

# **Ruins of Corrosa City**

## **Software Requirements Specification**

Final Version

2015-05-31

Jacob Duvander, Joel Hultin, Jennifer Linder, Jenny Orell

## **1. Introduction**

### **1.1 Purpose of application**

The purpose of our application is pure entertainment.

### **1.2 General characteristics of application**

The application will be a desktop, singleplayer platform game for Windows.

The game will have some different levels. On these levels, there will be various objects that the player can pick up such as other characters and objects that increase the player's score.

While playing, the player will be able to switch which character to play as between the ones he or she has picked up. The characters have different characteristics and weapons. There will also be a timer counting upwards on the levels, but there is no time limit for the level.

There will also be various types of enemies that deal different amount of damage to the player. The player can kill these enemies.

### **1.3 Scope of application**

The game will be two dimensional (2D). There are a few various levels, characters and enemies. Each level only take a few minutes each to complete. Characters have different characteristics and weapons. A mouse will be required for aiming weapons.

There will be no feature to save game progress between sessions, but settings will be possible to save.

Game content can easily be added or modified by the community.

### **1.4 Objectives and success criteria of the project**

1. A functional start menu.
2. Pause menu.
3. The player can configure the controls.
4. Collect and switch between at least three characters with different characteristics.
5. Different weapons for characters.
6. The player can shoot.
7. At least two different types of close combat enemies.
8. The player can deal damage to enemies.
9. The enemies can deal damage to the player.
10. The characters and enemies can die.
11. Score.

12. Timer.

## **1.5 Definitions, acronyms and abbreviations**

2D - two dimensions.

Java - platform independent program language.

JRE - Java Runtime Environment.

.jar-file - a compressed directory containing a Java application and a main class.

.zip-file - a compressed directory.

## **2. Requirements**

### **2.1 Functional requirements**

As a user you are able to:

1. Start a new game.
2. Configure controls.
3. Be able to pick up objects like pizza and characters.
4. Move the character.
5. Switch character to play as while in-game.
6. Die as a character.
7. Kill enemies.
8. Shoot.
9. Pause the game.
10. Be able to win the game.
11. Exit the game.

### **2.2 Non-functional requirements**

#### **2.2.1 Usability**

The user will be able to configure the game controls for easy usability. Users will be able to easily start a game and exit. It will be clear of which characters are enemies and which are allies. When the active character or an enemy receives damage the user will be notified visually.

#### **2.2.2 Reliability**

NA

#### **2.2.3 Performance**

NA

### **2.2.4 Supportability**

It is possible to expand the application without modifying the code itself.  
The application can be launched on Windows.

### **2.2.5 Implementation**

To run the application, Java Runtime Environment must be installed on the computer. For availability of full content the application must have access to file reading and writing in the directory where it's installed.

### **2.2.6 Packaging and installation**

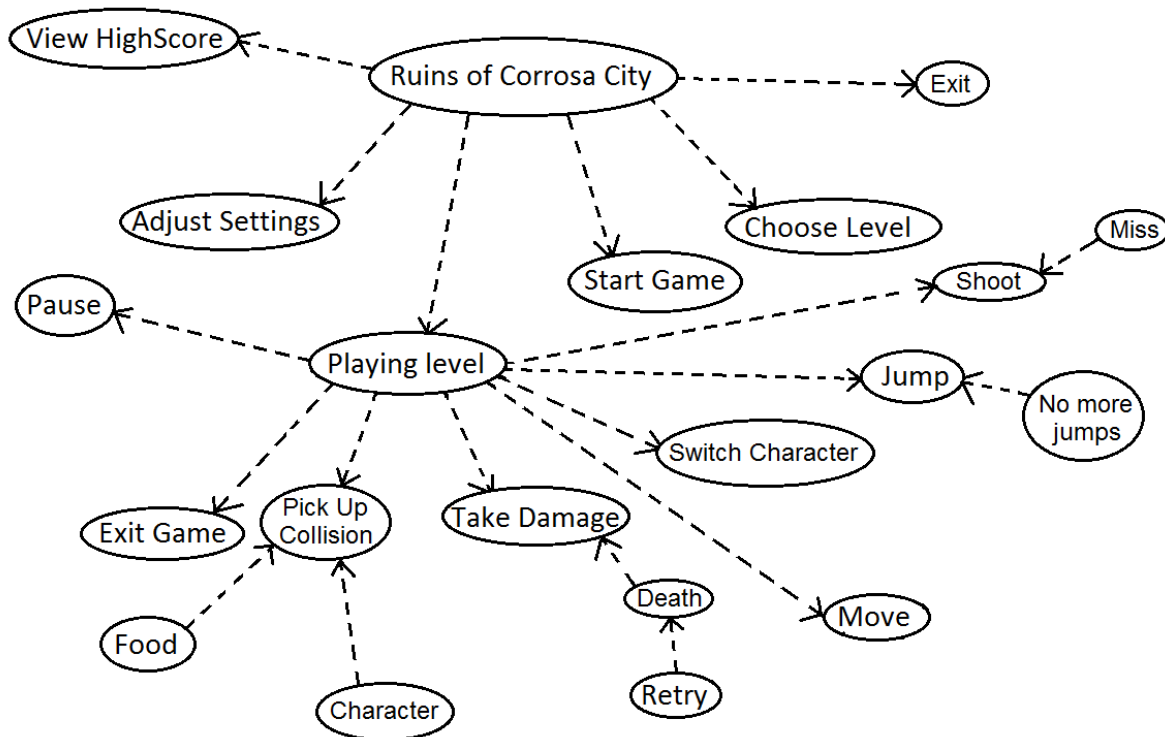
The application will be given in a .zip-file containing a "How to" text-file, a .jar-file and a script "Play" for executing that jar-file. For saving of settings between sessions the jar-file must be unpacked as described in the "How to" file and launched with the "Play\_if\_unzipped" script.

