Software Requirements Specification

for

eMarket

Version 1.0 approved

Prepared by Hudson Hargrove, Matthew Crosno,

Andrea Ambrose, Jeremiah Ickom

Department of Computer Science, Mississippi State University

26 August 2024

1. Introduction

1.1 Purpose

This Software Specifications and Requirements (SRS) document provides the requirements specification for all implemented systems featured in the release of eMarket Version 1.0, an online

shopping platform built on principles of minimalism and simplicity. The purpose of this document is to define all software features, functions, and nonfunctional requirements associated with Version 1.0, developed in accordance with the stakeholders' preliminary high-level requirements as defined in Section 2. The document pertains only to systems and subsystems for Version 1.0, any subsequent releases will include updated SRS documents.

1.2 Document Conventions

This document uses a tiered section format, with primary section titles written in bold, size eighteen font with a numerical format that follows the scheme 'X. Title', where X represents the section number, and Title represents the title of the section. Subsections are conveyed using the 'X.Y Title' scheme, where X represents the section number, Y represents the subsection index, and Title represents the subsection title. Unless otherwise stated, all detailed requirements inherit the priority of the encompassing high-level requirements. All acronyms are defined in Appendix A: Glossary.

1.3 Intended Audience and Reading

The SRS is intended to serve as a guide for developers, users, testers, and documentation writers. While each subgroup of the intended audience may find unique value in the information contained herein, it is suggested that all readers begin by becoming familiar with 'Section 2: Overall Description', which provides a high-level overview of the software release and its functions, classes, and constraints. Users, and documentation writers will also find the information contained in 'Section 3: System Features' to be helpful for understanding the capabilities and features available in eMarket, Version 1.0. Developers should consult Sections 3, 4, and 5 as a guide to workflow formation, as these sections provide insight into the specified system features and requirements. Those tasked with testing the software should read Sections 3, 4, and 5 to develop test strategies that ensure all requirements are satisfied for each feature.

1.4 Product Scope

eMarket is an online shopping app designed around the principles of minimalism and user-centric simplicity. Unlike sprawling e-commerce platforms that overwhelm users with ads, endless widgets, and relentless upselling, eMarket offers a refreshing alternative focused on ease of use and genuine engagement.

The app features a clean, intuitive interface that makes navigation effortless. Instead of presenting users with an overwhelming array of choices, eMarket provides a curated selection of high-quality products, streamlining the decision-making process. This curated approach ensures that users find what they need quickly and confidently.

In line with its minimalist ethos, eMarket delivers an ad-free experience, eliminating distractions and keeping the focus on the products themselves. Personalized recommendations are based on user preferences, avoiding manipulative tactics often seen in other platforms.

Privacy is a cornerstone of eMarket's design. The app limits data collection and maintains transparency about data usage, empowering users with control over their information. Additionally, eMarket emphasizes ethical shopping by featuring products from responsible brands, allowing users to make conscientious choices.

By prioritizing a minimalist, user-first approach, eMarket sets itself apart from current e-commerce giants. It represents a thoughtful evolution in online shopping, focusing on user satisfaction and well-being rather than engagement metrics and commercialism (ChatGPT).

1.5 References

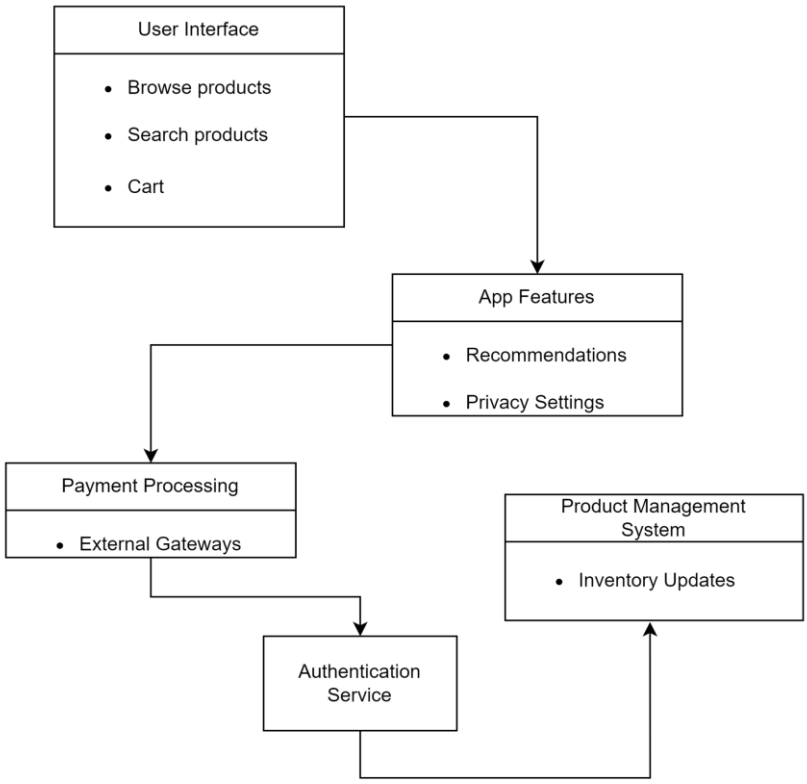
“ChatGPT.” OpenAI, 26 Aug. 2024, chat.openai.com.
“Nonfunctional Requirements: Examples, Types and Approaches.” AltexSoft, 30 December 2023, https://www.altexsoft.com/blog/non-functional-requirements/. Accessed 2 September 2024.

