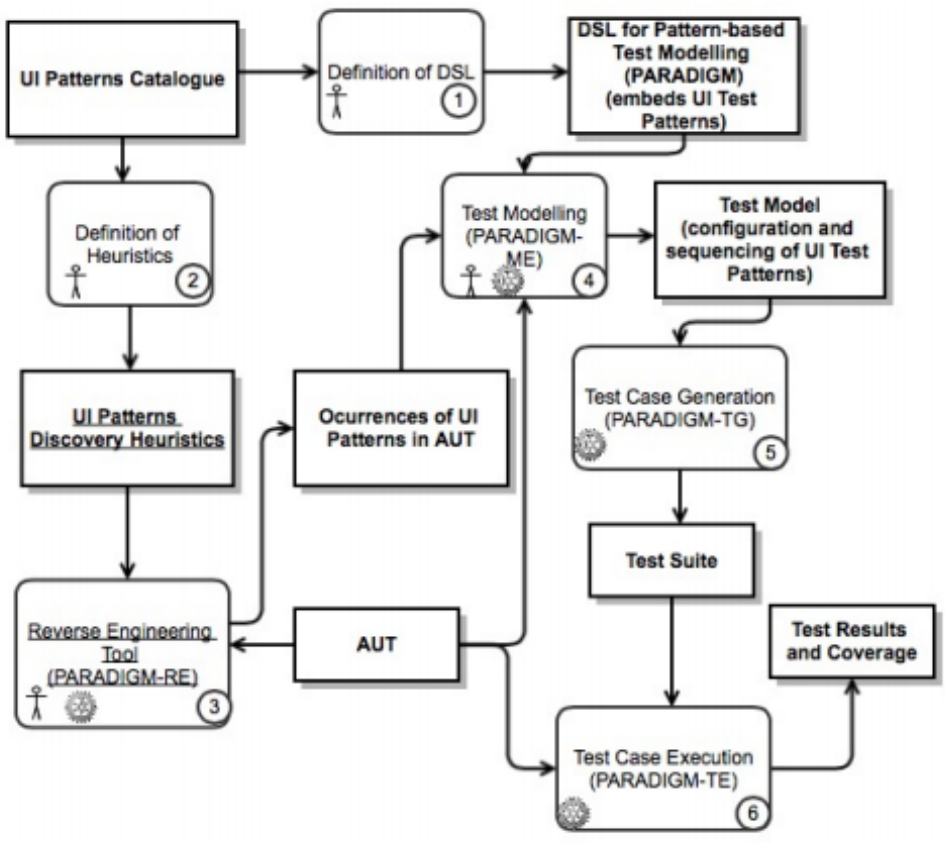

Figure 1.1: An overview of the PBGT project [NPCF13]

The proposal aims to continue the work done on PARADIGM-RE [NPCF13]. This tool identifies interface patterns using Machine Learning inference with the Aleph ILP system [1] running on user interaction execution traces produced using Selenium [2]. It was deemed necessary then to transform the whole process into an iterative one, with the model being updated at every iteration.

This was accomplished in [NPF14], where the tool was extended with a pattern identifying module using heuristics. The current structure of the tool can be seen in Figure 1.2.

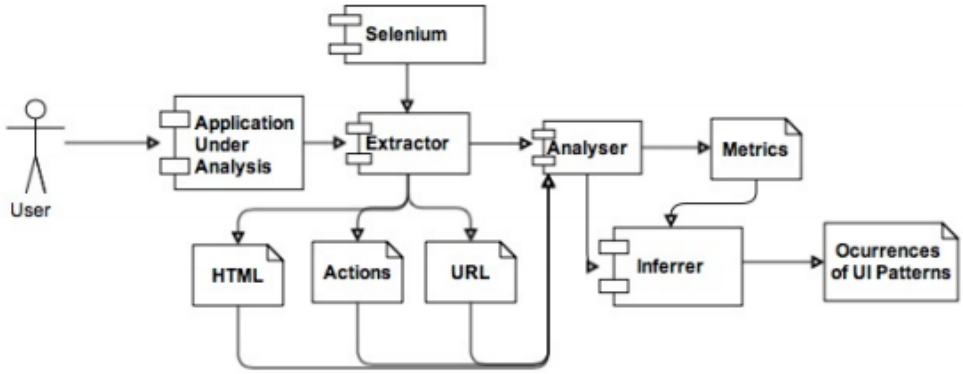

Figure 1.2: Structure of the PARADIGM-RE tool [NPF14]

The user interacts with the Web application, using Selenium to save the actions taken. An example of execution traces saved by Selenium can be seen on table 1.1.

amazon		
actionType	**element**	**text**
open	/	
type	id=twotabsearchtextbox	tablet
clickAndWait	css=input.nav-submit-input	
select	id=sort	label=Most Popular
click	id=pagnNextString	
click	id=pagnNextString	
clickAndWait	link=Image	
clickAndWait	link=Detail	
click	link=android tablet	
click	css=li.refinementImage >a.. >span.refinementLink	
clickAndWait	css=#result_3 >h3.newaps >a >span.lrg.bold	
clickAndWait	link=Explore similar items	

Table 1.1: An example of execution traces produced on the Amazon.com website.

