Trabalho Prático nº2

Configuration Management Plan

Version <1.0>

Table of Contents

1. Introduction 3

1.1 Purpose 3

1.2 Scope 3

1.3 Definitions, Acronyms, and Abbreviations 3

1.4 References 3

1.5 Overview 3

2. Software Configuration Management 3

2.1 Organization, Responsibilities, and Interfaces 3

2.2 Tools, Environment, and Infrastructure 3

3. The Configuration Management Program 3

3.1 Configuration Identification 3

3.1.1 Identification Methods 3

3.1.2 Project Baselines 3

3.2 Configuration and Change Control 3

3.2.1 Change Request Processing and Approval 3

3.2.2 Change Control Board (CCB) 3

3.3 Configuration Status Accounting 3

3.3.1 Project Media Storage and Release Process 3

3.3.2 Reports and Audits 3

4. Milestones 3

5. Training and Resources 3

6. Subcontractor and Vendor Software Control 3

# Introduction

[The introduction of the **Configuration Management Plan** provides an overview of the entire document. It includes the purpose, scope, definitions, acronyms, abbreviations, references, and overview of this **Configuration Management Plan**.]

## Purpose

[Specify the purpose of this **Configuration Management Plan.**]

O projeto consiste na elaboração da parte prática no âmbito da disciplina de Engenharia de Software II, trabalho elaborado em época contínua de avaliação.

## Scope

[A brief description of the scope of this **Configuration Management Plan**; what model it is associated with and anything else that is affected or influenced by this document.]

o projeto consiste na elaboração de um motor de pesquisa criado pelo nosso grupo que através da introdução de uma frase, irá processar a informação e será capaz de filtrar as informações de modo a encontrar ficheiros com grande grau de apoximação com o pesquisado.

## Definitions, Acronyms, and Abbreviations

[This subsection provides the definitions of all terms, acronyms, and abbreviations required to properly interpret the **Configuration Management Plan**. This information may be provided by reference to the project’s Glossary.]

CCB Configuration Control Board

CI Configuration Item

CM Configuration Management

CSA Configuration Status Accounting

PDL Project Documentation Library (BitBucket Repository)

PSL Project Software Library (BitBucket Repository)

Commit Apply staged changes to a branch

Merge Combine 2 branches

Branch A separate path of development

Master The development branch from which release branches emerge

Repo The code and document repository (aka PSL/PDL)

Repo Owner The CCB Librarian

BitBucket Remote repository host

Push Move changes from a local repository to the remote repository

Pull Move changes from the remote to a local repository

Testers List An email list of approved pre-official-release users.

Merge Request A Merge Request is a re-submittal of a change request by a developer once the

changes that are indicated on the change request are done on a branch.

CM Leader The head of the configuration management team.

Configuration Control Board The CCB are those responsible for enforcing the following of change control procedures.

CCB Leader The head of the CCB chosen by the CM Leader.

CCB Librarian A member of the CCB who controls access to the PSL and PDL.

Change Request Basically, a BitBucket Issue.

Change Request Form A form used by developers to request changes to the PSL or PDL. This form is accessed via the BitBucket Downloads Page. The form is copied into BitBucket tickets/issues to be filled out and submitted.

Version Number Unique identifier of a release.

Configuration Identification The process of tracking project components.

Configuration Item Any item associated with the project, i.e. files.

Developer All project members who develop or design code.

Project Documentation Library The official location of all project documentation.

Project Software Library The official location of all project code.

Project Teams The project teams consist of a Project Management Team, Configuration management Team, and Quality Assurance Team.

## References

[This subsection provides a complete list of all documents referenced elsewhere in the **Configuration Management Plan**. Identify each document by title, report number (if applicable), date, and publishing organization. Specify the sources from which the references can be obtained. This information may be provided by reference to an appendix or to another document.]

