



## **TERM PROJECT ANALYSIS REPORT**

**CSE 344 - SOFTWARE ENGINEERING**

**Spring 2022**

**Project Name:** Golden Railways

March 15, 2022

Şeyda Gürsel
Defne Su Özbudak
Merve Çalışkan
Efe Ömer Erdem
Fatih Tutcu
Mert Celikan



**Yeditepe University  
Faculty of Engineering  
Department of Computer Engineering**

## **TABLE OF CONTENTS**

<b>1. INTRODUCTION</b>	<b>3</b>
1.1. Purpose	3
1.2. Background	3
1.3. Motivation	5
1.3.1. Statement of the Problems with the Existing System	5
1.3.2. The New System	5
1.4. Structure of the Document	6
<b>2. FUNCTIONAL REQUIREMENTS</b>	<b>6</b>
2.1. Description of the System Functionalities	6
2.2. Description of the System Users	8
2.3. Specific Requirements	8
2.3.1. Use Case Diagram	8
2.3.2. Use Case Priority List	9
2.3.3. Use Case Specifications	11
<b>3. NON-FUNCTIONAL REQUIREMENTS</b>	<b>22</b>
3.1. Volere Template	22
<b>4. SYSTEM MODELS</b>	<b>26</b>
4.1. Object and Class Model	26
4.2. User Interface	28
<b>DEFINITIONS, ACRONYMS &amp; ABBREVIATIONS</b>	<b>30</b>
<b>GLOSSARY &amp; REFERENCES</b>	<b>30</b>

# 1. INTRODUCTION

## 1.1. Purpose

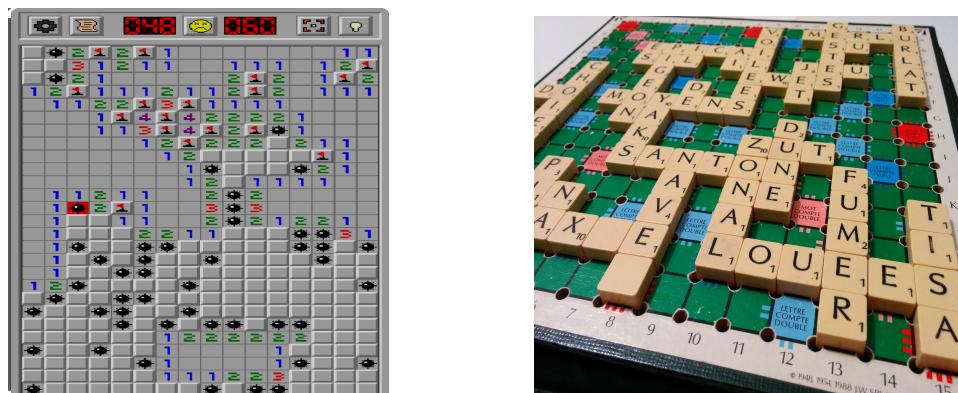
The purpose of this document is to explain the functional and non-functional requirements of the game to be designed, as well as to provide information about the content and construction process of the game.

## 1.2. Background

The domain of software we will develop is mobile game development. When we started this game project, we first got information about the creation process of the game from the VCD team. Since the aim is to create a simple one-handed game that can be played anywhere, puzzle games are preferred, so Golden Railways is simply a puzzle game with a western theme.



Puzzle games are among the most preferred games in the world. It is preferred by users in many different age ranges because it is more preferred for time evaluation and tests the problem-solving capacity of the player, and it is also a more popular choice by game creators due to its wide variety and relatively easy development and adaptability. Some of the types of puzzle games are: Action puzzle, hidden object game, piece matching game and traditional puzzle games.



Minesweeper is a very popular old puzzle game released in 1989. Scrabble is sold in 121 countries and is available in 30+ languages.

Piece matching games, that is, puzzle word origin means to find the unknown. Puzzles consist of many pieces and these pieces need to be combined in a certain order in order to be completed, which makes games such as puzzles good for some health problems of people.

It has been proven to prevent the formation of diseases such as Alzheimer's. In addition, it provides skills such as patience, which is part of personal development. For all these reasons, puzzle games are among the most preferred games.



Golden Railways is a puzzle game in which the train travels by connecting the tracks of a train. Although it initially started with a creation such as arranging city lines, it was thought that adding a story would make the game more preferable.

The idea of having some simple tasks in the game, such as collecting gold, emerged as a result of examining the candy crush game. Six levels have been added to increase the difficulty level in the game, and the difficulty of these levels has been developed according to the target audience of the game, which is 13-25 years old.



**Golden Railways**



**Candy Crush Saga**

### **1.3. Motivation**

#### **1.3.1. Statement of the Problems with the Existing System**

- ❖ 2D train track puzzle games commonly focus on the complexity of the puzzle rather than the competition.
- ❖ Puzzle games dynamics are almost the same; however, level difficulties change as the chapters move on. Even though the gaming dynamics are the same, players would lose their attention after a couple of hours. Game should attract the players to continue the gameplay as much as they can.
- ❖ Arcade games such as *Raillands* and *Conduct THIS*, do not own their player login system which allows players to be unique as they can create their nicknames.
- ❖ These types of games do not have their own ranking systems which encourage and challenge players to be in high-end ranks! This will attract them as they want more points.
- ❖ Having stories in puzzle games will keep players' attention and attraction. Stories will allow players to be integrated with the game.
- ❖ There are not enough sound effects in puzzle games.

