Software Requirements Specification

for

<Wbay>

Version 1.0 approved

Prepared by <author>

<organization>

<date created>

Table of Contents

Table of Contents ii

Revision History ii

1. Introduction 1

1.1 Purpose 1

1.2 Document Conventions 1

1.3 Intended Audience and Reading Suggestions 1

1.4 Product Scope 1

1.5 References 1

2. Overall Description 2

2.1 Product Perspective 2

2.2 Product Functions 2

2.3 User Classes and Characteristics 2

2.4 Operating Environment 2

2.5 Design and Implementation Constraints 2

2.6 User Documentation 2

2.7 Assumptions and Dependencies 3

3. External Interface Requirements 3

3.1 User Interfaces 3

3.2 Hardware Interfaces 3

3.3 Software Interfaces 3

3.4 Communications Interfaces 3

4. System Features 4

4.1 System Feature 1 4

4.2 System Feature 2 (and so on) 4

5. Other Nonfunctional Requirements 4

5.1 Performance Requirements 4

5.2 Safety Requirements 5

5.3 Security Requirements 5

5.4 Software Quality Attributes 5

5.5 Business Rules 5

6. Other Requirements 5

Appendix A: Glossary 5

Appendix B: Analysis Models 5

Appendix C: To Be Determined List 6

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

This document describes the requirements of developing an online e-commerce website.

## Document Conventions

<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>

## Intended Audience and Reading Suggestions

This document is intended for the stakeholders such as:

1. Supervisor.
2. Development Team.
3. Quality Assurance Team.
4. Anyone with an interest in the Project.

## Product Scope

The scope of an e-commerce website outlines features, functionalities and boundaries the website will provide.

* Secure and user-friendly experience.
* Seamless interaction with the website for users to browse and purchase and add products.

## References

<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>

# Overall Description

## Product Perspective

This website is a new self-contained product that will serve as a platform for online retail transactions connecting vendors and customers.  
this product perspective provides a clear understanding of how the website operates in its environment.

## Product Functions

* User management.
* Product management.
* Shopping cart & checkout.
* Search & navigation.

## User Classes and Characteristics

This section will identify the user classes and their characteristics by categorizing the users based on Their roles.

**2.3.1 Customers:**

They are the general users whose primary goal is browsing products and making purchases.

**2.3.2 Vendors:**

They are the product owners who will use this website to list and sell their products, their primary goal is managing their products and inventory.

**2.3.3 Admin:**

They are responsible for overseeing the entire website, their primary goal is to manage content, vendors, users and overall website operation.

Defining these user classes and their characteristics, the website can be designed to meet the needs of all its users

## Operating Environment

<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>

## Design and Implementation Constraints

This section will outline the limitations and guidelines that must be followed during the development phase

* **Programming languages:** must use a be predefined language to ensure full compatibility
* **Security:** encrypting data such as private user information (home address, payment details)
* **User interface:** Must be compatible with major browsers such as Chrome and Firefox
* **Development constraint:** The project must user a version control system such as Git for tracking code changes and seamless team collaboration.

## Assumptions and Dependencies

This section defines the conditions that are required for the system to function correctly.

**Assumptions:**

* Internet connection.
* Up-to-date browsers.
* Product information and content accuracy.
* Database and server availability.

**Dependencies:**

* Hosting infrastructure.

# External Interface Requirements

## User Interfaces

The website will provide a user-friendly interface for various users including customers, vendors and administrators. The site will prioritize ease of use and performance to enhance user engagement

**Interface Components:**

1. **User interfaces:**

* **Login/Sign-Up Page:** This will be the main entry point for the product users.
* **Home Page:** This will be the main page to showcase products and categories.
* **Product Detail Page:** Displays detailed information about the selected product.
* **Shopping Cart:** Displays all products that the user added to their cart to potentially purchase.
* **Checkout Page:** Facilitates the order completion process.

1. **Vendor interfaces:**

* **Vendor Dashboard:** The main page for vendors to view their listed products and pending orders.
* **Product Management Page:** Gives the vendors the ability to add delete or edit information about the listed product.
* **Order Management Page:** Enables the vendors to view their orders and process them.

1. **Admin interfaces:**

* **Admin Dashboard:** Provides a comprehensive view of the website performance and user activity.
* **User Management Page:** Allows the admin to view every registered user/vendor and the ability to view and disable their account and reset their password.

## Hardware Interfaces

This section describes the infrastructure and devices needed for the website to fully operate on both server and client side. Which includes host servers, database servers, user devices.

1. **Server-Side Hardware:**

* Web Servers.
* Database Servers.

1. **Client-side Hardware:**

* Desktop.
* Laptop.

The hardware interface must ensure high performance and reliability to handle user interactions with efficiently.

## Software Interfaces

This section describes the Software Interfaces that the website will interact with which includes the Server-Side and Client-Side OS and also the Database schema. Which will ensure a seamless integration between various components to provide a comfortable experience for the user.

1. **Server-Side OS:**

* Supported OS: Windows

1. **Client-Side OS:**

* Supported Desktop OS: Windows, Linux, macOS.
* Browser Compatibility: Chrome, Firefox.

1. **Database Interfaces:**

Manage all of the website data, including user information, products and orders.

**Database Type:**

* Relational Database: MySQL.

**Communication Protocols:** SQL/SQLite queries.

## Communications Interfaces

The Communication Interface defines how the website will interact with other systems and users. These interfaces ensure a reliable data exchange.

**Network Communication Interface:**

* **Internet Protocol:** ease of data exchange between the website and users.
* **Network Ports:** allow specific traffic type to flow through the server.

# System Features

<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>

## System Feature 1

<Don’t really say “System Feature 1.” State the feature name in just a few words.>

4.1.1 Description and Priority

<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>

4.1.2 Stimulus/Response Sequences

<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>

4.1.3 Functional Requirements

<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>

<Each requirement should be uniquely identified with a sequence number or a meaningful tag of some kind.>

REQ-1:

REQ-2:

## System Feature 2 (and so on)

# Other Nonfunctional Requirements

## Performance 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.>

## Safety Requirements

<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>

## Security Requirements

<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>

## Software Quality 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.>

## Business Rules

<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>

# 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.>