Software Requirements Specification

Project BomberMan

Brett Leighton
Leotard Niyonkuru
Marc-André Cataford
Martin Zhang
Mete Kemertas

9th February, 2014

Table of Contents

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the Bomberman game. It will explain the interface and features of the game as well as the constraints under which the game will operate. These requirements will be outlined through text descriptions, visual diagrams, and tables.

The intended audience is the customer looking to purchase the BomberMan game. The customer will have access to software developers, quality assurance personnel, as well as regular consumers to test the game.

1.2 Scope

The software product to be produced Bomberman style video game system named Project BomberMan. The main application of this system will be to relieve stress and procure enjoyment to its users. The system will appeal to users of all ages, and be simultaneously familiar to users with prior experience in Bomberman style games as well as welcoming to players with no experience and/or computer skills.

1.3 Definitions

System: The gaming system discussed in this document, *Project BomberMan*.

Bomberman: An arcade-style video game produced in 1983 by Hudson Soft.

Power-up: A temporary attribute that enhances the user's capabilities within the game.

Bomb: The user's tool to destroy enemies by burning up a region of two perpendicular lines with the user at the intersection. The user is allowed to drop only one bomb at a time unless they have picked up a relevant power-up. A dropped bomb detonates after a short period. The user will fail the level if they come into contact with the bomb blast.

Level: The game consists of a number of levels, each of which has a different map setup. The setup varies in the starting positions of characters, number of enemies, and location of obstacles. The user progresses through levels one by one as they clear the enemies from the map.

Character: In-game representation of the player. Each player controls one character that is differentiable from other characters on the map.

Obstacle: Objects in the game field that none of the characters or enemies are allowed to go through or over. Depending on their attributes, they may be resistant to bombs.

Enemy: Enemies are obstacles that have the ability to move and that are controlled by software artificial intelligence. A user will not be allowed to complete a level until all enemies have been eliminated through use of bombs. The user will fail the level if they come into contact with an enemy.

Multiplayer game: Playing mode that allow multiple players to play simultaneously using a single keyboard or by alternating turns on a keyboard. The former allows for a free-for-all game and the latter, a contest for the highest score.

Main Menu: The first screen to appear once the executable file is opened. It presents the user the following options to choose from: New Game, Load Game, Multiplayer Game, Instructions, High Scores, and Exit Game.

Escape Menu: This menu appears if the user presses "Esc" key while in the game. The game is paused once the Escape Menu appears. It presents the following options: Resume Game, Save Game, View Instructions, Quit Game, and Exit Game.

