**Project for obtaining the professional certificate at**

**INFORMATICS**

**Ruby's Adventure**

**2D Game**

**Coordinating teachers: Student:**

Brîndușa Morari Cristea Tudor

Daniela Chitiul

**Class 12 A – 2020-2021**

Table of Contents

1. General presentation of the theme ………………………..….…..…... 2
2. Making and using the application ……………………..........…..…… 2
3. Possible extensions of the application ……………………..…..…… 10
4. Bibliography………..………………………………………...…….. 10

Application overview

The main theme of the game I developed is adventure. The main character, controlled by the player, must go through a map consisting of 10 levels to reach the "finish line" and thus complete the game. The protagonist needs to talk to an NPC (character who only communicates with the player and does not participate, in any other way, in the action of the game) and help with the tasks that the NPC assigns to her. Along the way, she will face multiple challenges of varying difficulty that require focus and dexterity, but are also fun and challenging.

Making and using the application

The main menu

The main menu includes 3 buttons:

* New Game (to start the game)
* Instructions (to learn about the buttons used in the game and the functionality of each one)
* Quit Game (to exit the game)

Also, after completing a few levels, another button titled "Delete all saved progress" will appear on the main menu to give the player the option to delete their progress and start over. In addition, the "New Game" button will change to "Continue Game" (because once you start a new game, you can only continue it), and if you delete your progress, it will change back to "New Game".



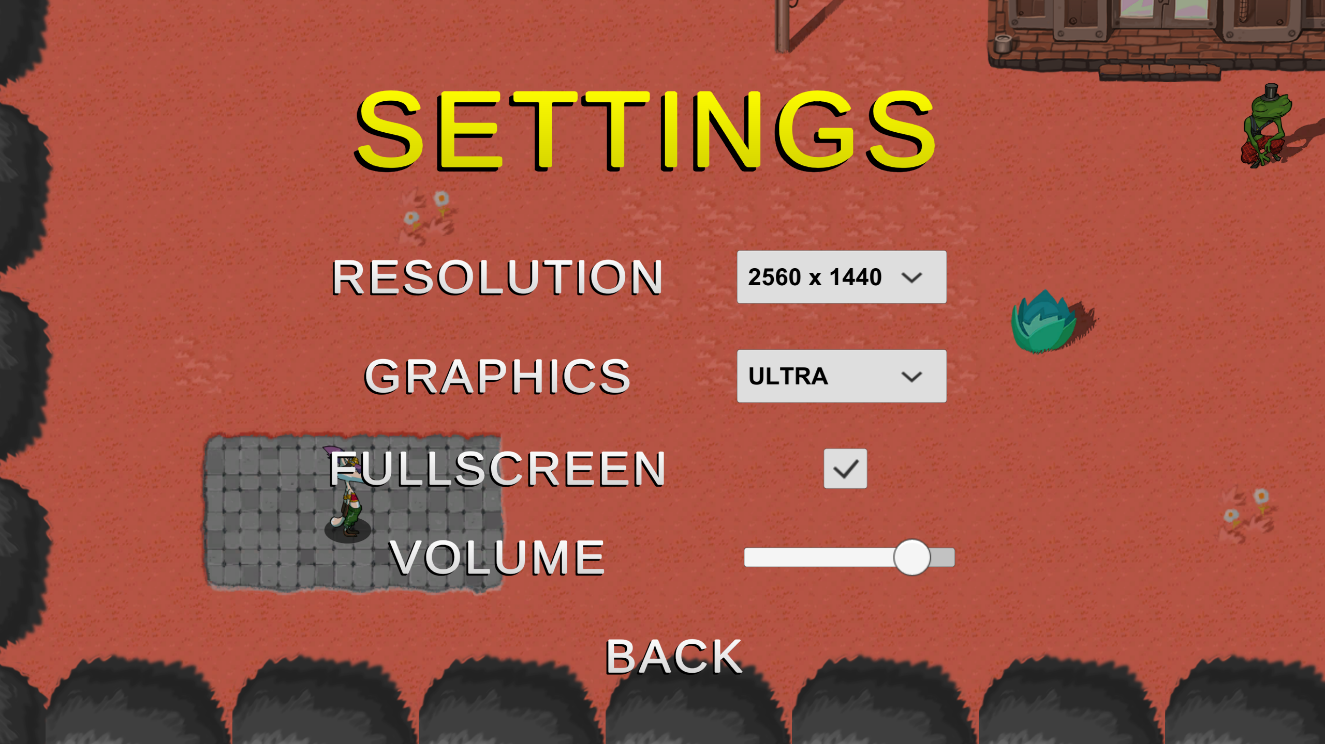


Instructions menu

The instructions menu contains details about the keys used to control the character, as well as other useful information regarding various aspects of the way to play the game.