#### **1.3.2. The New System**

Game features that we are planning to develop in the current system can be listed as;

- ❖ Puzzle levels will be hand made which makes the game more authentic than automatically generated levels.
- ❖ The game offers players a login system which leads them to have unique nicknames.
- ❖ Players can see the leaderboard which challenges them.
- ❖ The game will have a storyline throughout the game.

- ❖ New sound effects will be added, such as the sound when players collect gold.

## 1.4. Structure of the Document

We will describe the functional requirements of our game, which includes the description of the system functionalities, the description of the system users, and specific requirements.

After that, we have made our system's object and class model and UI, so that the system can be expressed more clearly. In order to express the system more clearly, we included our system's object and class model and UI. At the end of the document we are adding definitions, acronyms, abbreviations, glossary and references.

## 2. FUNCTIONAL REQUIREMENTS

### 2.1. Description of the System Functionalities

Our game is a 2D puzzle game. Thus the functionalities are listed below:

#### 2.1.1 What Inputs Should The System Accept

While in the main menu the user can interact with touch to choose an option, and then to execute it. Also, there are dynamic objects in the game such as the train rails and train.

#### 2.1.2. What Outputs The System Should Produce

**Login Menu has these options:**

→ **Login**

The user will login and be redirected to the main menu.

→ **Register**

The user will be redirected to the register screen.

**Register Menu has these options:**

→ **Registration**

The user will see the registration screen and be registered.

→ **Back**

When 'Back' is selected The user will return to the login screen

### **Main Menu has these options:**

#### **→ Play The Game**

When 'Play The Game' is selected the system will start the game.

#### **→ Options Menu**

When 'Options Menu' is selected the user can change game options such as sound etc.

#### **→ Leaderboard**

When 'Leaderboard' is selected the user will be able to see the scores of users worldwide.

#### **→ Learn How to Play**

When 'Learn How to Play' is selected then the user will see a visual that explains the game.

#### **→ Exit**

When 'Exit' is selected the game will be terminated and the application will be exited.

### **2.1.3. What Data The System Should Store That Other Systems Might Use**

The system stores 3 different data types. One is the user's name, the second is the user's password, and the third is the user's score.

### **2.1.4. What Computations The System Should Perform**

The game has to do a lot of computations. For example:

- Increasing the score proportional to the number of gold
- Position of train rails
- Order of the high scores
- Starting position
- Finish position
- Etc.