### **2.2.7 Legal**

Not covered.

## **2.3 Application models**

### **2.3.1 Use case model**



To see individual use cases, see appendix A.

### 2.3.2 Use cases priority

High:

- Playing level
- Shoot
- Switch character
- Move
- Jump
  - No more jumps
- Exit game

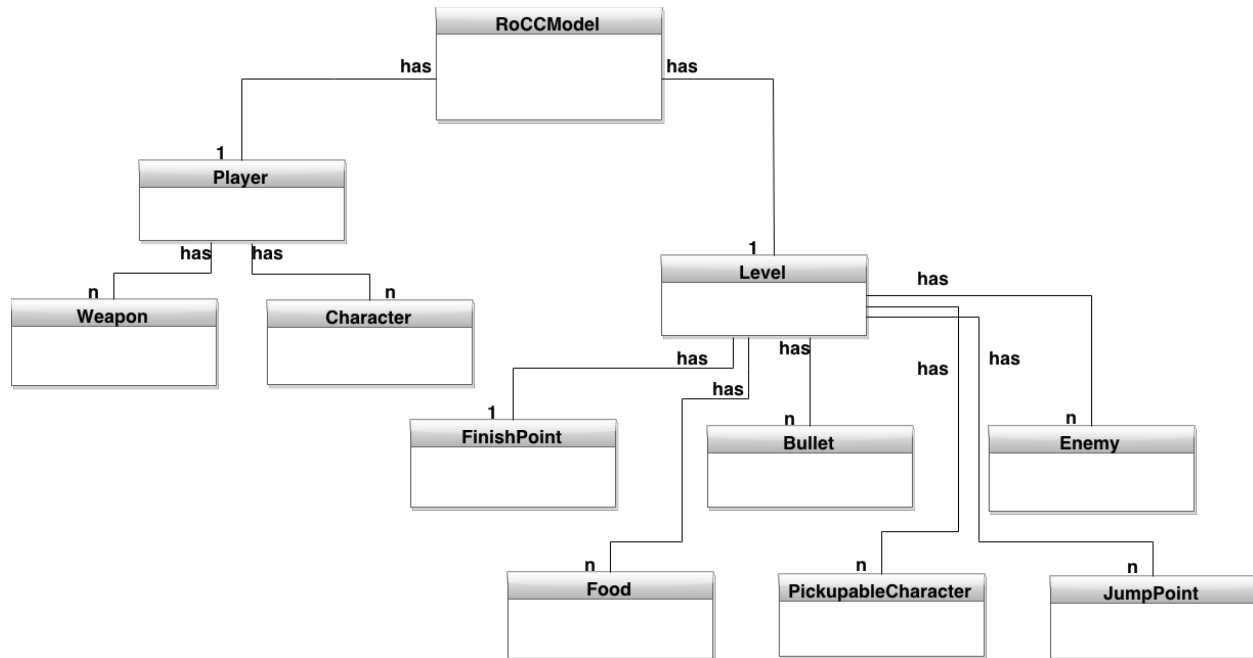
Medium:

- Start game
- Take damage
  - Death
- Pick up
  - Character
  - Food

Low:

- Choose level
- Adjust settings
- View highscore
- Exit
- Retry
- Pause

### 2.3.3 Domain model



### 2.3.4 User interface

The user interface is resizable and supports fullscreen mode. All essential information is clearly shown.



