Requirements and Analysis Document for Bomberman

Table of Contents

Version: 2.0
Date: 2012-05-08

Author: Andreas Rolén, Adrian Bjugård, Joakim Persson, Viktor Anderling

This version overrides all previous versions.

Table of content

1. Introduction

1.1 Purpose of application

The goal is to build a game application, similar to Bomberman. The main goal is to create the game just the way the real game looks like and then if there's any time over add extra features like saving your profile, play over network and more boosters for you character in game.

1.2 General characteristics of application

The game will be a desktop application with same-keyboard multiplayer support. The idea is to start with two players and then extend it to four, either on the same keyboard or with external Xbox controls. When a new game starts, every player is placed in each corner of the gamefield. Then the mission is to place out bombs to blast boxes and pillars to reach the other players, and then blast the players. In some of the boxes that you blast there is placed power-ups which can give you higher speed, more bombs, more powerful bombs etc. The pillars require more bomb power to destroy then the boxes.

1.3 Scope of application

The game will only support multiplayer, either at the same computer or by network, and not playing against a player controlled by the computer. The idea is that a game consists of a several rounds. Every round consists of three matches and between every match the players will be able to buy upgrades for the points they earned.

1.4 Objectives and success criteria of the project

The objective is to be able to play a game with more then one round with two players in any GUI based platform.

1.5 Definitions, acronyms and abbreviations

All definitions and terms regarding the core Monopoly game are as dened in the references section.

- GUI, graphical user interface.
- Java, platform independent programming language.
- JRE, the Java Run time Environment. Additional software needed to run an Java application
- Host, a computer where the game will run.
- Round, one complete game ending in a winner.
- Match, several complete rounds ending in a winner.
- Score, based on several different factors.

2. Requirements

2.1 Functional requirements

The players should be able to:

1. Select how many players for the game. Two, three or four.
2. Select if the music and sound effects are turned on or off.
3. Start a new match.
    a. Move the character.
    b. Place a bomb (Which will explode).
    c. Pick up an item.
    d. Get an “item-boost” (Speed, bombrange, numbers of bombs).
    e. Get hit by a bomb.
4. Enter the shop.
    a. Buy an endless “boost” (Speed, bombrange, number of bombs etc.).
    b. Buy an ending “boost” (Hit points, life, more score/item etc.) that only last for a match.
5. Save the score.
6. See the highscore of the game.

2.2 Non-functional requirements

2.2.1 Usability

The usability is very important in this game. A new user should be able to start a new game in a short time and understand how it works in less then a half minute.
Tests should be performed by two persons to see if the games usability is clear fashion. The language of the game is english and we don’t see any need of another version since there’s not much text. Also it should be an english manual how to play the game.

2.2.2 Reliability

NA

2.2.3 Performance

Any actions by the players should not exceed a 300 ms. response time in worst case.

2.2.4 Supportability

The application should be implemented so that the GUI is easily modifiable to suit the web. Also, after we implemented the network based version it should be able to modify it to other platforms such as mobile apps, pads, etc.

2.2.5 Implementation

To achieve platform independence the application will use the Java environment. All hosts must have the JRE installed and congured. The application needs to be installed on all hosts where it will run (possibly downloaded).

2.2.6 Packaging and installation

2.2.7 Legal

There could be legal issues regarding rights to the Bomberman game and trade mark. This is not covered here.

3. Application models

3.1 Use case model

See appendix for UML and textual description.

3.2 Use cases priority

1. Move
2. Place bomb
3. Bomb explode
4. Destroy block
5. Hit player
6. Kill player
7. Turn end
8. Pick up item
9. Boost player
10. Match end
11. Scoring

3.3 Domain model

See appendix.

3.4 User interface

The GUI will be a 800x600. See the appendix for print screens.

3.5 References