Software Design 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**
Nuratiqha Binti Abd Razak	**38098**	**nuratiqharazumi@gmail.com**
Tan Sheu Yeu	**39049**	**sheuyeu@gmail.com**
Vernon Chien	**39233**	**vernonchienchien@gmail.com**

Instructor: Madam Nurfauza Binti Jali

Course: Software Engineering Laboratory

Lab Section: *Friday (2:30 – 4:30 PM)*

Date: 1st December 2014

Contents

Lists of Tables, Figures and Diagrams

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Five Stars	Software Requirement Specification (SRS) for Mario Game Application that described on the introduction of game application, overall description process, specification requirement either functional or non-functional and other requirements that use in this application.	12nd November 2014
2.0	Five Stars	Software Requirement Specification (SRS) for Mario Game Application that described on the introduction of game application, overall description process, specification requirement either functional or non-functional and other requirements that use in this application.	1st December 2014

1 Introduction

This document provides a complete example of first version of a Software Design Specification (SDS) document for a small game application (namely Mario). This document is primarily based on the developed earlier Mario SRS document. In the following sections, we specify the purpose of this document, the overview of the small game application and the sources used in the production of this document.

1.1 Purpose

The purpose of this document is to provide a high-level overview of the architecture and a low-level design for the small game application. In addition, it outlines the different strategies and methods used to produce the most effective architectural structure for the software.

1.2 System Overview

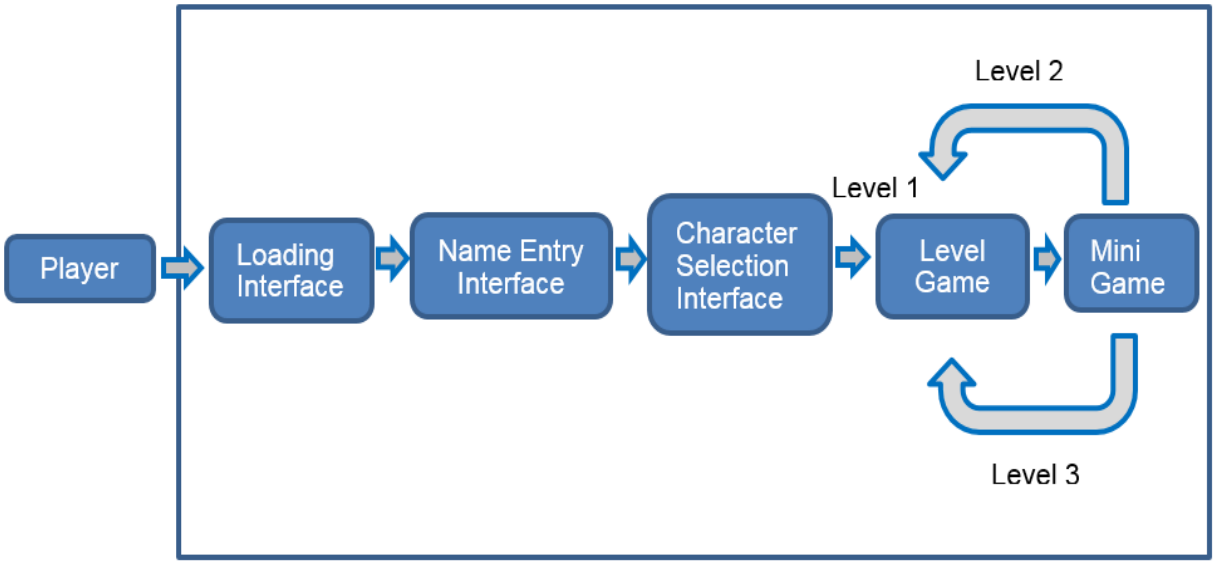

Diagram 1.0: Mario Game Overview

Mario Game application runs on computer with Java compiler installed. The minimal requirements related to the computer, shall adhere to the ones specified in section 2.4 (Operating Environment) of the Mario SRS v. 2.0 document. Refer to Figure 2.0, the application does not have restricted access, which means everyone are able to play the game. Player can name their character, select their favorite character (Mario or Luigi) and play the level game and mini game.