Figure: The start menu.

As shown in appendix B all menu screens have a uniform style with a back button.



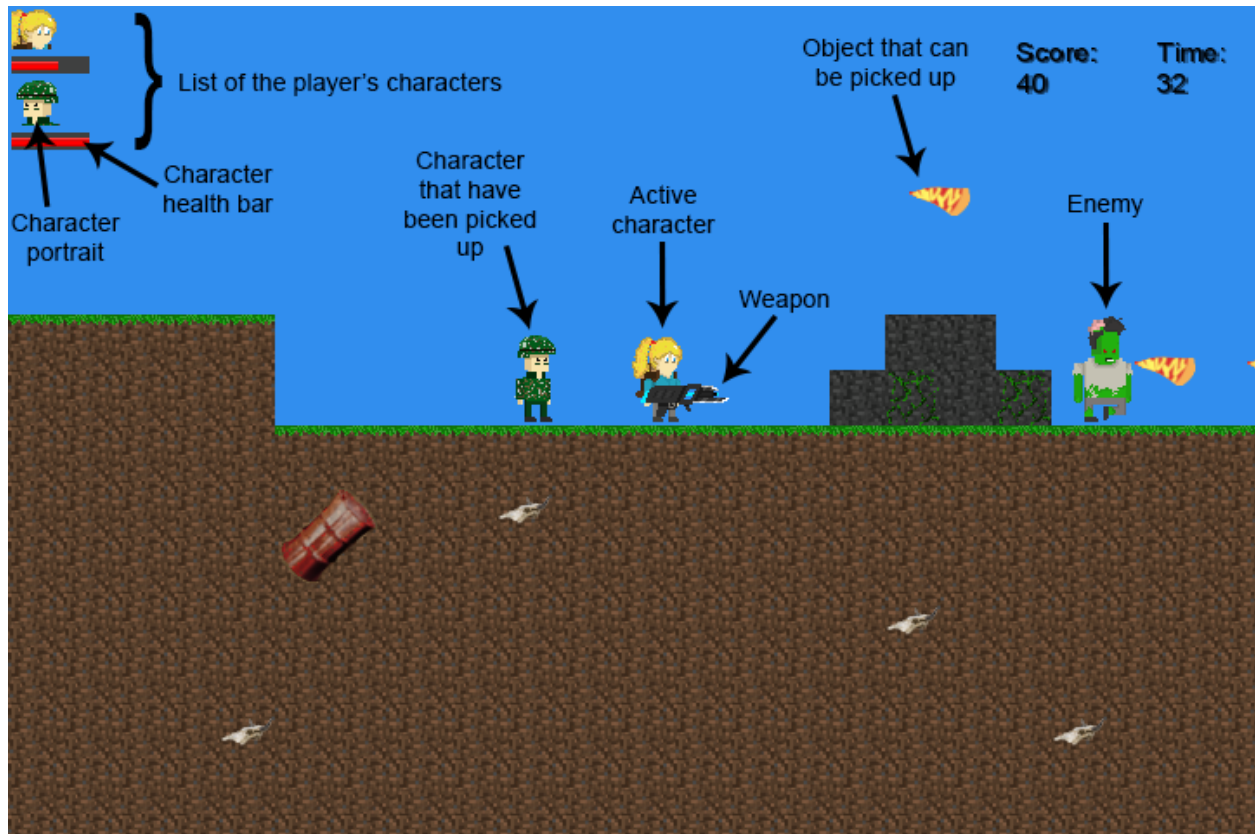


Figure: The graphical interface in-game with explanations.

The list to the top left shows all characters the player has collected so far. The characters are represented with a portrait and a health bar showing its health status.

The character the player is currently playing as is always shown in the middle of the screen, and is the only character wielding a weapon.

Other collected characters are following the active character at a distance.

Enemies are evil- and non-human-looking creatures while all allies are human.

The player's score along with a timer is shown in the interface's upper right corner.

When a character or enemy takes damage, they will flash red for a few seconds.

