Bilkent University

Department of Computer Engineering

CS319 PROJECT – GROUP #2

Analysis Report

Bombalamasyon

Oğuz Demir – 21201712

Anıl Sert – 21201526

Kaya Yıldırım – 21002071

Kaan Kale - 21000912

Progress Report

Feb 20, 2016

This report is submitted to the Department of Computer Engineering of Bilkent University in partial fulfillment of the Analysis Report of CS319 Object-Oriented Software Engineering course project.

# Introduction

We decided to create a game like the classic arcade game called Bomberman and our games’ name is Bombalamasyon. It is a game that basically player/s try to reach the other players by exploding the walls and try to kill other bombers to be the last man standing.

This report contains an overview of the game, basic gameplay and rules of the game. Then it describes the requirements of the project which are functional, non-functional and pseudo requirements. Lastly, system models of the game will be described and Glossary&references will be given.

# Overview

Bombalamasyon is a strategic, maze-based game like the other alternatives such as Bomberman and Fireman. It is an easy and fun game to play but it is hard to master it. Player/s play the game in a labyrinth shaped map which consists of different kinds of walls and paths. The player/s objective is the reach and kill the other bombers and be the last man standing in a competitive killing race in a limited time.

## Single Player Game

The single player mod of the game is a level based bomberman game which player needs to complete 5 levels to finish the game. Every level is different than each other and levels are become harder as player continues. At each level, the player’s objective is the same as the whole game’s which is kill the other bombers in a limited time. If the player is killed or time elapses, the user loses that level and move on the next one if there is any. At the end of the single player game if the player takes greater score than computer opponents and earn better point than the lowest of the high score list, s/he deserves to enter her/his name to the high score list.

## Multiplayer Game

The multiplayer mod of the game is a race between two players and the computer opponents. There is three different possible results in a multiplayer game’s levels. The first one is both players are dead and this will end the level, second is one of the players be the last man standing in the maze and wins that level, the last one is time elapses and highest scored bomber becomes the winner of the game. If one of the players wins the game and earn better point than the lowest of the high score list, s/he deserves to enter her/his name to the high score list.

## Gameplay

The player/s play the game from the keyboard. There is specific keys for each action in the game like moving the bomber, drop the bomb, or pause the game and see the pause menu. If users play a multiplayer game because they will use the same keyboard to prevent hand conflicts we assign these action buttons separate as much as we can. To interact with the menus of the game like main menu and pause menu, player/s need to use the mouse buttons to choose their selections.

## Bomberman and Bombs

The main character of our game is the bombermen and to differentiate each bomberman from the others we are going to use colors. Every bomberman will have a different uniform color and their bombs color will also be same with their uniforms. The player/s control the bombermen and interact with game with these characters. They can run the bomberman in four different directions and can place a bomb to their current position. Bombs are the weapons of the bombermen and have explosion magnitudes. At the start of every game bombermen can put only one bomb at the time. However, the explosion magnitude and the bomb quantity can be increased with the powerups. Bombs can break some kinds of walls and kill the opponents, so they are essential tool for every bomberman.

## Walls

Since players will play the game in a maze, they are surrounded with the walls. However, players can use their bombs to break the walls and approach their opponents in different ways. Breaking a wall may also drop some powerups randomly that players can use to reach their goal faster and easier. However, not every wall is breakable. There are 3 types of the walls:

***Brick Walls:*** They are the classic wall that easily can be exploded by the bombs.

***Strong Brick Walls:*** They are stronger walls than the bricks walls and players need to explode it two times to fully break the wall. In the first explosion, it transforms to normal brick walls.

***Steel Wall:*** They are the strongest walls of the game and cannot exploded by the bombs which means they are permanent in the game area.