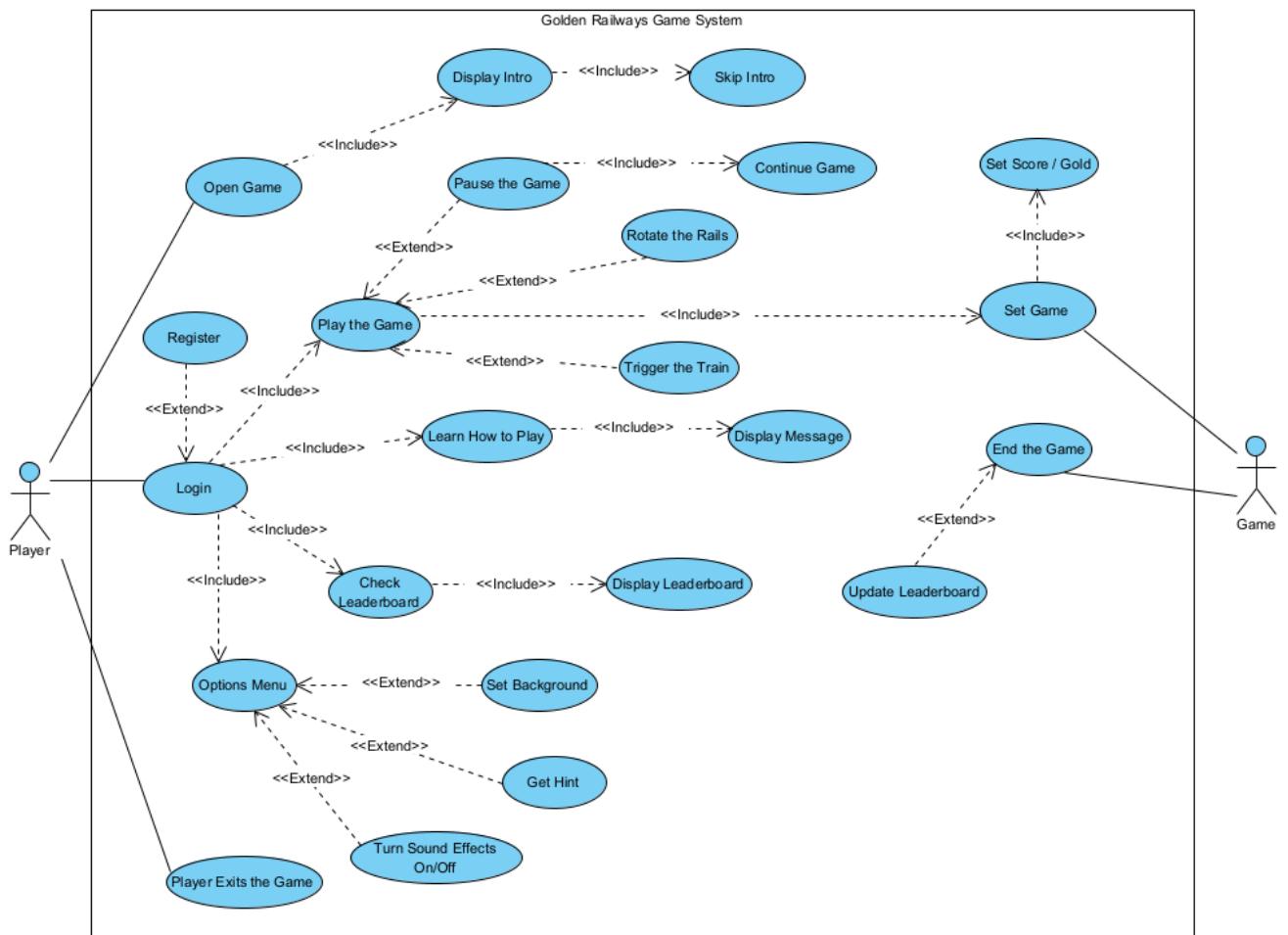
## 2.2. Description of the System Users

The system users of the game who love puzzle games with a simple structure, based on the reward system, score-based with details that do not tire the player.

Age ratings ensure that entertainment content is clearly labeled for the minimum age group for which it is suitable. We decided due to the content of our game and some inconvenient images such as guns and the game is forbidden roughly for ages under 13.

## 2.3. Specific Requirements

### 2.3.1. Use Case Diagram



### 2.3.2. Use Case Priority List

Highest Priority: 1 >> Lowest Priority: 10

Use Case #	Use Case	Priority	Explanation
1	Login	1	Must be done before playing the game.
2	Register	1	Must be done before playing the game.
3	Play the Game	1	Players should press the 'Start' button to play the game.
4	Pause the Game	4	Players must press the 'Pause' button to pause the game.
5	Learn How to Play	7	Users should learn about how to play the game by clicking the 'How to Play' button.
6	Rotate the Rails	1	Required for the player to rotate the rails and complete the level successfully.
7	Trigger the Train	1	It is required to move the train when the player connects the rails.
8	Set Score / Gold	3	It is required to set the score and gold when the player completes a phase of the game.
9	Get Hint	2	Players can get hints while playing.
10	Set Game	1	System must set the game before the game starts.

<b>11</b>	Display Leaderboard	<b>4</b>	Made to display the leaderboard.
<b>12</b>	Update Leaderboard	<b>3</b>	Made to update high scores on the leaderboard.
<b>13</b>	Options Menu	<b>6</b>	Users can change some game settings by clicking the “Options” button.
<b>14</b>	Player Exits the Game	<b>10</b>	It is optional if the user wants to exit the game.
<b>15</b>	Turn Sound Effects On/Off	<b>10</b>	Required to turn sound off and on.
<b>16</b>	End the Game	<b>5</b>	Occurs when the player successfully completes all levels.
<b>17</b>	Set Background	<b>9</b>	Required so that the player can change the game background image.
<b>18</b>	Display Message	<b>8</b>	Game displays a message that contains information about the game.
<b>19</b>	Check Leaderboard	<b>3</b>	Players can display the leaderboard.
<b>20</b>	Continue Game	<b>4</b>	Players should press the ‘Resume’ button to continue the game.
<b>21</b>	Display Intro	<b>8</b>	Made to display the game intro before the game starts.
<b>22</b>	Skip Intro	<b>9</b>	Players can skip the game intro by clicking the ‘>> Skip’ text.

### 2.3.3. Use Case Specifications

#### 1) Login

<b>Use Case</b>	Login
<b>Actor</b>	User
<b>Description</b>	For users to log in to the game using their username and password.
<b>Precondition</b>	The user must open the game and should have been registered before.
<b>Postcondition</b>	The user will be logged in.
<b>Main flow</b>	<ol style="list-style-type: none"> <li>1. System shows a login page.</li> <li>2. User fills the username and password fields with the correct information.</li> <li>3. The user will be directed to the main menu. (inclusion point: Choosing an option from the main menu)</li> </ol>
<b>Alternative path</b>	<p>2.a. -User enters the wrong username or password.          -System displays a warning.          -Use case resumes at step 1.</p>

#### 2) Register

<b>Use Case</b>	Register
<b>Actor</b>	User
<b>Description</b>	Users specify their username and password for the game.
<b>Precondition</b>	The user must open the game
<b>Postcondition</b>	The user will have created an account.
<b>Main flow</b>	<ol style="list-style-type: none"> <li>1. System shows a register page.</li> <li>2. User fills the username and password fields according to his choice.</li> <li>3. The user will be directed to the main menu.</li> </ol>
<b>Alternative path</b>	<p>1.a. User enters a username that was previously taken.          1.b. System displays a warning.          1.c. Use case resumes at step 1.</p>

### 3) Play the Game

<b>Use Case</b>	Play the Game
<b>Actor</b>	User
<b>Description</b>	For users to play the game
<b>Precondition</b>	The user must be logged in to the game and press the play button.
<b>Postcondition</b>	The user will play the game.
<b>Main flow</b>	<ol style="list-style-type: none"> <li>1. User presses the play game button.</li> <li>2. System starts the game system. (inclusion point: Start the game system)</li> <li>3. User controls the rails by clicking the mouse.</li> </ol>
<b>Alternative path</b>	-

