****

**DEPARTMENT OF COMPUTER SCIENCE & INFORMATION TECHNOLOGY**

Spring 2018 – 2022

Document Format Validator

Software Requirements Specification

01

24 June 2021

Prepared By

Zohaib Mehmood (70062940)

&

Adil Bashir (70065622)

# Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Description** | **Author** | **Comments** |
| 24 Jan,2021 | The initial version of SRS. | Zohaib Mehmood  Adil Bashir | This is first version of SRS. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Document Approval

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

|  |  |  |  |
| --- | --- | --- | --- |
| **Signature** | **Printed Name** | **Title** | **Date** |
|  | Mr. Ijaz Khan Baig | Supervisor |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

[Revision History 2](#_Toc75257593)

[Document Approval 2](#_Toc75257594)

[1. Introduction 1](#_Toc75257595)

[1.1 Purpose 1](#_Toc75257596)

[1.2 Scope 1](#_Toc75257597)

[1.3 Definitions, Acronyms, and Abbreviations 1](#_Toc75257598)

[1.4 References 2](#_Toc75257599)

[1.5 Overview 2](#_Toc75257600)

[2. General Description 2](#_Toc75257601)

[2.1 Product Perspective 2](#_Toc75257602)

[2.2 Product Functions 2](#_Toc75257603)

[2.3 User Characteristics 2](#_Toc75257604)

[2.4 General Constraints 3](#_Toc75257605)

[2.5 Assumptions and Dependencies 3](#_Toc75257606)

[3. Specific Requirements 3](#_Toc75257607)

[3.1 External Interface Requirements 3](#_Toc75257608)

[3.1.1 User Interfaces 3](#_Toc75257609)

[3.1.2 Hardware Interfaces 3](#_Toc75257610)

[3.1.3 Software Interfaces 3](#_Toc75257611)

[3.1.4 Communications Interfaces 3](#_Toc75257612)

[3.2 Functional Requirements 3](#_Toc75257613)

[3.2.1 Register 3](#_Toc75257614)

[3.2.2 Login/Logout 3](#_Toc75257615)

[3.2.3 File Uploading 4](#_Toc75257616)

[3.2.4 Set Parameters 4](#_Toc75257617)

[3.2.5 Upload Parameters 4](#_Toc75257618)

[3.2.6 Show Results 4](#_Toc75257619)

[3.3 Use Cases 5](#_Toc75257620)

[3.3.1 Use Case #1 5](#_Toc75257621)

[3.3.2 Use Case #2 6](#_Toc75257622)

[3.3.3 Use Case #3 6](#_Toc75257623)

[3.3.4 Use Case #4 6](#_Toc75257624)

[3.3.5 Use Case #5 7](#_Toc75257625)

[3.4 Non-Functional Requirements 7](#_Toc75257626)

[3.4.1 Usability 7](#_Toc75257627)

[3.4.2 Reliability 7](#_Toc75257628)

[3.4.3 Availability 7](#_Toc75257629)

[3.4.4 Security 8](#_Toc75257630)

[3.4.5 Performance 8](#_Toc75257631)

[3.4.6 Portability 8](#_Toc75257632)

[3.4.7 Accurate Results 8](#_Toc75257633)

[3.5 Inverse Requirements 8](#_Toc75257634)

[3.6 Design Constraints 8](#_Toc75257635)

[3.7 Logical Database Requirements 8](#_Toc75257636)

[3.8 Other Requirements 8](#_Toc75257637)

[4. Analysis Models 9](#_Toc75257638)

[4.1 Sequence Diagrams 9](#_Toc75257639)

[4.2 Data Flow Diagrams (DFD) 10](#_Toc75257640)

[4.3 State-Transition Diagrams (STD) 11](#_Toc75257641)

[5. Change Management Process 12](#_Toc75257642)

[A. Appendices 12](#_Toc75257643)