The Mario Game will manage the condition of every level game and mini game to determine whether the player can continue his/ her journey, success or fail in his/ her journey.

1.3 Definitions, Acronyms and Abbreviations

Term	Description
IEEE	Institute of Electrical and Electronics Engineers
SDS	Software Design Specification
SRS	Software Requirements Specification
UI	User Interface

Table 1.0: The Abbreviations in Mario Game Application

1.4 Supporting Materials

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

The following document has been used in the process of developing this document:

[1] Five Stars. (2014). *Mario Software Requirements Specification.*

[2] Morpheus Page. (2014). SDS Example – Mini Thermostat.

1.5 Document Overview

The next section of the Mario SDS v. 1.0 provides the architectural view of the system. It illustrates how the game application was decomposed into several main components and what were the reasons for this particular decomposition. Subsections of section two, describe in detail the components and their corresponding interfaces. Section three of the document, provides a Control View of the small game application and explanations regarding the different state transitions. Finally, in section four the reader will find the low-level design of the system. Low-level design includes an overview of the modules and a detailed design of every module.

2 Architecture

This section provides the architecture design of the Mario Game application. It includes the final version of the game application Component Diagram which illustrates the different components, their interfaces and dependencies between components. The main principles that led to the proposed design are:

- Maintainability
- Understandability
- Anticipation of changes

Section 2.2 discusses in detail the different components and how they adhere to the listed above principles. The next section provides the Component Diagram and the details about the proposed architecture.

