

Software Requirements Specification

for

MULTI-USER TEXT EDITOR

Version 1.0 approved

Prepared by

**Sidhant Arora 101403176
Soham Ghosh 101403178
Soumik Pradhan 101403179
Sudhanshu Kaul 101403180**

Nov 26th, 2016

TABLE OF CONTENTS

1. Introduction

- 1.1 Purpose of this Document
- 1.2 Scope of the Development Project

2. Overall Description

- 2.1 Product Perspective
- 2.2 Product Function
- 2.3 User Characteristics
- 2.4 General assumptions, constraints & dependencies
- 2.5 Apportioning of requirements

3. Specific Requirements

- 3.1 External Interface Requirements
- 3.2 Detailed Description of Functional Requirements
- 3.3 Software Quality Attributes

4. Change History

5. Document Approvers

6. References

1. Introduction

1.1 Purpose of this Document

The purpose of this SRS document is to provide a detailed overview of our software product, its parameters and goals. This document describes the project's target audience and its user interface, hardware and software requirements. It defines how our client, team and audience see the product and its functionality.

1.2 Scope of the Development Project

The software can be extended to all Integrated development Environment where developers can collaborate to form a product as a group. This can be even included in most modern text editors like Sublime Text, Atom, Brackets etc.

It can even further be extended to web based UI for collaborating text documents online.

2. Overall Description

2.1 Product Perspective

Earlier, people used to code in text editors such as Notepad or Word pad, and their corresponding equivalents on Linux and macOS. Now, with the advent of UI/UX and technology, developers and software engineers actually care how their code looks like, indentation, syntax highlighting, and so on.

So, keeping in mind this current trend, our team has made a fully functional, multi-user text editor, complete with the features listed above.

The main aim of our project is not to compete with the current market (Sublime Text, CodeBlocks, Atom, etc., which stands well above our project. Our aim is to simply show that anybody with the right mindset, right skill set, and determination can actually build one, without too much toil.

2.2 Product Functions

The software must be able to perform following operations :

1. **Create and Edit Programming Language Source Code** : It must be able to open and edit the source code of any programming language.
2. **Host File Online** :It must be able to host file online and generate a unique ID to the user.
3. **Join Hosted File Online** : It must be able to join multiple users who have the same ID as the file being hosted online.
4. **Syntax Highlighting** : It must be able to highlight the syntax of any source code upon selecting the type of programming language.
5. **Text Highlighting** : Specific text can be highlighted by the user.

2.3 User Characteristics

We have identified five potential classifications of users of our system:

Users : These are the people that take the model generated by the designers and implement it in code. They may also use the system to identify the design of an existing system in order to maintain it.

2.4 General assumptions, constraints & dependencies

In general it is assumed that the user is well versed with the fundamentals of programming and hence has used text editor earlier.

The software does not have compiling options for the source code.

Dependencies used in this project is sockets, a javascript library.

2.5 Apportioning of requirements

The Multi-user Text Editor is to implemented in following eight phases:

1. **Preparation:** This phase involves identifying the goal and overall strategy, design client questionnaire, preparation of feasibility report etc.
2. **Planning:** This phase involves the how the development of the software will take place and assigning various tasks to different team members.
3. **Requirement Gathering:** This phase involves the gathering of requirements, research on official documentation and various external libraries like sockets.
4. **Modelling:** This phase shall involve designing the actual system and identifying the requirement to develop it. In this phase we need to design the interface of the software and enlist all the functionalities to be included. Also it includes the designing of all the UML diagrams like Activity diagram, Class diagram, Use-Case diagram and scenario, DFD diagram etc.
5. **Construction:** This phase witness the actual development of the project and turning the project into actual program. In this actual coding will be done for each functionality and testing will be conducted so all basic functionality should be working correctly and efficiently.
6. **Testing:** This phase will engage in testing and debugging according to test cases and test plan created and verify that each feature is indeed producing the necessary result.
7. **Deployment:** Actual deployment of the project will happen in this phase. Here project will be set up in the production environment and after final testing and enhancements will be setup on the main website.
8. **Maintenance:** This phase mean testing the working of project in real time environment and remove any bugs that may occur whatsoever.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

The user interface is clean and simple so that all developers can dive right in to product development rather than wasting time learning technicalities of the software.

1. Toolbar : Simple Toolbar options File, Edit, View and Help options.
2. Dropdown Menu : Simple dropdown menu options such as New, Open, Save and Save As
3. Text Highlighting : Highlight current line option under View
4. Go Online : Toggle button on top right corner.
5. Unique ID : Once document gets hosted to server a unique ID gets generated effortlessly.
6. Themes : Various built in themes for formatting text
7. Syntax Highlighting : Automatic detection of programming language and syntax highlighting.

3.1.2 Hardware Interfaces

The project requires INTEL/ AMD 32 bit processor, minimum 512 MB RAM, 512 MB disk space, and it uses other hardware devices such as mouse and keyboard as positioning devices, and monitors for showing the information and the whole system interacts with the internet server for getting and updating the information of patients and analyzing data online from anywhere.

3.1.3 Software Interfaces

The project has software requirements of programming language like Python, Javascript ;external libraries like socks and developing tools which are Pycharm, atom

3.2 Detailed Description of Functional Requirements

3.2.1 Performance Requirements

The software is built using python programming language and to use real time editing features it use sockets which is an extremely responsive bidirectional user and server side communication javascript library.

Hence all edits occur in real time without any noticeable lag to the user.

3.2.2 Safety Requirements

Source code in the text editor would never be shared to any unauthenticated sources. Every time a user hosts file online a unique ID gets generated and only those who have the same ID can join. Without the permission of the user who hosted the file, no unauthenticated person can join the document.

3.2.3 Security Requirements

The software provides a unique ID to the user who hosts the file online. This ID is required by any other user who wants to join the document for collaboration. Hence no information can be shared to any unauthenticated user.

3.3 Software Quality Attributes

4. Change History

2016	Version 1.0 - Initial Release
------	-------------------------------

5. Document Approvers

SRS for Multi-user Text Editor is approved by:

Name :

Designation :

Date :

6. References

1. Python Documentation <https://docs.python.org/3/tutorial/>
2. Tkinter Graphics Library <https://docs.python.org/3/library/tk.html>
3. Text Editor Features Documentation jqueryte.com/documentation
4. Sockets Documentation socket.io/docs/