POLITECNICO DI MILANO

Computer Science and Engineering

Requirements Analysis and Specifications Document

CodeKataBattle

Software Engineering 2 Project

Academic year 2023 - 2024

20 November 2023

Version 1.0

Authors:
Massara Gianluca
Nguyen Ba Chiara Thien Thao
Nicotri Flavia

Professor:
Matteo Giovanni Rossi

Contents

Chapter 1

Introduction

1.1 Purpose

In the last years, more and more students are becoming interested in programming and many educators have realised the need of new innovative methods to improve coding skills.
CodeKataBattle aims to assist students in enhancing their programming skills by challenging them with creative tasks in a competitive and stimulating environment.

The following document want to describe the system focusing on the requirements and specification, providing scenarios and use case to specify what the system must do and how it should interact with the stakeholders.

1.1.1 Goals

In the following table we describe the main goals that our system want to achive.

Goal	Description
G.1	Allow students to compete in a tournament
G.2	Allows educators to create challenges for students.
G.3	Allows educators to grade students' projects.
G.4	Allows students to collect badges.

1.2 Scope

The main actors of the system are students and educator. Educator can:

- **Create a tournament:** decide which colleague can create battles within the tournament and defines badges that represent the achievements of individual students;
- **Create a battle:** set configurations and rules for that battle;
- **Evaluate:** manually assign a personal score to the students' works.

Students can:

- **Join a tournament**;
- **Partecipate on a battle:** create a team and complete the project with his code;
- **Collect badges:** based on the rules of the tournament and his performance.

1.2.1 World Phenomena

W.P.	Description
WP.1	Educator wants to create a tournament.
WP.2	Educator wants to create a new battle.
WP.3	A student want to join a tournament.
WP.4	A student wants to join a battle.
WP.5	An educator evaluates the works done by students.

1.2.2 Shared Phenomena

S.P.	Description	Controlled by
SP.1	The system notifies the student about upcoming battles.	Machine
SP.2	The student commits his code.	World
SP.3	The educator configures the tournament rules.	World
SP.4	The student forms a team.	World
SP.5	The educator grants other colleagues the permission to create battles within a tournament.	World
SP.6	The educator configures the battle.	World
SP.7	The student joins a team.	World
SP.8	The system creates a GitHub repository.	Machine
SP.9	The student forks and sets up the GitHub repository.	World
SP.10	GitHub Actions notifies the system about students' commits.	World
SP.11	The system shows the battle score of the team.	Machine
SP.12	The system notifies when the final battle rank becomes available.	Machine
SP.13	The system shows the tournament rank.	Machine
SP.14	The system shows the list of ongoing tournaments.	Machine
SP.15	The system shows the student's badges.	Machine
SP.16	The educator defines the badges of a tournament.	World
SP.17	All users can visualize the profile of a user.	World

1.3 Definitions, Acronyms, Abbreviations

Abbreviations	Meaning
WP	World Phenomena
SP	Shared Phenomena
G	Goal
R	Requirement
NFR	Non Functional Requirement
D	Domain Assumption
w.r.t.	with reference to
e.g.	exempli gratia
i.e.	id est
etc.	etcetera

1.4 Revision History

1.5 Reference Documents

- Course slides on WeBeep.
- RASD assignment document.

1.6 Document Structure

The structure of this RASD document is the following:

1. **Introduction**: In this section is presented the purpose of this document highlighting in particular the main goals, the audience which is referred to, the identification of the product and application domain describing world and shared phenomena and, lastly, the terms definitions.
2. **Overall Description**: This chapter describes the possible scenarios of the platform, the shared phenomena presented at the beginning of the document and assumptions on the domain of the application.
3. **Specific Requirements**: Includes all the requirements in a more specific way than the "Overall Description" section. Moreover, it is useful to show functional requirements in terms of use cases diagrams, sequence/activity diagrams.
4. **Formal Analysis Using Alloy**: Includes Alloy models which are used for the description of the application domain and his properties, referring to the operations which the system has to provide.
5. **Effort Spent**: This section shows the effort spent in terms of time for each team member and the whole team.
6. **References**: Includes all documents that were helpful in drafting the RASD.

Chapter 2

Overall Description

2.1 Product perspective

2.2 Product functions

2.3 User characteristics

2.4 Assumptions, dependecies and constraints

Chapter 3

Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

3.1.2 Hardware Interfaces

3.1.3 Software Interfaces

3.1.4 Communication Interfaces

3.2 Functional Requirements

3.3 Performance Requirements

3.4 Design Constraints

3.4.1 Standards compliance

3.4.2 Hardware limitations

3.4.3 Any other constraint

3.5 Software System Atttributes

3.5.1 Reliability

3.5.2 Availability

3.5.3 Security

3.5.4 Maintainability

3.5.5 Portability

Chapter 4

Formal Analysis

Chapter 5

Effort Spent

Chapter 6

References