“PCI Quick Reference Guide.” PCI Security Standards Council, 2008, https://listings.pcisecuritystandards.org/documents/pci_ssc_quick_guide.pdf. Accessed 2 September 2024.

2. Overall Description

2.1 Product Perspective

eMarket is a new independent application providing a different approach than the current options of e-commerce platforms. It will be centered around a more secure, simplified, and privacy-oriented approach to shopping online.



2.2 Product Functions

eMarket will have many functions to properly satisfy customer needs and improve overall experience using the application. Below is a description of those functions to be used.

- **Authentication**
  - Enables users to create accounts and manage them.
  - The function should verify the credentials of the user before giving access to the account.

- **Browsing and Search**
  - Allow user to browse products
  - Have filters and sorting options when searching for products

- **Shopping Cart**
  - Allow user to add and delete products from cart
  - Have a checkout process with options on payment and shipping

- **Order Management**
  - Allow user to view old orders
- **Privacy**
  - Show data usage and give access to data management tools
  - Allow user to customize privacy settings and sharing of data

- **Ethical Shopping**
    - Provides information on products history
    - Highlights more high rated products

- **Customer Support**
    - Offer support resources

2.3 User Classes and Characteristics

eMarket will have a variety of different classes to ensure the comfortability of the user is a consumer or some kind of administrator.

- **General User**
    - For regular users to use for personal shopping

  Characteristics:
    - The most frequently used
    - Functions Used: Browsing and Search, Recommendations, Shopping Cart, Order Management
    - General users varying from low to intermediate technical skills
    - Will have standard access to account management and transaction history
    - No specific educational background required

- **Registered User**
    - For users with a created account for more personalization

  Characteristics:
    - Uses all function used by General User plus more functionality
    - May require more tech experience for more advanced features
    - High level privilege than General User
    - Has more detailed personal data and order history management

- **Administrator**
    - This is for Users than manage everything within the application

  Characteristics:
    - Regular use with access to management tools
    - Has access to manage products, app settings, and user accounts
    - Expected to have a high technical background to manage backend systems and be able to solve issues
    - Has access to all systems including user data and configuration settings
    - Has expertise in application or system administration
- **Customer Support**
    - For users that assist customers with any issues

  Characteristics:
    - Has access to user accounts, order history, and support tools
    - Limited access to admin functions
    - Should have some experience with troubleshooting and application features

- Only allow to view and assist users with accounts with limited admin control
- **Suppliers**
  - External user that manages inventory **Characteristics:**
  - Access is limited to only product management features
  - Individuals that has experience with inventory management
  - A more basic to intermediate technical expertise
- **Most Important Users**
  - **General User:** Main class to contribute to application success
  - **Registered User:** Needed to make personalized experience better; will influence application functionality
- **Less Important**
  - **Administrators:** meant for managing the app and its systems from a admin's perspective
  - **Customer Support:** to provide assistance and keep customer satisfied
  - **Suppliers:** a more specialized role to update product information

2.4 Operating Environment

eMarket is an online shopping site to work with many environments to be available to as many users as possible. The following will be a description of the hardware, operating systems, and software components that must be used.

- **Hardware Platform**
  - **Client Devices:**
    - **Desktops/Laptops:** The user can access eMarket from any standard desktop or laptop computer with any typical web browser.

      Specification:

- Processor: Intel Core i3 or equivalent
- RAM: 4 GB minimum
- Storage: enough for browsing and caching
- Display: 1366x 768 resolution or higher

    - **Mobile Devices:** eMarket will also be available to be accessed through smartphones and tablets.