The **extractor** saves the HTML of all pages visited, their URLs, and the actions taken in each page. All that information is passed along to the **analyzer**, whose purpose is to produce

[1] Aleph: http://www.cs.ox.ac.uk/activities/machlearn/Aleph/aleph_toc.html
[2] Selenium: http://docs.seleniumhq.org/

metrics like page ratios, differences between consecutive pages, and others, from the data given. Those metrics are passed to the **inferrer**, who runs the heuristics suite, identifies existing patterns, and produces a XMI file with the occurrences found.An example of the contents of such a file, produced consuming the execution traces in Table 1.1, can be found on Listing 1.1.

```
<?xml version="1.0" encoding="UTF-8"?>
<Paradigm:Model xmi:version="2.0" xmlns:xmi="http://www.omg.org/XMI"
  xmlns:Paradigm="http://www.example.org/Paradigm" title="patterns"/>
  <nodes xsi:type="Paradigm:Init" name="XInit" number="1"/>
  <nodes xsi:type="Paradigm:Sort" name="Sort1" number="2"/>
  <nodes xsi:type="Paradigm:End" name="End" number="3"/>
</Paradigm:Model>
```

Listing 1.1: An example of a .paradigm file with identified patterns

This approach was evaluated on several worldwide used Web applications and the results were deemed satisfactory, since the tool identified most of the occurring patterns and their location on the page. However, there are some patterns the tool struggles with, such as the Menu pattern, and the heuristics are considered to be still in an incipient state.

1.3 Expected Contributions

As stated before, the main goal for this research work is to improve and continue the work on the PARADIGM-RE. The primary task is to adapt a learning algorithm for the tool, in order to improve the existing pattern identifying heuristics, and the other goals are extending the existent identification of patterns and implementing the prodution of a PARADIGM model for the PARADIGM-ME tool to process. As such, either an existing data mining tool will be used in conjunction with the PARADIGM-RE tool to create a data model, or a data mining framework will be integrated into the tool. Examples of tools and frameworks considered can be found in Section **??**.

For the identification of patterns, the information available is the execution traces produced by a user and the HTML code and URLs of all the visited pages. Since execution traces can be considered paths in which we wish to extract sub-paths, **association rule learning** has been considered, along with its subtype, **sequential pattern mining**. Sequential pattern mining is preferred because general association rule mining typically does not consider the order of items, as opposed to sequential pattern mining. Other possible venues to pursue are **classification** algorithms, in which the classes would be the current identifiable patterns, and **clustering** algorithms. These alternatives probably won't be followed, since the data is mostly text and most classification and clustering algorithms deal with numeric values; consequently, to follow this alternative some transformation of the data would be required to pursue this approach. All study done related to data mining can be found on the Section **??**.

There is also the fact that the previously considered paths would only mine execution traces and ignore the other available data (HTML pages, URLs, metrics) so further research is needed

to find a way to include that data into the pattern mining process. A possible way could be to use Inductive Logic Programming (ILP) in which one of the steps run would be the sequential pattern mining algorithm, but it would possibly require more time than is allocated for this dissertation. Further study is needed to choose the optimal approach to follow.

1.4 Structure of the Report

This document is structured into four main chapters. In this first section, Chapter 1, we start by introducing the theme to be developed during the course of the dissertation, starting by defining the context, motivations, goals of this dissertation and the issue at hand.

Chapter 2 introduces essential concepts to understand the problems with which this document deals, presents the state of the art of approaches that reverse-engineer Web applications and popular capture-replay tools, and lastly, gives some context about data mining algorithms and presents relevant data mining tools and frameworks and how they will be applied to this work.

Chapter **??** outlines the main steps in the development of this dissertation (and the respective software prototype) and attempts to provide a feasible schedule for the execution of the work to be done.

Chapter **??** sums up the what has been defined in the report, emphasizing the problem that the dissertation addresses and the work that will be executed towards solving that problem. It will also give a brief idea of what are the expected results at the end of the project.

Chapter 2

Revisão Bibliográfica

Neste capítulo é descrito o estado da arte e são apresentados trabalhos relacionados para mostrar o que existe no mesmo domínio e quais os problemas em aberto. Deve deixar claro que existe uma oportunidade de desenvolvimento que cobre alguma falha concreta .