### 4) Pause the Game

<b>Use Case</b>	Pause the Game
<b>Actor</b>	User
<b>Description</b>	For users to pause the game.
<b>Precondition</b>	The user must have started the game.
<b>Postcondition</b>	The user will pause the game.
<b>Main flow</b>	<ol style="list-style-type: none"> <li>1. User presses the ESC key on the keyboard.</li> <li>2. System pauses the game system.</li> <li>3. If the user presses the ESC key on the keyboard while the game is in progress System pauses the game. Else System resumes the game</li> </ol>
<b>Alternative path</b>	-

## 5) Learn How to Play

<b>Use Case</b>	Learn How to Play
<b>Actor</b>	User
<b>Description</b>	For users to get information about how to play the game.
<b>Precondition</b>	The user should be in the main menu.
<b>Postcondition</b>	The user can see and learn how to play the game.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. The user clicks the 'How To Play' button.</li><li>2. The system shows the rules.</li></ol>
<b>Alternative path</b>	-

## 6) Rotate the Rails

<b>Use Case</b>	Rotate the Rails
<b>Actor</b>	User
<b>Description</b>	For users to rotate the rails.
<b>Precondition</b>	The game system must have started.
<b>Postcondition</b>	The user will control the character.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. System starts the game system.</li><li>2. User presses the screen on the phone.</li><li>3. System changes rotation rails.</li></ol>
<b>Alternative path</b>	-

## 7) Trigger the Train

<b>Use Case</b>	Trigger the Train
<b>Actor</b>	User
<b>Description</b>	For users to move the train.
<b>Precondition</b>	The game system must have started
<b>Postcondition</b>	The user will move the train.
<b>Main flow</b>	<ol style="list-style-type: none"> <li>1. System starts the game system.</li> <li>2. User presses the “Train” button on the screen.</li> <li>3. If the rail directions will be true, System forward the train. Else Sound alert for wrong directions.</li> </ol>
<b>Alternative path</b>	-

## 8) Set Score (Gold)

<b>Use Case</b>	Set Score
<b>Actor</b>	Control Manager
<b>Description</b>	For changing the user’s score.
<b>Precondition</b>	The user earns gold and achieves the game score.
<b>Postcondition</b>	The gold count will increase.
<b>Main flow</b>	<ol style="list-style-type: none"> <li>1. Check collected coins.</li> <li>2. Set available score and coins.</li> </ol>
<b>Alternative path</b>	-

## 9) Get Hint

<b>Use Case</b>	Get Hint
<b>Actor</b>	User
<b>Description</b>	Get clues to solve the puzzle
<b>Precondition</b>	The user must have gold to get hints.
<b>Postcondition</b>	The user will get hint and gold will decrease.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. Check if the user has enough gold.</li><li>2. Show the hint.</li><li>3. Decrease the user's gold.</li></ol>
<b>Alternative path</b>	-

## 10) Set Game

<b>Use Case</b>	Set Game
<b>Actor</b>	Control Manager
<b>Description</b>	Set the current level.
<b>Precondition</b>	The user must have started the game
<b>Postcondition</b>	Train rails and golds are set in accordance with the episode.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. Set the rails.</li><li>2. Start and end points are determined.</li><li>3. Set the golds.</li></ol>
<b>Alternative path</b>	-

## 11) Display Leaderboard

<b>Use Case</b>	Display Leaderboard
<b>Actor</b>	Control Manager
<b>Description</b>	For users to see the leaderboard.
<b>Precondition</b>	The user must be in the main menu.
<b>Postcondition</b>	The user will see the leaderboard.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. System shows the worldwide leaderboard.</li><li>2. The user can view the top 5 players' high scores and the user's own score.</li></ol>
<b>Alternative path</b>	-

## 12) Update Leaderboard

<b>Use Case</b>	Update Leaderboard
<b>Actor</b>	Database Manager
<b>Description</b>	For sorting the users with their latest high score saved in the database
<b>Precondition</b>	The game must have ended.
<b>Postcondition</b>	The leaderboard will contain the latest data and rankings.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. Database Manager gets the user's high score.</li><li>2. If the high score has changed, Database Manager places the user's new high score.</li></ol>
<b>Alternative path</b>	-

### 13) Options Menu

<b>Use Case</b>	Options Menu
<b>Actor</b>	User
<b>Description</b>	To show users the buttons they can select.
<b>Precondition</b>	The user must be logged in to the game.
<b>Postcondition</b>	The user will be directed to their choice.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. System shows the menu.</li><li>2. The user chooses</li><li>3. Guidance is made according to the selection made by the user.</li></ol>
<b>Alternative path</b>	<ol style="list-style-type: none"><li>1. User will select 'Learn How To Play'</li><li>2. User will select 'Sound Options'</li></ol>

### 14) Player Exits the Game

<b>Use Case</b>	Player Exits the Game
<b>Actor</b>	User
<b>Description</b>	The user terminates the game.
<b>Precondition</b>	The user must be in menu options.
<b>Postcondition</b>	The game is terminated and closed.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. The user clicks the 'Exit' button.</li><li>2. It is recorded in which chapter the user is in.</li><li>3. The game is terminated.</li></ol>
<b>Alternative path</b>	-