2.1 Overview

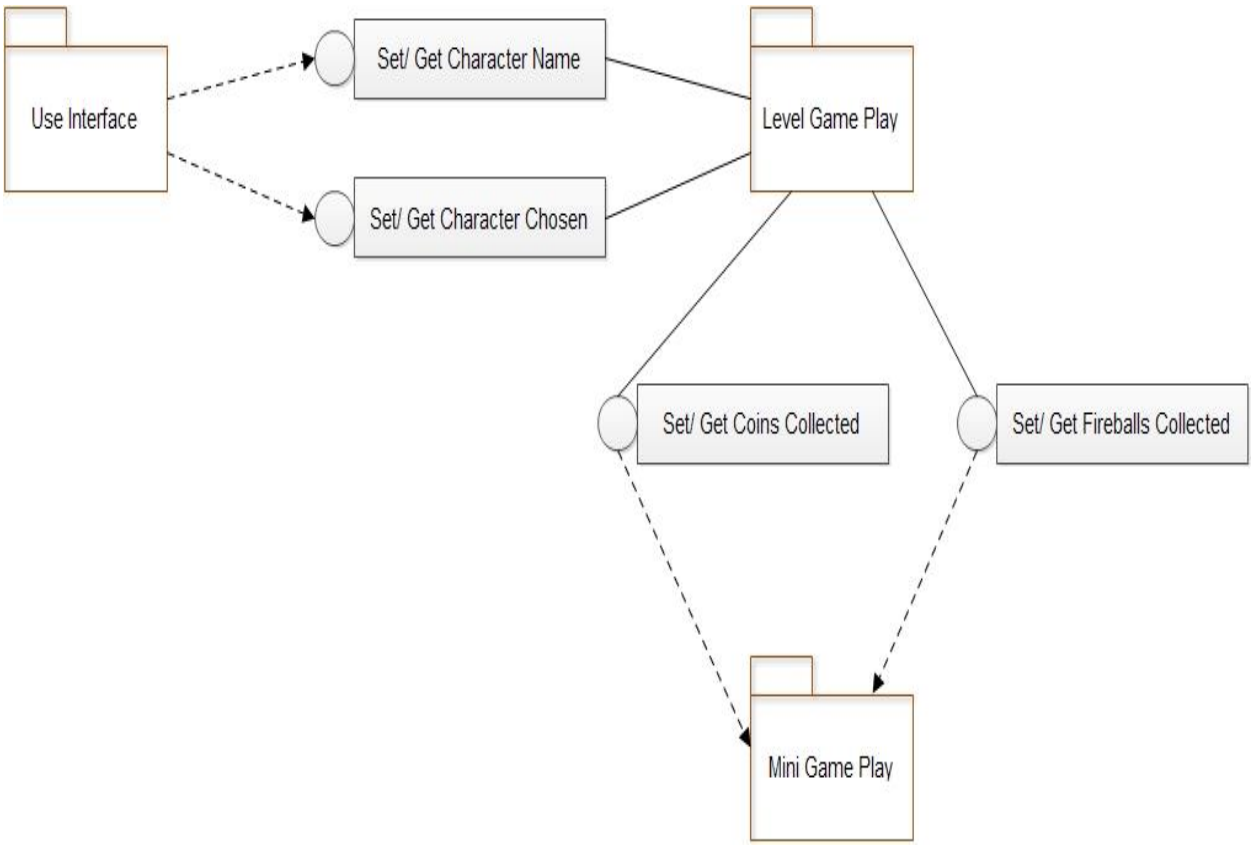

Figure 2.0: Mario Game Component Diagram

As illustrated in Figure 2.0, Mario Game Application consists of three main components:

1. User Interface (UI)
2. Level Game Play
3. Mini Game Play

In order to make the architecture more understandable, maintainable and anticipation for changes it was decomposed according to the different functional areas of the entire game application. All three components cover specific functional areas, as specified in the Mario SRS document. By decomposing the game application to only three main components, the architecture of the Mario Game application becomes more understandable and the game is easier to maintain.

2.2 Component of User Interface (UI)

User interface (UI) component provides graphical interfaces to end-user. It depends on:

2.2.1 Set Character Name/ Get Character Name

Interface provided by Level Game Play component. Through this interface, UI provide the end-user with the ability of enter the name for the chosen character (Mario or Luigi).

2.2.2 Set Character Chosen/ Get Character Chosen

Interface provided by Level Game Play component. Through this interface, UI provide the end-user with the ability of select their favorite character (Mario or Luigi).

2.3 Component of Level Game Play

- Level Game Play component is responsible to display the chosen character and the coins and fireballs collected corresponding to the different level game play. Level Game Play component does not depend on any other component of the Mario Game application and provides two interfaces for this component.
- The interfaces are:

2.3.1 Set Character Name/ Set Character Set Coins Collected/ Set Fireballs Collected

Interfaces provide the ability to store any changes that related to the different level game play.

2.3.2 Get Character Name/ Get Character Chosen/ Get Coins Collected / Get Fireballs Collected/ Get

Interfaces provide the ability to retrieve the current interfaces to end-user.

2.4 Component of Mini Game Play

Mini Game Play component depends on two interfaces. It depends on:

2.4.1 Set/ Get Coins Collected

- Interfaces provided by the Level Game Play component. Mini Game Play component use this interface to obtain coins collected in the previous level game. Thus, the game application can manage and display the number of coins collected after the completion of mini game.

2.4.2 Set/ Get Fireballs Collected

- Interfaces provided by the Level Game Play component. Mini Game Play component use this interface to obtain fireballs collected in the previous level game. Thus, the game application can manage and display the number of fireballs collected after the completion of mini game.

3 High Level Design

The High-Level Design section describes in further details the interactions between the system components and their corresponding interfaces. To illustrate the dynamic behaviour of the Mario Game application, Control view using Statecharts has been used. Statecharts model the behaviour from the perspective of a single entity. The entity that is modelled in Figure 2.0 is the complete Mario Game application.

3.1 View / Model Component

- At the highest level of abstraction, the architecture can be thought of as a composition of a small set of modules, each related to a specific core functional area.
- The state diagrams are used by developers to understand better any complex or unusual functionalities of specialized areas of the application.
- Behavioural elements of state diagram are brief described below.

3.1.1 Player

- Can create character name.
- Can select character (Mario or Luigi).
- Can view coins and fireballs collected.