- Smartphones: iOS(iPhone 5 or later) and Android (version 9.0 or later)
- Tablets: iOS (iPad 5 gen or later) and Android (version 9.0 or later)

  - **Server Structure:**
    - **Web Servers:** The application will be hosted on a local server
      Specifications:
      - Processor: Multi-core CPUs
      - RAM: 4 GB or enough to support capacity of application
      - Storage: HDDs or SSDs

- Network: sufficient high-speed internet

- **Operating System ■**

Client Side:

- Windows: Windows 10 or later
- macOS: macOS Mojave (10.14) or later
- Linux: Ubuntu 18.04 or later
- ChromeOS: 126.0.6478.252

- **Software Components ■**

Web Browsers:

- Google Chrome: version 89 or later
- Mozilla Firefox: version 89 or later
- Safari: version 14 or later (macOS)
- Microsoft Edge: version 89 or later ■ **Backend:**
- Web Server Software: HTTP Server
- Application Server: Python
- Frameworks: Django
- Database Management System: MySQL for storing and managing application data

■ **Frontend:**

- Design Language: CSS ● Structure Language: HTML

2.5 Design and Implementation Constraints

When developing eMarket, there are several constraints and limitations that must be put into place. The constraints may impact many parts of the software.

- **Corporate Policies:**
  - must follow policies related to ethical practices

- **Hardware Limitations**

Client Devices:

- the application must work efficiently on low and high end devices
- must be usable on varying device screen sizes and resolutions

Server Requirements:

- must be able to handle various loads of user traffic
- Server must have sufficient resources allocated for data storage

- **Technologies, Tools, and Databases**
  Technology Stack:
  - Must use specific frontend frameworks

- Must use certain databases or backend technologies

Development Tools:

- Must user version control systems such as Git for collaboration and management
- For further development GitHub will be used to keep continuous workflow

3. System Features

3.1 Account Management

3.1.1 Description and Priority

Display a login window that contains a box for a user to input both a username or email as well as a password. Provide functionality for the user to create a new account if one does not exist. Additionally, the user should be able to see information about their account, such as their account balance, listed products, and recent orders. Finally, administrators should be able to accept or deny new account creation.

Priority: High

3.1.2 Stimulus/Response Sequences

Trigger	Response
User inputs login information	If user account exists, continue with sign in
User clicks create account button	Display required information, obtain login credentials, store for account approval/denial
User clicks sign out button	Sign out of user account

3.1.3 Functional Requirements

USER-LOGIN: Users must be able to have a displayed section to input user credentials to login to the system or to create an account to register with the system.

Error	Response

User inputs incorrect credentials	Notify user with popup, deny access
User attempts to create account with duplicate credentials	Notify user with popup, deny access

3.2 Main Product Display

3.2.1 Description and Priority

Products must be displayed in organized sections for the user to browse through. When items are displayed in a list, show necessary information that can be seen without clicking on the product such as its name and price.

Priority: High

3.2.2 Stimulus/Response Sequences

Trigger	Response
User clicks “home” button (design and position TBD)	Redirect user to main display page
User clicks category button	Update product display to match selected category
User selects filter(s) of price, date added, etc.	Update product display to match selected filter(s)

3.2.3 Functional Requirements

CATEGORIES: A menu containing various categories must be available for the user to select from. When a category is selected, the product display page must update and reflect the selection

TBD: Menu design and category amounts

Error	Response

User selects a category that does not match any currently available products	Notify the user with a message or popup that informs them that no products are currently available in that category

FILTERS: In addition to categories, filters need to be available for the user to select from. When a filter or multiple filters are selected, the product page needs to be updated to show the changes.

TBD: Filter types; can include but not limited to price, date added, and color.

Error	Response
User selects filters that do not match any currently available products	Notify the user with a message or popup that informs them that no products are currently available with the selected filters.
User selects multiple filters with conflicting criteria (e.g., selecting both a red color filter and a blue color filter).	Display products that meet at least one of the conflicting criteria as well as any additional selected filters.

DISPLAY: Products must be displayed in a neat grid or list and include basic information about name, price, etc.

TBD: Size of product, organization of list/grid

3.3 Product Sort and Search

3.3.1 Description and Priority

Provide basic tools to sort products by certain criteria and search for a specific product or type of product.

Priority: Medium

3.3.2 Stimulus/Response Sequences

Trigger	Response

User clicks on sort dropdown menu	Display different options for sorting (name, price, date added, etc.)
User selects new sort style	Display updates items in desired order
User searches for item	Display updates and shows items that include inputted text

3.3.3 Functional Requirements

SORT: Users must be able to have a displayed menu to select how to arrange products on the main display. Additionally, when a user selects a different sorting option, the display must update to reflect the changes.

TBD: Design of sort menu (dropdown, list in category menu, etc.)

Error	Response
Two products share the same value for a given criterion	Program the sorting algorithm to handle scenarios like this efficiently

SEARCH: Users must be able to input text that filters the displayed products accordingly. At minimum, should search for text in the product's name.

TBD: Provide other methods of search (e.g., if the user searches "phone", and the product name is "Samsung Galaxy S24", it should still pop up). One way this could be completed is to allow the seller to include keywords when creating the listing of the product, and the search algorithm also cross-checks any keywords that match.