## 15) Turn Sound Effects On/Off

<b>Use Case</b>	Turn Sound Effects On/Off
<b>Actor</b>	User
<b>Description</b>	Users can turn the sound effects on and off.
<b>Precondition</b>	The user should be in the main menu.
<b>Postcondition</b>	The user will turn the sound effects on or off.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. User presses the on/off button.</li><li>2. If there is a tick on the button then the sound is on, if there is a cross then the sound is off.</li></ol>
<b>Alternative path</b>	-

## 16) End The Game

<b>Use Case</b>	End The Game
<b>Actor</b>	Control Manager
<b>Description</b>	For the control manager to end all processes in the game.
<b>Precondition</b>	Users should complete all the levels successfully.
<b>Postcondition</b>	Game progress is completed for the user.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. The user passes the last level successfully.</li><li>2. The game progress is completed.</li></ol>
<b>Alternative path</b>	-

## 17) Set Background

<b>Use Case</b>	Set Background
<b>Actor</b>	User
<b>Description</b>	The user will change the background image of the game.
<b>Precondition</b>	The user must be logged in to the game.
<b>Postcondition</b>	Background image will change.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. The user selects the background.</li><li>2. Background is changed.</li></ol>
<b>Alternative path</b>	-

## 18) Display Message

<b>Use Case</b>	Display Message
<b>Actor</b>	Control Manager
<b>Description</b>	Game will display a message.
<b>Precondition</b>	The user must be in the menu.
<b>Postcondition</b>	The user sees messages about the game.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. Messages are displayed to the user.</li><li>2. Clicking the cross icon will remove the messages.</li></ol>
<b>Alternative path</b>	-

## 19) Check Leaderboard

<b>Use Case</b>	Check Leaderboard
<b>Actor</b>	User
<b>Description</b>	For users to check the leaderboard.
<b>Precondition</b>	The user must be in the menu.
<b>Postcondition</b>	The user will see the leaderboard.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. System shows the leaderboard.</li><li>2. The user sees high score players.</li></ol>
<b>Alternative path</b>	-

## 20) Continue Game

<b>Use Case</b>	Continue Game
<b>Actor</b>	User
<b>Description</b>	For users to continue the game.
<b>Precondition</b>	The user must be in the menu and pause the game.
<b>Postcondition</b>	The game continues
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. The user presses the “Continue” button.</li><li>2. System starts the game again.</li></ol>
<b>Alternative path</b>	-

## 21) Display Intro

<b>Use Case</b>	Display Intro
<b>Actor</b>	Control Manager
<b>Description</b>	User will see the intro screen.
<b>Precondition</b>	The user must be logged in to the game.
<b>Postcondition</b>	The user sees the intro screen.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. Intro is shown to the user.</li><li>2. When the intro is finished, it navigates to the main menu.</li></ol>
<b>Alternative path</b>	-

## 22) Skip Intro

<b>Use Case</b>	Skip Intro
<b>Actor</b>	User
<b>Description</b>	The user skips the intro part.
<b>Precondition</b>	The user must see the intro.
<b>Postcondition</b>	The intro is terminated.
<b>Main flow</b>	<ol style="list-style-type: none"><li>1. The user presses the 'Skip intro' button.</li><li>2. The intro is terminated.</li></ol>
<b>Alternative path</b>	-

### **3. NON-FUNCTIONAL REQUIREMENTS**

A non-functional requirement is a specification portraying the system's operation capabilities and limitations that upgrade its usefulness. NFRs consist of quality, platform and process requirements. Each system has its own unique set of requirements. We determined the NFRs of our software design according to the important features such as usability, performance, reliability, compatibility, resource usage and maintainability.

#### **3.1. Volere Template**

<b>Requirement No:</b> 1	<b>Requirement Type:</b> NFR (Security)	<b>Use Cases:</b> 1,2
--------------------------	---	-----------------------

**Description:** Any personal information should not be asked to the user.

**Rationale:** The game should not require the user's personal information to be played.

**Source:** GoRa Team

**Fit Criterion:** The only needed information is username and password to save scores.

<b>Customer Satisfaction:</b> 4	<b>Customer Dissatisfaction:</b> 3
---------------------------------	------------------------------------

<b>Supporting Material:</b> None	<b>History:</b> March 18, 2022
----------------------------------	--------------------------------

<b>Requirement No:</b> 2	<b>Requirement Type:</b> NFR (Usability)	<b>Use Cases:</b> -
--------------------------	--	---------------------

**Description:** The game should have a simple UI and the user shall play the game with only one hand.

**Rationale:** The game shall provide ease of use and a simple user interaction.

**Source:** GoRa Team

**Fit Criterion:** The number of interactions with the game can be observed.

<b>Customer Satisfaction:</b> 5	<b>Customer Dissatisfaction:</b> 5
---------------------------------	------------------------------------

<b>Supporting Material:</b> None	<b>History:</b> March 18, 2022
----------------------------------	--------------------------------

