Software Requirements Specification

for

Mario

Version <1.0>

Prepared by

Group Name: Five Stars

Jadecrystal Tang Ming Mei	36309	Jade92crystal@live.com
Lee Fui Yee	36618	fannyken@live.com
Nuratigha Binti Abd Razak	38098	nuratigharazumi@gmail.com
Tan Sheu Yeu	39049	sheuyeu@gmail.com
Vernon Chien	39233	Vernonchienchien@gmail.com

Instructor: Madam Nurfauza Jali

Course: Software Engineering Lab

Lab Section: *X*

Teaching Assistant: *X*

Date: 12 November 2014

Contents

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
		.	

1 Introduction

This document provides a complete example of a Software Requirements Specification (SRS) document for a small game application (namely Mario) that is meant to improve the popularity and quality of current version of Mario. In the following sections, we specify the purpose of this document, its intended audience, the scope of the end-product and all sources used in the production of this document.

1.1 Document Purpose

The purpose of this document is to specify and establish the functional and non-functional requirements associated with the Mario game software version 1.0. Mario game version 1.0 SRS will serve as the backbone for any other documents to be developed for this project in the future. Moreover, it will provide the basis for the future software verification and testing by specifying the behaviour requirements of the application in the form of a Use Case Diagram and State Machines.

1.2 Product Scope

Mario game is a computer based program which is intended to provide different version of Mario game for game players. Mario game will allow player to choose a character and name the chosen character before the game is started. Then, player will continue with the level game which is same as the traditional Mario game. Finally, the program will give another task for player before the player can proceed to the next level.

Mario program will be the game that can improve the popularity and quality of traditional version of Mario game. Furthermore, Mario game will provide some educational purpose to the player. In addition, this Mario game is also suitable for everyone regardless of age. On the other hand, Mario game can also save valuable childhood memory.

1.3 Intended Audience and Document Overview

The SRS document is the result of interview between the game application architect and the client and represents the mutual agreement between these two parties to bind a contract. Therefore, this SRS document should be carefully read and approved by the client, the development team and the Quality Assurance (QA) team. Section 2 of this document provides an overall description of the product by specifying product perspective, high level functionality of the product and other general product related requirements. Section 3 describes the specific requirements of the software including: External Interfaces, Functional Requirements and Behavioural Requirements in the form of use cases and state diagrams. This section will be the most interest to the project architects, developers and the QA team. Section 4 describes the non-functional requirements of the product and focuses on the different software attributes such as maintainability, security, performance, etc.

1.4 Definitions, Acronyms and Abbreviations

GUI – Graphical User Interface
QA – Quality Assurance
UI – User Interface

1.5 Document Conventions

1.5.1 Formatting Conventions

Several formatting conventions have been followed throughout the entire document:

1. All text contained in this document is 11pt Arial font.
2. Section titles are 18pt Arial font.
3. Subsection titles are 14pt Arial font.
4. Any further subsection breakdown is 12pt Arial font.
5. All sections and subsection are numbered using the x.x.x…format, where x represents numbers.

1.5.2 Naming Conventions

The naming conventions are the means of making the SRS more understandable and easier to follow. They are also used to build a structure for the whole game application. The conventions are used for variables, function names, packages, etc.

The following naming conventions have been used to define the different variables in this SRS document:

1. All constants are CAPITALIZED.
2. All variables representing input are prefixed with i_.
3. All variables representing output are prefixed with o_.

1.6 References and Acknowledgments

The following standards apply:

J-STD-016-1995	IEEE/EIA Standard for Information Technology, Software Lifecycle Processes, Software Development, Acquirer-Supplier Agreement
IEEE-STD-P1063	IEEE Standard for Software User Documentation

2 Overall Description

2.1 Product Perspective

Mario is a game designed and based from the original ‘Super Mario’ game. It is free and open source using Java as its programming language. Mario is aimed towards young and old generations alike who can be potential users. This project helps users to be reminded how addictively fun Mario can be. Mario is intended to be non-dependent on other software and should run on devices which can run Java.

2.2 Product Functionality

Mario game provides the following functions:

- Character naming system, means we can name the character chosen at the beginning of the game.
- Allows players to choose their favourite character. For example, player can choose either Mario or Luigi before enter game level 1.
- Allows players to move around in the 2-D environment. For example, the character that has been chosen can move forward, move backward and jump.
- Provides players with the game music.
- Displays the players collected coins for the level.

2.3 Users and Characteristics

Developers:	People with good knowledge of programming languages, able to understand and improve the game's source code.
General users:	People of all age groups without much experience because the project's interface is user friendly and the functions are simply and easy to use.

2.4 Operating Environment

Mario game is not platform dependence and can be installed in any operating system capable to run Java environment.