[A.1 Appendix 1 12](#_Toc75257644)

[A.2 Appendix 2 12](#_Toc75257645)

***List of Tables***

[TABLE 1 5](#_Toc75258778)

[TABLE 2 6](#_Toc75258779)

[TABLE 3 6](#_Toc75258780)

[TABLE 4 6](#_Toc75258781)

[TABLE ***5*** 7](#_Toc75258782)

# 1. Introduction

In this day and age of digitization, our professional and non-professional lives are filled with documents whether they are assignments of a students, ads for recruitments, a proposal for business project or legally binding documents. All of these documents are being made electronically on computer or laptops. Moreover, the most important thing in all these types of documents is their format and styling (How data is presented). Although there are, tools for such work but there are no tools that can check the formatting and styling of said documents. There is need of platform where user can check document’s format by setting custom parameters to be checked in given file. System will use OCR to read file that have been uploaded.

## 1.1 Purpose

The purpose of this web base project is to provide an easy way to check format/style of documents by making a platform where users can upload their files and check their formatting/styling against one of the pre-set parameters or parameters set by themselves.

At the end, system will highlight any mistakes that do not match the given parameters if they exist or display a message that the document is properly formatted against the given parameters.

## 1.2 Scope

This system will be helpful for students, project supervisors and project coordinators of University of Lahore to check the room of error in SRS document or any university document.

## 1.3 Definitions, Acronyms, and Abbreviations

|  |  |
| --- | --- |
| **Term** | **Definition** |
| HTML | The Hypertext Markup Language or HTML is the standard markup language for documents designed to be displayed in a web browser. |
| PHP | It is a widely-used open source general-purpose scripting language that is especially suited for web development and can be embedded into HTML |
| CSS | Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML. |
| Office JS | The JavaScript API for Office enables you to create web applications that interact with the object models in Office host applications. ... The office. js library loads the object models that are applicable to the Office application that is running the add-in. |
| OCR | Optical character recognition (OCR) refers to both the technology and process of reading and converting typed, printed or handwritten characters into machine-encoded text or something that the computer can manipulate. It is a subset of image recognition. |

## 1.4 References

1. [https://www.grammarly.com](https://www.grammarly.com/)
2. Kang, N., Gelbukh, A., & Han, S. (2006, September). PPChecker: Plagiarism pattern checker in document copy detection. In *International Conference on Text, Speech and Dialogue* (pp. 661-667). Springer, Berlin, Heidelberg.
3. [Understanding the Office JavaScript API - Office Add-ins | Microsoft Docs](https://docs.microsoft.com/en-us/office/dev/add-ins/develop/understanding-the-javascript-api-for-office)
4. Eikvil, L. (1993). OCR-optical character recognition.

## 1.5 Overview

In section 2 SRS contains the functions and objectives of the system. It also includes the requirement and constraints of the system. In Section 3 SRS contains the external interface, functional, non-functional, use cases, domain, inverse, database, design constraint, and other requirements of the system. Section 4 describes the Analysis Models such as sequence diagram, Use case diagram, Dataflow diagram, etc.

# 2. General Description

This section will describe the function and objectives of the system. It also includes the requirements and constraints of the system.

## 2.1 Product Perspective

The product will be web base implemented with PHP, html, CSS, office JS and OCR. It will create a platform where user can check format of document. Website will focus on university document like SRS.

## 2.2 Product Functions

Website will allow new user to register and then after successful registration he can login. After login, the user will be in homepage of website where he will upload his file and set some parameter to be checked. It will use OCR to read file data and then after matching given parameter and file, data user will be given report of error in file*.*

## 2.3 User Characteristics

A system that has a user-friendly interface will enable user to interact with it easily. It is considered that the user does have the basic knowledge of working with the websites and have need access to the internet because it works online.

## 2.4 General Constraints

This system is provisioned to be built on WordPress using html, PHP, CSS, Office JS. MySQL will be used to store data for registered user. Only workable if your network connection is working.

