

# Software Requirements Specification

Version 0.1

August 09, 2022

**Online Shopping Portal**  
**Software Testing Course**

## VERSION HISTORY

Version #	Implemented By	Revision Date	Approved By	Approval Date	Brief description of changes
0.1	Arif	09/08/2022			SRS of Online Shopping Portal-draft

## Table of Contents

<b>1. Introduction .....</b>	<b>4</b>
1.1 Purpose .....	4
1.2 Intended Audience .....	4
1.3 Product Scope .....	4
1.4 References .....	4
<b>2. Overall Description .....</b>	<b>4</b>
2.1 Interfaces .....	4
2.2 Constraints .....	4
2.3 Assumptions and Dependencies .....	4
2.4 User Documentation .....	4
2.5 List of Role .....	4
2.6 List of Business Features .....	5
<b>3. Specific Requirements .....</b>	<b>5</b>
3.1 System Features .....	5
3.1.1 Business Feature – User Login .....	5
3.1.2 Business Feature – User Registration .....	7
3.1.3 Business Feature – Forgot Password .....	9
3.2 External Interfaces .....	11
3.3 Performance Requirements .....	11
3.4 Logical database Requirements .....	11
3.5 Design and Implementation Constraints .....	11
3.6 Software Quality/System Attributes .....	11
<b>4. Use Cases .....</b>	<b>11</b>
4.1 Use Case 1 - <Name of Use case> .....	11
<b>5. Other Non-Functional Requirements .....</b>	<b>12</b>
5.1 Safety Requirements .....	12
5.2 Security requirements .....	12
<b>6. Change Management Process .....</b>	<b>12</b>
<b>7. Document Approvals .....</b>	<b>12</b>
<b>8. Other Requirements .....</b>	<b>12</b>
 <b>APPENDIX A</b>	
<b>APPENDIX B</b>	
<b>APPENDIX C</b>	

# 1. Introduction

## 1.1 Purpose

The purpose of this document is to outline all the requirements for Online Shopping Portal (SQA Shopping)

## 1.2 Intended Audience

- Developer who will make changes in the application by further developing it or fix existing bugs.
- Testers who will perform different testing on the application.
- Other relevant stakeholders

## 1.3 Product Scope

- Brief description of the software to be tested, how end user will interact with this system.

## 1.4 References

- Link of the system to be tested or any other relevant link or document can be placed.

# 2. Overall Description

## 2.1 Interfaces

- Hardware Interfaces
- Software Interfaces
- Operating Environment

## 2.2 Constraints

- Mention the constraints here

## 2.3 Assumptions and Dependencies

- Mention the assumptions and dependencies here

## 2.4 User Documentation

List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards

## 2.5 List of Role

- Normal Users
- Administrator

## 2.6 List of Business Features

SL No.	Group	Functionality	Brief Description
1	User Account	User Login	Provides User Login functionality
		User Logout	Provides User Logout functionality
		Forgot Password	Enables user to get new password in case he forgets current one.
		Change Password	Enables logged user to change current password
		Edit Profile	Enables logged user to update his profile information

## 3. Specific Requirements

### 3.1 System Features

#### 3.1.1 Business Feature – User Login

##### 3.1.1.1 Description and Priority

This functionality will help users to log into the system providing their valid credentials.

##### 3.1.1.2 Stimulus/Response Sequences

- User enters the URL in the browser to fetch the login page.
- User enters his **Email** and **Password**.
- User clicks the “Log In” Button to log into the system.
- User will be logged in to the system and navigated to the Dashboard screen.

### 3.1.1.3 Sample Screen

## SIGN IN

Hello, Welcome to your account.

Email Address \*

Password \*

☐ Remember me

[Forgot your Password?](#)

LOGIN

### 3.1.1.4 User Login – Header Data

R – Required, O – Optional

S. No	Field	Data Type	R/O*	Input Type	Possible Values
1.	User Email	Char(50)	R	Text	
2.	User Password	Char(6-20)	R	Text	
3.	Remember me	Boolean	O	Checkbox	

### 3.1.1.5 Functional/Nonfunctional Requirements

Sl. No	Requirement	Requirement Type	Dependencies
1.	REQ-1	Functional	The Email and Password Field should be visible so that user can enter his credentials.

2.	REQ-2	Functional	Forgot Password link should be available so that user can proceed to Forgot Password Functionality if he forgets his credentials.
3.	REQ-3	Functional	If Remember me is ticked then user will be remembered in the device
4.	REQ-4	Functional	Mandatory fields should be validated from front-end, Email address should be validated. Back-end validation should be available

### 3.1.2 Business Feature – User Registration

#### 3.1.2.1 Description and Priority

Using this functionality, user would be able to register in this shopping portal.