2.5 Design and Implementation Constraints

Mario is written in Java. Its user interface is written with Java programming language, so anyone who wishes to further develop the project has to have adequate knowledge in this programming language. Multilingual support is not available in this game.

2.6 User Documentation

At the present time no user documentation is available for this product. This document should serve as the basis for all user documentation to follow. Each step in the development of the product shall be documented and a detailed user manual shall be compiled during the development process.

2.7 Assumptions and Dependencies

The following is a list of assumed factors that could significantly affect the requirements stated in this document.

- The minimum hardware must be in place.
- A working Java Runtime Environment is necessary.

3 Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The user interface shall consist of three main GUI screens. The three interfaces are: Loading Screen – UI01, Name Entry Screen – UI02 and Character Selection – UI03.

3.1.1.1 Loading Screen – UI01

The Loading Screen shall serve as the main UI for the game.

UI01 Screen shall display the following things:
- The welcoming message: “Welcome to Mario”.
- Loading message: “loading…you have to wait for good things”.

Player has to wait until loading is finished and the screen then will switch to Name Entry Screen – UI02.

An example screenshot of UI01 is provided in Figure 1:

Figure 1: Mario game sample UI01

3.1.1.2 Name Entry Screen – UI02

The player should be able to enter the name for the character chosen.

UI02 shall display the following things:
- The welcoming message: “Welcome to Mario”.
- The prompting message: “Enter your name:”.

UI02 shall have the following list of textbox and button in its interface:
- Name textbox: shall allow the player to input i_Name.

- OK button: shall allow the Name Entry screen – UI02 switch to Character Selection screen – UI03.

An example screenshot of UI02 is provided in Figure 2:

Figure 2: Mario game sample UI02

3.1.1.3 Character Selection Screen – UI03

The player should be able to choose their favourite character.

UI03 shall display the following things:

- The welcoming message with name given: “Welcome XXX,”, where XXX is the name given in UI02.
- The prompting message: “Please choose one character”.

UI03 shall have the following buttons in its user interface:

- Character 1 button: shall allow the player to choose character 1 (Mario).
- Character 2 button: shall allow the player to choose character 2 (Luigi).

An example screenshot of UI03 is provided in Figure 3.

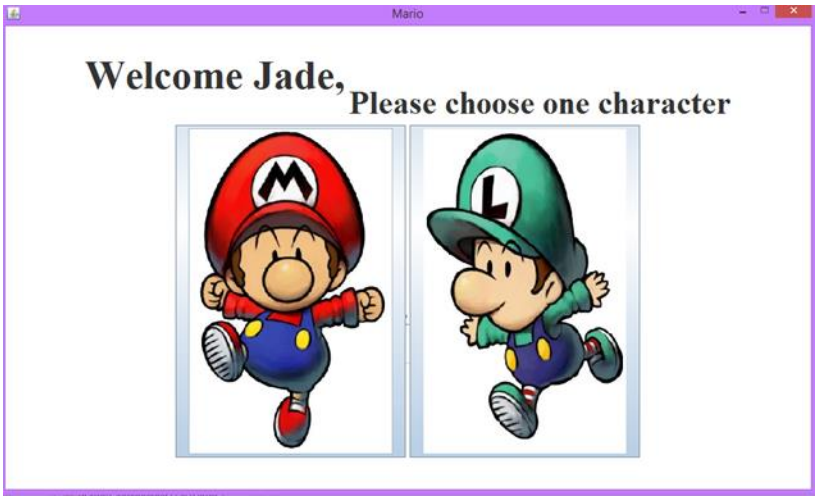


Figure 3: Mario game sample UI03

3.1.2 Hardware Interfaces

Mario game runs on any computer hardware meeting the following criteria:

- Includes a mouse
- Includes a keyboard

3.1.3 Software Interfaces

Mario game integrates some external software to provide functionality.

Client: Mario game interfaces with the user computer and expect that it have java environment installed.

3.1.4 Communications Interfaces

Mario game is not a web application, so no communications functions and communication standards are required by this program.

3.2 Functional Requirements

3.2.1 Character Naming

- Players shall be able to name the character after finish loading.
- When “OK” button is pressed, current Character Naming screen shall be switched to next screen which is Character Selection screen.

3.2.2 Character Selection

- Players shall be given the option in selecting the character, they can either choose Mario or Luigi.
- When character is selected, current Character Selection screen shall be switched to next screen which is the screen where the level 1 of the game starts.

3.2.3 Move Around

- Players shall be able to move forward when forward key on keyboard is pressed.
- Players shall be able to move backward when backward key on keyboard is pressed.
- Players shall be able to jump when up key on keyboard is pressed.

3.2.4 Shooting

- Players shall be able to shoot the enemies in Mario game when space bar on the keyboard is pressed.