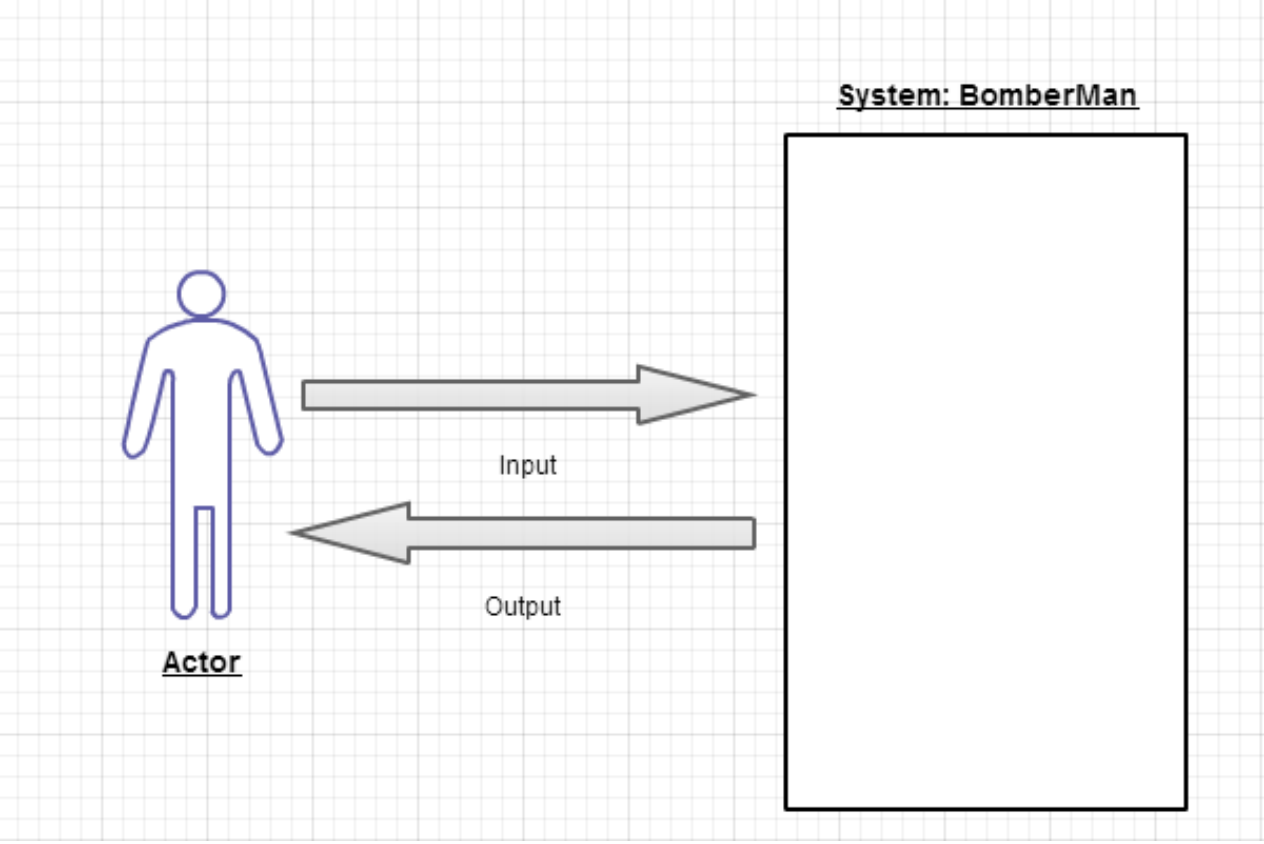
1.4 Document Overview

This document contains a product perspective as a diagram, as well as major use cases divided into tables. User characteristics are discussed, followed by a list of constraints, assumptions and dependencies. The document concludes with a detailed list of both functional and nonfunctional requirements.

2. Overall Description

2.1 Product Perspective

Project BomberMan will have one active actor and one cooperating system, which are independent.



2.2 Product Functions/ Use Cases

Use Case Name	New Game (1p)
Participating Actors	Player, System
Entry Conditions	Player selects “New Game (1p)” from Main Menu
Flow of Events	System asks the user to select a difficulty level
	User moves arrow near a difficulty level (“Easy”, “Medium”, “Hard”)
	User presses "Enter"
Exit Conditions	System initializes a new game with selected difficulty
Quality Requirements	System starts a new game at Level 1
	The difficulty levels are appropriately varied relative to each other
Exceptions	The user closes the game without choosing a difficulty level
	The user chooses to go back to the Main Menu

Use Case Name	New Game (2p)
Participating Actors	Player, System
Entry Conditions	Player selects “New Game (2p)” from Main Menu
Flow of Events	System asks the user whether they would like to play simultaneously or in alternating turns
	User selects either one of “Alternating Turns” and “Simultaneous Game”
	System asks the user to select a difficulty level
	User moves arrow near a difficulty level (“Easy”, “Medium”, “Hard”)
	User presses "Enter"
Exit Conditions	System initializes a new multiplayer game with the selected difficulty and mode

Quality Requirements	System starts a new game at Level 1
	The difficulty levels are appropriately varied relative to each other
Exceptions	The user closes the game without choosing a difficulty level
	The user chooses to go back to the Main Menu

Use Case Name	Load Saved Game
Participating Actors	Player, System
Entry Conditions	Player selects “Load Game” from Main Menu
Flow of Events	System displays a list of saved games
	User moves arrow to the game they wish to load
	User presses "Enter"
	System retrieves the level and difficulty from selected file
Exit Conditions	System initializes game with information retrieved from the selected file
Quality Requirements	System remembers the difficulty level of the game to be loaded
	System remembers the level user stopped at accurately
Exceptions	The user closes the game without choosing a file
	The user chooses to go back to the main menu

