Software Requirements Specification Template

CS 258 Software Engineering

January 2018

The following annotated template shall be used to complete the Software Requirements Specification (SRS) assignment of CS 258.

**Template Usage:**

Text contained within angle brackets (‘<’, ‘>’) shall be replaced by your project-specific information and/or details. For example, <Project Name> will be replaced with either ‘Smart Home’ or ‘Sensor Network’.

Italicized text is included to briefly annotate the purpose of each section within this template. This text should not appear in the final version of your submitted SRS.

This cover page is not a part of the final template and should be removed before your SRS is submitted.

**Acknowledgements:**

Sections of this document are based on the IEEE Guide to Software Requirements Specification (ANSI/IEEE Std. 830-1984). The SRS templates of Dr. Orest Pilskalns (WSU, Vancover) and Jack Hagemeister (WSU, Pullman) have also been used as guides in developing this.

Development of a Game on Landslide disasters in North-eastern states of India

Software Requirements Specification

1.0

25th January 2018

160001004 - Amit Meena

160001011 - Ashutosh Bang

160001023 - Joshi Niranjan Suhas

160001033 - Mahesh Kumar

160001034 - Anish Sri Chandra

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| <date> | <Version 1> | <Your Name> | <First Revision> |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  | <Your Name> |  |  |
|  |  |  |  |
|  |  |  |  |

**Table of Contents**

Revision History ii

Document Approval ii

1. Introduction i

1.1 Purpose i

1.2 Scope i

1.3 Definitions, Acronyms, and Abbreviations i

1.4 References i

1.5 Overview i

2. General Description ii

2.1 Product Perspective ii

2.2 Product Functions ii

2.3 User Characteristics ii

2.4 General Constraints ii

2.5 Assumptions and Dependencies ii

3. Specific Requirements ii

3.1 External Interface Requirements iii

3.1.1 User Interfaces iii

3.1.2 Hardware Interfaces iii

3.1.3 Software Interfaces iii

3.1.4 Communications Interfaces iii

3.2 Functional Requirements iii

3.2.1 <Functional Requirement or Feature #1> iii

3.2.2 <Functional Requirement or Feature #2> iii

3.3 Classes / Objects iii

3.3.1 <Class / Object #1> iii

3.3.2 <Class / Object #2> iii

3.4 Non-Functional Requirements iii

3.4.1 Performance iv

3.4.2 Reliability iv

3.4.3 Availability iv

3.4.4 Security iv

3.4.5 Maintainability iv

3.4.6 Portability iv

3.5 Inverse Requirements iv

3.6 Design Constraints iv

3.7 Logical Database Requirements iv

3.8 Other Requirements iv

A. Appendices iv

A.1 Appendix 1 iv

A.2 Appendix 2 iv

# 1. Introduction

This document lays out a project plan for the development of “Salvos”, an open source repository. This is a game for awareness on landslides.

## 1.1 Purpose

The purpose of this document is to describe the purpose and functionality of the software product requested by Dr. Neelima Satyam of The Department of Civil Engineering, IIT Indore. The SRS will include the details of the project's requirements, interface, design issues and components. The intended readers of this document are the current and future developers working on the project and proposer of this project.

## 1.2 Scope

The “Salvos” project is a learning tool created to help improve the awareness among the children of age group between 6 to 10. The product will be an interactive multi-platform game. The Application will be released both as a native Windows desktop application and an android based app. At the end of the game, students will be given feedback based on their game scores. The game will focus on both prevention and survival during the landslide.

## 1.3 Definitions, Acronyms, and Abbreviations

*This subsection should provide the definitions of all terms, acronyms, and abbreviations required to properly interpret the SRS. This information may be provided by reference to one or more appendixes in the SRS or by reference to other documents.*

## 1.4 References

* *Dr. Neelima Satyam, Department of Civil Engineering, IIT Indore*
* *Dr. Abhishek Srivastav, Department of Computer Science & Engineering, IIT Indore*
* *Overview of Game Design :* [*https://github.com/maheshcool/Group-H-Project-4/blob/master/Documentation/Game\_design.md*](https://github.com/maheshcool/Group-H-Project-4/blob/master/Documentation/Game_design.md)
* *Literature on Landslides : https://github.com/maheshcool/Group-H-Project-4/blob/master/Documentation/literature.md*

## 1.5 Overview

# This document is designed to provide information to both the client and the technical designers of the software. Section one is a brief overview of the product, including definitions and references. The definitions section is intended to assist the technical designers as well as the client in clarifying the terms used throughout the document. Section two is a general description of the product requirements from a user's perspective. This section includes information such as functional and data requirements, general constraints, and assumptions. Section three is a detailed requirements specification targeted toward technical designers. Specific requirements and expectations regarding the components of the product are given in this portion of the SRS document.

