

Escola Superior de Tecnologia e Gestão

Engenharia de Software II

LEI_ESII2022_TP2_GRUPO19 Configuration Management Plan

Version <2.0>

LEI_ESII2022_TP2_GRUPO19	Version: 2.0
Configuration Management Plan	Date: 03/01/2022
SCMP#1	

Revision History

Date	Version	Description	Author
24/12/2021	1.0	draft	João,Pedro,Francisco
03/01/2022	2.0	Final version	João,Pedro,Francisco

LEI_ESII2022_TP2_GRUPO19	Version: 2.0
Configuration Management Plan	Date: 03/01/2022
SCMP#1	

Table of Contents

Conteúdo

1. Introduction	4
1.1 Purpose.....	4
1.2 Scope.....	4
1.3 Definitions, Acronyms, and Abbreviations	4
1.4 References	4
2. Software Configuration Management.....	5
2.1 Organization, Responsibilities, and Interfaces	5
2.1.1 Papeis e responsabilidades SCM	5
2.2 Tools, Environment, and Infrastructure	6
3. The Configuration Management Program	6
3.1 Configuration Identification.....	6
3.1.1 Identification Methods.....	6
3.1.2 Project Baselines.....	7
3.2 Configuration and Change Control	7
3.2.1 Change Request Processing and Approval	7
3.2.2 Change Control Board (CCB).....	7

LEI_ESII2022_TP2_GRUPO19	Version: 2.0
Configuration Management Plan	Date: 03/01/2022
SCMP#1	

Configuration Management Plan

1. Introduction

1.1 Purpose

Este documento tem como objetivo descrever as ferramentas e recursos a utilizar, o processo e métodos de todas as atividades a desenvolver pela organização, bem como os papéis e os seus respetivos responsáveis.

1.2 Scope

O plano SCM será aplicado ao enunciado proposto pelos docentes, na elaboração de um software de gestão de encomendas, tirando partido de um ficheiro JSON e de uma API do trabalho prático 1. Aqui está registado todas as atividades individuais e/ou coletivas do grupo na elaboração do projeto e toda a documentação produzida pela equipa de CM.

1.3 Definitions, Acronyms, and Abbreviations

CM – Configuration Management.

CCB – Configuration Control Board

DEV – Developers

SCM – Software Configuration Management

IDE - Integrated Development Environment

ES2 - Engenharia de Software 2

API - Interface de Programação de Aplicações

JSON - JavaScript Object Notation

1.4 References

IEEE. (s.d.). *Standard for Software Configuration Management Plans*.

Repositorio Gitlab - https://gitlab.estg.ipp.pt/8140232/lei_esii2022_tp2_grupo19

LEI_ESII2022_TP2_GRUPO19	Version: 2.0
Configuration Management Plan	Date: 03/01/2022
SCMP#1	

2. Software Configuration Management

2.1 Organization, Responsibilities, and Interfaces

2.1.1 Papeis e responsabilidades SCM

Scrum Master: Pedro

O scrum master é responsável por assegurar que o projecto é executado de acordo com as práticas, valores, regras do Scrum e metodologia Agile, bem como se progride como planeado.

Product Owner: João

O product owner é responsável pela gestão, controlo do projeto e gerir os backlogs.

QA leader: Francisco

O QA leader é responsável por assegurar a qualidade do software.

Code Reviewer: João, Francisco, Pedro

O Code reviewer é responsável pela revisão do Código já desenvolvido.

2.1.1.1 Equipa CM

A equipa CM é responsável pela manutenção de toda a documentação oficial, Código e software subjacente. Os membros da equipa CM são: João, Francisco, Pedro.

2.1.1.1.1 CM Leader

Líder CM é João.

O Líder CM Compõe o plano CM e é responsável por procedimentos de alterações e convenções de Código. É da responsabilidade do líder CM informar os membros da equipa CM das suas responsabilidades. É também responsável por realizar reuniões para manter a equipa dentro das tarefas e dentro dos prazos. Pedidos de alteração principais necessitam de revisões de código realizadas pelo líder CM.

2.1.1.2 Configuration Control Board

CCB é responsável por aprovar alterações ao software e documentação descritas anteriormente durante o projeto. É da responsabilidade da CCB direcionar os desenvolvedores sobre como estão envolvidos na aprovação da alteração.

2.1.1.2.1 Lider CCB

CCB Leader: Pedro

O CCB leader é responsável por aprovar ou reprovare change requests ou realizar meetings para rever o mesmo.

Os membros CCB são João, Francisco, Pedro.

LEI_ESII2022_TP2_GRUPO19	Version: 2.0
Configuration Management Plan	Date: 03/01/2022
SCMP#1	

2.2 Tools, Environment, and Infrastructure

Para o desenvolvimento do projeto, é necessário que a equipa de desenvolvimento, possua experiência e capacidades nas seguintes ferramentas:

IntelliJ como IDE principal de desenvolvimento do software;

Java como linguagem para desenvolver o software;

Gitlab para gerir e controlar as versões e alterações do projeto;

Junit para a realização dos testes unitários

Gradle para a gestão do projeto;

Pmd para análise de código;

Jacoco para verificar o code coverage.

3. The Configuration Management Program

3.1 Configuration Identification

3.1.1 Identification Methods

User Stories: As users storys devem ter um título, descrição detalhada para a fácil compreensão e um peso baseado na importância e prioridade do mesmo de acordo com a sequência de Fibonacci (1 – 13). Para user storys de funcionalidades do software, a descrição deve possuir o seguinte esquema:

As a *<type of user>*

I want to *<perform some task>*

so that I can *<achieve some goal>*

Acceptance criteria

Given *<some context>*

When *<some action is carried out>*

Then *<a set of observable outcomes should occur>*

Change Requests: Para a criação dos change requests, deve ser criada uma nova issue e o título da mesma deve começar por “Change Request”

LEI_ESII2022_TP2_GRUPO19	Version: 2.0
Configuration Management Plan	Date: 03/01/2022
SCMP#1	

3.1.2 *Project Baselines*

Para a elaboração do projeto foram estabelecidas 4 sprints.

Sprint 1 - Nesta sprint será realizado o planeamento e configuração base do projeto.

Sprint 2 - Nesta Sprint será realizado a correção da API fornecida e será adicionado as configurações para auxilio do desenvolvimento do Código (jacoco e pmd).

Sprint 3 - Nesta sprint será desenvolvido o software.

Sprint 4 – Nesta Sprint será concluído a fase de desenvolvimento e será realizado a revisão e balanço de todo o projeto.

3.2 **Configuration and Change Control**

3.2.1 *Change Request Processing and Approval*

O processo para uma change request é o seguinte: 1. Registrar uma Change Request; 2. CCB avalia e decide se a change request é aprovada; 3. . No caso de ser aprovada a change request passa para a fase de desenvolvimento; 4. É criada uma branch e é adicionado essa change request para este mesmo branch;5. A change request passa para review; 6. A change request é aprovada ou não; 7. No caso de ser aprovada é feito o merge request; 8. É feito o merge para o master; 9. Change request é fechado.

3.2.2 *Change Control Board (CCB)*

A equipa de CCB é composta por Pedro, João, Francisco. Esta equipa vai avaliar e por sua vez aprovar/reprovar os change requests dependendo se cumprem todas as normas estabelecidas, e se é relevante e adequado.