1. [1] Los Alamos National Lab Software Configuration Management Plan for the Sequence Annotation Toolkit, Robert Leach, February 2, 2001
2. [2] Los Alamos National Lab Software Configuration Management Plan for XBase Internet Application, Thomas Brettin, May 23, 2000
3. [3] IEEE Std 828-1990 IEEE Standard for Software Configuration Management Plans
4. <https://www.infoq.com/br/articles/spotify-escalando-agile?fbclid=IwAR0P-hZANvRDKeic-C3GKGOXP2j3dfLgPRWb8_z0vBMCHueSv9BzXboRFDw>
5. <https://blog.crisp.se/wp-content/uploads/2012/11/SpotifyScaling.pdf?fbclid=IwAR0hL2Rd92BZpzqwWMGwk01J0BJgwVNOgcwqVf0tD-33HuJyfriENQWycuw>
6. <http://www.scrumcasestudies.com/adobe/?fbclid=IwAR1Zwe9zWlhmFjc2N624v3Rvq4Zia5Iuorz3c_jtBsMJQsTSBjp4C1SkNyU>
7. <http://blogs.adobe.com/agile/files/2012/08/Adobe-Premiere-Pro-Scrum-Adoption-How-an-agile-approach-enabled-success-in-a-hyper-competitive-landscape-.pdf?red=a&fbclid=IwAR2WlzLNpq6HEIT_d_6ENsXQajiZqCSjkD499gTDuQBOi03QJZ25P0k2nPg>
8. <http://www.scrumcasestudies.com/intel/?fbclid=IwAR0IXl48kEonTa25423vps-L0R9Ush4oWlm8vcZuMSUuHcGyNB7-nlc9VxI>
9. IEEE Std 828-1990 IEEE Standard for Software Configuration Management Plans

## Overview

[This subsection describes what the rest of the **Configuration Management Plan** contains and explains how the document is organized.]

Este documento irá conter toda a organização feita pelo nosso plantel de forma a conseguir elaborar o projeto de forma concreta e dentro dos prazos estabelecidos, ira conter também todas as ferramentas que utilizaremos para a resolução do projeto e em que situações irão ser utilizadas as mesmas. Irão também aqui ser descritas todos os métodos a ser implementados, bem como as baselines do projeto e todas as configurações do mesmo.

# Software Configuration Management

## Organization, Responsibilities, and Interfaces

[Describe who is going to be responsible for performing the various Configuration Management (CM) activities described in the CM Process Discipline.]

O nosso grupo de trabalho vai tratar da configuração das ferramentas em conjunto. Uma vez que esta configuração envolve uma máquina virtual que mantem as ferramentas utilizadas uniformizadas faz sentido que todos os membros estejam envolvidos da sua configuração de modo a todos saberem como trabalhar e gerir as ferramentas.

Como tal, todos os membros do grupo têm a responsabilidade de saber não só configurar como tambem trabalhar com as ferramentas utilizadas.

## Tools, Environment, and Infrastructure

[Describe the computing environment and software tools to be used in fulfilling the CM functions throughout the project or product lifecycle.

Describe the tools and procedures required used to version control configuration items generated throughout the project or product lifecycle.

Issues involved in setting up the CM environment include:

* anticipated size of product data
* distribution of the product team
* physical location of servers and client machines]

A infraestrutura utilizada neste projeto consiste numa maquina virtual que vai servir para centralizar as ferramentas necessárias para a realização do mesmo.

Como ferramentas vamos ter:

-O YouTrack, ferramenta issue tracker para desenvolvimento usando metodologias ágeis;

-O Jenkins que vai permitir implementar conceitos de Continuous Integration como builds e testes automáticos;

-O IntelliJ como ambiente de desenvolvimento;

-A linguagem Java que vai ser utilizada para o desenvolvimento do motor de pesquisa e especificação de testes;

-A framework jUnit para codificação dos testes especificados;

-A plataforma GitHub que serve para hospedagem do codigo-fonte.

A equipa vai trabalhar em conjunto na configuração de modo a ser mais facil obter um uso uniforme das ferramentas implementadas.