## 2.4 References

NA

# Appendix A

## Use case: Start Game

**Summary:** Selecting the “new game” button in menu.

**Priority:** Medium

**Extends:** -

**Includes:** Start application

**Participator:** User

**Normal flow of events**

	Actor	System
1	Clicks on "new game"	
2		Load new game

## Use case: Choose level

**Summary:** A user can start a level by selecting “choose level” button in the menu and then selecting a level to play. This removes the choose level screen and displays the selected level.

**Priority:** Medium

**Extends:** -

**Includes:** Start application, Select Choose level

**Participator:** User

**Normal flow of events**

The selected level is loaded.

	Actor	System
1	Clicks on "choose level" button	
2		choose level screen appears
3	Click on level button	
4		Load chosen level

## Use case: Adjust Settings

**Summary:** The user is in the “options” view and adjusts a setting.

**Priority:** Low

**Extends:** -

**Includes:** Start application, Select Options

**Participator:** User

### Normal flow of events

The user adjusts a setting.

	Actor	System
1		

## Use case: Configure Controls

**Summary:** The user interacts with the “configure controls” interface.

**Priority:** Low

**Extends:** Adjust Settings

**Includes:** -

**Participator:** User

### Normal flow of events

The user changes the keybindings.

	Actor	System
1	Left click "configure controls" button	
2		Configure Controls screen appear
3	Left click "key" button	
4		"Press key" appears
5	Press key	
6		Key is set

## Use case: Fullscreen

**Summary:** User check fullscreen checkbox.

**Priority:** Low

**Extends:** Adjust Settings

**Includes:** -

**Participator:** User

### Normal flow of events

The application enters fullscreen mode.

	Actor	System
1	Left click "fullscreen" checkbox	

2		Application enters fullscreen
---	--	-------------------------------

## Use case: View Highscore

**Summary:** The user enters the highscore screen.

**Priority:** Low

**Extends:** -

**Includes:** Start application

**Participator:** User

**Normal flow of events**

	Actor	System
1	Left click "highscore" button	
2		Highscore screen appears

## Use case: Exit

**Summary:** User clicks exit button in the start menu.

**Priority:** High

**Extends:** -

**Includes:** Start application

**Participator:** User

**Normal flow of events**

Application closes.

	Actor	System
1	Left click "exit" button	
2		Application exits

## Use case: Move

**Summary:** The users presses the selected left or right button.

**Priority:** High

**Extends:** -

**Includes:** Select level, Start level

**Participator:** User

**Normal flow of events**

The active character moves in the right direction.

	Actor	System
1	Press down "d" ("a")	
2		Character moves right (left)
3	Release "d" ("a")	

4	Character stops
---	-----------------

### Use case: Obstacle

**Summary:** The character tries to move but there is an obstacle in the way.

**Priority:** High

**Extends:** Move

**Includes:** -

**Participator:** User

**Normal flow of events**

	Actor	System
2.1.1		Character stands still

### Use case: Take Damage

**Summary:** The active character takes damage.

**Priority:** Medium

**Extends:** -

**Includes:** Select level, Start level

**Participator:** User

**Normal flow of events**

	Actor	System
2.2.1		1: Lose HP
		2: Character flash red

### Use case: Death

**Summary:** The active character takes lethal damage and dies, change active character.

**Priority:** Medium

**Extends:** Take Damage

**Includes:** -

**Participator:** User

**Normal flow of events**

	Actor	System
2.3.1		Character dies
2.3.2		Switch active character.

**Use case: Retry****Summary:** The user retries the level.**Priority:** Low**Extends:** Death**Includes:** -**Participator:** User**Normal flow of events**

	Actor	System
1		Defeat screens appears
2	Left click "retry" button	
3		Level restarts

**Use case: Shoot****Summary:** A bullet is fired from the active character's weapon.**Priority:** High**Extends:** -**Includes:** Select level, start level**Participator:** User**Normal flow of events**

	Actor	System
1	Left click	
2		Fire bullet towards cursor
3		Enemy hit and takes damage

**Use case: Enemy dies****Summary:** Enemy takes lethal damage and dies.**Priority:** Medium**Extends:** Shoot**Includes:** -**Participator:** User**Normal flow of events**

