Web development As2

Report

Haniyya Sajid Suleman

Student ID: 23851879

Table of Contents

[1 AS2 Functionality Checklist 2](#_Toc197288685)

[2 Testing 4](#_Toc197288686)

[2.1 How did you test that your code worked? 4](#_Toc197288687)

[2.2 Could you test certain aspects of the code without running the entire game and waiting for the correct condition to be met? 4](#_Toc197288688)

[2.3 What tests did you carry out, and what were the outcomes? 4](#_Toc197288689)

[Black Box Testing 4](#_Toc197288690)

[White-Box Testing 9](#_Toc197288691)

[2.4 What bugs did you discover during testing? 9](#_Toc197288692)

[3 Evaluation 10](#_Toc197288693)

[3.1 A list of known bugs/weaknesses in the game 10](#_Toc197288694)

[3.2 What works well? 10](#_Toc197288695)

[3.3 What improvements could be made? 10](#_Toc197288696)

[3.4 What else would you have done if you had more time? 10](#_Toc197288697)

[3.5 How easy would it be to extend the game to add more functionality? 10](#_Toc197288698)

[3.6 If you had to build a similar game in the future, what would you do differently and why? 10](#_Toc197288699)

[4 Video Demonstration URL 11](#_Toc197288700)

[5 References 11](#_Toc197288701)

# 1 AS2 Functionality Checklist

|  |  |
| --- | --- |
| D+ to D- | |
| The start Game Button is hidden when clicking. |  |
| * The player cannot move until the start button has been clicked. |  |
| Maze wall collision with the player |  |
| * Multiple points of collision for the maze walls (top left and top right for the up direction) |  |
| Point element collision with the player. |  |
| * Points are hidden from the maze. |  |
| The score p tag is updated for every point the player collects. |  |
| A game-over message appears after collecting all the points in the maze. |  |
| Once the game has ended, the player can no longer move. |  |
| The game is over when the player collides with an enemy character. |  |
| * Display the death animation upon enemy collision (dead CSS class) |  |
| C+ to C- | |
| Randomise the position of enemies at the start of the game. |  |
| Prevent the enemies from being created outside of the maze. |  |
| * Prevent enemies from being created where there are walls. |  |
| Enemies randomly move around in the maze. |  |
| * Enemy movement has wall collision (cannot move through walls) |  |
| * The enemy does not stop upon collision with the wall instead it moves in a new direction. |  |
| Enemies stop moving when the game-over state has been reached. |  |
| Reset button instead of game over (resets game state) |  |
| Implement the arrow buttons. The player will continue moving in that direction when an arrow button is clicked. |  |
| * The Arrow GUI button movement does not impact the arrow key movement. |  |
| B+ to B- | |
| At the end of the game, ask the player to enter their name. |  |
| * Save the name and score using local storage. |  |
| Display the scores of all the players on the leaderboard. |  |
| The leaderboard should be organised in order from the highest score to the lowest score. |  |
| Add the lives through JavaScript (not the HTML) at the start of the game. |  |
| Remove a life instead of the Game Over/restart button when the player collides with an enemy. |  |
| * Add the hit class and prevent the player from moving for 1.5 seconds while the animation plays. |  |
| Display the game over/restart button when all three lives are lost. |  |
| A+ to A- | |
| Once all the points are collected, reconfigure the maze and create a new layout for each level. |  |
| * Implement increasing difficulty. As the game goes on, it should get more challenging. |  |
| Create an infinite number of levels (not premade mazes but randomly created) |  |
| * Prevent an impossible-to-solve maze. |  |
| Add two unique features to the game. The better the feature, the more marks |  |
| * Feature One (replace with the feature) |  |
| * Feature Two (replace with the feature) |  |

# 2 Testing

## 2.1 How did you test that your code worked?

**(***make sure to put through validators)*

## 2.2 Could you test certain aspects of the code without running the entire game and waiting for the correct condition to be met?

## 2.3 What tests did you carry out, and what were the outcomes?

### Black Box Testing

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Requirement | Expected output | Actual output | Result | Notes |
| The start Game Button is hidden when clicking. |  |  |  |  |
| The player cannot move until the start button has been clicked. |  |  |  |  |
| Maze wall collision with the player |  |  |  |  |
| Multiple points of collision for the maze walls (top left and top right for the up direction) |  |  |  |  |
| Point element collision with the player. |  |  |  |  |
| Points are hidden from the maze. |  |  |  |  |
| The score p tag is updated for every point the player collects. |  |  |  |  |
| A game-over message appears after collecting all the points in the maze. |  |  |  |  |
| Once the game has ended, the player can no longer move. |  |  |  |  |
| The game is over when the player collides with an enemy character. |  |  |  |  |
| Display the death animation upon enemy collision (dead CSS class) |  |  |  |  |
| Randomise the position of enemies at the start of the game. |  |  |  |  |
| Prevent the enemies from being created outside of the maze. |  |  |  |  |
| Prevent enemies from being created where there are walls. |  |  |  |  |
| Enemies randomly move around in the maze. |  |  |  |  |
| Enemy movement has wall collision (cannot move through walls) |  |  |  |  |
| The enemy does not stop upon collision with the wall instead it moves in a new direction. |  |  |  |  |
| Enemies stop moving when the game-over state has been reached. |  |  |  |  |
| Reset button instead of game over (resets game state) |  |  |  |  |
| Implement the arrow buttons. The player will continue moving in that direction when an arrow button is clicked. |  |  |  |  |
| The Arrow GUI button movement does not impact the arrow key movement. |  |  |  |  |
| At the end of the game, ask the player to enter their name. |  |  |  |  |
| Save the name and score using local storage. |  |  |  |  |
| Display the scores of all the players on the leaderboard. |  |  |  |  |
| The leaderboard should be organised in order from the highest score to the lowest score. |  |  |  |  |
| Add the lives through JavaScript (not the HTML) at the start of the game. |  |  |  |  |
| Remove a life instead of the Game Over/restart button when the player collides with an enemy. |  |  |  |  |
| Add the hit class and prevent the player from moving for 1.5 seconds while the animation plays. |  |  |  |  |
| Display the game over/restart button when all three lives are lost. |  |  |  |  |
| Once all the points are collected, reconfigure the maze and create a new layout for each level. |  |  |  |  |
| Implement increasing difficulty. As the game goes on, it should get more challenging. |  |  |  |  |
| Create an infinite number of levels (not premade mazes but randomly created) |  |  |  |  |
| Prevent an impossible-to-solve maze. |  |  |  |  |
| Add two unique features to the game. The better the feature, the more marks |  |  |  |  |
| Feature One (replace with the feature) |  |  |  |  |
| Feature Two (replace with the feature) |  |  |  |  |

### White-Box Testing

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | Test | Result | Notes |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

## 2.4 What bugs did you discover during testing?

# 3 Evaluation

## 3.1 A list of known bugs/weaknesses in the game

## 3.2 What works well?

## 3.3 What improvements could be made?

## 3.4 What else would you have done if you had more time?

## 3.5 How easy would it be to extend the game to add more functionality?

## 3.6 If you had to build a similar game in the future, what would you do differently and why?

# 4 Video Demonstration URL

# 5 References