## 2.5 Assumptions and Dependencies

The system will work when device is connected to stable internet connection. Every User will register himself and all his belongings will be login with valid Id Provided by Admin.

# 3. Specific Requirements

## 3.1 External Interface Requirements

### 3.1.1 User Interfaces

The user interface of this website has been specifically designed with their users in mind. User interface of the website will be very simple and easy to use so that a non-technical user can easily understand everything. User must use laptop or mobile to open a browser. He will login if he is an old user and register if he is a new user. After login, homepage of website will be shown to user. He will be asked to upload a file he wants to be checked. The he will be asked for parameters to be checked in file. Then after processing and matching result will be shown,

### 3.1.2 Hardware Interfaces

User need to have a Mobile or laptop.

### 3.1.3 Software Interfaces

To run the website user, need an internet browser and a web-browser.

### 3.1.4 Communications Interfaces

This application does require a network connection to work properly.

## 3.2 Functional Requirements

1. Register.
2. Login/ Logout.
3. File uploading.
4. Set parameters.
5. Upload parameters.
6. Show results.

### 3.2.1 Register

Users will be able to register themselves to the website by providing their email address and setting up a unique password, which will allow them to use its web-based services.

### 3.2.2 Login/Logout

After registering users will able to log-in with the e-mail and password set at the time of registration.

User will be able to log-out as well.

### 3.2.3 File Uploading

A drag-and-drop box as well as an “upload” button will allow the users to easily upload their desired file on the website and check its format.

### 3.2.4 Set Parameters

A set of format/styling options e.g. font size, font style, margin, line spacing etc. will be provided so that the user can set the desired parameters to check the format/style of their file.

### 3.2.5 Upload Parameters

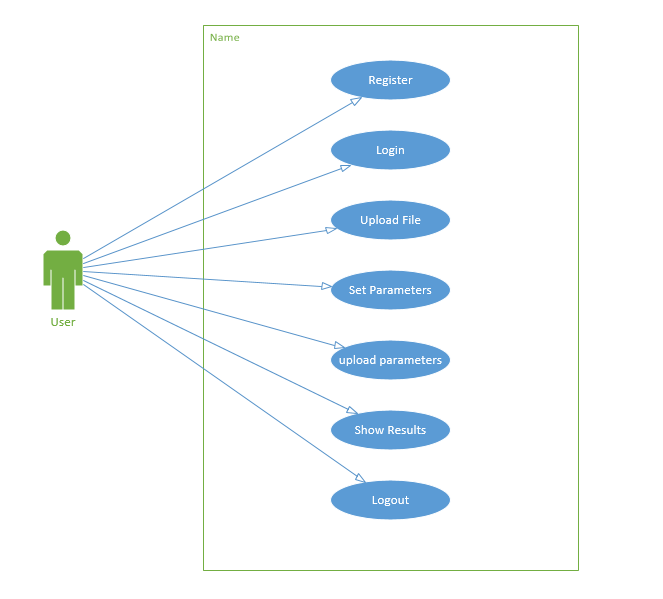
A user will be able to create a shortcut for their desired parameters and save it, thus saving them the hassle of setting up parameters each time they want to test a file.

E.g. a user such as a professor from University of Lahore can create and save a shortcut for testing the document of their students by setting the parameters according to their University guidelines and saving it as shortcut so when they want to test a file they will simply click the shortcut and the parameters will be automatically set.

### 3.2.6 Show Results

After scanning the file against set parameters the website will point out the shortcoming if any to the user. It will mark the pages where the document doesn’t comply with the set parameters and also point out which parameters aren’t a match.

