Software Requirements Specification

for

Doodle & Stick

Version 1.0 approved

Prepared by

Group Name: Sleepy Dinos

Gracie Breazeale	**903367171**	**egb208@msstate.edu**
Alexander Yates	**902779886**	**ahy16@msstate.edu**
Hayden Hunt	**903189621**	**hah379@msstate.edu**

Instructor: Nisha Pillai

Course: CSE-4214

Lab Section: Monday 3:30-4:45

Teaching Assistant: Tirian Judy

Date: August 25, 2025

1. Introduction

1.1 Purpose

The purpose of this SRS document is to house all of the requirements of our e-commerce website that the user may need. The SRS document will describe every aspect of the system, including the backend processes needed for the website to function correctly. The document also serves as a loose forecast of what requirements will be needed, but also of ideas that may not be used in the final version. In summary, the purpose of the document is to give an in-depth overview of our e-commerce website, the parameters and overarching goals of our system, as well as our intended use cases for the product. This document also defines any system requirements, user-interface design, and how our team views Doodle & Stick as a whole.

1.2 Document Conventions

In the following document, if red text is observed, then the topic of discussion is likely important. Any highlighted text should be considered high priority and should be accomplished quickly. If there is no significant marking on a piece of text, it should be considered a normal priority.

1.3 Intended Audience and Reading

The primary audience is the faculty and the teaching assistant for CSE 4214. It is also intended for our project group, who will use it as a reference during design and development. In addition, the document may be useful to anyone interested in understanding the goals, features, and technical details of the Doodle & Stick project.

The SRS is organized into sections that outline a short explanation of the project's purpose and design, describe the overall system, and specify the detailed requirements. Since the platform supports Buyers, Sellers, and Admins with distinct features, the requirements section is significant for clarifying their functionalities. Readers should begin with the introduction and overall description, then focus on the sections most relevant to what they want explained.

1.4 Product Scope

Doodle & Stick is an e-commerce website that sells artwork from local artists. Doodle & Stick's goal is to showcase art from small artists online, giving sellers an audience for their talent. Doodle & Stick will also let the user view the "Doodle Den," which will feature special/limited artwork on the front page. Our website will be available online and designed to be user-friendly.

1.5 References

No current document or Web addresses are currently in reference.

2. Overall Description

2.1 Product Perspective

The classification of this product is of the self-contained type. Doodle & Sticks came about because of a Lab project for the CSE 4214 class. We decided on a sticker/art store that would allow theoretical users to sell, purchase, and administer on an E-commerce webpage. The system itself will have a frontend and a backend. The frontend will allow said users to view different available stickers and arts that they can purchase, while also having a page that will allow other users to sell their own stickers and arts they have created. The backend will be where we are able to store the data from the purchases and update any inventories that are created by users.

2.2 Product Functions

Buyer Functions:
- Search and browse products
- Add items to the cart and checkout
- View order history

Seller Functions:
- List and manage products for sale
- Update inventory
- Track sales

Admin Functions:
- Manage user accounts (Buyer and Seller)
- Oversee platform activity

System-Level Functions:
- Responsive web interface for desktop and mobile
- Secure authentication and authorization
- Reliable data management for products, orders, and user accounts

2.3 User Classes and Characteristics

The user classes will be Buyer, Seller, and Admin. Buyers will make up the largest and most frequent user group, as they interact with the system to browse and search for products, add items to their cart, complete purchases, and view their order history. They are expected to have varying levels of technical expertise, so the interface must be intuitive and easy to navigate. Buyers operate with the lowest privilege, level limited to purchasing and personal account management, but they represent the most important class to satisfy since the success of the platform depends on their adoption and usability. Sellers, by contrast, are focused exclusively on offering products for sale. They will use the system to create and manage product listings, update inventory, and track their sales. Their technical expertise is expected to be moderate, as they must understand how to manage listings and handle orders effectively. Sellers operate with a higher privilege level than Buyers, but their participation directly impacts the variety and quality of products available to Buyers. Finally, Admins serve as the highest privilege class, responsible for managing user accounts, monitoring transactions, handling disputes, and maintaining platform security. Admins will use the system less frequently than Buyers or Sellers, but their role is critical to ensuring the integrity and smooth operation of the platform. While Buyers' and Sellers' satisfaction is the highest

priority, reliable and secure Admin functionality remains essential to the long-term stability of the project.

2.4 Operating Environment

The software will operate as a web-based application accessible through modern web browsers. We will host the website via Netlify. The application should be compatible with all major browsers and responsive across desktop and mobile devices. No specialized hardware is required beyond standard devices accessible to the client for access to the site. The software should coexist peacefully with any common technologies that may be used during development.

2.5 Design and Implementation Constraints

The development of the E-Commerce platform will be subject to several constraints that may limit design and implementation choices. The website is intended to be hosted via Netlify initially, with the possibility of changing hosts at a later time. The platform must be compatible with modern web browsers and must function well on both desktop and mobile devices, imposing constraints on UI and design and responsive layout.