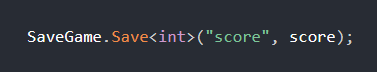
Settings menu

During the playthrough of the game, the player has the possibility to pause and adjust the game settings, including: resolution, quality level (of textures and details), volume of sounds and background music, but also if he wants the game to be displayed on the whole screen (fullscreen) or only as a window (windowed).



Progress save system

To facilitate the progressive aspect of the game, a save system is implemented. In this way, the player no longer has to worry about having to start the game from the beginning every time they start it, because at the start of each of the 10 levels (checkpoints), the player's progress up to that point will be saved (protagonist's position, health level and ammo amount, which enemies are defeated and which are not, etc.). This is done with the help of a special library that contains some useful functions to save, reload or delete the desired variables, via text files.

 **Library Functions:**

function to save the value of a variable to a text file

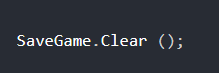


function to reload the value of a variable saved in a text file by assigning it to another variable

function to delete a saved text file



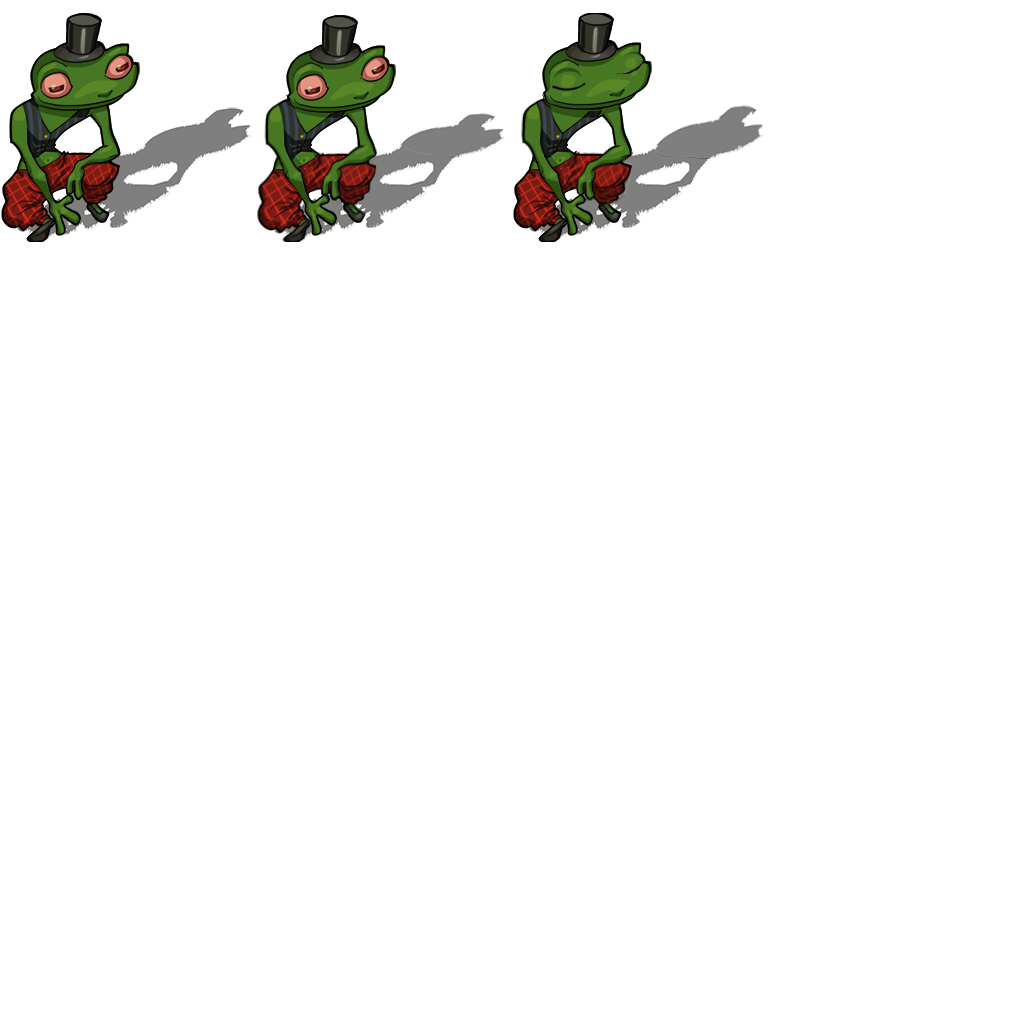
function to check for the existence of a text file



function to delete all created text files

How Ruby Works (Protagonist)