# 2. General Description

## 2.1 Product Perspective

Desktop:

* OS: Windows XP SP2+.
* Graphics card: DX9 (shader model 3.0) or DX11 with feature level 9.3 capabilities.
* CPU: SSE2 instruction set support.
* Android: OS 4.1 or later; ARMv7 CPU with NEON support or Atom CPU; OpenGL ES 2.0 or later.

*The program requires an internet connection only for downloading the application.*

*The game can run offline.*

## 2.2 Product Functions

This subsection of the SRS should provide a summary of the functions that the software will perform.

## 2.3 User Characteristics

## The target clients for our software are the children of the age group in between 6 to 10. These students are in the process of learning how to tackle the problem of landslides both during the disaster and pre-disaster stages. Moreover, these students (as well as the teacher) are assumed to have basic computer and Internet skills that will enable them to use this software. For the mobile app, they are assumed to know how to operate a mobile phone. The game will request the user to enter his/her username and a character will be assigned to them. The game will be played using this character.

2.3 Overview of Functional Requirements

The "Salvos" game will have the following functional components:

1. An introductory movie to set up the storyline.

2. A main menu, including a brief help and settings section

3. A series of missions (testing survival skills and awareness) that sequentially form a storyline related to the introduction.

The missions will be in the form of a path where the next mission is unlocked only after the completion of the previous mission

4. The user’s score is calculated at the end of every mission and is in the form of stars. The user can get a maximum of 3 stars.

At any time during the game, the user has an option to quit the game.

In between or at the end of each mission, the user has an option to retry the game.

## 2.4 General Constraints

This program will run offline on any computer or a mobile phone that satisfies the above mentioned requirements.

## 2.5 Assumptions and Dependencies

It is assumed that the player is age group in between 6 to 10 without any prior knowledge on landslides. He/she should know how to operate a computer. It is assumed that the game is properly installed on the device. The device must satisfy the above mentioned requirements.

**2.6 Flow chart for Game flow**

**2.7 Game Design**

# 3. Specific Requirements

## 3.1 External Interface Requirements

3.1.1 User Interfaces

We will have to provide a 3d video interface to the user so that he can get himself more and more

into the game. A first player controller, hence, would be the best suited for the game.

3.1.2 Hardware and Software Interfaces

In order to play the game, we will facilitate the users to use their keyboards, touchpad, mouse or

joystick to input the commands and real time effects of the input will be reflected in the game.

The user should have a good graphic card installed in his/her machine. Basically, mouse shall be

used to change rotate the screen and arrow keys to move in the respective directions.

## 3.2 Functional Requirements

**3.3 Non-Functional Requirements**

**3.3.1 Performance**

All the modules should be in a single game and should be followed by a single story or theme of

the game. The games should be as much interactive as look like a game environment not a test.

Different tasks in the game should require the above mentioned parameters.Responses should be

added by mouse or keyboard and a log file should be generated having the score information.

**3.3.2 Availability**

The game should be of distributable over PCs and Android devices. And, since we’re publishing it on the web it will be accessible and available for downloading.

**3.3.3 Security**

The game will be entirely offline. So, there are no threats from potential hackers.

**3.3.4 Maintainability**

Collaboration using version control systems like Git and storage of the code on cloud of github

and proper comments and description for every code will ensure that a new programmer who gets to work or improve this very piece of code faces minimal hurdles. A proper documentation

for every module shall be included.

**3.3.5 Portability**

All the laptop/PC machines that posses Windows XP SP2 or versions newer than that are capable to run this.

## 3.6 Design Constraints

## *The outdated assets of unity in the unpaid version might be a constraint but it shall be compensated by making the modules more interactive and user friendly to drag the player more into the story of the game.*