Use Case Name	View High Scores
Participating Actors	Player, System
Entry Conditions	User selects “High Scores” from Main Menu
Flow of Events	System retrieves highest scores for each level
Exit Conditions	System displays a table with high scores

Quality Requirements	System displays an easily readable table with appropriately sized font
Exceptions	The user has a number of records that are less than 5. System fills those with a "-" next to the index

Use Case Name	View Instructions
Participating Actors	Player, System
Entry Conditions	Player selects “Instructions” from Main Menu or Escape Menu
Flow of Events	System finds the image file of Instructions
Exit Conditions	System displays the Instructions image to the screen
Quality Requirements	The image is legible
	Instructions themselves are clear and easy to understand
Exceptions	N/A

Use Case Name	Save Game
Participating Actors	Player, System
Entry Conditions	Player selects “Save Game” from Escape Menu
	System prompts user for file name
	User enters file name
	System retrieves the current level and difficulty
	System asks user for filename
	User types a filename and selects “Enter”
Exit Conditions	System displays “Game Saved Successfully”
Quality Requirements	System accurately stores the level and difficulty
Exceptions	The user is on Level 1, there is nothing to save. System displays “Nothing to save!”

Use Case Name	Exit Game from Escape Menu
Participating Actors	Player, System
Entry Conditions	Player selects “Exit Game” from Escape Menu
	System asks user if they want to save their game
	User selects “No”
	System asks user if they are sure they want to exit
	User selects “Yes”
Exit Conditions	System exits game
Quality Requirements	System exits in less than 3 seconds once the user states they are sure they want to exit
Exceptions	N/A

Use Case Name	Exit Game from Main Menu
Participating Actors	Player, System
Entry Conditions	Player selects “Exit Game” from Main Menu
	System asks user if they are sure they want to exit
	User selects “Yes”
Exit Conditions	System exits game
Quality Requirements	System exits in less than 3 seconds once the user states they are sure they want to exit
Exceptions	N/A

Use Case Name	Quit Game
Participating Actors	Player, System
Entry Conditions	Player selects “Quit Game” from Escape Menu
	System asks user if they want to save their game

	User selects “No”
	System asks user if they are sure they want to quit
	User selects “Yes”
Exit Conditions	System goes back to Main Menu
Quality Requirements	System quits and displays Main Menu in less than 3 seconds once the user states they are sure they want to quit
Exceptions	N/A

Use Case Name	Pause Game
Participating Actors	Player, System
Entry Conditions	Player presses “Esc” button while playing
Flow of Events	System pauses execution of code, keeping objects in position
Exit Conditions	System displays Escape Menu
Quality Requirements	The program keeps the game “Paused” as long as the Escape Menu is open
Exceptions	N/A

Use Case Name	Resume Game
Participating Actors	Player, System
Entry Conditions	Player selects “Resume Game” from Escape Menu
Flow of Events	Escape menu closed, code execution resumed
Exit Conditions	Game is initialized according to the information
Quality Requirements	The game should continue exactly as it was paused
	The game should “resume” within 2 seconds after user hits “Enter” key for “Resume Game” option
Exceptions	N/A

Use Case Name	Move Character
Participating Actors	Player, System
Entry Conditions	Player is in the game
Flow of Events	Player presses assigned keyboard key to move in one of the 4 directions: +x, -x, +y, -y
	Systems makes the character turn to the given direction
	System checks if there is an obstacle on the way
Exit Conditions	Condition 1: There is an obstacle on the way. Character does not move Condition 2: The way is clear. Character moves by 1 unit length towards the given direction
Quality Requirements	The character's change of position should take place simultaneously as the player presses the key
Exceptions	Character contacts an enemy upon movement. Game terminates, system displays "You lose." message.

Use Case Name	Drop Bomb
Participating Actors	Player, System
Entry Conditions	Player is in the game and the character has either no bombs waiting to detonate or a power-up allowing to drop at least one more bomb
Flow of Events	The user presses the "Space" key
	System checks if the user's character is allowed to drop a bomb
Exit Conditions	A bomb appears at the location where character was when the user pressed "Space" key
Quality Requirements	The bomb should appear simultaneously as the user presses the key
Exceptions	The user is not allowed to drop a bomb: Nothing happens and no bomb is dropped.