From a technology perspective, the platform will rely on HTML, C++, and JavaScript as core technologies, with potential use of front-end frameworks or libraries as appropriate. Data storage, authentication, and order tracking may need to interface with third-party services or APIs, which will require adherence to their protocols and limitations. Security considerations, including but not limited to user authentication, authorization, and data privacy, will constrain how sensitive information is handled, stored, and transmitted.

Development will follow standard web development conventions and coding best practices to ensure maintainability, especially if the customer or team members will be responsible for ongoing maintenance. Any features requiring real-time updates, parallel processing, or heavy computation must be carefully evaluated due to the hosting environment. Finally, all design and implementation decisions must respect potential regulatory policies governing user data, privacy, and e-commerce operations.

3. System Features

3.1 Buyer Functionality

3.1.1 Description and Priority

Buyers can browse and search products, add items to a cart, complete purchases, and view order history. This feature is a high priority as it represents the core functionality driving the platform's commerce.

3.1.2 Stimulus/Response Sequences

Stimulus: Buyer searches for a product -> Response: System displays a list of matching products

Stimulus: Buyer adds an item to the cart -> Response: System updates the cart and confirms the addition of new items
Stimulus: Buyer completes checkout -> Response: System processes the order, updates inventory, and generates an order confirmation

3.1.3 Functional Requirements

REQ-1: The system shall allow Buyers to search for products by name, category, or keyword.
REQ-2: The system shall display product details such as name, price, description, availability, etc.
REQ-3: The system shall allow Buyers to add or remove items from their cart
REQ-4: The system shall process checkout securely, including payment and confirmation
REQ-5: The system shall allow Buyers to view a history of their past orders
REQ-6: The system shall provide clear error messages if a product is out of stock or payment fails.

3.2 Seller Functionality

3.2.1 Description and Priority

Sellers can list products for sale, update inventory, and track sales. This feature is a high priority as Seller participation directly impacts product availability.

3.2.2 Stimulus/Response Sequences

Stimulus: Seller creates a new product -> Response: System saves the listing and displays it in the Seller's inventory
Stimulus: Seller updates product inventory -> Response: System reflects the new quantity and availability
Stimulus: Seller views sales analytics -> Response: System displays total sales, revenue, and other relevant metrics

3.2.3 Functional Requirements

REQ-7: The system shall allow Sellers to create, manage, and modify product listings.
REQ-8: The system shall allow Sellers to update inventory quantities, and product details, and product price.
REQ-9: The system shall provide Sellers with access to basic sales analytics.
REQ-10: The system shall prevent Sellers from accessing other Sellers' data.

3.3 Admin Functionality

3.3.1 Description and Priority

Admins oversee platform activity, manage user accounts, search all product listings, and ensure security. This feature is a medium priority, but critical for system integrity.

3.3.2 Stimulus/Response Sequences

Stimulus: Admin updates a user account -> Response: System saves changes, confirms the update, and notifies the user.
Stimulus: Admin monitors transactions -> Response: System provides a summary of activity, including flagged issues.
Stimulus: Admin responds to a reported issue -> Response: System records the resolution and updates the report status.
Stimulus: Admin searches and removes an inappropriate listing -> Response: System removes the product listing.

3.3.3 Functional Requirements

REQ-11: The system shall allow Admins to manage Buyer and Seller accounts.
REQ-12: The system shall provide Admins with an overview of platform activity.
REQ-13: The system shall allow Admins to review and resolve disputes or flagged transactions.
REQ-14: The system shall enforce role-based access control to restrict sensitive data.
REQ-15: The system shall allow Admins to search all product listings and remove invalid/inappropriate listings.

3.4 Authentication and Authorization

3.4.1 Description and Priority

The platform must provide secure authentication and authorization to ensure that users can only access features appropriate to their role. This is a high-priority feature because it outlines system security and the proper function of Buyer, Seller, and Admin roles.

3.4.2 Stimulus/Response Sequences

Stimulus: user attempts to log in with valid credentials-> Response: System grants access and loads the user's dashboard based on role
Stimulus: User attempts to log in with invalid credentials -> Response: System denies access and displays an error message
Stimulus: Logged-in user attempts to access unauthorized functionality -> Response: System blocks the action and shows an access denied message

3.4.3 Functional Requirements

REQ-15: The system shall require users to log in before accessing restricted features
REQ-16: The system shall enforce role-based access control

REQ-17: The system shall prevent unauthorized access to restricted data or actions
REQ-18: The system shall lock accounts or prompt additional verification after repeated failed login attempts

3.5 Data Management

3.5.1 Description and Priority

The platform must reliably store and retrieve data related to products, orders, users, and transactions. This feature is of high priority since accurate data persistence is critical for platform integrity.

3.5.2 Stimulus/Response Sequences

Stimulus: Buyer places order -> Response: System stores the order details and updates product inventory
Stimulus: Seller updates product details -> Response: System saves the changes and displays the updated information
Stimulus: Admin deletes a user account -> Response: System removes associated data as appropriate

3.5.3 Functional Requirements

REQ-19: The system shall store user account information securely
REQ-20: The system shall store and update product listings and inventory
REQ-21: The system shall record all completed orders and maintain order history

3.6 Notifications and Error Handling

3.6.1 Description and Priority

The system must provide clear feedback to users for key actions, errors, and status updates. This is a medium-priority feature, but necessary for usability.

