CS319 Object-Oriented Software Engineering
Section 2
Group: 2.E
Analysis Report 1

Maze Runner

Group members: **Ali Sabbagh, Mehmet Furkan Dogan,**
Umitcan Hasiboglu, Xheni Caka

Course Instructor: **Ugur Dogrusoz**

Date: 25 Feb 2017

Contents

1- Introduction

The project group decided to implement an arcade game because arcade games have an advantage which is the models can be drawn easier than the other programs. Thus, we decided to create a game which is a combination of "Pac-man" and "Bomberman". We decided to implement a stealth game "Maze Runner" that combines these two games and offers many new features. It will give you the present enjoyment of gaming plus the lovely nostalgia to the old days. The game can be played as a single player or multiplayer and the game purpose is to reach the end as soon as possible.

The goal of our game is to solve a sophisticated maze as fast as possible with the least possible attraction of soldiers. Both single and multiplayer modes can plant a bomb and try to keep the patrols out of your way or damage your enemy (in multiplayer version). When the player reaches the end, he will receive bonus points with respect to remaining time.

Overview section includes more details about the content of the project. The report contains overview, functional requirements, nonfunctional requirements and use case model chapters. The following chapter is overview that gives more information about the game.

2- Proposed System

2.1   Overview

The game consists of a Pac-man like moving object which goal is to finish the maze as soon as possible without being killed by the patrols which are continuously moving around and trying to stop the player from reaching the exit of the maze. The player has abilities that help him escape from the patrols when he is very close to them. The game has a time limit and also there are time related features of the bombs. The game can be paused and continued where it was left afterwards. A high score system will be accessible allowing the player to check his last five scores. In both single and multiplayer versions, the game will be played by using the keyboard. Player can access the help menu to get instructions on how to play the game.

2.1.1 Play Mode

- Single player mode: In this mode the player's goal is to reach to the exit as soon as possible. There will be the patrols trying to stop him and the challenge is finishing the maze before the time ends. The player can get rid of the patrols by throwing bombs to them but needs to escape in a short amount of time from the explosion range

- Multiplayer mode: In this, the challenge is finishing before your opponent. The game is won from the player who made it first to the exit or, if the time limit is up and none of the players makes it to the finish line, then the winner is the one who has collected more points.

2.1.2 Bomb

- Area:
  Bomb in own area: This kind of bomb is dropped in the current position of the player and has a small explosion area and the target is the patrols. It has an explosion time of 2 seconds which is enough for the player to escape from the explosion range.

  Bomb in opponent's area: This kind of bomb differs from the other type because it is used to kill your opponent and resets the opponent's position to the beginning position unless the opponent makes it away untouched.

- Bomb usage ability: Planting a bomb on his maze costs specific amount of points (100) but planting on the other side costs double (200). Each player has 200 points when "MazeRunner" starts and the score of the player increases as the time runs out such as gets 100 points in every 15 seconds. He can spend these points according to his position in the maze. For instance, in the multi-player mode, if he is ahead of the other player, uses bombs to pass the patrols or if he is not ahead, uses bombs to bring back his enemy to initial point unless the enemy gets away of the explosion rate on time.

2.1.3 Time

Limit:There will be a time limit to finish the game and it will vary from level to level. In the first level the time limit is 1 minute 30 seconds. In the second level the time limit is 2 minutes 30 seconds and in the third level the time limit is 3 minutes 30 seconds.

Bonus Score: When the game is done the remaining time in seconds will be multiplied by 10 and it will be added to the score.

2.1.4 Patrols

- Blue: These patrols just move in one direction and they do not follow. They kill only if the player hits them.

- Yellow: These patrols move back and forth and they have the ability to follow the player when he is in a certain range. The player then has to escape or kill them in order to get rid of them.

2.2   Functional Requirements

- Player(s) should be able to move freely except their direction is blocked by obstacles.

- Players(s) should be able to plant bombs in their area as well as in other player's area.

- Players(s) should die if a bomb explodes near them or any of computer controlled enemies touch them.

- Game should end if any of the players dies or reaches the end point.

- If game ends, player(s) should be able to restart the game.

- Game should be customizable such as number of bots, number of bricks that can be exploded by bombs or the speed of the bots or the resolution.

- Bombs that planted in other player's side should be placed in where the other player is.

2.3   Non-Functional Requirements

- Bombs should explode after 2 seconds + randomly determined time up to 1 second.

- Player(s) movement should not be blocked by bombs.

- By default, player(s) speed should be 0.66 times of enemies.

- Approximately, games FPS should be reasonable such as 100 FPS.

- Player(s) should be able to plant bomb in their sides after 5 seconds they plant a bomb on their side and plant a bomb after 10 seconds they plant a bomb on other player's side.

2.4   System Models

2.4.1 Use Case Model

Use case name:	SelectDifficulty
Contributors:	Player
Entry condition:	Game launched and main menu is entered
Flow of events:	Player presses the select difficulty button, three buttons appear, (Easy-Regular-Hard), Player selects one, and automatically goes back to main menu.
Exit condition:	Player is in main menu, with difficulty selected.
Exceptions:	No exceptions.

|	
Use case name:	CheckLeaderboard
Contributors:	Player
Entry condition:	Game launched and main menu is entered
Flow of events:	Player presses "Leaderboard" button, enters to the table saving names and scores of the top 5 players.
Exit condition:	Player is back to main menu.
Exceptions:	In case game is newly installed table will be empty. "Take the chance and be on top of the leaderboard!" message will appear.

