Innocent Nsabimana

Scalable Software

Project 3

**System Requirements for “Frogar Game” App**

**Goal**: Interactive android Frogar Game for a single player

**Scope**: Single player game to run on Android smartphones

**Actors and Goals**:

1. User: wants to play the Frogar Game
2. The app: allows the user to play the game

**In/ Out list**

|  |  |  |
| --- | --- | --- |
| **Topic** | **In** | **Out** |
| Host one player | In |  |
| Hosting multiple players |  | Out |
| Creating a user account |  | Out |
| Playing multiple levels | In |  |

**Use Cases:**

**Use Case 1 – Start Frogar Game App**

**Precondition:** Frogar GameApp is installed on the android device

**Trigger:** User starts Frogar Game App

**Main Success Scenario**:

1. The user clicks on the app’s icon.
2. The system shows a welcome screen and directs a user to the game screen after a few seconds.

**Minimal Guarantee:** program closes and exits on failure.

**Success Guarantee:** The system opens and directs the user to the game screen.

**Use Case 2 – Start the Game**

**Precondition:** The user is on the game screen

**Trigger:** The user clicks on the “start game” button

**Main Success Scenario**:

1. The system activates the obstacles to move from top to the bottom
2. The system activates the frog to move left to right and right to left, as the user moves the device.

**Minimal Guarantee:** program closes and exits on failure.

**Success Guarantee:** The frog and the obstacles get activated.

**Use Case 3 – Score the Game**

**Precondition:** The user is on the game screen, and has some lives left

**Trigger:** The frog dodges, or get hit by the obstacle

**Main Success Scenario**:

1. The user is moving the frog left to right and vice versa.
2. The system detects a collision between an obstacle and a frog, player’s lives get reduced by one.
3. The system detects the frog dodging an obstacle, player’s scores get increased by 2.

**Minimal Guarantee:** program closes and exits on failure.

**Success Guarantee:** The frog and the obstacles get activated.

**Use Case 4 – Ending the Game**

**Precondition:** The user is on the game screen

**Trigger:** The use has reached 100 points or 0 lives

**Main Success Scenario**:

1. The system stops the game and prints the player’s scores.
2. The user reached 100 points, go to use case 6
3. The user reached 0 lives, go to use case 5

**Minimal Guarantee:** The program closes and exits on failure.

**Success Guarantee:** The player’s score is printed.

**Use Case 5 – Restarting the Game**

**Precondition:** The user is on the game screen

**Trigger:** The use has reached 100 points or 0 lives

**Main Success Scenario**:

1. The system prompts the user to exit or restart the game.
2. The user selects restart.
3. Go to use case 2

**Minimal Guarantee:** The program closes and exits on failure.

**Success Guarantee:** The game is restarted.

**Use Case 6 – Next Game Level**

**Precondition:** The user is on the game screen

**Trigger:** The use has reached 100 points and still have lives left.

**Main Success Scenario**:

1. The system prompts the user to exit, restart, or go to the next level.
2. The user selects the next level.
3. The system multiplies the speed of the obstacles by 2
4. Go to use case 2

**Minimal Guarantee:** The program closes and exits on failure.

**Success Guarantee:** The player continues to a more difficult level.

**Glossary**

**Lives**: a set number of chances that the frog can be hit without ending the game. In this project, the player will start with 3 lives.

**Levels**: is the game experience, which is influenced by the speed of the obstacles.

* Easy level: the obstacles are moving at a slow speed.
* Medium level: the obstacles are moving at a relatively faster speed.
* Hard level: the obstacles are moving at a very high speed.