## 3.3 Use Cases



### 3.3.1 Use Case #1

|  |
| --- |
| TABLE 1 |
|  |

|  |  |
| --- | --- |
| **ID** | UC01 |
| **Description** | User Registration |
| **Actors** | First time visitors/non-registered user |
| **Preconditions** | Working Internet , Web browser |
| **Basic Steps** | 1. Enter the website URL in the web browser’s search bar. 2. Click on “Register” button. 3. Provide a valid email address. 4. Set up a unique password. 5. Click on “Sign-up” button. |
| **Alternate Steps** | NILL |
| **Exceptions** | * Invalid password type( e.g. weak password) |
| **Post conditions** | * User is able to login afterwards. |

### 3.3.2 Use Case #2

|  |
| --- |
| TABLE 2 |

|  |  |
| --- | --- |
| **ID** | UC02 |
| **Description** | Login |
| **Actors** | Registered User |
| **Preconditions** | Working Internet , Web browser, Registered |
| **Basic Steps** | 1. Enter the registered e-mail address in e-mail given space. 2. Enter the set password in password given space. 3. Click “Log-in” button. |
| **Alternate Steps** | NILL |
| **Exceptions** | * Invalid Username or password. |
| **Post conditions** | * User is led to home-page. |

### 3.3.3 Use Case #3

|  |
| --- |
| Table 3 |

|  |  |
| --- | --- |
| **ID** | UC03 |
| **Description** | Upload File |
| **Actors** | Registered User |
| **Preconditions** | Working Internet , Web browser, Registered, Logged-in, |
| **Basic Steps** | 1. Click “Upload” button. 2. Select desired file. 3. Click “Ok” button. |
| **Alternate Steps** | Drag and drop the required file in the “Drag and Drop Box”. |
| **Exceptions** | * Slow Internet. * Invalid File type. |
| **Post conditions** | * File is accepted. |

### 3.3.4 Use Case #4

|  |
| --- |
| TABLE 4 |

|  |  |
| --- | --- |
| **ID** | UC04 |
| **Description** | Set Parameters |
| **Actors** | Registered User |
| **Preconditions** | Working Internet , Web browser, Registered, Logged-in |
| **Basic Steps** | 1. Select the desired parameters. 2. Set their values. 3. Click “Check” button |
| **Alternate Steps** | Select a shortcut.  Click “Check” button. |
| **Exceptions** | * Parameters not set successfully. |
| **Post conditions** | * File is scanned. * Results are shown. |

### Use Case #5

|  |
| --- |
| ***TABLE 5*** |

|  |  |
| --- | --- |
| **ID** | UC05 |
| **Description** | Upload Parameters |
| **Actors** | Registered User |
| **Preconditions** | Working Internet , Web browser, Registered, Logged-in |
| **Basic Steps** | 1. Select the desired parameters. 2. Set their values. 3. Click “Create Shortcut” button. |
| **Alternate Steps** | NILL |
| **Exceptions** | * Parameters not saved. |
| **Post conditions** | * Shortcut is saved. * User is able to use it afterwards. |

## 3.4 Non-Functional Requirements

### 3.4.1 Usability

The website shall be used friendly and does not require any guidance to be used. In other words, the website will be as simple as possible, so its users shall use it easily. Actually, the interface is quite simple and straight forward so that anyone can understand it.

### 3.4.2 Reliability

Under the given pre-conditions of have a stable internet connect and a web-browser the website service won’t experience any failure for 95% of the week.

### 3.4.3 Availability

As this website attracts users from different aspects of fields and different time zones thus the website will be made available to them 99% of the time day.

### 3.4.4 Security

The website will be secure no unauthorized person can use it. The website will have an encrypted login page and will use a secure host. The uploaded file will also be deleted as soon as the results of the file are scanned and results are shown. The uploaded file will not be kept in any form of storage once the results are shown, everything will be deleted. A filter will be used to make sure no unauthorized person can peep at the file while it is being uploaded.

### 3.4.5 Performance

All the elements for the website e.g. buttons and parameters will be made quite responsive (e.g. give response to a click in 0.001sec) to give the user a comfortable experience.