	Actor	System
3.1.1		Enemy is hit and dies

3.1.2		Increase score
-------	--	----------------

### Use case: Miss

**Summary:** The fired bullet doesn't hit anything.

**Priority:** High

**Extends:** Shoot

**Includes:** -

**Participator:** User

**Normal flow of events**

	Actor	System
3.2.1		

### Use case: Jump

**Summary:** The active character jumps.

**Priority:** High

**Extends:** -

**Includes:** Select level, Start level

**Participator:** User

**Normal flow of events**

	Actor	System
1	Press down space	
2		Character jumps
3		Character lands

### Use case: No more jumps

**Summary:** The active character can't jump anymore.

**Priority:** High

**Extends:** Jump

**Includes:** -

**Participator:** User

**Normal flow of events**

	Actor	System
2.1.1		

**Use case: Collision between active character and pick up-able object****Summary:** The active character collides with a pick up-able object.**Priority:** Medium**Extends:** -**Includes:** Select level, Start level**Participator:** User**Normal flow of events**

	Actor	System
1	Character collides with pick up-able object	
2		Delete object

**Use case: Collision between active character and food****Summary:** The active character picks up the food and the player's score increases.**Priority:** Low**Extends:** Collision between active character and pick up-able object**Includes:** -**Participator:** User**Normal flow of events**

	Actor	System
2.1.1		Increase player score

**Use case: Collision between active character and pick up-able character****Summary:** The active character picks up the character who is then added to the playable characters list.**Priority:** Medium**Extends:** Collision between active character and pick up-able object**Includes:** -**Participator:** User**Normal flow of events**

	Actor	System
2.1.1		Add character to list



**Use case: Victory**

**Summary:** Active character collides with victory point and enters victory screen.

**Priority:** Medium

**Extends:** Collision between active player and victory point

**Includes:** -

**Participator:** User

**Normal flow of events**

	Actor	System
1	Walk on victory point	
2		Victory screen appears

**Use case: Pause**

**Summary:** The user pauses the game.

**Priority:** Low

**Extends:** -

**Includes:** Select level, Start level

**Participator:** User

**Normal flow of events**

The game pauses and a pause window appears on top of the game.

	Actor	System
1	Press down "esc"	
2		Game pauses
3		Pause screen appears

## Appendix B



Figure: Screen for the level chooser.

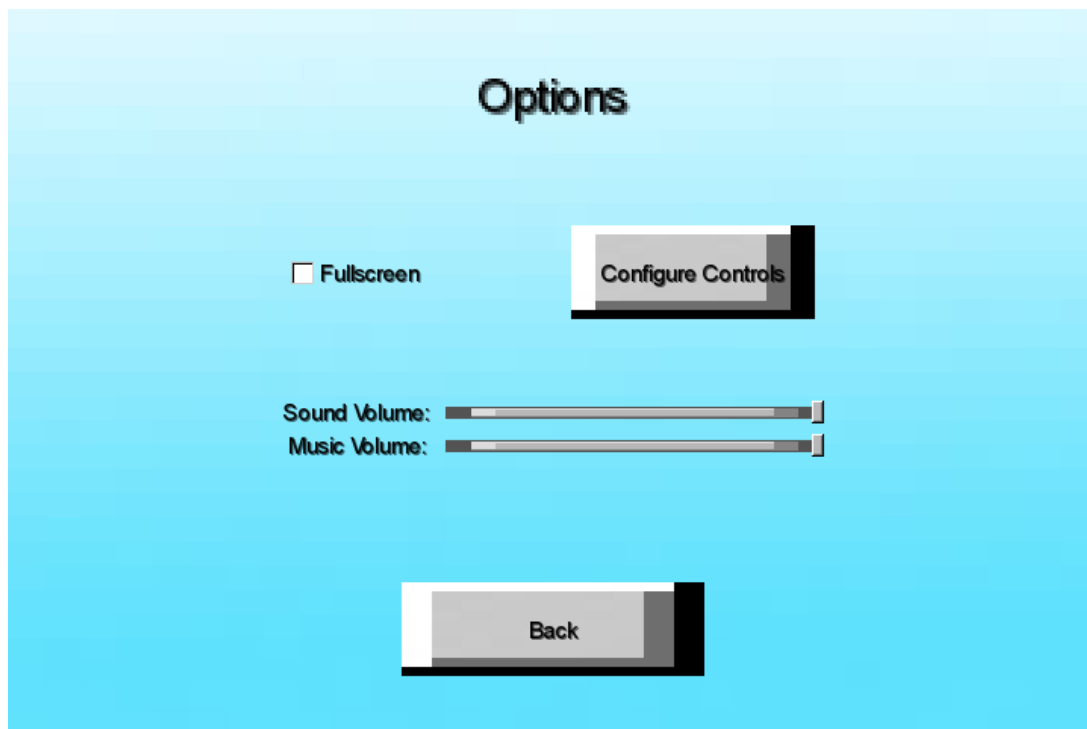


Figure: The options screen with its adjustable settings.