**Requirement No:** 3      **Requirement Type:** NFR (Response Time)      **Use Cases:** -

**Description:** The maximum response time between user interaction and game response should be 0.5 seconds.

**Rationale:** The game shall be played without an important delay.

**Source:** GoRa Team

**Fit Criterion:** The response time can be measured by division of total time and user clicks in the game.

**Customer Satisfaction:** 5

**Customer Dissatisfaction:** 5

**Priority:** Essential

**Conflicts:** None

**Supporting Material:** None

**History:** March 18, 2022

**Requirement No:** 4      **Requirement Type:** NFR (Performance)      **Use Cases:** 10

**Description:** The game should load in 7 seconds at startup.

**Rationale:** The game should start fast enough in order to provide user satisfaction.

**Source:** GoRa Team

**Fit Criterion:** The game start time can be observed by timer easily.

**Customer Satisfaction:** 4

**Customer Dissatisfaction:** 4

**Priority:** Essential

**Conflicts:** None

**Supporting Material:** None

**History:** March 18, 2022

**Requirement No:** 5    **Requirement Type:** NFR (Maintainability)    **Use Cases:** -

**Description:** The game code must be compatible with constant change.

**Rationale:** New changes and additions shall be made easily without changing the most part of the code.

**Source:** GoRa Team

**Fit Criterion:** It can be observed when new features are added to the game.

**Customer Satisfaction:** 4

**Customer Dissatisfaction:** 4

**Priority:** Desirable

**Conflicts:** None

**Supporting Material:** None

**History:** March 18, 2022

**Requirement No:** 6    **Requirement Type:** NFR (Reusability)    **Use Cases:** -

**Description:** The game programming should consist of clean code and must be clear to other developers..

**Rationale:** Complex and unalterable structure should be avoided in programming.

**Source:** GoRa Team

**Fit Criterion:** The ability of multiple developers to easily understand and develop the code can be used as a benchmark.

**Customer Satisfaction:** 2

**Customer Dissatisfaction:** 2

**Priority:** Desirable

**Conflicts:** None

**Supporting Material:** None

**History:** March 18, 2022

**Requirement No: 7      Requirement Type: NFR (Resource Usage)      Use Cases: -**

**Description:** The disk usage should not be more than 1GB.

**Rationale:** The disk usage that the user would not prefer should be avoided.

**Source:** GoRa Team

**Fit Criterion:** It can be measured by looking at the size of game code and materials such as photos, videos and sounds.