3.1.2 Operation

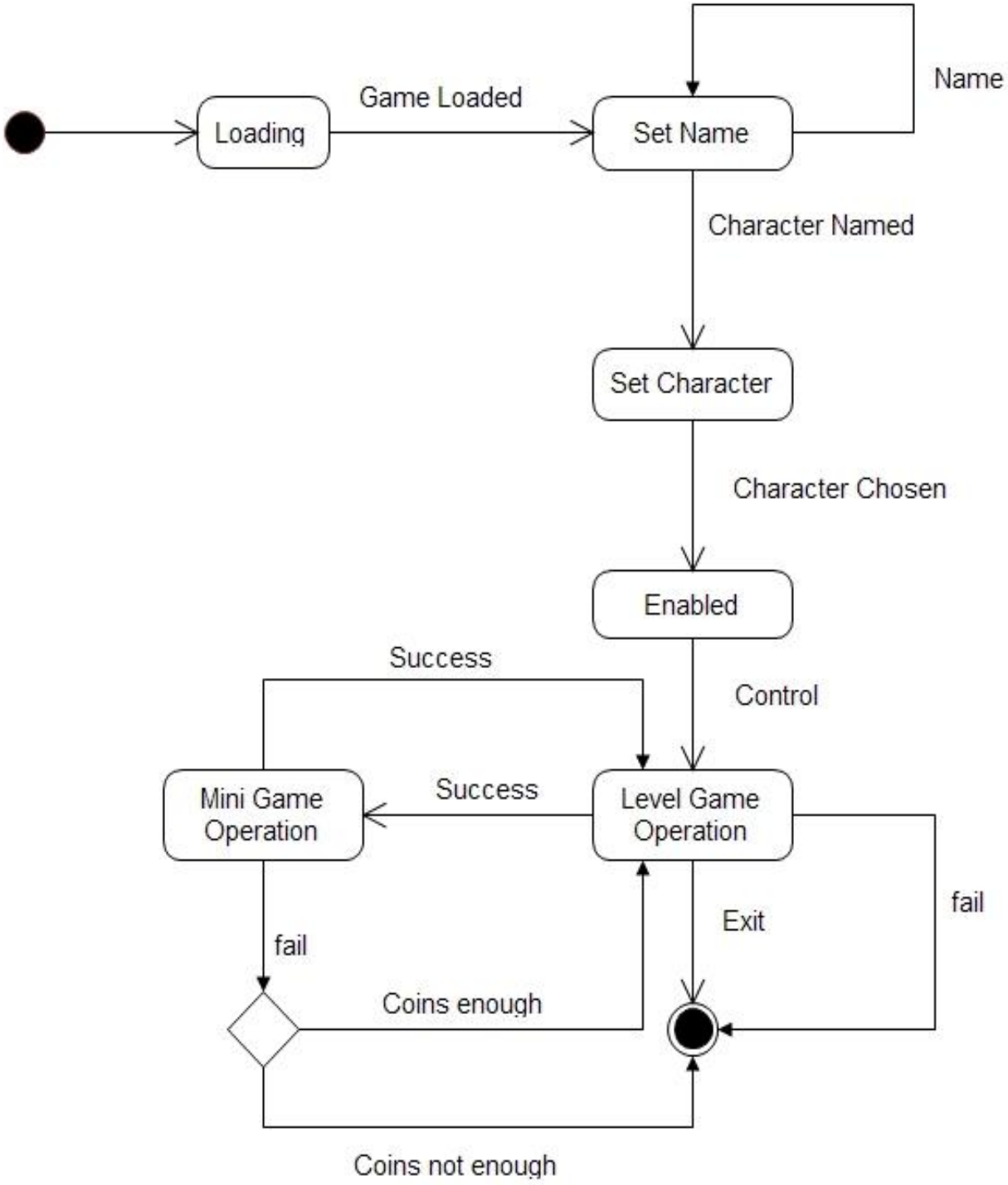

Diagram 3.0 Statechart for Mario Game Application

State	Description
Loading	The game is loading. The display shows welcoming message.
Set Name	The character name is set to the player's input value. The display shows "welcome," and the name given.
Set Character	The character is set to the player's selected character.
Enabled	Level game operation is enabled. The game is ready for player. The display shows the scene of level one and the character chosen.
Level Game Operation	Level game in operation. The display shows the movements of the chosen character which are move forward, move backward and jump. Music of the game is played.
Mini Game Operation	Mini game in operation. The display shows the problem-solving questions.

