

**Code Kata Battle - Eusebio Alberto,
Martini Marcello**



POLITECNICO
MILANO 1863

Requirement Analysis and Specification Document

Deliverable:	RASD
Title:	Requirement Analysis and Verification Document
Authors:	Eusebio Alberto, Martini Marcello
Version:	1.0
Date:	22-December-2023
Download page:	https://github.com/martinimarcello00/EusebioMartini
Copyright:	Copyright © 2023, Eusebio Alberto, Martini Marcello – All rights reserved

Contents

Table of Contents	3
List of Figures	5
List of Tables	5
1 Introduction	6
1.1 Purpose	6
1.1.1 Goals	6
1.2 Scope	6
1.2.1 World Phenomena	6
1.2.2 Shared Phenomena	6
1.3 Definitions, Acronyms, Abbreviations	6
1.4 Revision history	6
1.5 Reference Documents	6
1.6 Document Structure	6
2 Overall Description	7
2.1 Product perspective	7
2.1.1 Scenarios	7
2.1.2 Domain models	7
2.2 Product functions	7
2.2.1 Requirements	7
2.3 User characteristics	7
2.4 Assumptions, Dependencies and Constraints	7
2.4.1 Assumptions	7
2.4.2 Dependencies	7
2.4.3 Constraints	7
3 Specific Requirements	8
3.1 External interface requirements	8
3.1.1 User interfaces	8
3.1.2 Hardware interfaces	8
3.1.3 Software interfaces	8
3.1.4 Communication interfaces	8
3.2 Functional requirements	8
3.3 Performance requirements	8
3.4 Design constraints	8
3.4.1 Standards compliance	8
3.4.2 Hardware limitations	8
3.4.3 others	8
3.5 Software systems attributes	8
3.5.1 Reliability	8
3.5.2 Availability	8
3.5.3 Security	8
3.5.4 Maintainability	8
3.5.5 Portability	8
4 Formal Analysis Using Alloy	9

5 Effort Spent	10
-----------------------	-----------

List of Figures

List of Tables

1 Introduction

1.1 Purpose

The purpose of this Requirements Analysis and Specification Document is to present the CodeKataBattle platform. CodeKataBattle (CKB) is a platform where students can improve their software development skills, by solving programming exercises, in this context called as katas. Each kata is proposed by a single instructor through the platform, and can be addressed by several groups of students in a programming language of choice. katas are composed of the following elements:

- A brief textual description of the problem
- A set of test cases that will be the base to evaluate the proposed solution

The platform allows Educators to define deadlines on subscription and submission as long as other constraints such as the maximum and minimum number of students per group and other mechanisms for scoring.

The problems are addressed in battles, organized in tournaments, where the groups of students can participate.

The participation in a tournament can improve a student's rank on the platform and grant them badges, based on the performance measured.

The thresholds for badges, their release and the creation of other variables are features available only to educators, but their visualization is open to every other user.

The platform relies on GitHub actions for the creation of repositories containing code katas and the automatic testing of the proposed solutions. The students are required to fork and setup an automatic workflow through GitHubActions

1.1.1 Goals

Here we identify the goals of the system:

- [G1] Students are able to practice on katas through participating to tournaments in groups
- [G2] Students obtain an automatic evaluation of their proposed solutions
- [G3] Instructors are able to create, manage and award tournaments
- [G4] Instructors are able to provide a manual evaluation of the proposed solutions

1.2 Scope

1.2.1 World Phenomena

1.2.2 Shared Phenomena

1.3 Definitions, Acronyms, Abbreviations

1.4 Revision history

1.5 Reference Documents

1.6 Document Structure

2 Overall Description

Here you can see how to include an image in your document.

2.1 Product perspective

2.1.1 Scenarios

2.1.2 Domain models

2.2 Product functions

2.2.1 Requirements

2.3 User caratheristics

2.4 Assumptions, Dependencies and Constraints

2.4.1 Assumptions

2.4.2 Dependencies

2.4.3 Constraints

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

TODO: use case diagrams, use cases and associated sequence/activity diagrams, and mapping on requirements

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 others

3.5 Software systems attributes

3.5.1 Reliability

3.5.2 Availability

3.5.3 Security

3.5.4 Maintainability

3.5.5 Portability

4 Formal Analysis Using Alloy

Organize this section according to the rules defined in the project description.

5 Effort Spent

Provide here information about how much effort each group member spent in working at this document. We would appreciate details here.