POLITECNICO
MILANO 1863

Department of Electronics, Information and Bioengineering
Doctoral Programme In Information Technology

Software Engineering 2 Requirements Analysis and Specification Document

Author(s): **Filippo Balzarini -**
Christian Biffi -
Michele Cavicchioli -

Academic Year: 2023-2024

Download Page: https://github.com/filomba01/BalzariniBiffiCavicchioli

Contents

1 | Introduction

1.1. Purpose

The CodeKata is a learning method that takes inspiration from the Kata techniques and is based on continuous practice which became very popular in those years.

CodeKataBattle delineates an innovative platform geared towards enhancing students' software development skills through collaborative learning using CodeKata's fundamentals. Facilitated by educators, CKB provides a dynamic environment where students engage in code kata battles, refining their programming proficiency and embracing best practices such as the test-driven development approach.

Similar to recent initiatives addressing global challenges, CKB empowers educators to orchestrate challenges within tournaments, fostering healthy competition and cultivating an environment for skill enhancement. The platform enables educators to define battle parameters, set deadlines, and configure scoring criteria, fostering a tailored and effective learning experience.

At its core, a code kata battle presents students with programming challenges within specific language frameworks, coupled with exhaustive test cases. Teams collaboratively tackle these exercises, adhering to a test-first methodology and submitting solutions to the platform upon battle completion.

CKB's automated evaluation system ensures an impartial assessment of student submissions. Automated scrutiny covers mandatory factors, including functional aspects, timeliness, and source code quality, offering an unbiased representation of team performance. Educators can further enhance evaluations with optional manual assessments, providing nuanced insights into student work.

1.1.1. Goals

The CKB platform aims to provide a collaborative environment for students to practice and refine their software development skills. The platform enables educators to orchestrate challenges within tournaments, fostering healthy competition and cultivating an environment for skill enhancement. The platform enables educators to define battle parameters, set deadlines, and configure scoring criteria, fostering a tailored and effective learning experience.

The platform will be used by two types of users: Educators (ED) and Students (ST). The ED will be able to create competitions and battles within competitions. The ST will be able to create teams and join battles as a team or individually. The platform will provide a dashboard for code submission and automated evaluation of the code submitted. The platform will also provide a ranking of the competition and battles.

Below there is the table of goals that the platform will achieve:

#	**Goal**
G1	Enable ED to Create New Competitions
G2	Enable ED to Create Code Battles within Competitions
G3	Enable ST to Create Teams by Inviting Other STs
G4	Enable ST to Join Teams for Which They Have Been Invited
G5	Allow STs to Join Battles as a Team
G6	Allow STs to Join Battles Individually
G7	Send Notifications to STs about New Competitions and Closing of Competitions
G8	Automatically Create GitHub Repositories for Every Battle in a Competition
G9	Synchronize the Submission of Each Candidate with Their GitHub Repository
G10	Provide a Dashboard for Code Submission
G11	CBK Provides an automated evaluation of the code submitted
G12	Provide Automated Evaluation of Submitted Code
G13	Assign Points to STs Based on Code Evaluation
G14	Allow STs to View Rankings of Competition
G15	Allow STs to View Rankings of battles only in competition for which are subscribed

Table 1.1: Goals

1.2. Scope

1.2.1. World phenomena

ID	Definitions
WP1	ED wants to create a competitions
WP2	ED wants to create a battle
WP3	ST wants to participate in a competition
WP4	ST wants to participate in a battle
WP5	ST set up GitHub actions

Table 1.2: List of the world phenomena

1.2.2. Shared phenomena

ID	Definitions
SP1	ST creates an account on the platform
SP2	ED creates an account in the platform
SP3	ST logs in to the platform
SP4	ED logs in to the platform
SP5	ST registers for the competitions before the deadline
SP6	ED creates a badge with certain rules
SP7	ED manually evaluates the code submitted by students
SP8	ED creates a competition
SP9	ED creates a battle within a competition
SP10	ED closes a competition
SP11	ST pushes new commit(s) into their GitHub repository before the deadline
SP12	ST invites other STs to participate in a battle as a team
SP13	ST subscribes as a single/team for an incoming battle before the deadline
SP14	CKB sends a notification that a competition is available to ST
SP15	CKB sends a notification that a battle is created inside a competition to ST
SP16	CKB sends a notification that a competition has ended to ST
SP17	CKB sends a notification that a battle has ended to ST
SP18	CKB sends links to the GitHub repository to all the ST subscribed
SP19	CKB updates scores for each ST
SP20	CKB gives badge to ST