Ruby is the protagonist of the game, and thus is controlled directly by the player using the WASD keys to move her around the map, the SPACE key to make her throw wheels at enemies, and the F key to talk to Jambi the Frog (who helps her throughout the levels). It starts with 5 hit points and 0 cogs (ammo). However, she will receive 10 cogs immediately after starting the game, which are useful for learning the throwing mechanism.

Also, in the case of losing a health point, she can recover it by consuming a "strawberry".

Jambi the Frog (NPC)

Bag of 20 cogs

Strawberry that gives 1 HP

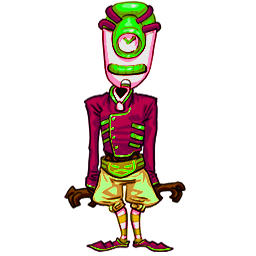


Strawberry that gives 2 HP

Bag of 10 cogs

Ruby (protagonist)

How normal enemies work

 Normal enemies operate based on simple movement routines (up-down, left-right, or forming a square). Normal enemies do not attack Ruby in any way, but if the protagonist touches them, she will lose a health point.

Fast enemy

Slow enemy

Otherwise, if they are hit once by a cog, they stop and no longer affect Ruby's health.

How BOSS enemies work

BOSS-type enemies operate based on more complex routines that involve both moving in multiple directions and attacking the protagonist. Like normal enemies, if the protagonist touches them, she will lose a health point. To defeat BOSS type enemies, it is necessary to hit them with several cogs depending on their health level (10, 15 and 20 respectively). Also, each of them has different attacks, so the player gets a new challenge when meeting each BOSS.

BOSS 2

BOSS 3 (Final)

BOSS 1

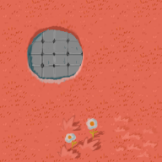
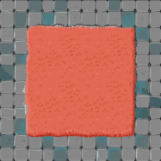
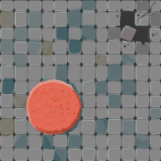
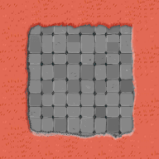
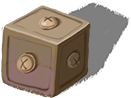
Static enemies

The game's static enemies are represented by spikes that sprout from the ground and will deduct points from the protagonist's health.

Spikes that subtract 2 HP

Spikes that subtract 1 HP

Game map

The game map is made using a "Tilemap". It allows the game space to be divided into squares that can be filled, arranged, modified or deleted at will. Also, the map includes elements corresponding to each level, such as: trees, houses, boxes, etc.

Environmental elements

The "chunks" of land used to create the map

The main elements of the interface

The interface (or UI) consists of 4 components corresponding to the 4 corners of the screen. The protagonist's life level appears in the upper-left corner, the protagonist's current number of wheels appears in the upper-right corner, the current level appears in the lower-left corner, and the health level of each BOSS (while fighting one of them) appears in the lower-right corner. Also, when the protagonist talks to Jambi the frog, at the beginning of each level, his words will appear in the middle-lower part of the screen.

Possible extensions of the application

* Introducing more levels
* Improved progress save system
* Developing the map on the vertical direction as well
* Adding more enemy types
* Extending the story
* Implementing difficulty levels (easy, medium, hard)
* Adding temporary abilities for the protagonist (increased speed, throwing multiple cogs at once, etc.)
* Improving the graphics quality

Bibliography

* Tutorials from [www.learn.unity.com](http://www.learn.unity.com)
* Various videos from [www.youtube.com](http://www.youtube.com)
* Useful tips from [www.stackoverflow.com](http://www.stackoverflow.com) (website dedicated to programmers for solving errors or other problems)
* Images and animations from [www.shutterstock.com](http://www.shutterstock.com)
* Other assets on [www.craftpix.net](http://www.craftpix.net), [www.sketchfab.com](http://www.sketchfab.com),[www.itch.io/game-assets](http://www.itch.io/game-assets), and from [www.assetstore.unity.com](http://www.assetstore.unity.com)
* Sound effects on [www.storyblock.com](http://www.storyblock.com), [www.pond5.com](http://www.pond5.com), and from [www.motionarray.com](http://www.motionarray.com)