Software Requirements Specification

for

SoleLegacy

Version 1.0 approved

Prepared by Charles Virden, Kevin McDonald, Aayam Raj Shakya, Conner Roberson

Group 7

August 28, 2024

1. Introduction

1.1 Purpose

<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.>

The purpose of this document is to give readers a detailed description of the requirements for our "SoleLegacy" e-commerce shoe store. It will illustrate the purpose and complete declaration for the development of the system. It will also explain system constraints and program interface. This document is primarily intended to be proposed to our customer, Mr. Abhilash Kanduri, for his approval and a reference for developing the first version of the system for the development team.

1.2 Document Conventions

1.2.1

<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.>

This SRS document uses Times New Roman as its font. Headings and subheadings are bolded and set to 18 pt and 14 pt, respectively. The text under subheadings is set to 12 pt.

1.3 Intended Audience and Reading

1.3.1

<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>

The intended audience for this document consists of the four members of our development team, as well as the teaching staff, Dr. Charan Gudla and Mr. Abhilash Kanduri. This SRS is divided into five sections, each covering different aspects of the project. The first section, which you're currently reading, covers only superficial information, providing details about our project and its scope. The second section, **Overall Description**, provides a more detailed overview of our project, covering its context, major functions, intended user classes, operating environment, and any constraints that might affect the project's design and implementation. The third section, **System Features**, outlines the functional requirements of our project based on major services or features it provides. It describes what happens when a feature is called/triggered and the corresponding system responses, as well as how the system will handle errors or invalid inputs. This section also categorizes the features according to their priority level. After this comes the **Other Nonfunctional**

Requirements section, which discusses performance requirements, safety protocols, security measures, and quality attributes such as reliability and usability. Finally, the **Other Requirements** section at the end mentions any additional requirements not covered elsewhere in the SRS.

1.4 Product Scope

<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. Relate the software to corporate goals or business strategies. If a separate vision and scope document is available, refer to it rather than duplicating its contents here.>

SoleLegacy is a user-friendly e-commerce store designed to cater to sneaker enthusiasts. It allows users to browse new sneaker releases and purchase them at affordable prices. We at SoleLegacy Inc. aim to provide an excellent user experience by offering a seamless interface and competitive pricing to attract and retain customers. We offer Nike's quality at Temu's price. Come shop with us!

1.5 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.>

Loosely based on
https://www.se.rit.edu/~co-operators/Semester2/SoftwareRequirementsSpecification.pdf

2. Overall Description

2.1 Product Perspective

<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>

The product mentioned earlier in this SRS document was a common idea to sell shoe products through our system, which will be developed through Python (backend), React.js (frontend), SQLite or Django for database management. It is currently an idea held between Group-7 members. SoleLegacy falls under Introduction to Software Engineering (CSE 4214). Hence, the availability of this product will be limited to members and the concerned teaching staff only. No further information has been supplied pertaining to the subject of copyright or plagiarism.

2.2 Product Functions

<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary (such as a bullet list) is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, is often effective.>

Functional Requirements: —

User functions:

- Create account
- Login
- Logout
- Change account info
    - Adding payment info and relative data required for successful purchases
- View and add to or remove from the cart
- Browse through the inventory
- Checkout

Seller Functions:

- Add/Remove Product to/from Inventory
- Change Product info (Name, Price)

Admin Functions:

- View and edit inventory and stock info
- Edit other users account info with reason
- Add/remove users from the system

Non-Functional requirements: —

- Label accounts to have a buyer status - added due to security requirements to prevent bots
- verification (possibly) - could be email based, number based, etc…

2.3 User Classes and Characteristics

<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important to satisfy.>

- Admin
    - Has full access to the system - (inventory management, store management, user account management, etc…)
- User
    - Can view inventory and product info
    - Add/remove items to/from their respective carts’
    - Can add items to their respective wishlists
    - Change their own account info
    - Create product reviews

2.4 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.>

Since SoleLegacy is a cloud-based shopping platform and not a downloadable application, it should run on any device and browser. For a smoother experience, we recommend that users have the latest versions of browsers they are using. During development phase, the system shall be deployed locally on a machine with following specifications:

- OS: Fedora Workstation 40
- RAM: 16 GB
- CPU: AMD Ryzen 5 6600HS Creator Edition

2.5 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 (for example, if the customer's organization will be responsible for maintaining the delivered software).>

Constraints being:

- Since we are using SQLite, which is a lightweight database, its capacity for managing data is limited.
- We are using React.js for frontend, which might limit the use of alternative front-end frameworks.
- With less than a semester left, we will not be able to provide multiple language support on our platform. We will also have a basic UI due to limited design time.

3. System Features

3.1 User Registration

3.1.1 Description and Priority

This feature allows users to create an account if they don't have one. This is essential for granting access to the website and its features.

Priority: High

3.1.2 Stimulus/Response Sequences

- Stimulus: User clicks the "register account" button
  Response: User will be redirected to a registration form where they will enter their personal information

- Stimulus: User clicks on "submit" button
  Response: User information will be stored in database and they will be logged in and taken to the homepage

3.1.3 Functional Requirements

REQ-1:The system will allow users to enter a username and email address.
REQ-2:The system will validate passwords to ensure proper strength requirements.
REQ-3:The system will check for duplicate email addresses in the database.
REQ-4:The system will not allow the user to register if the password and confirmation password do not match.
REQ-5:The system will direct the user to the login page once registration is successful.

3.2 User Login

3.2.1 Description and Priority

This feature allows the user to login to their accounts, allowing them to browse through the store, add items to their cart, and make purchases.

Priority: High

3.2.2 Stimulus/Response Sequences

- Stimulus: The user enters their login credentials and submits.
  Response: The system checks the user's input against the database. If the input is correct, the user is logged in; if the input is incorrect, the system will display an error message and ask the user to try again.