Table 1.3: List of the shared phenomena

1.3. Definitions, Acronyms, Abbreviations

Acronyms	Definitions
ST	Student
ED	Educator
CKB	CodaKataBattle
RASD	Requirements Analysis and Specification Document
SAT	Static Analyzer Tool
WPX	World Phenomena X
SPX	Shared Phenomena X
GX	Goal Number X
DX	Domain Assumption X
UCX	Use Case X

Table 1.4: List of the world phenomena

1.4. Revision history

1.5. Reference Documents

1.6. Document Structure

2 | Overall Description

2.1. Product perspective

2.1.1. Scenarios

Scenario 1: Professor Harry is a professor teaching at Politecnico di Milano together with professor Donald. Harry would like to encourage his students to study during the course, instead of having to study everything a few days before the exam. To do so he came up with the idea to create a challenge where the students can test their preparation and earn some extra points in the exam. While talking with other colleagues, Prof. Harry discovered CKB and he thought it was the perfect fit to implement his idea. The first thing he does is to go to the webpage of CKB and create an account by clicking the ***Sign up*** button and providing some information about himself. Afterwards he is redirected to the home page of the platform where he can click the button ***Create Competition***, and finally he inserts the name of the competition and the subscription deadline. At this point he wants to invite his colleague Donald to manage the competition with him; since he is an ED in the competition he can click on ***Invite Educator*** in the competition page, then provides the email of Donald's account, who will be part of the competition once he accepts the invite.

Scenario 2: Professor Harry is an ED of a competition, within which he wants to create a battle. To do so he enters the dashboard of the competition, clicks on the button ***Create Battle*** and provides everything the platform needs: description, test cases, build automation scripts, deadlines, accepted sizes of groups. Marco, a ST who subscribed to the competition, received the notitication about the newly created battle via email. Outside of the platform Marco agreed with a couple of friends to participate in the battle together. Marco then goes on the competition page and finds the newly created battle, here he finds two buttons, ***Participate as: 1. Loner 2. Team***; he clicks the second button to participate as a team and invites his friends by providing the platform his friends' account email. Once Marco's friends accepted the invite the subscription to the battle will be automatically finalized by the platform.

Scenario 3: Professor Harry wants to give credit to the hardest working student, so while creating the competition he decided to create a new badge. The hardest working ST is the one that has written the highest amount of code lines among all the battles in the same competition. To implement this badge Prof. Harry must create a new variable ***hardest_ worker*** and provide the code that defines how to compute the value of such variable. Some time after the specification of this new badge, a ST, participating in the competition and in the current battle, called Marco, pushes a commit to his repository. Since all students are supposed to setup *GitHub Actions*, CKB is notified about Marco's commit, so it proceeds to run the required processes to calculate the new score, but also checks if Marco acquired new badges by checking their rules. Assuming that with the last commit Marco has now the most written lines of code, CKB assigns to him the *hardest_ worker* badge.

Scenario 4: Marco and his team participated in a battle provided by Prof. Harry, one of the ED of the competition. Since the battle ends the next day, Marco wants to look at the partial rankings of the battle, so he goes on the page related to the battle and clicks on the *Results* section, and sees his team at the bottom of the chart. Understandably, Marco's team resumes to work on the problem and they are able to commit a new version of their solution, which increased their placement in the partial rankings of the battle. The submission deadline now expired and the EDs now want to manually check the work of their STs to assign manually a score to each team; to do this Prof. Harry goes on the battle page and clicks on ***Perform Manual check***, which will redirect the ED to another page where he can inspect the source code of each team and give a score to each final work. Once this consolidation phase has been declared finished by an ED, CKB sends to all the STs subscribed to the competition a notification that the battle's results are available and the global scores of the competition have been updated.

2.2. Product perspective

2.3. User characteristics

2.4. Assumptions, dependencies and 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

3.2. Functional Requirements

3.3. Performance Requirements

3.4. Design Constraints

4 | Formal Analysis Using Alloy

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

5 | Effort Spent

Members of group	**Effort spent (hours)**	
Filippo Balzarini	Introduction	$0h$
	Overall description	$0h$
	Specific requirements	$0h$
	Formal analysis	$0h$
	Reasoning	$0h$
Christian Biffi	Introduction	$0h$
	Overall description	$0h$
	Specific requirements	$0h$
	Formal analysis	$0h$
	Reasoning	$0h$
Michele Cavicchioli	Introduction	$0h$
	Overall description	$0h$
	Specific requirements	$0h$
	Formal analysis	$0h$
	Reasoning	$0h$

Table 5.1: Effort spent by each member of the group

Bibliography

List of Figures

List of Tables

List of Symbols

Variable	Description	SI unit
$\boldsymbol{u}$	solid displacement	m
$\boldsymbol{u}_f$	fluid displacement	m