O capítulo deve também efetuar uma revisão tecnológica às principais ferramentas utilizáveis no âmbito do projeto, justificando futuras escolhas.

2.1 Introdução

Neste capítulo é ilustrada a utilização de macros LaTeX para definir entradas no índice remissivo e são feitas diversas referências bibliográficas, usando-se texto de um artigo apresentado na Conferência XATA2006 [**?**].

Nos últimos tempos têm surgido diversas soluções, apresentadas por empresas do sector Automação de Sistemas para a disponibilização de sistemas *SCADA/DMS* na *Web*.

Aliquam sollicitudin facilisis sapien. Mauris tincidunt tristique diam. Mauris sollicitudin pede at tellus varius volutpat. Integer vel leo. Nunc massa diam, egestas eu, venenatis at, porttitor ac, sapien. Sed magna elit, vulputate in, lacinia sed, lobortis ac, urna. Proin cursus massa id risus. Vestibulum libero. Curabitur venenatis augue. Mauris eu libero eget lectus tempus tempor. In tincidunt, justo in varius adipiscing, ipsum enim gravida massa, eget ornare ante lacus id est. Praesent vitae est ut elit convallis convallis. Aenean tincidunt, purus id consectetur volutpat, sem leo pulvinar libero, nec semper sem purus ultricies nibh [**?**].

Fusce risus mi, tristique eu, consectetuer id, auctor sed, elit. Donec laoreet. Duis consectetuer interdum libero. Etiam eu orci. In eu arcu. Fusce luctus diam eget lectus. Duis interdum lacus sed ligula. Proin vestibulum felis eget lacus. Vivamus vestibulum, tellus ut congue viverra, mauris lacus tempor turpis, eu congue nisi magna at dolor. Ut molestie vehicula libero. Praesent in neque sed risus tempus ornare. Donec hendrerit, erat eu semper aliquam, pede nulla dapibus risus, ut

pretium orci pede et neque. Etiam eget tortor a metus convallis viverra. Quisque eget nisi sed orci facilisis interdum. Aliquam non felis.

2.2 Secção Exemplo

Scalable Vector Graphics é uma linguagem em formato XML que descreve gráficos de duas dimensões. Este formato padronizado pela W3C (*World Wide Web Consortium*) é livre de patentes ou direitos de autor e está totalmente documentado, à semelhança de outros W3C standards [**?**].

Sendo uma linguagem XML, o *SVG* herda uma série de vantagens: a possibilidade de transformar *SVG* usando técnicas como XSLT, de embeber *SVG* em qualquer documento XML usando *namespaces* ou até de estilizar *SVG* recorrendo a CSS (*Cascade Style Sheets*). De uma forma geral, pode dizer-se que *SVG*s interagem bem com as atuais tecnologias ligadas ao XML e à Web, tal como referido em [**?**, **?**].

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Donec a eros. Phasellus non nulla non massa venenatis convallis. In porta. Mauris quis magna. Proin mauris eros, aliquet id, eleifend vitae, semper quis, erat. Aliquam id lectus non odio dignissim blandit. Vestibulum porttitor arcu ut ligula. Nunc quis erat. Curabitur ipsum tortor, ornare vitae, dapibus pretium, hendrerit sed, urna. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Phasellus bibendum, nulla eget varius aliquam, tortor nulla sollicitudin quam, vel vestibulum nisl magna at sem. Aliquam velit sapien, ultrices viverra, tempus quis, ultrices at, dui. Aliquam sit amet justo.

Quisque tristique, metus eu iaculis sagittis, urna leo bibendum diam, a ultricies sem diam a augue. Mauris consectetuer, libero vel euismod tincidunt, nisi metus viverra ante, quis pretium sapien odio nec risus. Nunc semper auctor nulla[1].

2.2.1 Subsecção Exemplo

Batik é um conjunto de bibliotecas baseadas em *Java* que permitem o uso de imagens *SVG* (visualização, geração ou manipulação) em aplicações ou *applets* [**?**]. O projeto Batik destina-se a fornecer ao programador alguns módulos que permitem desenvolver soluções especificas usando *SVG* [**?**].

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Nunc eu nulla. Pellentesque vitae nibh ultrices quam iaculis convallis. Aliquam purus eros, varius eget, volutpat sodales, imperdiet nec, lacus. Curabitur in elit sed sem rutrum posuere. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Duis sem. Praesent ultricies odio vel sapien. Integer faucibus malesuada libero. Cras semper, dolor id ullamcorper varius, magna risus volutpat felis, id pellentesque nulla ante at erat. Integer sodales.