## Powerups

The powerups are the doping of the bombermen and they increase the abilities of bombermen. Powerups can be dropped by the breaking walls and players can pick them by walking over them. Powerups make the game more funny and unpredictable because it becomes hard to guess the explosion size of a bomber’s bomb or guess how many bomb they can drop. There are 4 kinds of powerups:

***Increase Bomb Quantity:*** If player takes this powerup, the number of consecutive bomb drops by the player increased by 1, but it can be maximum 5.

***Increase Bomb Magnitude:*** If player takes this powerup, the bomber’s bombs explosion magnitude will increase by 1 square.

***Increase Bomberman Speed:*** If player takes this powerup, their characters walk speed will increase 1 unit.

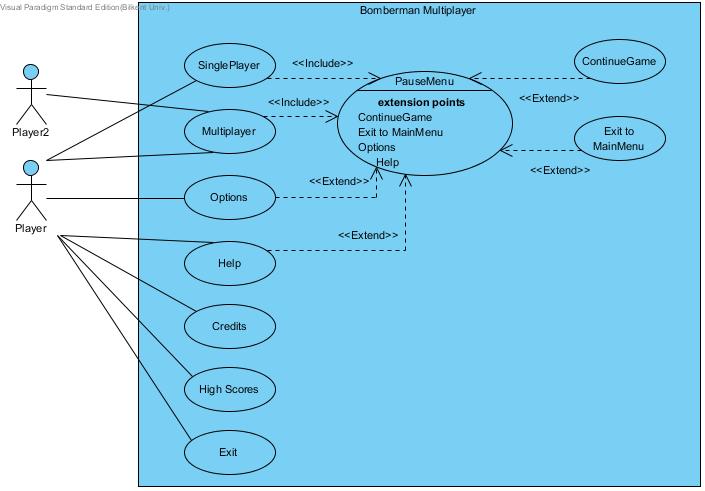
***Shield:*** If player takes this powerup, they are protected for the next damaging explosion.

# Functional requirements

# Nonfunctional requirements

# System models

## Use case model

****

## Use cases

**Use case #1:**

**Use case name:** SinglePlayer

**Participating actors:** Player

**Interests:** Player aims to be the last man standing with a high score.

**Entry condition:** Player had executed the game and he is on the main menu, presses the “Single Player” button.

**Exit condition:** Player presses the “Pause Game” button or game is over.

**Pre-condition:** The default or last saved settings is applied.

**Post-condition:** The score of the player is added to high score chart if he wins and makes enough score.

**Main Flow of Events:**

1. Player starts the game and clicks “Single Player” button.

2. System creates the game area for the first level.

3. Player starts playing corresponding level.

4. The game continues until only one player remains. If the player is died, the level directly ends without waiting computer opponents.

5. At the end of level, next level is loaded.

*Steps 2-5 is repeated until all levels are completed or player exits.*

6. At the end, the System brings the game over screen front and if the player has a high score around previous high scores, its score is written on high scores chard with the Player’s name.

7. The player is brought back to main menu screen.

**Alternative Flows:**

1. More than one players are still alive when time limit is reached.

A.1. Bomber with highest score so far is selected to be winner.

A.2. System brings the game over screen front and if the player is winner same procedure is applied.

B. User clicks the “Pause Game” button.

B.1. Pause menu is brought front.

B.2. Case 3 is applied.

**Use case #2:**

**Use case name:** Multiplayer

**Participating actors:** Player1 & Player2

**Interests:** Players aims to be the last man standing by defeating both other player and computer opponents with a high score.

**Entry condition:** Player had executed the game and he is on the main menu, presses the “Multiplayer” button.

**Exit condition:** Player presses the “Pause Game” button or game is over.

**Pre-condition:** The default or last saved settings is applied.

**Post-condition:** The score of the winner player is added to high score chart if he makes enough score.

**Main Flow of Events:**

1. Player starts the game and clicks “Multiplayer” button.

2. System creates the game area for the first level.

3. Players start playing corresponding level.

4. The game continues until only one player remains. If both 2 players are died, the game directly ends without waiting computer opponents.