#### 3.1.2.2 Stimulus/Response Sequences

- User enters the Registration URL in the browser.
- User enters the 'Full Name'.
- User enters the 'Email address'.
- User enters the 'Contact No.'.
- User enters the 'Password'.
- User enters the 'Confirm Password'.
- User clicks on the "SIGN UP" Button.
- User should get registered and redirected to the dashboard.

#### 3.1.2.3 Sample Screen

## CREATE A NEW ACCOUNT

Create your own Shopping account.

Full Name \*

Email Address \*

Contact No. \*

Password. \*

Confirm Password. \*

SIGN UP

### 3.1.2.4 User Registration – Header Data

**R – Required, O – Optional**

SL. No	Field	Data Type	R/O*	Input Type	Possible Values
1.	Full Name	Char(50)	R	Text	
2.	Email Address	Char(50)	R	Text	
3.	Contact No.	Varchar2	R	Text	
4.	Password	Char (6,20)	R	Password	
5.	Confirm Password	Varchar2	R	Password	
6.	SIGN UP	-	-	Button	



### 3.1.2.5 Functional/Nonfunctional Requirements

SL. No	Requirement	Requirement Type	Dependencies
1.	REQ-1	Functional	The Full Name, Email Address, Contact No. fields should be visible so that user can provide the input.
2.	REQ-2	Functional	The Password & Confirm Password fields should have values shown as stars.
3.	REQ-3	Functional	Email address should be properly validated in both front-end and back-end.
4.	REQ-4	Functional	Contact No. should have 11 digits.

### 3.1.3 Business Feature – Forgot Password

#### 3.1.3.1 Description and Priority

Using this functionality, user can reset his/her password if forgotten.

#### 3.1.3.2 Stimulus/Response Sequences

- User goes to the Login URL.
- User clicks on the “Forgot your Password” link.
- User enters his Email Address.
- User enters his Contact No.
- User enters his Password.
- User enters Confirm Password
- User then clicks on the “CHANGE” Button.
- User will be able to change his password.

### 3.1.3.3 Sample Screen

#### FORGOT PASSWORD

Email Address \*

Contact no \*

Password. \*

Confirm Password. \*

CHANGE

### 3.1.3.4 Forgot Password – Header Data

R – Required, O – Optional

Sl. No	Field	Data Type	R/O*	Input Type	Possible Values
1	Email Address	Char(50)	R	Text	
2	Contact No.	Varchar2	R	Text	
3	Password	Char (6,20)	R	Password	
4	Confirm Password	Varchar2	R	Password	

### 3.1.3.5 Functional/Nonfunctional Requirements

Sl. No	Requirement	Requirement Type	Dependencies
1	REQ-1	Functional	Email Address and Contact no. must match
2	REQ-2	Functional	Email address must be validated in both front-end and back-end

### 3.2 External Interfaces

- Mention the external interfaces (if any)

### 3.3 Performance Requirements

- Mention the performance requirements

### 3.4 Logical database Requirements

This section specifies the logical requirements for any information that is to be placed into a database. This may include:

- Types of information used by various functions
- Frequency of use
- Accessing capabilities
- Data entities and their relationships
- Integrity constraints
- Data retention requirements

### 3.5 Design and Implementation Constraints

Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards

### 3.6 Software Quality/System Attributes

Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.

## 4. Use Cases

### 4.1 Use Case 1 - <Name of Use case>

#### Use Case Format

<Use Case Figure>

Use Case Name	
Actors	
Role	
Trigger	
Description	

Preconditions	
Assumption	
Goal	
Normal Flow/Use Case Steps	
Alternate Flow	
Failure	
Post Conditions	
Frequency of Use	
Business Rules/Constraints/Dependencies (Functional & Non-functional)	

**\*\* Use Case steps should be written in details**

## 5. Other Non-Functional Requirements

If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.

### 5.1 Safety Requirements

### 5.2 Security requirements

## 6. Change Management Process

*Identify the change management process to be used to identify, log, evaluate, and update the SRS to reflect changes in project scope and requirements. How are you going to control changes to the requirements. Can the customer just call up and ask for something new? Does your team have to reach consensus? How do changes to requirements get submitted to the team? Formally in writing, email or phone call?*

## 7. Document Approvals

*Identify the approvers of the SRS document. Approver name, signature, and date should be used.*

## 8. Other Requirements

Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.

## **Appendix A: Glossary**

*<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>*

## **Appendix B: Analysis Models**

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams.>*

## **Appendix C: To Be Determined List**

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*