Quisque sit amet odio. In at risus sit amet turpis interdum posuere. Maecenas iaculis vehicula sem. Ut leo arcu, malesuada vel, imperdiet id, dignissim a, purus. Duis eleifend, lectus non venenatis dignissim, risus libero imperdiet mi, nec gravida massa libero sed mauris. Nullam lobortis

[1] Exemplo de nota de rodapé.

libero non sapien. Integer convallis iaculis erat. Morbi dictum. Ut ultrices pellentesque velit. Cras ac ante. Etiam in neque tincidunt lacus gravida vehicula. Proin et nisi.

2.2.2 Subsecção Exemplo

Loren ipsum dolor sit amet, consectetuer adipiscing elit. Praesent sit amet sem. Maecenas eleifend facilisis leo. Vestibulum et mi. Aliquam posuere, ante non tristique consectetuer, dui elit scelerisque augue, eu vehicula nibh nisi ac est. Suspendisse elementum sodales felis. Nullam laoreet fermentum urna.

Loren ipsum dolor sit amet, consectetuer adipiscing elit. Praesent sit amet sem. Maecenas eleifend facilisis leo. Vestibulum et mi. Aliquam posuere, ante non tristique consectetuer, dui elit scelerisque augue, eu vehicula nibh nisi ac est. Suspendisse elementum sodales felis. Nullam laoreet fermentum urna.

Duis eget diam. In est justo, tristique in, lacinia vel, feugiat eget, quam. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Fusce feugiat, elit ac placerat fermentum, augue nisl ultricies eros, id fringilla enim sapien eu felis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Sed dolor mi, porttitor quis, condimentum sed, luctus in.

2.3 Resumo ou Conclusões

No final do capítulo deverá ser apresentado um resumo com as principais conclusões que se podem tirar.

Vivamus non nunc nec risus tempor varius. Quisque bibendum mi at dolor. Aliquam consectetuer condimentum risus. Aliquam luctus pulvinar sem. Duis aliquam, urna et vulputate tristique, dui elit aliquet nibh, vel dignissim magna turpis id sapien. Duis commodo sem id quam. Phasellus dolor. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos.

Chapter 3

Visualização de Sinópticos SVG

Este capítulo deve começar por fazer uma apresentação detalhada do problema a resolver[1] podendo mesmo, caso se justifique, constituir-se um capítulo com essa finalidade.

Deve depois dedicar-se à apresentação da solução sem detalhes de implementação. Dependendo do trabalho, pode ser uma descrição mais teórica, mais "arquitetural", etc.

3.1 Secção Exemplo

Neste capítulo apresentam-se exemplos de formatação de figuras e tabelas, equações e referências cruzadas.

Apresenta-se de seguida um exemplo de equação, completamente fora do contexto:

$$CIF_1: \quad F_0^j(a) = \frac{1}{2\pi\imath}\oint_\gamma \frac{F_0^j(z)}{z-a}dz \tag{3.1}$$

$$CIF_2: \quad F_1^j(a) = \frac{1}{2\pi\imath}\oint_\gamma \frac{F_0^j(x)}{x-a}dx \tag{3.2}$$

Na Equação 3.2 lorem ipsum dolor sit amet, consectetuer adipiscing elit. Suspendisse tincidunt viverra elit. Donec tempus vulputate mauris. Donec arcu. Vestibulum condimentum porta justo. Curabitur ornare tincidunt lacus. Curabitur ac massa vel ante tincidunt placerat. Cras vehicula semper elit. Curabitur gravida, est a elementum suscipit, est eros ullamcorper quam, sed cursus velit velit tempor neque. Duis tempor condimentum ante.

Phasellus imperdiet, orci vel pretium sollicitudin, magna nunc ullamcorper augue, non venenatis dui nunc quis massa. Pellentesque dolor elit, dapibus venenatis, viverra ultricies, accumsan cursus, orci. Aliquam erat volutpat. Mauris ornare tristique leo. Maecenas eros. Curabitur velit nunc, tincidunt vitae, dictum posuere, pulvinar nec, diam. In suscipit mauris a nunc. Pellentesque gravida. Morbi quam lacus, pretium eget, tincidunt vulputate, interdum sed, turpis. Curabitur quis

[1] Na introdução a apresentação do problema foi breve.