### 3.4.6 Portability

As this is a web-based service thus any device that has a stable internet connection and a web-browser can use this service.

### 3.4.7 Accurate Results

As this service’s whole focus is on accurate results, we used office.js along with OCR to maximize accuracy and minimize mistakes.

## 3.5 Inverse Requirements

Non-registered users cannot use the web-based service. Website will not accept any file type other then “.doc”. The website should not offer more than 10 input blocks. It should not take more than 2 minutes to process a document. It should not include harsh colors. It should not be static.

## 3.6 Design Constraints

This website will be developed using “WordPress”.

PHP and Office.js are used for scripting.

## 3.7 Logical Database Requirements

The System will store the account info. Of its registered users along with any shortcuts they make.

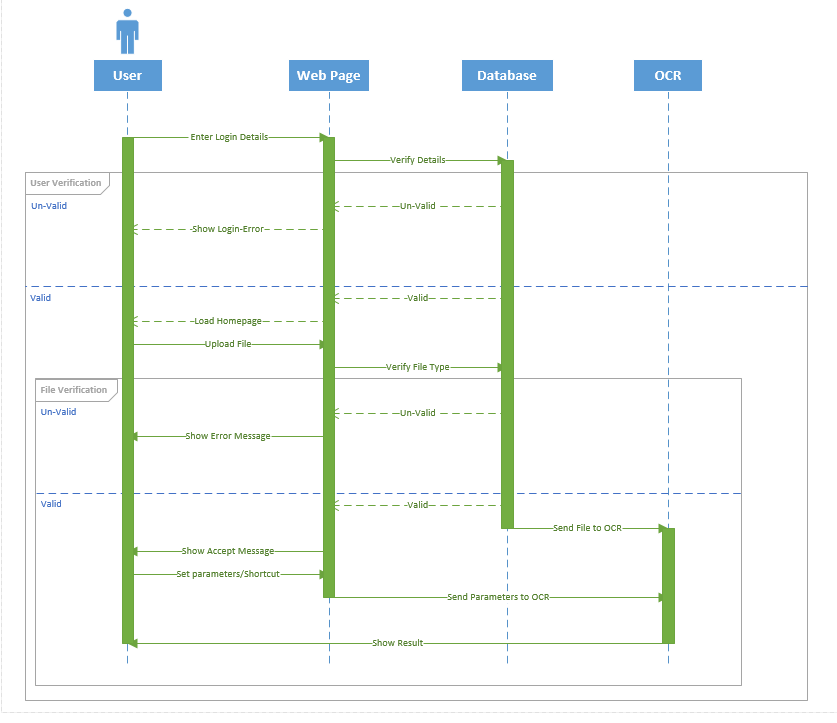
The account information (e-mail, password) will be saved in a text-based file. Account holders will be sorted numerically in the database. Each attribute will be separated by a semicolon (;). For each account there will be another file that contains the shortcuts saved by that account holder. The shortcuts will be saved a command script. The data will be stored static. The data will be protected using real time database monitoring and using firewalls. Only the developers will access the database.