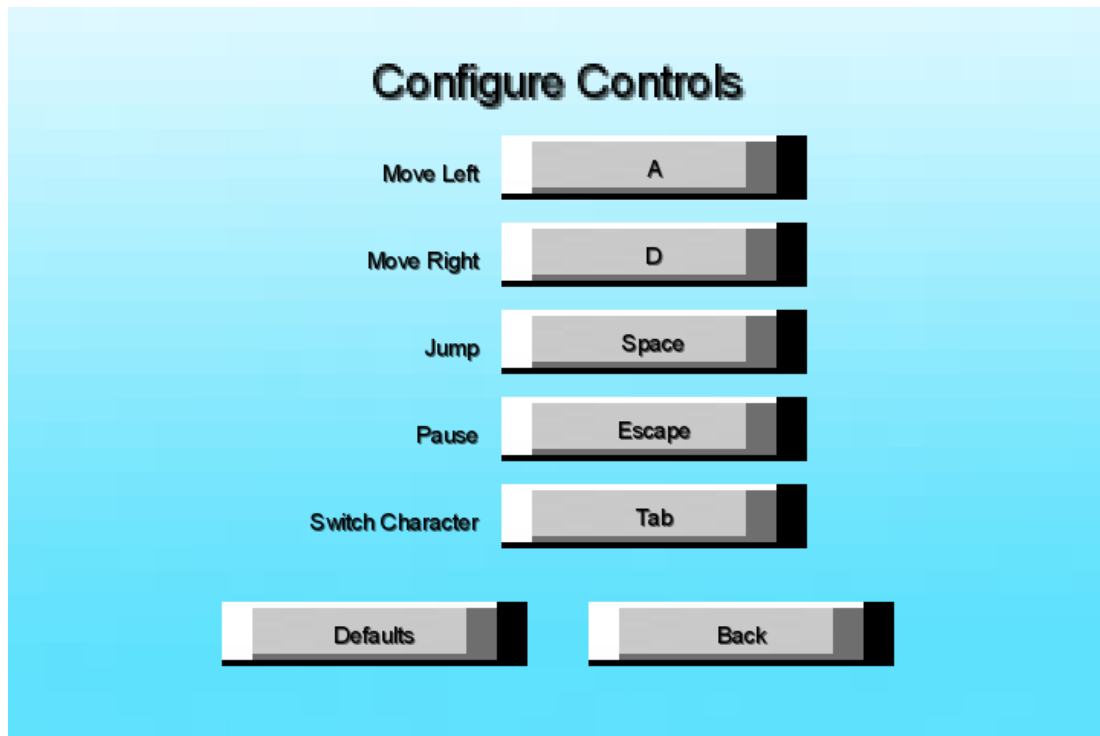


Figure: The screen with the customizable controls in the options menu.



Figure: The highscore screen.