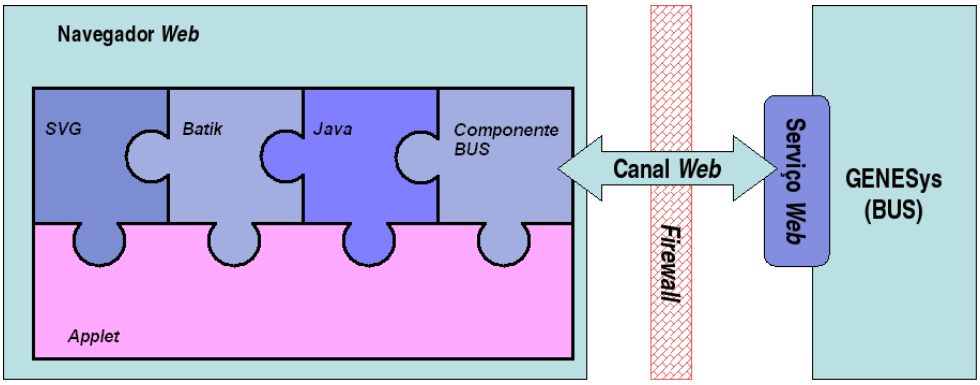

Figure 3.1: Arquitectura da Solução Proposta

est. Sed lectus lorem, congue vel, dignissim laoreet, blandit a, nisi. Aenean nunc ligula, tincidunt eu, hendrerit vel, suscipit non, erat. Aliquam gravida. Integer non pede. In laoreet augue id leo. Mauris placerat.

A arquitetura do visualizador assenta sobre os seguintes conceitos base [**?**]:

- **Componentes** — Suspendisse auctor mattis augue *push*;
- **Praesent** — Sit amet sem maecenas eleifend facilisis leo;
- **Pellentesque** — Habitant morbi tristique senectus et netus.

3.1.1 Exemplo de Figura

É apresentado na Figura 3.1 um exemplo de figura flutuante.

Loren ipsum dolor sit amet, consectetuer adipiscing elit. Praesent sit amet sem. Maecenas eleifend facilisis leo. Vestibulum et mi. Aliquam posuere, ante non tristique consectetuer, dui elit scelerisque augue, eu vehicula nibh nisi ac est. Suspendisse elementum sodales felis. Nullam laoreet fermentum urna.

Duis eget diam. In est justo, tristique in, lacinia vel, feugiat eget, quam. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Fusce feugiat, elit ac placerat fermentum, augue nisl ultricies eros, id fringilla enim sapien eu felis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Sed dolor mi, porttitor quis, condimentum sed luctus.

3.1.2 Exemplo de Tabela

É apresentado na Tabela 3.1 um exemplo de tabela flutuante e na Tabela 3.2 um exemplo de tabela flutuante, um pouco mais complicada.

Integer quis pede. Fusce nibh. Fusce nec erat vel mi condimentum convallis. Sed at tortor non mauris pretium aliquet. In in lacus in dolor molestie dapibus. Suspendisse potenti. Pellentesque

Table 3.1: Uma Tabela Simples

Acrónimo	**Significado**
ADT	*Abstract Data Type*
ANDF	*Architecture-Neutral Distribution Format*
API	*Application Programming Interface*

sagittis porta erat. Mauris sodales sapien id augue. Nam eu dolor. Donec sit amet turpis non orci rhoncus commodo. Etiam condimentum commodo libero.

Mauris pede. Curabitur faucibus dictum nibh. Proin tincidunt diam vitae mauris. Sed hendrerit dolor vel ipsum. Nullam dapibus. Vivamus tellus diam, egestas sit amet, vulputate non, vulputate id, eros. Nunc sit amet nibh eget nibh imperdiet ornare. Cras vehicula mattis ipsum. Sed diam arcu, semper at, gravida vitae, fermentum et, nulla. Aenean massa orci, tristique nec, rutrum id, fringilla eget, erat. Curabitur nulla ipsum, aliquam sed, rutrum vitae, semper quis, ante. Fusce at nunc in dolor condimentum tempor. Duis sit amet massa.

Curabitur convallis nulla quis risus. Nulla mollis porttitor purus. Fusce ultricies odio at ligula pellentesque suscipit. Nulla velit libero, blandit a, aliquet quis, hendrerit id, arcu. Phasellus porttitor porttitor purus. Suspendisse velit tortor, fringilla sit amet, commodo a, ultrices et, mi. Donec eu metus in erat ornare adipiscing. Praesent varius mi ac nunc. Vestibulum leo lacus, elementum in, vestibulum sit amet, hendrerit at, justo. Sed sit amet neque. Donec libero risus, commodo sit amet, dignissim ut, tincidunt a, eros. Ut non lacus quis tortor mattis ullamcorper. Vivamus consequat augue vel erat. Sed tincidunt. Sed leo eros, ornare a, pulvinar non, mattis quis, nibh. Aliquam faucibus mi ac nisi.

Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Duis aliquet, libero sit amet ornare viverra, augue erat interdum dolor, vitae tincidunt lorem erat a lacus. Sed lectus nisi, auctor in, hendrerit a, molestie vel, lectus. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Duis lacinia tempor dui. Vivamus rhoncus, tellus a viverra dignissim, pede dui adipiscing odio, non faucibus metus mi gravida eros. Nullam a tellus ut velit elementum tempus. Aenean rutrum convallis tellus. Vestibulum nulla ante, dapibus ut, lobortis ut, varius sed, nisl. Fusce lobortis. Sed ac lorem. Nulla tincidunt nulla eget leo. Maecenas ac lectus eu neque ultrices pharetra. Curabitur a risus nec arcu placerat tempor. Suspendisse magna nisl, viverra a, adipiscing eget, ornare ultricies, ligula. Maecenas eu ligula vitae eros convallis dignissim.

