Software Requirements Specification

for

E-Commerce Store

Version 1.0 approved

Prepared by Granville Alpha  
Kevin Lopez  
Evertte Berkyaw  
Gavin Seiler

Intro to Software Engineering

February 4th, 2025

# Introduction

## 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.>

## 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

This document is intended for the developers of software development team and the project stakeholder (i.e. the T.A for the CSE 4214/6214 course). The SRS contains the requirements for an e-commerce webstore. The document is organized as follows: Section 1 introduces the software project and the SRS document. Section 2 denotes the overall description of the software project. Section 3 describes the features the software system to be produced will include. Section 4 denotes the non-functional requirements the software system should meet. It is recommended that this document is read from start to finish. Developers who already understand the project’s scope may begin reading Section 3 of the SRS to find intended features of the software project.

## Product Scope

The software project aims to create an e-commerce webstore for the client organization. The webstore will allow customers to purchase items from the webstore and vendors to manage the item inventory of the webstore. Furthermore, system administrators from the webstore should be able to login to the webstore, manage user accounts and products. The business hosting e-commerce benefits from this software project because they would be unable generate revenue without a platform to sell products on. Consequently, this software project will represent a mission critical asset to the client as a conduit for the client organization’s online commerce.

## References

Python Django: <https://docs.djangoproject.com/en/5.1/>

<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

<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.>

## Product Functions

* All users (buyers, sellers, and admins) will be able to access and login into the website via a graphical user interface.
* Users, subject to admin approval, should be able to make accounts.
* Buyers will be able to browse items for sale on the website
* With the exception of a buyer’s ability to browse items, users must authenticate the system to perform all actions related to their respective user classes.
* Buyers should be able to add items to a shopping cart and purchase those items.
* Authenticated sellers must be able to manage an inventory of items up for sale.
* Authenticated administrators should be to approve and manage buyers, sellers, and products.
* The product must be able to store user information and product information in a mySQL database.
* The produce must store sensitive information securely (i.e. via hashing).

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.>

## 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.>

## Operating Environment

The software will be accessible via modern web browsers, for example, Chrome, Safari, or Firefox. It will be hosted on a Linux-based web server. The database that will be used is PostgreSQL while using Python 3.x with Django

## Design and Implementation Constraints

<Describe any items or issues that will limit the options available to the developers.

Design: The project will utilize Python with Django version 4.1

The website will use the terms-of-service along with a pop-up to ensure all users are from the United States. This is due to accounting for the regulatory posturing of other countries not being within the scope of the project as defined by the client.

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).>

# 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.>

## Buyer Functionality

3.1.1 Description and Priority

Enable buyers to search for, compare, purchase, and return products.

Priority: High

3.1.2 Stimulus/Response Sequences

* Stimulus: A buyer enters a search query.

Response: The system returns a list of matching products.

* Stimulus: A buyer selects products to compare

Response: The system displays the product details side by side

* Stimulus: A buyer adds a product to the shopping cart.

Response: The system updates the cart and displays the current subtotal.

* Stimulus: A buyer initiates checkout.

Response: The system processes payment via the payment gateway and confirms the order.

* Stimulus: A buyer initiates a return.

Response: The system allows the Buyer to submit a return request for the specified order.

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.>

# 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.>