# The Configuration Management Program

## Configuration Identification

### Identification Methods

[Describe how project or product artifacts are to be named, marked, and numbered. The identification scheme needs to cover hardware, system software, Commercial-Off-The-Shelf (COTS) products, and all application development artifacts listed in the product directory structure; for example, plans, models, components, test software, results and data, executables, and so on.]

|  |  |  |
| --- | --- | --- |
| **Artifact** | **To be named** | **Numbered** |
| Requisitos iniciais do projeto | Requisitos iniciais do projeto |  |
| Cenários do projeto | Cenários do projeto |  |
| Software Configuration Plan | SCM Plan - Grupo 24 |  |
| Tabelas BVA | Tabelas BVA |  |
| Tabelas equivalência e casos de teste | Tabelas equivalência e casos de teste |  |
| Diagrama Use Cases | Use Cases Diagram |  |
| Projeto de desenvolvimento do software | EXP |  |
| Resultados testes caixa preta | Resultados testes caixa preta |  |
| Resultados testes caixa branca | Resultados testes caixa branca |  |

### Project Baselines

[Baselines provide an official standard on which subsequent work is based and to which only authorized changes are made.

Describe at what points during the project or product lifecycle the baselines are to be established. The most common baselines would be at the end of each of the Inception, Elaboration, Construction, and Transition phases. Baselines could also be generated at the end of iterations within the various phases or even more frequently.

Describe who authorizes a baseline and what goes into it.]

As baselines devem ser definidas pelos integrantes do projeto e pré-aprovadas pelos mesmo e, de seguida, aprovadas pelo sponsor, neste caso os professores da Unidade Curricular.

Inicialmente deverá ser definida a baseline de âmbito do projeto com a definição dos requisitos do mesmo(artefactos a desenvolver, tecnologias a adotar) e, de seguida, baseline de tempo(backlog, duração de sprints, epics, user stories, ect).

A seguir, definição de baseline em termos de estabelecimento de mecanismos de suporte às principais tarefas de gestão de configuração de software, nomeadamente: i) controlo de versões, controlo de alterações, configuração de mecanismos de building, e auditorias;

Depois, deverá ser realizada a baseline de documentação do processo de análise, especificação de testes e de mecanismos de code review.

## Configuration and Change Control

### Change Request Processing and Approval

[Describe the process by which problems and changes are submitted, reviewed, and dispositioned.]

O processo de Changue Request relativo às baselines proporciona-se da seguinte forma:

Numa primeira fase deverá ser evitada qualquer mudança nas baselines definidas pela equipa e aprovas pelo sponsor.

De seguida, caso não se consiga evitar a mudança, deverá ser feita uma explicação detalhada do porquê de ser necessária a mudança(risco), qual a probabilidade do mesmo bem como o seu impacto caso este não seja minimizado ou eliminado através de uma risk response.

Depois, deverão ser descritas as alterações propostas e quais as consequências positivas e negativas das mesmas.

Todas estas fases deverão ser seguidas e, de seguida, enviadas para aprovação do sponsor, que caso aprove, poderão ser postas em práticas as mudanças necessárias nas baselines propostas para alteração.

### Change Control Board (CCB)

[Describe the CCB membership and the procedures for processing change requests and approvals to be followed by the CCB.]

Os integrantes do projeto têm acesso aos Change Request submetidos e o seu estado de aprovação ou não por parte do Sponsor.

## Configuration Status Accounting

### Project Media Storage and Release Process

[Describe retention policies, and the back-up, disaster, and recovery plans. Also describe how the media is to be retained—online, offline, media type, and format.

The release process describes what is in the release, who it is for, and whether there are any known problems and any installation instructions.]

Em termos de back-ups, em caso de erros, avarias de hardware, entre outros, todos os artefactos estão guardados e por versões dos mesmo no repositório online Github criado para o projeto.

Estes artefactos estão guardados offline nos computadores dos integrantes do projeto e quando estes se encontram online e efetuarem alterações nos artefactos, efetuam os respetivos commits de forma a colocar essas alterações online, prevenindo o risco de avaria do hardware.