3.6.2 Stimulus/Response Sequences

Stimulus: Buyer attempts to purchase an out-of-stock product -> Response: System displays an error message and prevents the purchase
Stimulus: Seller successfully updates inventory -> Response: System displays a confirmation message
Stimulus: Admin resolves a dispute -> Response: System updates the record and confirms resolution

3.6.3 Functional Requirements

REQ-22: The system shall provide error messages for invalid user inputs.
REQ-23: The system shall confirm successful actions such as orders, updates, and account changes.

REQ-24: The system shall provide status updates for pending actions.
REQ-25: The system shall log errors for Admin review and debugging.

4. Other Nonfunctional Requirements

4.1 Performance Requirements

- The system should handle at least 20 concurrent users without noticeable slowdown.
- Page load times should be under 3 seconds under normal network conditions
- Database queries should execute in under 2 seconds, using pagination for typical data volumes.
    - Product searches, order history retrieval, etc.
- The system should scale to accommodate future hosting upgrades if moved off Netlify to a more robust platform

4.2 Safety Requirements

- The system should prevent data loss through routine data backup. The backup data will be a complete backup of the entire site, backing up all user information and database information.
    - Manual or automated
- Input validation must prevent crashes due to malformed inputs
    - SQL injections, strings, excessively long fields, etc.
- No part of the system should overwrite or corrupt core system files on the hosting server
- The system should follow safe coding practices to prevent unintentional exposure of sensitive information
- The system must ensure that only authorized users can make changes to orders, listings, or accounts

4.3 Security Requirements

- All user accounts shall be protected by a unique username and password combination.
- The system shall support role-based access control to ensure users only see authorized features.
- User passwords shall be stored in hashed format.
- The system shall use HTTPS for all communications once deployed in a production environment.
- The system shall restrict administrative functions to Admin accounts only
    - I.e., user management, dispute resolution
- Sensitive user data shall not be stored in plain text.
    - I.e. payment info

4.4 Software Quality Attributes

- Usability

    - The system should be easy for non-technical users to navigate with a clean, intuitive UI
- Reliability
    - Core functions should work consistently and without failure in normal conditions
        - I.e., login, product search, checkout
- Maintainability
    - The code should be modular and documented so that future developers can easily make updates
- Portability
    - The application should run on standard modern web browsers.
    - The application should run on mobile and desktop devices.
- Flexibility
    - The design should allow for future expansion
        - I.e., integrating with a real payment gateway, adding analytics
    - The system should allow adding new categories in the future
        - I.e., stationery, prints
- Testability
    - Features should be testable via unit testing and functional testing for verification
- Availability
    - Since it is hosted on Netlify, the system should aim for high availability with an uptime goal of at least 95%

5. Other Requirements

Appendix A: Glossary

Admin (Administrator) *– User with highest privileges; manages accounts, disputes, and platform security.*

Authentication *– Verifying a user's identity (e.g., login credentials).*

Authorization *– Granting access to features based on user role.*

Backend *– Server-side system handling data, orders, and security.*

Buyer *– User who searches, browses, and purchases products.*

Cart *– Temporary collection of items selected by a Buyer.*

Checkout *– Process of finalizing a purchase, including payment.*

Concurrent Users *– Multiple users active on the system at once.*

Data Management *– Storing, retrieving, and updating system data.*

Doodle & Stick *– The e-commerce platform described in this SRS.*

Doodle Den *– Featured section for special or limited artwork.*

Error Handling *– System responses to invalid input or failed actions.*

Frontend *– User-facing part of the system (web pages, UI).*

Functional Requirement (REQ) *– A required feature, numbered REQ-1, REQ-2, etc.*

Inventory *– List of Seller's products with stock and details.*

Netlify *– Hosting platform for initial deployment of the site.*

Nonfunctional Requirement *– Requirement about system qualities (e.g., performance).*

Operating Environment *– Hardware, software, and browsers used to run the system.*

Order History *– Record of a Buyer's past purchases.*

Reliability *– Consistent and correct system performance.*

Role-Based Access Control (RBAC) *– Restricts access by user role.*

Scalability *– System's ability to handle growth in users or data.*

Security Requirement *– Requirement ensuring data confidentiality and integrity.*

Seller *– User who lists, updates, and tracks products for sale.*

Software Quality Attributes *– Nonfunctional system properties (e.g., usability, maintainability).*

Software Requirement Specification (SRS) *– This entire requirements document.*

System Features *– Grouped system capabilities (Buyer, Seller, Admin, etc.).*

Testability *– Ability to verify requirements through testing.*

Transaction *– Exchange where a Buyer purchases from a Seller.*

UI (User Interface) *– Visual and interactive elements users engage with.*

Usability *– Ease with which users learn and use the system.*

User Classes *– Defined roles: Buyer, Seller, Admin.*

Web Browser *– Software (e.g., Chrome, Firefox) used to access the site.*