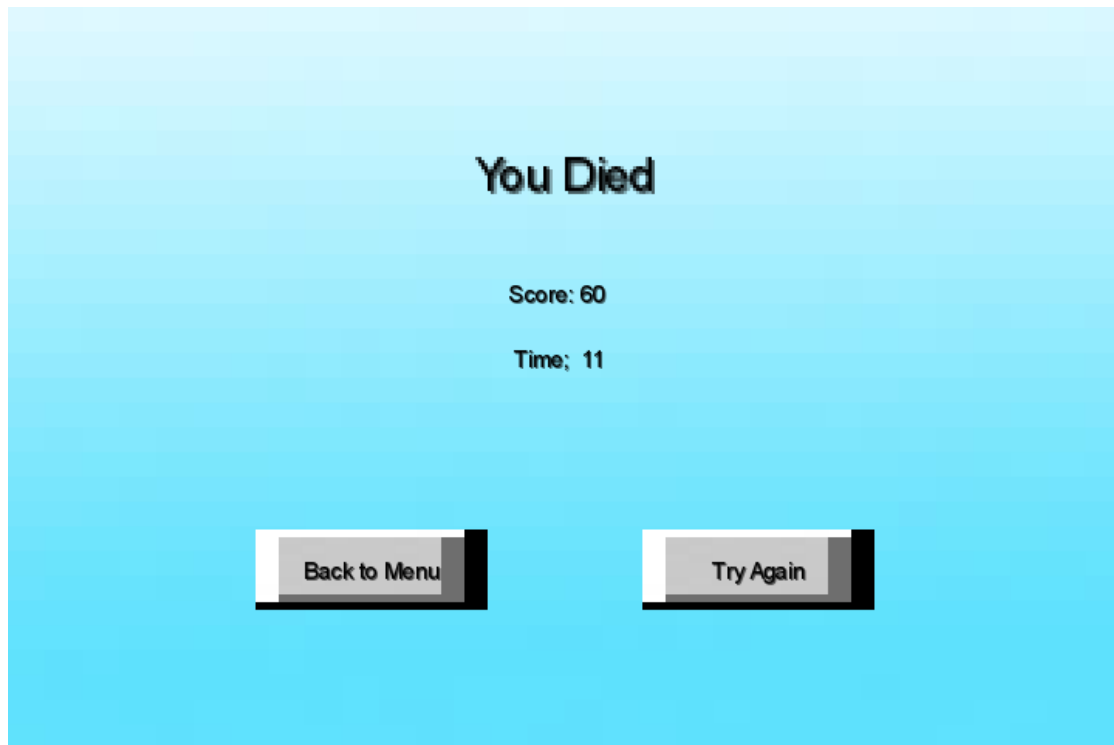


Figure: The “Game over” screen showing the player’s score and time of the failed level.

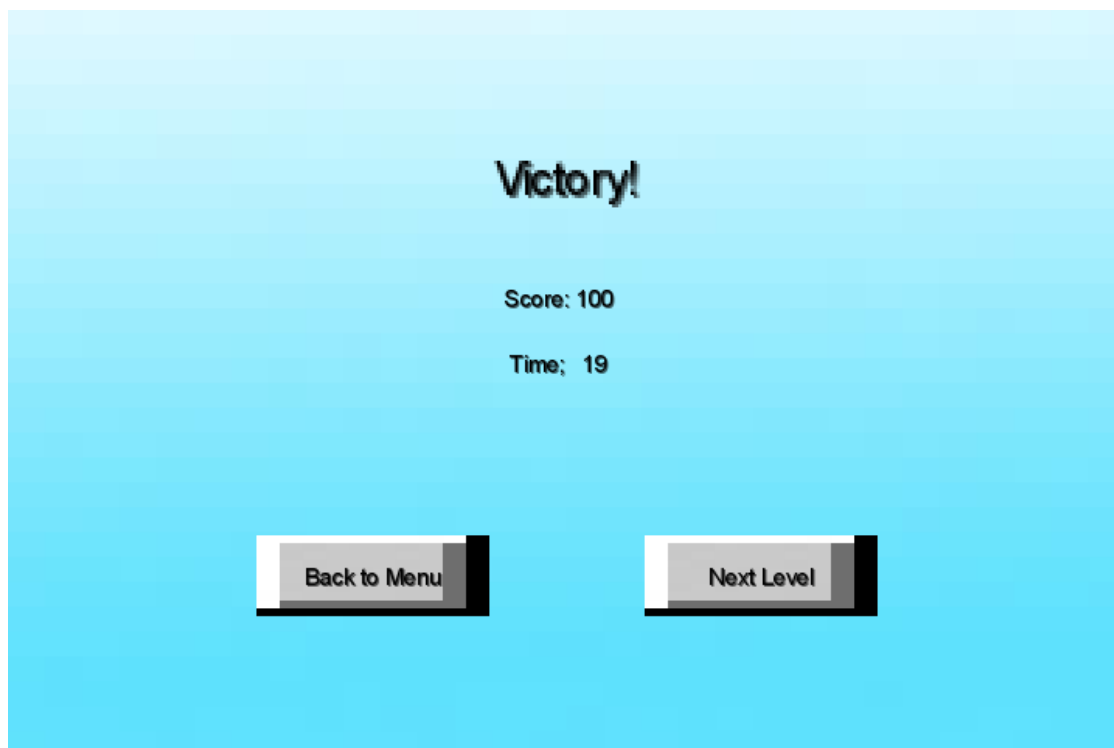


Figure: The “Victory” screen showing the player’s score and time of the passed level.