3.2.3 Functional Requirements

REQ-1: The system will allow users to login using their email address or username and their password.
REQ-2: The system will validate the user's entered information against the data stored in the database to authenticate their credentials
REQ-3: The system will display an error message if the login attempt is unsuccessful and let the user try again.

3.3 Adding Payment information

3.3.1 Description and Priority

This feature allows the user to add and manage their payment information.

Priority: High

3.3.2 Stimulus/Response Sequences

- Stimulus: The user navigates to the payment info page and enters their credit card information.
  Response: The system validates the information and stores it for future use.

3.3.3 Functional Requirements

REQ-1: The system will allow the user to add credit card information such as card number, expiration date, and CVV.
REQ-2: The system will store the payment information.
REQ-3: The system will allow the user to remove or edit the payment info.

3.4 Product Search and Browsing

3.4.1 Description and Priority

This feature allows the user to search for and browse the products in the store.

Priority: High

3.4.2 Stimulus/Response Sequences

- Stimulus: The user clicks on the search bar and enters a keyword.
  Response: The system processes the keyword and displays a list of products that match the search criteria or fall under the selected category.

3.4.3 Functional Requirements

REQ-1: The system will provide a search bar that allows users to search for products.
REQ-2: The system will display relevant products to the user based on their search criteria.
REQ-3: The system will allow users to sort search results based on price, size, and color.
REQ-4: The system will return a “No Results Found” prompt if no products in the store match the search criteria.

3.5 Items wishlist

3.5.1 Description and Priority

This feature allows the user to save products to a wishlist for future purchase. It allows the user to keep track of their favorite items that they wanted to buy but couldn’t, so they don’t have to search for them again. The user has to be logged in to the system to use this feature.

Priority: Low

3.5.2 Stimulus/Response Sequences

- Stimulus: The user clicks the “save” or “heart” button under the product information.
  Response: The system adds the product to the user’s wishlist.

- Stimulus: The user navigates to their wishlist page
  Response: The system displays all the products in the wishlist, which the user can either remove or add to the cart for purchase."

3.5.3 Functional Requirements

REQ-1: The system will provide an option for users to add products to their wishlist.
REQ-2: The system will allow users to view their wishlist.
REQ-3: The system will allow users to remove products from the wishlist.

3.6 Checkout Process

3.6.1 Description and Priority

This feature allows the user to finish their purchase by reviewing the cart, filling out shipping information, and finally confirming the payment.

Priority: High

3.6.2 Stimulus/Response Sequences

- Stimulus: The user will click the checkout button, review the order, enter shipping details, fill out payment info, and confirm the purchase.
  Response: The system will process the order information, confirm payment, and show a confirmation message with an order number.

3.6.3 Functional Requirements

REQ-1: The system will allow users to review their cart before proceeding further.
REQ-2: The system will have the user enter their shipping details.
REQ-3: The system will have the user select a payment method.
REQ-4: The system will confirm the order details with an order number.

3.7 Delete account

3.7.1 Description and Priority

This feature will allow users to close their account and delete all their personal information from the system database.

Priority: High for commercial e-stores but low in our case

3.7.2 Stimulus/Response Sequences

- Stimulus: The user navigates to the “Delete Account” section and clicks the delete button.
  Response: The system will ask for the user to enter their credentials to prevent unauthorized people from deleting others' accounts.

- Stimulus: The user enters their login credentials and other information to confirm their intent to close their account permanently.
  Response: The system processes the deletion request, shows a confirmation message, and removes the user’s details from the database.

3.7.3 Functional Requirements

REQ-1: The system will allow users to delete their account.
REQ-2: The system will ask the users for their information to prevent unauthorized deletions.
REQ-3: The system will process the deletion request and delete all the associated details from the database.
REQ-4: The system will confirm the deletion process by displaying a message.

4. Other Nonfunctional Requirements

4.1 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.>

1. The latency, or the time between the user sending requests and the requests being delivered, should be very low. We are planning to minify our codebase to achieve low latency and improve overall responsiveness.
2. Our platform should also be able to handle multiple users at a time while maintaining performance quality for all.
3. We are thinking of deploying the website using Vercel or Netlify to showcase our project beyond the class, which, if we do, we will ensure the website is mobile-optimized.

4.2 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.>

1. Login credentials are stored securely in the database, so the users need not worry about their credentials being easily accessible.
2. Non-buyers won't be able to post item reviews. This feature helps prevent paid fake reviews from bots and is based on practices used by modern website platforms like Amazon and Walmart.
3. A user's cart will be visible to him only (and possibly the admin).
4. A feature to hide important information, such as passwords and credit card details, which will be obscured by asterisks (*).

4.3 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.>

1. Every confidential and personal info will be saved in a separate database, which will be accessible to authorized users only.
2. Anti-DDoS and anti-Cross-Site-Scripting measures will also be implemented to secure our platform from various common attacks.

4.4 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.>

1. Our platform will run on any and all computers recommended by the university.
2. Our platform shall have an uptime of 90% and above.
3. We'll prioritize accessibility to ensure that our platform can be used by individuals of all kinds, shapes, and forms.
4. We're using React.js, known for its reusability and modularity, which will help us ensure both reusability and maintainability of our project.

5. 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.>

The time we have to work on this project is very limited, so not all features currently under consideration can be implemented.

Internationalization requirements: Our platform is available exclusively to English-speaking users.

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.>

DDoS - Distributed denial of service
uptime - amount of time that a website is operational; typically expressed as percentage
Vercel & Netlify - cloud platforms used for deploying and hosting modern web applications

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.>

1. Reset password feature
2. Product wishlist