Software Requirements Specification

for

Knockoff King

Version 1.0 approved

Prepared by Group 6.0

Mississippi State University

January 31, 2023

1. Introduction

1.1 Purpose

The purpose of our website project is to provide the ability to shop online, regulate users & products as well as manage current and previous orders. This SRS covers the general overall intent for (project name) and outlines our primary features such as the shopping cart and product catalog.

1.2 Document Conventions

For this document, the Times font was used throughout. For the title page, Times pt. 24 was used and in Bold. For titles, Times pt. 18 was used and in Bold. For subtitles, Times font pt. 14 was used and in Bold. For body text, Times pt. 12 was used. Important terms were italicized or underlined.

1.3 Intended Audience and Reading

This document is intended for the project manager. The reader should start with the overall description, and then move on to the system features. The features are organized by importance, so if seeking the critical features, the features list should be read in order.

1.4 Product Scope

Our website is designed for a seamless shopping experience, while incorporating critical actions for proper management of an ecommerce site. Admins will be able to regulate both users and products sold, and sellers able to list and sell allowed goods. Our primary goal is user experience, with an aim to create a smooth, visually appealing experience.

1.5 References

- *https://www.djangoproject.com/*
- *https://getbootstrap.com/*
- *https://developer.mozilla.org/en-US/docs/Web/HTML*

2. Overall Description

2.1 Product Perspective

Our ecommerce site Knockoff King is based on our aspirations to build a more ethical online market to rival existing sites. It is a part of the growing trend moving away from the traditional brick and mortar stores to an online platform. Knockoff King is aimed at helping small business owners maintain profitability by giving their employees new freedom and transitioning to an online platform.

The frontend will be designed with Bootstrap which incorporates HTML, JavaScript, and CSS. Frontend design will be the user interface that buyers, sellers and admins interact with when navigating/using Knockoff King. By contrast, the backend will be built with Python based Django, which will bridge the connection between HTML and our SQL database.

2.2 Product Functions

- Account Creation and Log-in
- Add, Delete, Save for Later, or Checkout Items in a Shopping Cart
- Search, Compare, Buy, and Return products
- Sellers will be able to add, sell, and receive payments for their products
- Admins will be able to oversee various functions

2.3 User Classes and Characteristics

The program will include three major user classes: Customers, Sellers, and Admins.

Customers are expected to have the least technical expertise, so their interface should be simple and well organized to facilitate universal user friendliness. Their permissions will be limited to logging in/out, creating and/or deleting their account, adding and/or removing items to their cart, and checking out (purchasing the items in their cart). The customers will be the most important user to satisfy, as if they are not satisfied, they will not use the site, which in turn will drive away sellers.

Sellers are the next most important user to satisfy, as without their products there is nothing for a customer to shop for. They will have the same basic permissions as customers (logging in/out, creating/deleting accounts), and they can also add/remove new items to their catalogue and edit their prices and stock. They will be able to receive payments from customers who make a purchase. A percentage of their sales will go to the owner of the program. If a seller deletes their account, their catalogue and all their products will be removed from the site immediately, and any pending transactions involving their products will be cancelled.

Admins will maintain order on the site. They will have the basic abilities of other users (creating/deleting accounts, logging in/out), but they will also be able to manage Customer and Seller accounts. They will be able to suspend all actions regarding any given Customer/Seller account, remove any product from the website, and delete any Customer/Seller account. Admins are expected to have a strong understanding of the interface and managing the database.

2.4 Operating Environment

The program will operate on Windows 10/11, MacBook, and Linux browsers, and should be able to cohabitate with any other website.

2.5 Design and Implementation Constraints

Design:
- Language: Python3 with Django, HTML, CSS
- Database: SQLite3
- Server: TBD

Constraints:
- Must run on the latest version of Google Chrome and Firefox
- Processing requirement: as little processing as possible

3. System Features

3.1 Account Creation / Log-In

3.1.1 Description and Priority

Account creation and login is of high importance as it will distinguish each individual user and their abilities on the site. Unauthorized access to admin accounts could cause considerable damage to the site's integrity.

3.1.2 Stimulus/Response Sequences

Enter the URL for the site and be greeted by a welcome page. Click on “login” and be brought to a new webpage with a username and password box. Once the user has entered their information and chosen “login” they are redirected back to the homepage.

3.1.3 Functional Requirements

- REQ-1: Ability to securely login
- REQ-2: Ability to log out of personal account
- REQ-3: Access and edit information in bio
- REQ-4: Account creation and deletion

3.2 Shopping Cart

3.2.1 Description and Priority

The shopping cart and all requirements/functions involving it are of high priority as it is the core of the product. The shopping cart will provide the user with an efficient way to purchase items.

3.2.2 Stimulus/Response Sequences

The user will be able to select a shopping cart button which will direct them to the shopping cart page. On the shopping cart page, each item currently in the user's shopping cart will be listed with a 'remove' button for each item, which will delete that item from the shopping cart. At the bottom of the page, a checkout button will direct the user to provide payment. Once payment has successfully been provided, the shopping cart will be cleared, and inventory will be updated.

