**2022**

THE RED CAPYBARAS

THE GAMES –ADVENTURE PROJECT

24.04.2021

**Table of contents**

[Introduction 3](#__RefHeading__483_813356745)

[Entry And Exit Criteria 3](#__RefHeading__485_813356745)

[Entry Criteria 3](#__RefHeading__487_813356745)

[Exit Criteria 3](#__RefHeading__489_813356745)

[Objectives And Tasks 3](#__RefHeading__491_813356745)

[Objectives: 3](#__RefHeading__493_813356745)

[Tasks: 3](#__RefHeading__495_813356745)

[Scope 5](#__RefHeading__497_813356745)

[Test Deliverables 5](#__RefHeading__499_813356745)

[STAGES OF IMPLEMENTATION 5](#__RefHeading__501_813356745)

Introduction

The job of the QA engineer is very important, as he has to check if the code is correct. It’s not an easy job, as you always have to think of new ways to “break” the program so most of the problems in the code are corrected.

Entry And Exit Criteria

**Entry Criteria**

When the QA engineer starts his work, the program isn’t optimized and “cleared” from any problems it may have.

**Exit Criteria**

When the QA engineer finished his work, the program is more optimized and up-to-date with less problems.

Objectives And Tasks

**Objectives:**

The goal is that we end up with a complete and working product.

**Tasks:**

The tasks a QA engineer has to face are:

1. Preparation of a QA documentation
2. Testing the app
3. Reports any problems
4. Making a documentation about the corresponding problem

Scope

The program has to be fully tested so any problems can be reported and solved.

Test Deliverables

The result of a tested program is that there will be fewer malfunctions and it will be more accessible for use.

STAGES OF IMPLEMENTATION

COMPOSITION OF THE TEAM

For maximum efficiency in the work process, our team is composed of participants with different abilities. Everyone takes care of a different part of the project development:

• Scrum Master - takes care of the good mood and spirit in the team, organizes meetings, assists in case of difficulties in the assigned tasks and writes documentation

• Back-end Developer - deals with the logical part of the program, various functions and data processing

• Front-end Developer - his work is related to the program interface, the way the menu is displayed, the result after performing any of the functions, etc.

• QA Engineers - create QA documentation, test program functions, report faults and bugs

DRAWING UP A SCHEDULE FOR WORK

In order to facilitate the work process, we have drawn up a schedule with deadlines for completing each task we have set. We used the gitHub glider to mark completed tasks.

PROJECT WORK

After the steps mentioned above, it's time for the actual code work. Everyone took on their tasks and we had periodic meetings during which we discussed how the work process is going and the various problems we have encountered.

PRESENTATION

The last stage of project development is its protection.

DIFFICULTIES

During the development of the project we encountered problems in communication and connecting the Back-end with the Front-end.

CONCLUSION

Within three weeks, we tried to create a working application.

TABLE OF FUNCTIONS

|  |  |  |  |
| --- | --- | --- | --- |
| **0** | **Type** | **Arguments** | **Description** |
| **funcSet()** | void |  | Sets function |
| **setEnemy** | void |  | Sets the game enemy and its weapon |
| **next** | void |  | Moves on to the next move |
| **hp0** | int |  | Check the current HP of the player |
| **playerStats** | string | int hp, string wpn, int dmg, int row | Displays player stats |
| **enemyStats** | string | int hp, string name, int dmg, int row | Displays enemy stats |
| **preRowFunc** | string | int row, int len, bool side | Checks the previous funtion |
| **stats** | void | int playerHpMain, int playerDmgMain, string wpnName, int enemyHpMain, int enemyDmgMain, string enemyNameMain, int actMain, int turnMain, bool stateMain | stats info |
| **checkHp** | int |  | Checks Hp |
| **empty** | String |  |  |
| **spaceFunc** | string | String row |  |
| **textAttack** | string |  | Reads text and interprets it as an attack |
| **text** | void |  | text info |
| **andIf** | bool |  |  |
| **checkFunc** | int |  | Checks functions |
| **funcExe()** | int |  |  |
| **enemyAttack()** | int |  | Enemy attack info |
| **newHp** | int | string funcMain | Displays the HP after an action |
| **damage** | int |  | Damage info |
| **Choose** | void |  | Allows the player to make a choice |
| **wrongFunc** | void |  |  |
| **Beaten** | Bool | Bool b | Displays a message after defeating an opponent |
| **trappedDev** | void |  | Displays the game name on the beginning screen |
| **rustGear** | void |  | Displays the text “Rust Gear Solid: Vengance” |
| **type()** | void |  | Type info |
| **gameOver()** | void |  | Game over credits |
| **Credits()** | Void |  | Displays the credits |