Loren ipsum dolor sit amet, consectetuer adipiscing elit. Praesent sit amet sem. Maecenas eleifend facilisis leo. Vestibulum et mi. Aliquam posuere, ante non tristique consectetuer, dui elit scelerisque augue, eu vehicula nibh nisi ac est. Suspendisse elementum sodales felis. Nullam laoreet fermentum urna.

Duis eget diam. In est justo, tristique in, lacinia vel, feugiat eget, quam. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Fusce feugiat, elit ac

Table 3.2: Uma Tabela Mais Complicada

Iteração k de $f(x_n)$				
$\boldsymbol{k}$	x_1^k	x_2^k	x_3^k	comentários
0	-0.3	0.6	0.7	-
1	0.47102965	0.04883157	-0.53345964	$\delta < \varepsilon$
2	0.49988691	0.00228830	-0.52246185	$\delta < \varepsilon$
3	0.49999976	0.00005380	-0.523656	N
4	0.5	0.00000307	-0.52359743	
$\vdots$	$\vdots$	$\ddots$	$\vdots$	
7	0.5	0.0	**-0.52359878**	$\delta < 10^{-8}$

placerat fermentum, augue nisl ultricies eros, id fringilla enim sapien eu felis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Sed dolor mi, porttitor quis, condimentum sed luctus.

3.2 Secção Exemplo

Loren ipsum dolor sit amet, consectetuer adipiscing elit. Praesent sit amet sem. Maecenas eleifend facilisis leo. Vestibulum et mi. Aliquam posuere, ante non tristique consectetuer, dui elit scelerisque augue, eu vehicula nibh nisi ac est. Suspendisse elementum sodales felis. Nullam laoreet fermentum urna.

Duis eget diam. In est justo, tristique in, lacinia vel, feugiat eget, quam. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Fusce feugiat, elit ac placerat fermentum, augue nisl ultricies eros, id fringilla enim sapien eu felis. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Sed dolor mi, porttitor quis, condimentum sed luctus.

3.3 Resumo e Conclusões

Resumir e apresentar as conclusões que se podem tirar no fim deste capítulo.

Chapter 4

Implementação

Este capítulo pode ser dedicado à apresentação de detalhes de nível mais baixo relacionados com o enquadramento e implementação das soluções preconizadas no capítulo anterior. Note-se no entanto que detalhes desnecessários à compreensão do trabalho devem ser remetidos para anexos.

Dependendo do volume, a avaliação do trabalho pode ser incluída neste capítulo ou pode constituir um capítulo separado.

4.1 Secção Exemplo

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Integer hendrerit commodo ante. Pellentesque nibh libero, aliquam at, faucibus id, commodo a, velit. Duis eleifend sem eget leo. Morbi in est. Suspendisse magna sem, varius nec, hendrerit non, tincidunt quis, quam. Aenean congue. Vivamus vel est sit amet sem iaculis posuere. Cras mollis, enim vel gravida aliquam, libero nunc ullamcorper dui, ullamcorper sodales lectus nulla sed urna. Morbi aliquet porta risus. Proin vestibulum ligula a purus. Maecenas a nulla. Maecenas mattis est vitae neque auctor tempus. Etiam nulla dui, mattis vitae, porttitor sed, aliquet ut, enim. Cras nisl magna, aliquet et, laoreet at, gravida ac, neque. Sed id est. Nulla dapibus dolor quis ipsum rhoncus cursus.

4.2 Mais uma Secção

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Quisque purus sapien, interdum ut, vestibulum a, accumsan ullamcorper, erat. Mauris a magna ut leo porta imperdiet. Donec dui odio, porta in, pretium non, semper quis, orci. Quisque erat diam, pharetra vel, laoreet ac, hendrerit vel, enim. Donec tristique luctus risus. Fusce dolor est, eleifend id, elementum sit amet, varius vitae, neque. Morbi at augue. Ut sem ligula, auctor vitae, facilisis id, pharetra non, lectus. Nulla lacus augue, aliquam eget, sollicitudin sed, hendrerit eu, leo. Suspendisse ac tortor. Mauris at odio. Etiam vehicula. Nam lacinia purus at nibh. Aliquam fringilla lorem ac justo. Ut nec enim.

```
map(String key, String value):
// key: document name
// value: document contents
for each word w in value:
EmitIntermediate(w, "1");

reduce(String key, Iterator values):
// key: a word
// values: a list of counts
int result = 0;
for each v in values:
result += ParseInt(v);

Emit(AsString(result))
```

Listing 4.1: Example map and reduce functions for word counting

Quisque ullamcorper. Aliquam vel magna. Sed pulvinar dictum ligula. Sed ultrices dolor ut turpis. Vivamus sagittis orci malesuada arcu venenatis auctor. Proin vehicula pharetra urna. Aliquam egestas nunc quis nisl. Donec ullamcorper. Nulla purus. Ut suscipit lacus vitae dui. Mauris semper. Ut eget sem. Integer orci. Nam vitae dui eget nisi placerat convallis.