3.2.3 Functional Requirements

- REQ-1: Ability to add an item to a shopping cart
- REQ-2: Ability to delete an item from a shopping cart
- REQ-3: Ability to add an item to a save-for-later menu
- REQ-4: Ability to check-out all items currently in the shopping cart

3.3 Product Catalog

3.3.1 Description and Priority

The product catalog is of medium-high importance. A functional catalogue is all that is required for proper website operation, but a visually appealing and easy to navigate catalogue will improve the user experience and can increase the customer traffic, and subsequently profit, for the website.

3.3.2 Stimulus/Response Sequences

The product catalogue will be navigated by customers.

User input to view a list of catalogs will make the program show all company catalogues. The user can then select a company catalog, and that catalog's products will be displayed. The user can then select a product, and they will be taken to the product page. From this page, the user can add the item to their cart (they can also select quantity). On all pages, the user has an option to return to the previous page.

3.3.3 Functional Requirements

- REQ-1: The software must be able to handle and display to customers when a product is out of stock.
- REQ-2: The software must be able to handle and/or prevent customers from attempting to purchase more items than are in stock.

3.4 Seller Menu

3.4.1 Description and Priority

The seller menu is of medium-high importance. Like the user catalogue, it only needs to be functional, but it should also be visually appealing and easy to use, as that will attract more sellers, which will improve the operation of the site.

3.4.2 Stimulus/Response Sequences

The seller menu will be navigated by sellers.

The user will have an interface with options for viewing their catalogue and editing their catalogue. If the user chooses to view, a preview of their catalogue page is displayed. If the user chooses to edit, they are taken to another page with options to add products, remove products, or edit product information.

If the user chooses to add a product, they are taken to a page where they will fill out the product information. This information includes the product name (required), the product model (required), a picture of the product (required), a brief description of the product (required), a product price (required), the number of the product in stock (required), the location the product was manufactured in (optional), and the product's warranty information (if applicable, optional).

If the user chooses to remove a product, they are taken to a list of their products, where they can choose which products to remove from their lineup. When they confirm the decision, the products are removed from the catalogue immediately.

If the user chooses to edit a product, they are taken to a list of their products to choose which product to edit. Once they choose a product, they are taken to a page much like the new product page wherein they can edit all the attributes of the product, including price and stock.

3.4.3 Functional Requirements

REQ-1: The software must be able to prevent sellers from posting a product without filling out all required fields (loop back to the page and display an error message).
REQ-2: The software must be able to handle seller attempts to set stock or price of a product to a negative number; such attempts should make the system loop back to the edit product or new product page and display the appropriate error message.

3.5 Admin Menu

3.5.1 Description and Priority

The admin menu is classified as a medium priority. Admins will have the abilities outlined in the functional requirements section. Admins will be able to provide the site with a variety of actions, such as blocking user accounts and removing products from the catalog.

3.5.2 Stimulus/Response Sequences

An admin will be able to view a list of user accounts, and will have the ability to block an unwanted account by selecting a 'block' button. They will also be able to remove a product from the catalog by selecting a 'remove' button.

3.5.3 Functional Requirements

- REQ-1: Ability to handle an admin blocking user accounts.
- REQ-2: Ability to handle an admin removing a product from the catalog.
- REQ-3: Ability to handle an admin overseeing a user's actions (i.e., adding to shopping cart and checking out).

3.6 Storage

3.6.1 Description and Priority

Storage of information needed for the website to function is vital, so it is considered a high priority. Databases will store data such as user information and shopping cart information.

3.6.2 Stimulus/Response Sequences

Users will need to be able to access their cart after they leave in order to streamline the experience, and that will need to be stored. Upon use of "save for later," for example, the user might receive a message that their cart has been stored.

3.6.3 Functional Requirements

- REQ-1: Storage of user information in a database
- REQ-2: Storage of shopping cart information in a database
- REQ-3: Storage of products/inventory in a database

4. Other Nonfunctional Requirements

4.1 Performance Requirements

The website should perform quickly. Buyers and Sellers should be able to move to new pages in under one second. More complex Admin commands such as deletion or suspension of an account can take more time.

4.2 Safety Requirements

The website must be able to prevent fraudulent purchases and products. This will be done by confirming the company's existence for sellers and checking the credit card databases for buyers. The credit card checks will confirm that cards A) exist and B) are not stolen.

4.3 Security Requirements

The website must have DDOS protection.

The website must handle sensitive user data carefully to prevent leaks.

There will be a user authentication page for confirming user identities and mitigating unauthorized site access. Admins MUST use a two-factor authentication to access the site, as an unauthorized user with admin permissions can be catastrophic to the website.

4.4 Software Quality Attributes

The site should be easy to maintain for the owner, which in turn will allow it to be available at more times (ideally all times). This will increase profit for the owner via increased user traffic. The site must also be correct in its operations, to prevent transactions from being carried out incorrectly (i.e., ordering one item and receiving another).

5. Other Requirements

The ecommerce site must store its data in a database and must access the database correctly to correctly respond to user inputs and actions/transactions.

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.>