Error	Response
No products exist within search	Display a message notifying the user and informing them that no products exist within the search criteria.

3.4 Individual Product Display

3.4.1 Description and Priority

When the user clicks on a product in the main display, redirect them to a webpage showing more information.

Priority: High

3.4.2 Stimulus/Response Sequences

Trigger	Response
User clicks on product in main display	Redirect user to individual product webpage

3.4.3 Functional Requirements

DISPLAY: Display product information (title, image(s), description, specifications, etc.)

Error	Response
Seller does not include some information	If needed, provide generic text. (“Description not provided.”)

3.5 Shopping Cart

3.5.1 Description and Priority

When the user clicks a provided button on an individual product display screen, add the product to the shopping cart, which stores a list of all the products you have added to the cart.

Priority: Medium

3.5.2 Stimulus/Response Sequences

Trigger	Response
User clicks on shopping cart icon	Redirect user to shopping cart screen

3.5.3 Functional Requirements

ADD: Add item to shopping cart through individual product page.

VIEW: View shopping cart through an icon visible throughout the website.

TBD: In addition to showing basic information about each product in the cart similar to the main product display page, show a sum of the prices as well as a count of the number of items presently in the cart.

Error	Response
No items added to shopping cart	Provide a message on the screen letting the user know the cart is empty

3.6 Purchases and Payment

3.6.1 Description and Priority

When the user is in the shopping cart, provide a button that initiates the purchase sequence. In the purchasing sequence, ask the user to input their information. Once inputted, charge the user the correct amount.

Priority: High

3.6.2 Stimulus/Response Sequences

Trigger	Response
User clicks on payment button	Redirect user to payment screen
User inputs payment information	Money is taken from buyer and given to seller

3.6.3 Functional Requirements

INFO: Click purchase button and input user information

Error	Response
No information inputted	Deny access to next step, ask the user to input information again
When user clicks button, product has already been purchased or product is out of stock	Notify user that the product is no longer available and either remove it from the shopping cart or notify the user to try again later

PAY: After buyer inputs information, pull money from buyer and give to seller

TBD: For the purpose of the project, the buyer will have an “unlimited” supply of money, and the seller can view their money through account management

Error	Response
Seller's account cannot be located	Cancel payment, notify user to try again

3.7 Returns and Recent Orders

3.7.1 Description and Priority

If the buyer decides that they no longer want a product, they should be able to view their recent orders and return them

Priority: Medium

3.7.2 Stimulus/Response Sequences

Trigger	Response
User clicks on recent orders button in account menu	Display recent orders that the user made, possibly up to a certain date or number
User clicks on button to return product	Refund purchase (possibly only allow up to a certain date)

3.7.3 Functional Requirements

VIEW: View recent orders in a list-like format similar to the main product display page

Error	Response
User has not made any previous orders	Provide a message on the screen letting the user know that previous orders will appear there

RETURN: Return previous orders, refund money to buyer, and return product to seller

TBD: If, for the purpose of the project, the buyer has an unlimited balance, then refunding is unnecessary

4. Other Nonfunctional Requirements

4.1 Performance Requirements

- The app will run on the user's local web server.
- Product search results should be displayed in under 5 seconds under normal conditions.
- Clicking any link should take under 3 seconds.
- Updating and refreshing the page should take under 3 seconds.
- New information should be saved and displayed in under 3 seconds.

4.2 Safety Requirements

- No one except for the user and permitted developers should know the user's login information.
- The system will not share confidential user information (passwords, payment details, etc.) with third parties.

4.3 Security Requirements

- The user's password should be hidden while typing in the web browser.
- Card information should never be fully displayed by the app on the web browser.
- User accounts and corresponding user data will be deleted upon request.
- Databases containing confidential user information should only be accessed by permitted developers.
- Software will be consistently updated to prevent exploitation and patch vulnerabilities.

4.4 Software Quality Attributes

4.4.1 Availability

- The app should be accessible 99.9% of the time.
- The app should consistently return correct information upon request. **4.4.2 Maintainability**
- Software issues will be fixed with new updates. ● New updates should take effect immediately.

4.4.3 Portability

- The app will be fully functional on major web browsers such as Microsoft Edge, Firefox, and Google Chrome.
- The app is capable of running on all operating systems.
- The app should adapt to maintain user experience across different devices such as mobile phones and laptops.

5. Other Requirements

Appendix A: Glossary

SRS

- System Requirements and Specifications - Document that defines all system features, functions, nonfunctional requirements, and classes in order to convey the objectives of the software and the manner in which the software will satisfy those objectives.

PCI DSS

- Payment Card Industry Data Security Standard - a set of policies and procedures to any entity or company that processes and stores credit/debit card data.