Sed id lorem. Proin gravida bibendum lacus. Sed molestie, urna quis euismod laoreet, diam dolor dictum diam, vitae consectetuer leo ipsum id ante. Integer eu lectus non mauris pharetra viverra. In feugiat libero ut massa. Morbi cursus, lorem sollicitudin blandit semper, felis magna pellentesque lacus, ut rhoncus leo neque at tellus. Sed mattis, diam eget eleifend tincidunt, ligula eros tincidunt diam, vitae auctor turpis est vel nunc. In eu magna. Donec dolor metus, egestas sit amet, ultrices in, faucibus sed, lectus. Etiam est enim, vehicula pharetra, porta non, viverra vel, nunc. Ut non sem. Etiam nec neque.

4.3 Resumo ou Conclusões

Proin vehicula pharetra urna. Aliquam egestas nunc quis nisl. Donec ullamcorper. Nulla purus. Ut suscipit lacus vitae dui. Mauris semper. Ut eget sem. Integer orci. Nam vitae dui eget nisi placerat convallis.

Chapter 5

Conclusões e Trabalho Futuro

Deve ser apresentado um resumo do trabalho realizado e apreciada a satisfação dos objetivos do trabalho, uma lista de contribuições principais do trabalho e as direções para trabalho futuro.

A escrita deste capítulo deve ser orientada para a total compreensão do trabalho, tendo em atenção que, depois de ler o Resumo e a Introdução, a maioria dos leitores passará à leitura deste capítulo de conclusões e recomendações para trabalho futuro.

5.1 Satisfação dos Objetivos

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam non felis sed odio rutrum ultrices. Donec tempor dolor. Vivamus justo neque, tempus id, ullamcorper in, pharetra non, tellus. Praesent eu orci eu dolor congue gravida. Sed eu est. Donec pulvinar, lectus et eleifend volutpat, diam sapien sollicitudin arcu, a sagittis libero neque et dolor. Nam ligula. Cras tincidunt lectus quis nunc. Cras tincidunt congue turpis. Nulla pede velit, sagittis a, faucibus vitae, porttitor nec, ante. Nulla ut arcu. Cras eu augue at ipsum feugiat hendrerit. Proin sed justo eu sapien eleifend elementum. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Vivamus quam lacus, pharetra vel, aliquam vel, volutpat sed, nisl.

Nullam erat est, vehicula id, tempor non, scelerisque at, tellus. Pellentesque tincidunt, ante vehicula bibendum adipiscing, lorem augue tempor felis, in dictum massa justo sed metus. Suspendisse placerat, mi eget molestie sodales, tortor ante interdum dui, ac sagittis est pede et lacus. Duis sapien. Nam ornare turpis et magna. Etiam adipiscing adipiscing ipsum. Fusce sodales nisl a arcu. Cras massa leo, vehicula facilisis, commodo a, molestie faucibus, metus. Suspendisse potenti. Duis sagittis. Donec porta. Sed urna. Maecenas eros. Vivamus erat ligula, pharetra sit amet, bibendum et, fermentum sed, dolor. Nullam eleifend condimentum nibh. Integer leo nibh, consequat eget, mollis et, sagittis ac, felis. Duis viverra pede in pede. Phasellus molestie placerat leo. Praesent at tellus a augue congue molestie. Proin sed justo eu sapien eleifend elementum. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas.

5.2 Trabalho Futuro

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aliquam tempor tristique risus. Suspendisse potenti. Fusce id eros. In eu enim. Praesent commodo leo. Nullam augue. Pellentesque tellus. Integer pulvinar purus a dui convallis consectetuer. In adipiscing, orci vitae lacinia semper, sapien elit posuere sem, ac euismod ipsum elit tempus urna. Aliquam erat volutpat. Nullam suscipit augue sed felis. Phasellus faucibus accumsan est.