Use case name:	WatchTutorials
Contributors:	Player
Entry condition:	Game launched and main menu is entered
Flow of events:	Player will enter a screen showing a picture explaining one of the game rules, a next arrow will take the player to next picture, a back one will go to the previous picture, pressing "Done" will take the player to main menu.
Exit condition:	Player is back to main menu.
Exceptions:	No exceptions.

Use case name:	PauseGame
Contributors:	Player(s)
Entry condition:	Player(s) should be already playing.
Flow of events:	Pressing the pause button, will display the Pause dialogue, game will stop, and then resume when the button is pressed again.
Exit condition:	Game will resume.
Exceptions:	Pause is not available after a player reaches the end of the maze

Use case name:	SelectPlayingMode
Contributors:	Player
Entry condition:	Game launched, "Play" button from main menu pressed
Flow of events:	Once player presses "Play" a screen asking for game mode selection appears, with two options: Single player, Multiplayer, pressing one will start the game accordingly.
Exit condition:	Player(s) in the game. Maze(s) are shown, game will start in 3 seconds.
Exceptions:	No exceptions

Use case name:	JoinLeaderboard
Contributors:	Winning player
Entry condition:	Player reached the exit, game finished, total score is calculated.
Flow of events:	Player is shown the leaderboard, with ability to write his name, to save it if he/she is among the top 5.
Exit condition:	Player(s) back to main menu
Exceptions:	No exceptions

Use Case Diagram
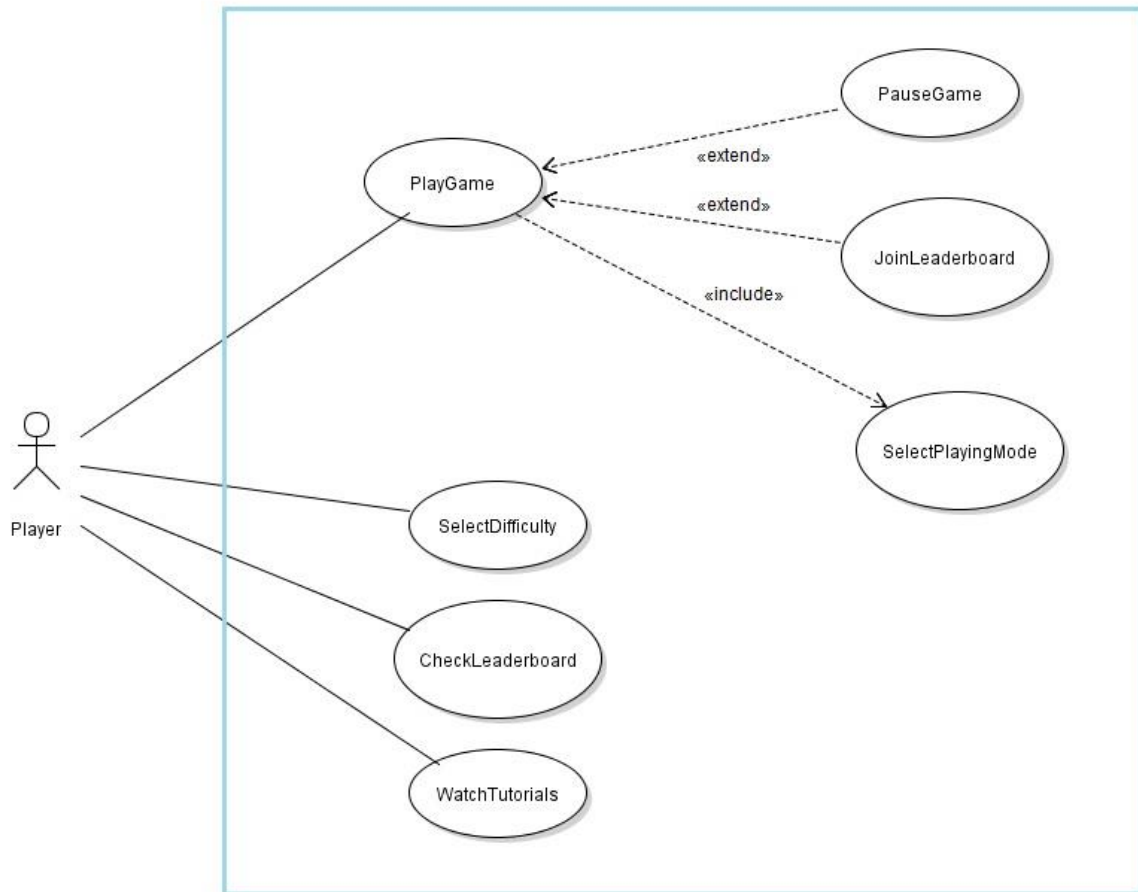
2.4.2 Object and Class Model

2.4.3 Dynamic Models

2.4.4 User Interface

3- Glossary

4- **References**