Table 2.0 States for the Mario Game Application

Stimulus	Description
Game Loaded	The game is loaded.
Name	The player has pressed character key.
Character Named	The player has entered the character named.
Character Chosen	The player has pressed the chosen character.
Control	The player has pressed control key. For example, forward key, backward key, up key or spacebar.
Fail	• The player failed the level game. • The player failed the mini game.
Success	• The player passed the level game. • The player passed the mini game.
Coins Enough	The player has enough money to be deducted.
Coins Not Enough	The player doesn't have enough money to be deducted.
Exit	The player pressed 'X' button.

Table 3.0 Stimulus for Mario Game Application

Appendix A – Group Log

Date	Time	Activity	Attendance
8/9/2014 – 14/9/2014, Monday – Friday	-	• Group project has released. • A group member has been formed. Total of 5-6 members.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
24/9/2014, Wednesday	(1400 – 1700)	• Assigned position of every member. • Defined the project scope. • Discussed about the project, identified the objectives and goals of the project.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
29/9/2014 – 3/10/2014, Monday – Friday	-	• Prepared the initial proposal. • Presented the initial proposal. • Initial proposal handed.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
6/10/2014, Monday	(1900 – 2230)	• Defined the target player. • Planned project timeline. • Distributed each task that need to be corrected before hand-in the edited copy of proposal.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
10/10/2014, Friday	(1430 – 1630)	• Final proposal handed.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
13/10/2014, Monday	(1300 – 1600)	• Discussion for the design of use case diagram, class diagram and sequence diagram of the game.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
20/10/2014, Monday	(1300 – 1600)	• Use case diagram, class diagram and sequence diagram have improved. • Discussion for the design of activity diagram and project interfaces.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
28/10/2014, Tuesday	(2000 – 0000)	• Activity diagram and project interfaces have improved.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak

		• Task distributed for each member for the SRS documentation.	• Tan Sheu Yeu • Vernon Chien
31/10/2014, Friday	(1630 – 1930)	• A basic interfaces of the game are coded.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
4/11/2014, Tuesday	(1300 – 1600)	• First draft of the SRS documentation completed. • Discussed regarding the documentation and checked for any wrong in the documentation.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
7/11/2014, Friday	(1630 – 1930)	• 50 % of the game has completed.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
10/11/2014 – 14/11/2014, Monday – Friday	-	• Presented the SRS documentation. • SRS documentation handed.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
17/11/2014, Monday	(1300 – 1500)	• Task distributed for each member for the SDS documentation.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
21/11/2014, Friday	(1630 – 1830)	• 70 % of the game has completed.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
24/11/2014, Monday	(1300 – 1500)	• Discuss and started the TSPi forms	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
26/11/2014, Wednesday	(1400 – 1600)	• Continue of the TSPi forms	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
28/11/2014,	(1630 – 1830)	• Whole game has completed.	• Jadecrystal Tang Ming Mei • Lee Fui Yee

Friday		• Discussion of the game flow and pointed out some game's problem.	• Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
1/12/2014, Monday	(1300 – 1430)	• All the problems have solved. • TSPi forms have completed.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
5/12/2014, Friday	(1430 – 1630)	• Final meeting for this course. • Any difficulties encountered solved.	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien
9/12//2014, Friday	(1630 – 1800)	• Compilation of the documentations: SRS, SDS and TSPi forms	• Jadecrystal Tang Ming Mei • Lee Fui Yee • Nuratiqha Binti Abd Razak • Tan Sheu Yeu • Vernon Chien