Os releases são controlados mediante alterações feitas em qualquer um dos artefactos e fazendo os respetivos commits anteriormente descritos, que deverão ser feitos de forma periódica de forma a prevenir os riscos de erros, bugs, entre outros problemas de conexão, hardware, ect.

Estes mesmos releases são feitos para futuro análise no final de cada sprint pelo sponsor de forma a verificar se o projeto está a ser bem desenvolvido.

Problemas de instalação ou instruções de instalação não são necessárias uma vez que o ambiente de desenvolvimento foi fornecido pelo sponsor.

### Reports and Audits

[Describe the content, format, and purpose of the requested reports and configuration audits.

Reports are used to assess the “quality of the product” at any given time in the project or product lifecycle. Reporting on defects based on change requests may provide some useful quality indicators and, thereby, alert management and developers to particularly critical areas of development. Defects are often classified by criticality (high, medium, and low) and could be reported on the following basis:

* Aging (Time-based Reports): How long have defects of the various kinds been open? What is the “lag time’’ between when defects are found in the lifecycle and when they are fixed?
* Distribution (Count Based Reports): How many defects are there in the various categories by owner, priority or state of fix?
* Trend (Time-related and Count-related Reports): What is the cumulative number of defects found and fixed over time? What is the rate of defect discovery and fix? What is the “quality gap” in terms of open as opposed to closed defects? What is the average defect resolution time?]

Os reports deverão ser feitos na ferramenta YouTrack e serão auditados pelo sponsor no final de cada sprint.

Más formulações deverão ser classificadas como high, medium ou low na mesma ferramenta de forma a poder ser controlada a qualidade do produto.

Deverá ser também efetuada uma análise no final das correções, caso estas sejam reportadas, de forma a perceber se o tempo de report e de resposta em relação a estes está a ser adequado ou não, bem como análise quantitativa de más formulações encontradas e categorizá-las pela prioridade dada pelo product owner às user stories ou tarefas.

# Milestones

[Identify the internal and customer milestones related to the project or product CM effort. This section includes details on when the **Configuration Management Plan** itself is to be updated.]

* Os milestones definidos com o sponsor são:
* Final do Sprint #1: em que deverão ser entregues a definição de requisitos, tabelas de BVA e equivalência e diagrama de casos de uso;
* Final do Sprint #2: em que deverá ser entregue o SCM Plan, codificação do software e respetivos testes em relação ao mesmo.
* Milestones internos:
  + Configuração inicial do projeto (repositório, ferramentas de issuetracking, automação de testes, ect);
  + Definição de requisitos;
  + Tabelas BVA e de equivalência e diagrama de casos de uso:
  + SCM Plan;
  + Codificação do software;
  + Testes de software;
  + Relatório do projeto;
  + Vídeo de apresentação do projeto.

# Training and Resources

[Describe the software tools, personnel, and training required to implement the specified CM activities.]

Como ferramentas vamos ter:

-O YouTrack, ferramenta issue tracker para desenvolvimento usando metodologias ágeis;

-O Jenkins que vai permitir implementar conceitos de Continuous Integration como builds e testes automáticos;

-O IntelliJ como ambiente de desenvolvimento;

-A linguagem Java que vai ser utilizada para o desenvolvimento do motor de pesquisa e especificação de testes;

-A framework jUnit para codificação dos testes especificados;

-A plataforma GitHub que serve para hospedagem do codigo-fonte.

Como pessoal de desenvolvimento vamos ter os 3 elementos que constituem o nosso grupo do projeto.

Este projeto não necessita nenhum treino para alem do conhecimento adequirido durante as aulas da disciplina.

# Subcontractor and Vendor Software Control

[Describe how software developed outside of the project environment will be incorporated.]

Vai ser utilizada uma maquina virtual desenvolvida pelos professores responsaveis pela cadeira e vão servir para incorporar todos os outros programas de forma centralizada.

O nosso grupo não tem qualquer influência sobre a funcionalidade desta maquina e ela apenas vai servir como plataforma de ligação para o projeto.