## 3.8 Other Requirements

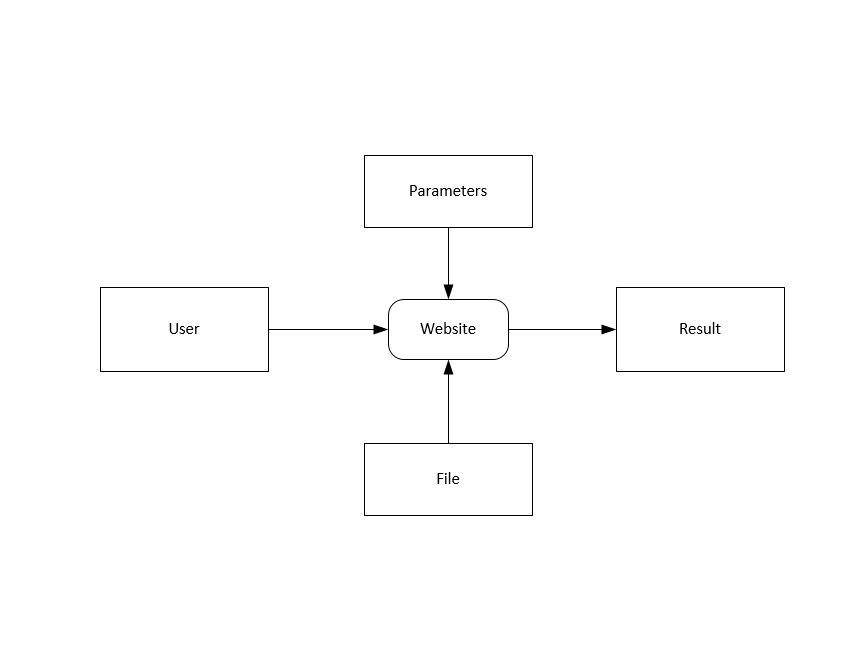
Scrum development model works on the sprints in which the changes are entertained whenever changes occur. The system should be developed under this model and the maximum time for the completion of one sprint is 3 weeks. Scrum development model involves product owner (the university), Scrum Master (team lead), and the team. The scrum master will be responsible for synchronizing work between the product owner and the team.

# 4. Analysis Models

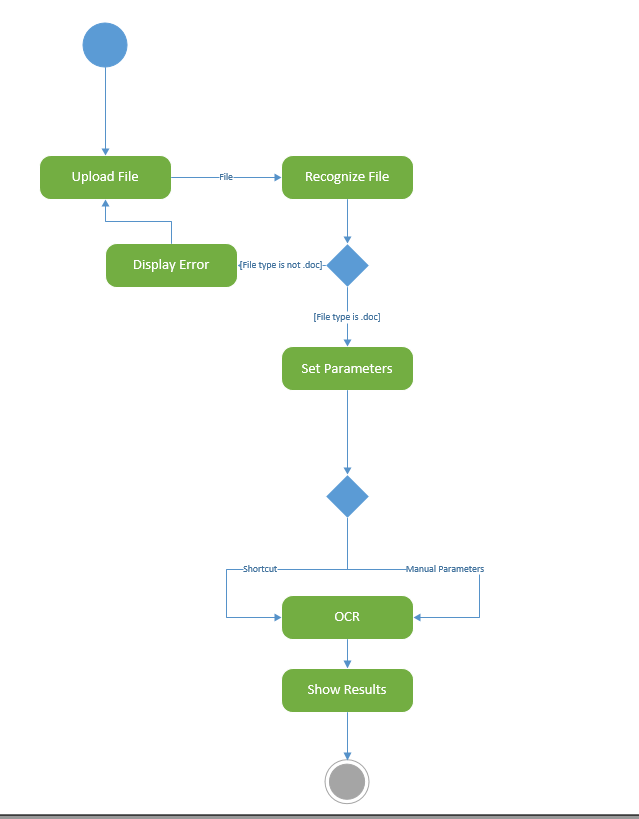
## 4.1 Sequence Diagrams



## 4.2 Data Flow Diagrams (DFD)



## 4.3 State-Transition Diagrams (STD)



# 5. Change Management Process

If any changes are authorized, then the following information will be mentioned/tracked in the second revision of the SRS.

• Who requested the change?

• What requirement is being changed?

• What requirement is being changed to?

• Who compiled the change draft?

• Who authorized the change?

# A. Appendices

## A.1 Appendix 1

1. Project Proposal was given to the FYP coordinator (Mr. Ijaz Khan Baig), approved by Supervisor (Mr. Ijaz Khan Baig), co-supervisor and the management committee.

## A.2 Appendix 2

1. All the diagrams are made in Microsoft Visio tool.