# **Software Requirement Specification**

# **Introduction**

The Software Requirements Specification is used to record and specify the customer and developer's agreement on the software product specs. Its primary purpose is to provide a concise and detailed "expression of user needs" that can be used as a guide for future software development. This article is structured into parts that logically separate the software needs into sections that may be easily accessed. This Software Requirements Specification attempts to describe the Functionality, External Interfaces, Attributes, and Design Constraints placed on the software system, which are specified throughout the rest of the document during development. Throughout the article, the terminology and vocabulary used to describe the software system should be clear and consistent.

# Purpose

The main purpose of this software requirement specification is to define the features and purpose of product recommendation system. The primary scope of this documentation is the system’s primary uses and required functionality as specified by our customer. Another one of the main purpose is that small company owner can easily grow their costumers online with little effort and also meet their needs.

# Scope

The software system or web application being created is called product recommendation system. It is being created with scope to a business owners who are intending to grow their market online. The product recommendation System is a web application which will allow users to create an account and order products through the website. On the bright side the web application will keep track of the users purchase data and will recommend product that user may be interested in the future visits. The main scope of this project is to develop an AI model that can analyze any user’s product preference and give similar movie recommendation to the user and also to help reduce time and effort for both user and web sites to give quality experience on the market. With all that said our project recommendation project is going to have some constrains such as not able to pay online via card such as credit and debit card processing. The scope is to meet the requirement of the user who wants to sell their product in the global market and provide quality experience for the user on the site as well.

## **Overall Description**

## User Needs

The web application is designed to be able to be used by anyone who has an access to a computer and internet through a web browser such as Chrome and Firefox. Beyond that there is not limitation on any hardware requirement nor does memory constrain. Some of the user needs and functionality that will be provided by Product Recommendation System are:

1. Maintain data associated with the product inventory

Product has a name, price and seller

Inventory keeps track of quantity of each product

1. Maintain records for many customers

Anyone can sign up as a new costumer

A customer has a username (unique across all users), password (no restrictions), email address (no restrictions).

Recommendation API keeps track of the user’s last buys and can recommend a new product for the user.

1. Show a listing of available products

Product will be listed in a grid format to be so that users can navigate through the products.

When the user clicks on the products the description of the product is shown to the user.

1. Allow customers and managers to log in and out of the system

The web application keeps track of all the user as well as admin who can control the product.

Can also login from any platform with just the help of web browser.

1. Shopping cart

After users pick the product we can check the product in cart page where all the selected items are listed for check out.

1. Checkout

In checkout page user can pay for the product and give delivery address.

1. Allow manager to update product quantities

This feature allows the admin to change the database of products from quantity.

1. Allow manager to change product price and quantity

## Assumptions and Dependencies

Client:

We have assumed that all of the computer systems can operate the system or web application and any user can operate the basic functionality of the system on a web browser.

Provider:

We have assumed that the Product Recommendation Project will be running on a properly working web server and database system with an Internet connection that allows this system to perform all communications with clients. Along with that the web application will have a working API which can recommend the product on user preference.

Assumptions:

• The user may not have any problems for purchasing as many products within a single transaction.

• The manager account’s username and password maybe hard coded.

• The manager can access information on databases and user details.

• Any user cannot edit other user account information.

## Design and Implementation Constrains

According to the user, privacy is not a problem with this platform. As a result, when there is a security issue, the system may not work correctly. These possibilities involve transmitting unencrypted connection and forcing consumers to use "strong" passwords, in addition to the ones mentioned above. A strong password is one that meets a set of criteria that prevents an attacker from easily guessing a user's password. In general, these requirements include making sure the password is lengthy enough and includes not only lowercase letters but also uppercase, digits, and, in certain cases, symbols. The system may not function correctly while utilizing other Internet browsers. This section also details all constrains upon the product being developed, weather customer mandated or policy/regulatory requirement.

## Preliminary Schedule and Budget

## **Features and Requirements (MoSCoW Prioritization)**

* Functional Requirements

This documentation thoroughly outlines the likely results of a software system, including the effects of program activity. All sensitive and suitable activities, such as calculations, data processing, and so on, are recorded in chronological order in this section. The following describes the functional requirements from the client’s perspective and need.

* Non-functional Requirements

This article discusses non-functional qualities required by software systems for improved performance. Some examples include security, portability, dependability, reusability, application compatibility, data integrity, scaling capacity, and so on.

* External Interface Requirements

1. User Interface
2. Hardware Interface
3. Software Interface
4. Communication Interface