**Customer Satisfaction:** 4

**Customer Dissatisfaction:** 4

**Priority:** Desirable

**Conflicts:** None

**Supporting Material:** None

**History:** March 18, 2022

**Requirement No: 8      Requirement Type: NFR (Computing Platform)      Use Cases: -**

**Description:** The game shall be played on a mobile platform.

**Rationale:** The platform should be available to use anywhere in daily life.

**Source:** GoRa Team

**Fit Criterion:** The game can be launched on a mobile platform.

**Customer Satisfaction:** 5

**Customer Dissatisfaction:** 4

**Priority:** Desirable

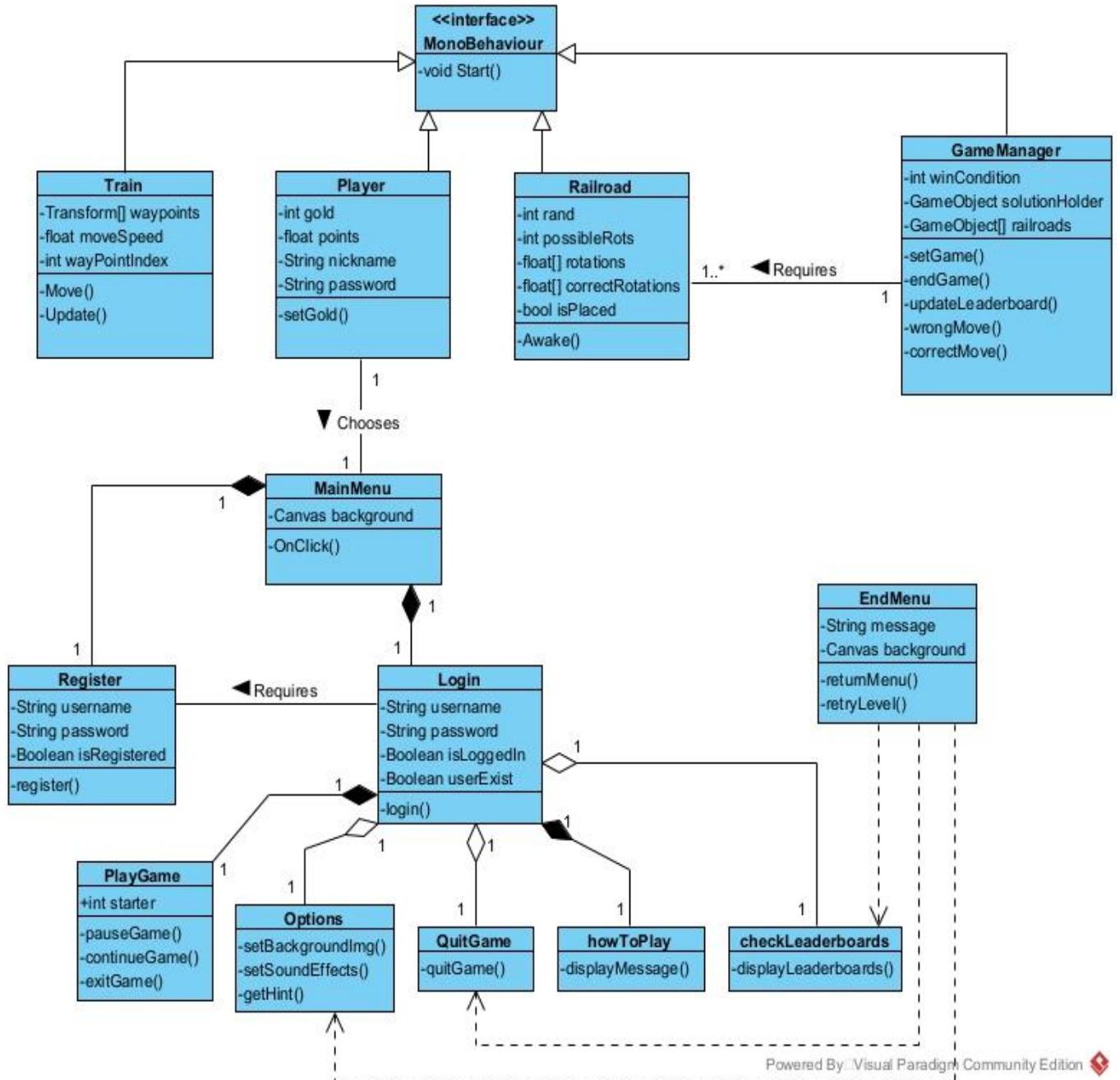
**Conflicts:** None

**Supporting Material:** None

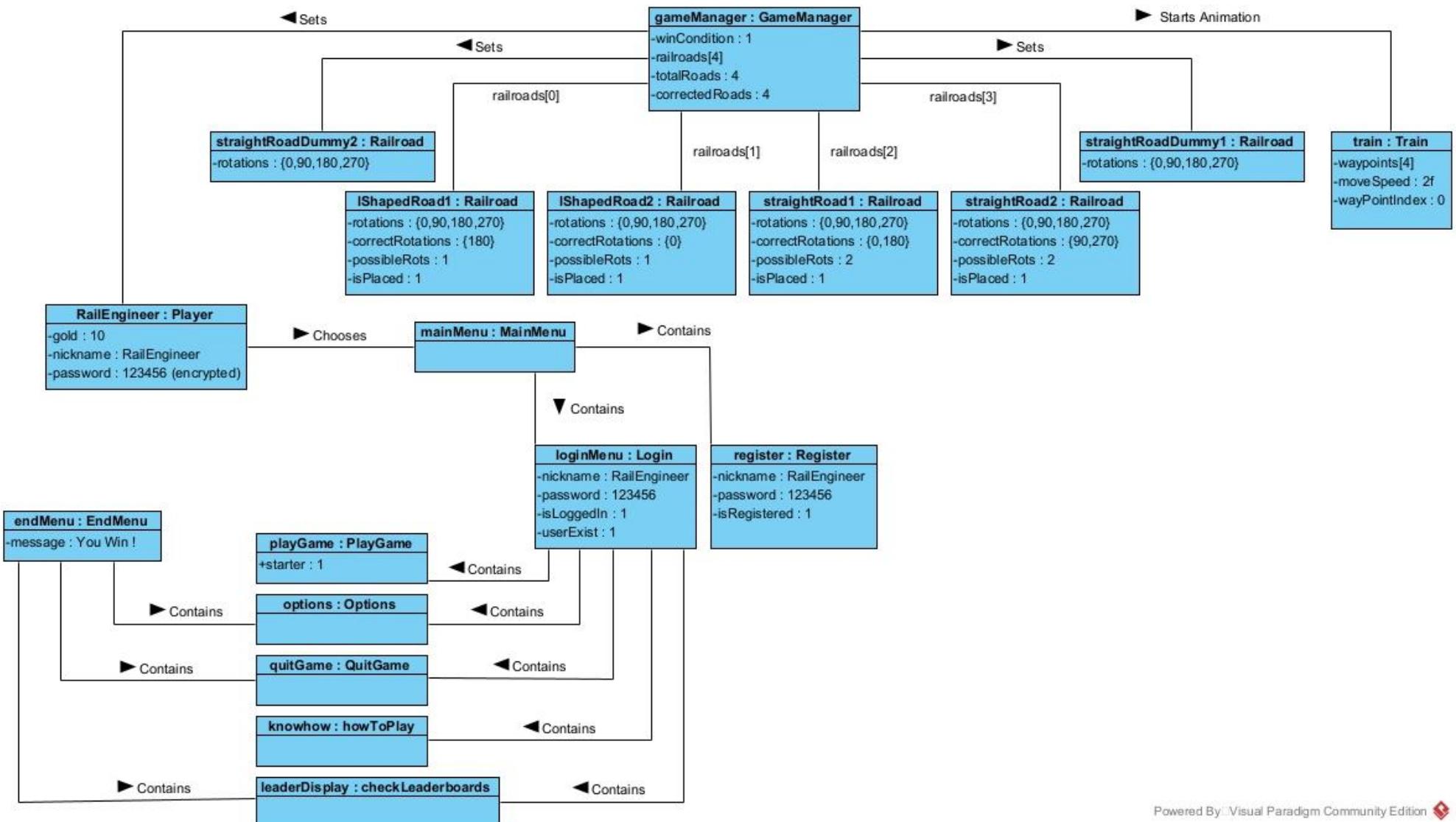
**History:** March 18, 2022

## 4. SYSTEM MODELS

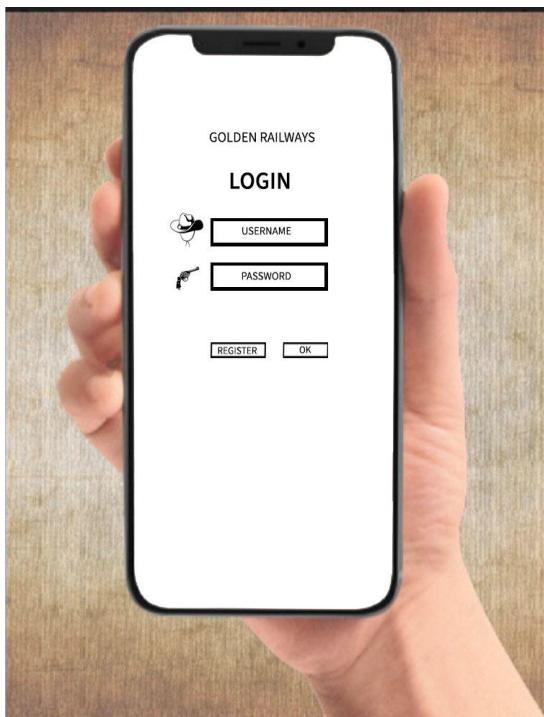
### 4.1. Object and Class Model



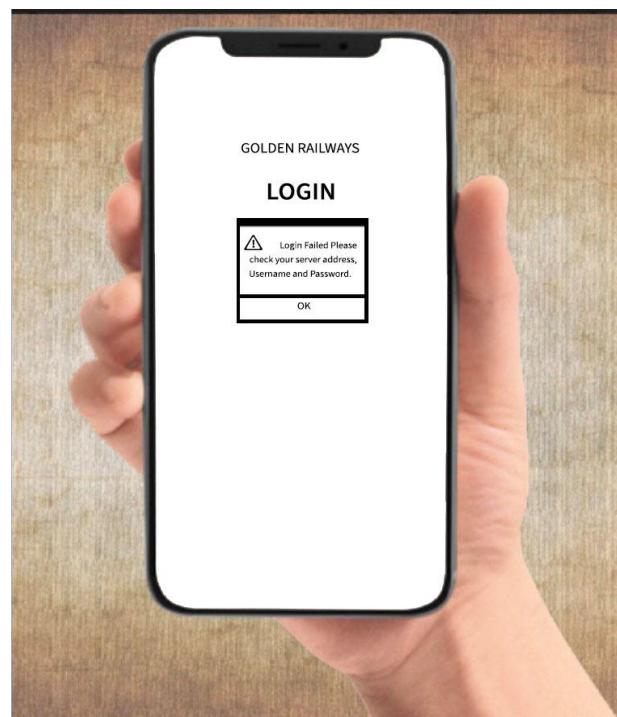
Powered By Visual Paradigm Community Edition



## 4.2. User Interface



Login Page



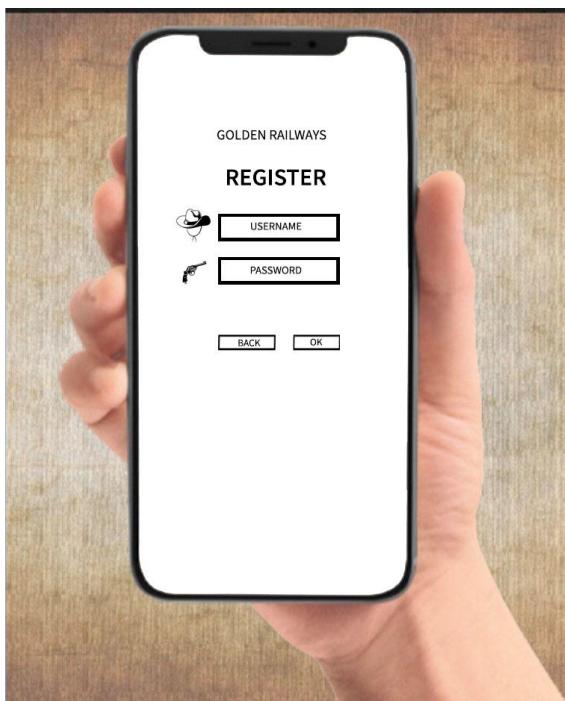
Login Fail Page



Main Menu



Game Menu



Register Page



Options Menu



Leaderboard Page

## **DEFINITIONS, ACRONYMS & ABBREVIATIONS**

**CSE:** Computer Science and Engineering

**VCD:** Visual Communication Design

**NFR:** Non-Functional Requirements

**GoRa:** Golden Railways

**2D:** Two-Dimensional

**UI:** User Interface

**GB:** Gigabyte

## **GLOSSARY & REFERENCES**

- [1] Lethbridge, T. C., Laganière, R. (2005). Object-Oriented Software Engineering: Practical Software Development using UML and Java, McGraw-Hill Education.
- [2] [Use Case Specifications, Visual Paradigm Community](#)
- [3] [Puzzle Games, Wikipedia](#)
- [4] [UML Class Diagram Tutorial, Visual Paradigm Community](#)
- [5] [Benefits of Puzzle Games, Morinaga Platinum](#)
- [6] [Puzzle Games, Hurriyet](#)
- [7] [Candy Crush Saga, Wikipedia](#)
- [8] [UML Association vs Aggregation vs Composition, Visual Paradigm Community](#)
- [9] [UML Archives, Visual Paradigm Community](#)
- [10] [Functional Requirements, Jama Software](#)
- [11] [Non-Functional Requirements, AltexSoft](#)