Aliquam felis justo, facilisis sit amet, bibendum ut, tempus ac, dolor. Sed malesuada. Nunc non massa. In erat. Nulla facilisi. Phasellus blandit, est in accumsan cursus, libero augue elementum leo, vitae auctor mauris nisl ac tortor. Cras porttitor ornare elit. Fusce at lorem. Sed lectus tortor, vestibulum id, varius a, condimentum nec, lectus. Maecenas in nisi et magna pretium aliquam. Pellentesque justo elit, feugiat nec, tincidunt a, dignissim vel, ipsum. Sed nunc. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae; Aliquam tempus rhoncus leo. Donec neque quam, cursus sit amet, ultricies varius, semper non, pede. Donec porttitor. Sed aliquet feugiat elit.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Phasellus tellus pede, auctor ut, tincidunt a, consectetuer in, felis. Mauris quis dolor et neque accumsan pellentesque. Donec dui magna, scelerisque mattis, sagittis nec, porta quis, nulla. Vivamus quis nisl. Etiam vitae nisl in diam vehicula viverra. Sed sollicitudin scelerisque est. Nunc dapibus. Sed urna. Nulla gravida. Praesent faucibus, risus ac lobortis dignissim, est tortor laoreet mauris, dictum pellentesque nunc orci tincidunt tellus. Nullam pulvinar, leo sed vestibulum euismod, ante ligula elementum pede, sit amet dapibus lacus tortor ac nisl. Morbi libero. Integer sed dolor ac lectus commodo iaculis. Donec ut odio.

References

[CL02] Larry L Constantine and Lucy AD Lockwood. Usage-centered engineering for web applications. *Software, IEEE*, 19(2):42–50, 2002.

[G+05] Jesse James Garrett et al. Ajax: A new approach to web applications, 2005.

[MPM13] Rodrigo MLM Moreira, Ana CR Paiva, and Atif Memon. A pattern-based approach for gui modeling and testing. In *Software Reliability Engineering (ISSRE), 2013 IEEE 24th International Symposium on*, pages 288–297. IEEE, 2013.

[NPCF13] Miguel Nabuco, Ana CR Paiva, Rui Camacho, and Joao Pascoal Faria. Inferring ui patterns with inductive logic programming. In *Information Systems and Technologies (CISTI), 2013 8th Iberian Conference on*, pages 1–5. IEEE, 2013.

[NPF14] Miguel Nabuco, Ana CR Paiva, and Joao Pascoala Faria. Inferring user interface patterns from execution traces of web applications. Manuscript submitted for publication, 2014.

[VWVDVE01] Martijn Van Welie, Gerrit C Van Der Veer, and Anton Eliëns. Patterns as tools for user interface design. In *Tools for Working with Guidelines*, pages 313–324. Springer, 2001.

REFERENCES

Appendix A

Loren Ipsum

Depois das conclusões e antes das referências bibliográficas, apresenta-se neste anexo numerado o texto usado para preencher a dissertação.

A.1 O que é o *Loren Ipsum*?

Lorem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum [**?**].

A.2 De onde Vem o Loren?

Contrary to popular belief, Lorem Ipsum is not simply random text. It has roots in a piece of classical Latin literature from 45 BC, making it over 2000 years old. Richard McClintock, a Latin professor at Hampden-Sydney College in Virginia, looked up one of the more obscure Latin words, consectetur, from a Lorem Ipsum passage, and going through the cites of the word in classical literature, discovered the undoubtable source. Lorem Ipsum comes from sections 1.10.32 and 1.10.33 of "de Finibus Bonorum et Malorum" (The Extremes of Good and Evil) by Cicero, written in 45 BC. This book is a treatise on the theory of ethics, very popular during the Renaissance. The first line of Lorem Ipsum, "Lorem ipsum dolor sit amet...", comes from a line in section 1.10.32.

The standard chunk of Lorem Ipsum used since the 1500s is reproduced below for those interested. Sections 1.10.32 and 1.10.33 from "de Finibus Bonorum et Malorum" by Cicero are also reproduced in their exact original form, accompanied by English versions from the 1914 translation by H. Rackham.

A.3 Porque se usa o Loren?

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem Ipsum is that it has a more-or-less normal distribution of letters, as opposed to using "Content here, content here", making it look like readable English. Many desktop publishing packages and web page editors now use Lorem Ipsum as their default model text, and a search for "lorem ipsum" will uncover many web sites still in their infancy. Various versions have evolved over the years, sometimes by accident, sometimes on purpose (injected humour and the like).

A.4 Onde se Podem Encontrar Exemplos?

There are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some form, by injected humour, or randomised words which don't look even slightly believable. If you are going to use a passage of Lorem Ipsum, you need to be sure there isn't anything embarrassing hidden in the middle of text. All the Lorem Ipsum generators on the Internet tend to repeat predefined chunks as necessary, making this the first true generator on the Internet. It uses a dictionary of over 200 Latin words, combined with a handful of model sentence structures, to generate Lorem Ipsum which looks reasonable. The generated Lorem Ipsum is therefore always free from repetition, injected humour, or non-characteristic words etc.