5. At the end of level, next level is loaded.

*Steps 2-5 is repeated until all levels are completed or player exits.*

6. At the end, the System brings the game over screen front and if one of the 2 players has a high score around previous high scores, his score is written on high scores chard with that Player’s name.

7. System displays the main menu screen.

**Alternative Flows:**

1. More than one players are still alive when time limit is reached.

A.1. Bomber with highest score so far is selected to be winner.

A.2. System brings the game over screen front and if winner is one of the players, same procedure is applied.

B. Users click the “Pause Game” button.

B.1. Pause menu is brought front, is controlled by only one player.

B.2. Case 3 is applied.

**Use case #3:**

**Use case name:** PauseMenu

**Participating actors:** Player

**Interests:** Player aims to pause the game

**Entry condition:** A single or multiplayer game is being played and user clicks the “Pause Game” button.

**Exit condition:** Player presses the “Continue Game”, “Options”, “Help” or “Exit to Main Menu” buttons.

**Pre-condition:** Current game time and scores are shown in screen.

**Post-condition:** -

**Main Flow of Events:**

1. User clicks “Pause Game” button from game screen.

2. System freeze the game timer and displays the game timer and current game scores.

3. If user clicks “Continue Game” button, previous game screen is brought back and continues.

4. If user clicks “Options” button, case 4 is applied.

5. If user clicks “Help” button, case 5 is applied.

6. If user clicks “Exit to Main Menu” button, the Main Menu is displayed.

**Use case #4:**

**Use case name:** Options

**Participating actors:** Player

**Interests:** Player aims to change the game configurations.

**Entry condition:** Player is in main menu or pause menu and clicks the “Options” button.

**Exit condition:** Player presses the “Save & Exit” or “Exit without Saving” buttons.

**Pre-condition:** The default or last saved settings is shown in configurations.

**Post-condition:** User’s changes is saved.

**Main Flow of Events:**

1. User clicks “Options” button from main menu or pause screen.

2. System displays the Options screen where the adjustable settings with current configurations are shown.

3. User configures the game speed and volume settings of sounds and music according to his desire.

4. User clicks “Save & Exit” button and System brings user to Pause Menu or Main Menu according to the state before the Options.

**Alternative Flows:**

1. User goes back without any change.

A.1. User clicks “Exit without Saving” button and System brings user to Pause Menu or Main Menu according to the state before the Options.

**Use case #5:**

**Use case name:** Help

**Participating actors:** Player

**Interests:** Player wants to be informed about game rules and playing.

**Entry condition:** Player is in main menu or pause menu and clicks the “Help” button.

**Exit condition:** Player presses the “Back” button.

**Pre-condition:** -

**Post-condition:** -

**Main Flow of Events:**

1. User clicks “Help” button from main menu or pause screen.

2. System displays the Help screen where user can find game information and instructions.

3. User clicks “Back” button and System brings user to Pause Menu or Main Menu according to the state before the Help.

**Use case #6:**

**Use case name:** Credits

**Participating actors:** Player

**Interests:** Player wants to be informed about the developers of the game.

**Entry condition:** Player is in main menu and clicks the “Credits” button.

**Exit condition:** Player presses the “Back” button.

**Pre-condition:** -

**Post-condition:** -

**Main Flow of Events:**

1. User clicks “Credits” button from main menu.

2. System displays the Credits screen where user can find information about the developers.

3. User clicks “Back” button and System brings user to Main Menu.

**Use case #7:**

**Use case name:** High Scores

**Participating actors:** Player

**Interests:** Player wants to see game’s high scores records.

**Entry condition:** Player is in main menu and clicks the “High Scores” button.

**Exit condition:** Player presses the “Back” button.

**Pre-condition:** -

**Post-condition:** -

**Main Flow of Events:**

1. User clicks “High Scores” button from main menu.

2. System displays the sorted high scores chart.

3. User clicks “Back” button and System brings user to Main Menu.