3.2.5 Game Music

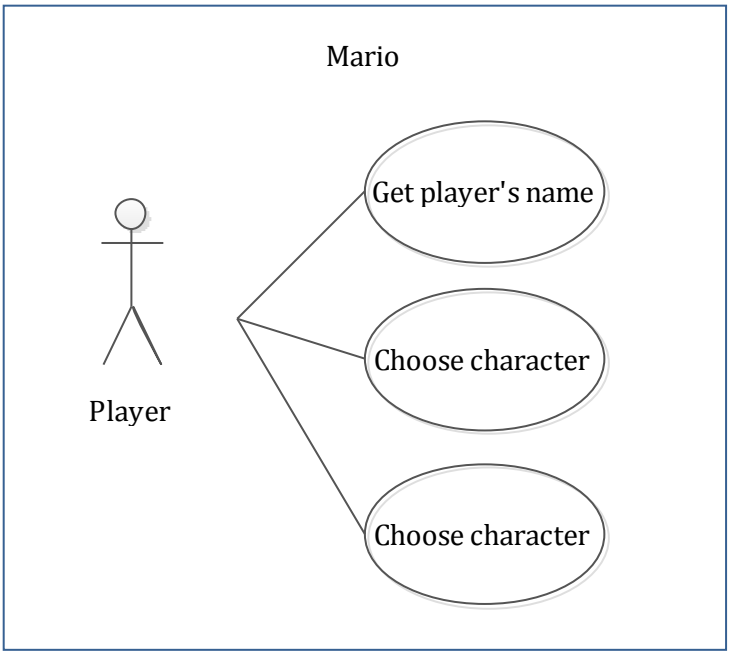
- Music shall be played once the players open Mario game.

3.2.6 Recording

- Coins collected by the player shall be recorded and displayed on the top of the windows.