2.3 User Characteristics

Those playing the game will be of all ages, and various technological competence and familiarity with the Bomberman game itself. They will have knowledge of basic computer hardware (keyboard, mouse). They will not have any physical impairment that will stop them from playing the game.

Maintenance and systems personnel will have advanced knowledge and experience with both Java and the Swing GUI.

2.4 General Constraints

- The game will not contain graphical violence that could be harmful.
- The product will have to be written in Java.
- The product will not run on machines without Java Virtual Machine.
- The code will make use of standard classes that are in the Java SDK.
- The game must not cause excessive visual stress to the point of causing epileptic seizures.
- The code must all be written from scratch.

- A required free space of 20 megabytes will be necessary on the hard disk for the application to be stored.

2.5 Assumptions and Dependencies

- It is assumed that the average user's system is not antiquated (i.e. the operating system is still maintained by its publisher and can operate on recent hardware).

- It is assumed that the user's computer is running a recent version of the Java Virtual Machine (JVM).

3. Specific Requirements

3.1 Functional requirements

- Essential

    - A1. The game shall display on a visual display window of 800*600 pixels. (Easy)
    - A2. The game shall have a main menu that allows navigation to the game screen, the options screen, or the statistics screen.

- A3. The game shall feature a player-controlled character that can drop bombs (Medium)
- A4. The game shall feature AI-controlled enemies that can kill the player-controlled character (Medium)
- A5. The game shall feature power-ups (Medium)
- A6. The game shall feature levels (Easy)
- A7. There shall be a graphical representations of the character and enemies, as well as power-ups, and other features of the game (Easy)

- Desirable

  - A8. The game should feature multi-player mode (Easy)
  - A9. The player should be able to select the game difficulty level (easy/medium/hard)
  - A10. The game graphics should be displayed in color. (Easy)
  - A11. The user should be able to save his/her game progress. (Medium)
  - A12. The user should be able to load saved games. (Medium)
  - A13. The game should feature a high score table (Medium)

- Optional

  - A14. The game should feature a multi-player mode features two players playing at the same time (Hard)
  - A15. The game should feature a full-screen and windowed option (easy)
  - A16. User can either use keyboard or mouse to select an option on each menu (medium)
  - A17. The game should feature a pause button during gameplay

3.2 Quality Requirements

- Essential

  - B1. The game shall have an instructions menu accessible through a button on the main screen.
  - B2. The instructions shall be of a resolution that is at least equivalent to a 10-point typeface.
  - B3. The game shall not use more than 50% of CPU resources for any given time.
  - B4. There should be a delay of no longer than .05 seconds between user command and execution of the command
  - B5. 75 out of 100 average players can fully complete the game
  - B6. 75 out of 100 average players can finish each level within 2 minutes
  - B7. The game shall run smoothly from startup 99% of the time

- B8. The game shall save the level completed, the difficulty selected, the score at the start of the last level started, and the powerups the player has accumulated

- Desirable
  - B9. After the user finishes a level, the next level should load in less than 5 seconds
  - B10. The game shall run at 60 frames per second on a basic 32 mb graphics chip
  - B11. The game can initialize within 2 seconds of launching the application
- Optional
  - B12. In the future, the bomberman game will be able to run in 1080p resolution
  - B13. In the future, the bomberman game can be ported to a mobile platform
  - B14. The user can enter a username to identify his save file and highscores to the game

3.3 Design constraints

- C1. The game shall be able to run on any Windows, OSX or Linux-based system
- C2. The user shall use a standard computer keyboard and mouse to interact with the game software.
- C3. The game shall output sounds corresponding to in-game events through the user's computer sound peripheral.
- C4. The game shall output graphics depicting in-game events through the user's computer screen.
- C5. The average user shall need basic computer operating skills to play the game.

3.4 Other requirements

- D1. The game requires the user to have functional eyes and limbs to use the computer and see the screen
- D2. The computer which the software is run on should have JVM installed
- D3. The computer should have 2 gigabytes of random access memory (RAM)
- D4. The computer should have at least a 32 megabyte graphics card
- D5. The computer should have a functional sound card