3.3 Behaviour Requirements

3.3.1 Use Case View



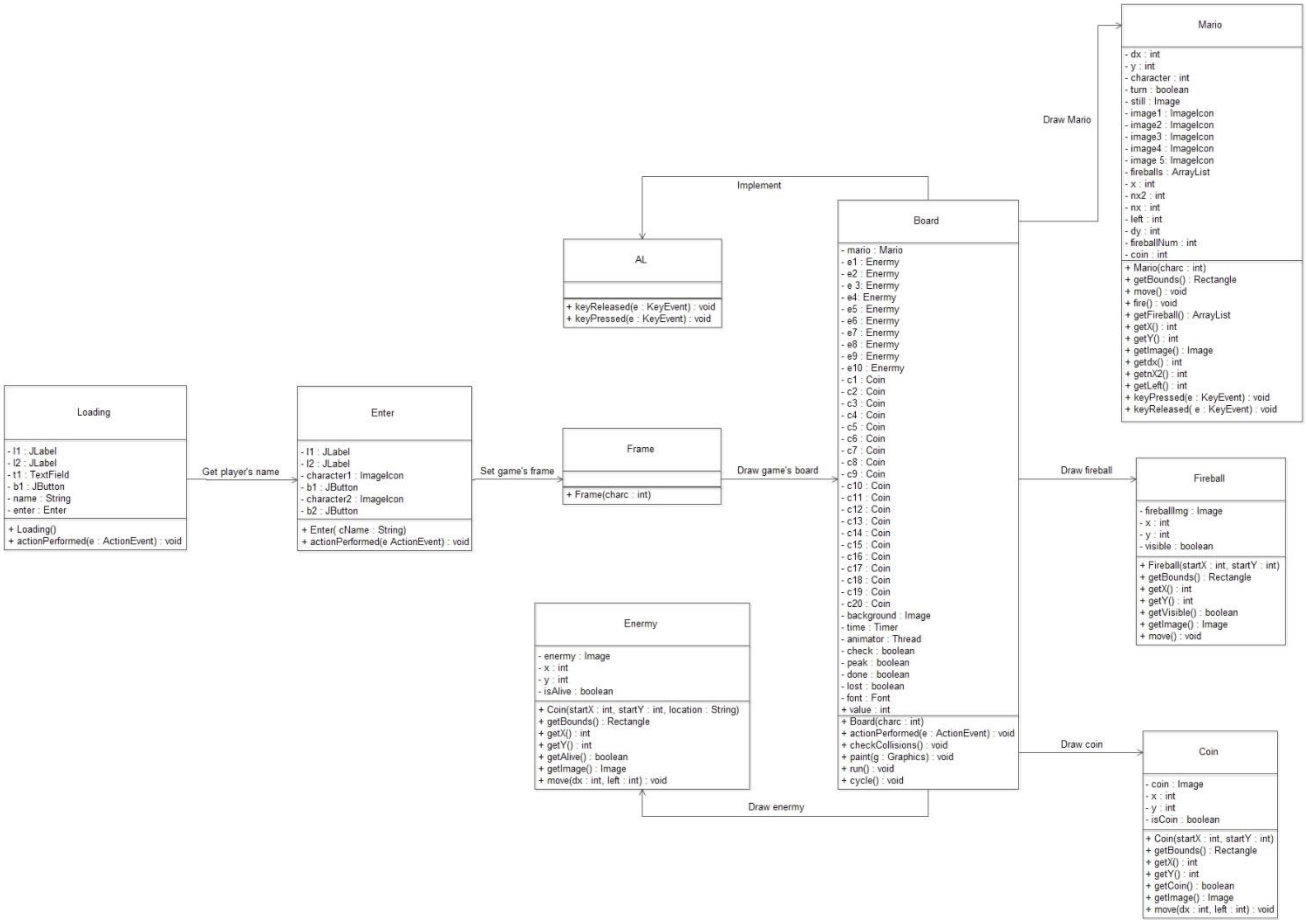
Mario: Get player's name

Actor	Player
Description	The game will provide an interface at every beginning whenever player wants to play the game. The game will prompt player to enter his or her name. After that, the game will continue after player clicks the OK button.
Data	Player's name
Stimulus	User command issued by player
Response	Continue the game
Comments	Player can only key in the name once, the name cannot be changed after the player clicks OK button.

Mario: Choose character	
Actor	Player
Description	The game will print out player's name and prompt player to choose the character. Player chooses the character by clicking one of the CHARACTER button. The game will continue after player clicks the CHARACTER button.
Data	Character
Stimulus	User command issued by player
Response	Continue the game with the chosen character
Comments	Player can only click one character. The chosen character will be used throughout the game.

Mario: Play game	
Actor	Player
Description	Player starts play the game. The Mario will jump after pressing the UP on the keyboard. Moreover, the Mario will move forward and backward by pressing the RIGHT and LEFT on the keyboard. A fireball will be fired each time when player presses SPACEBAR on the keyboard. Player has to kill the enemies in the game, collect coins and pass the mini games to win the game.
Data	Key event from keyboard Action event from mouse
Stimulus	User command issued by player
Response	The game end. The game exit.
Comments	The game will end before the players wins if the Mario is die.

3.3.2 Class View



4 Other Non-functional Requirements

4.1 Performance Requirements

Mario game consists of the following performance requirements:

- The loading time for Mario game will not take more than 10 seconds.
- The game shall respond to each player input within 3 seconds.
- Only single player gameplay is permitted.
- No internet connection is needed.
- The game must be able to run on all computer platform with java compiler installed.

4.2 Safety and Security Requirements

4.2.1 Safety

There are no safety requirements as the game has only three levels and is safe for children who are susceptible to flashing lights which are absent.

4.2.2 Security

- The program will not access any player data files or programs.
- The program will not alter or replace any system files.

4.3 Software Quality Attributes

4.3.1 Usability

- A new player should be able to play a complete game of Mario in less than 10 minutes.
- A new player should commit less than one error in use of the game every ten minutes.
- A user who is familiar with the traditional Mario game be able to correctly operate the program without any written documentation.

4.3.2 Portability

- The game is Java based and should be compatible to devices with Java implementations.

4.3.3 Reliability

- Since the program is purely for recreation and involves no user data, reliability is of low importance.

5 Other Requirements

There is no other requirements for Mario game.

Appendix A – Data Dictionary

Since Mario game does not include any database, so there is no data dictionary.

Appendix B - Group Log

MEETING MINUTES OF Five Stars
[Roles and Tasks Allocation]
Date : 19 September 2014
Time : 7:00 – 9:00 PM
Venue : Teaching Lab, FCSIT

ATTENDENCE

Project Manager: Jadecrystal Tang Ming Mei
Team Members: Nuratiqha Binti Abd Razak, Vernon Chien, Lee Fui Yee, Tan Sheu Yeu

1.0 INTRODUCTION

The meeting was called to distribute tasks to project team members by project manager, Jadecrystal Tang Ming Mei

2.0 AGENDA

- Opening and welcome
- Allocation of tasks
- Project Resources and Communications
- Date of next meeting

3.0 TOPIC DISCUSSED AND ACTION

No.	Topic of Discussion	Action Taken by
1.	Distribution of tasks according to TSP Student Information Sheet	Jadecrystal Tang Ming Mei
2.	Determination in project resources and communications	Jadecrystal Tang Ming Mei

Prepared by:

Nura

(Nuratiqha Binti Abd Razak)

Endorsed by:

Jade

(Jadecrystal Tang Ming Mei)