Software Requirement Specifications

[ShopFLow]

Version: [1.6]

Project Code	*SF2024*
Supervisor	*Munawar Shereen*
Co Supervisor	*Huzaifa Faran*
Project Team	*Munawar Shereen* *Huzaifa Faran* *Faraz Iqbal*
Submission Date	*23th-March-2024*

Document History:

Version	Name of Person	Date	Description of change
1.0	*Munawar Shereen*	*17th-Feb-2024*	*Initial Creation of Document*
1.1	*Huzaifa Faran*	*25th-Feb-2024*	*Added Non-functional requirements*
1.2	*Faraz Iqbal*	*29-Feb-2024*	*Added UseCase UC004*
1.3	*Huzaifa Faran*	*15th-March-2024*	*Added UseCase UC007*
1.4	*Munawar Shereen*	*22th-March-2024*	*Review and finalize the document for submission*

Distribution List

Name	Role
Munawar Shereen	*Supervisor*
Huzaifa Faran	*Co- Supervisor*

Document Sign-Off:

Version	Sign-off Authority	Sign-off Date
1.0	*Huzaifa Faran*	*18th-Feb-2024*
1.6	*Munawar Shereen*	*23th-March-2024*

Table of Contents

1. Introduction

1.1. Purpose of Document

The purpose of this document is to present a detailed description of the "Shopping Flow" project. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, the constraints under which it must operate, and how the system will react to external stimuli.

1.2. Intended Audience

These are the different readers for whom this SRS is intended:

Developer:
Developers will need to have extensive programming and software development experience.They will be in charge of writing actual software code that complies with the specifications listed in this SRS.They must be conversant with the technologies and programming languages utilized in the creation of the ShopFlow.

Project Manager:
Project Manager will be in charge of overseeing the project's finances, schedules, and resources.They must be quite familiar with the goals and specifications specified in this SRS.They will be in charge of making sure the project is finished on schedule, within budget, and to the satisfaction of all parties involved.

Marketing Staff:
Marketing Staff will be in charge of informing potential users and clients about ShopFlow.They must have a thorough awareness of the system's advantages and features as well as the demands and tastes of the intended audience.They will be in charge of coming up with marketing strategies and collateral that clearly convey the system's benefits.

Users:
Users will engage with the system to make use of the system, their technical maturity might differ and some might want to read the SRS for better understanding of the system they are using.

Testers:
Testers will be in charge of confirming that the software complies with the specifications listed in this SRS.They must be well-versed in software testing approaches and resources.They are in charge of finding and reporting any flaws or problems with the system.

Documentation Writers:
These people are in charge of writing user guides, instructional materials, and other documentation for ShopFlow.They must be well-versed in both the requirements and preferences of the intended audience as well as the system's capabilities and features.They will be in charge of writing comprehensible documentation that is short and to the point.

1.3 Abbreviations:

- ***E-commerce:*** *Electronic commerce*
- ***FTP:*** *File Transfer Protocol*
- ***GUI:*** *Graphical user interface*
- ***HTTP:*** *Hypertext Transfer Protocol*
- ***HTTPS:*** *Hypertext Transfer Protocol Secure*

- ***IEC:*** *International Electrotechnical Commission*
- ***ISO:*** *International Organization for Standardization*
- ***REQ:*** *Requirement*
- ***SRS:*** *Software Requirement Specification*
- ***QA:*** *Quality Assurance*
- ***UC:*** *Use Case*
- ***UI:*** *User Interface*

1.4 Document Convention

The font size used throughout the document is Arial, 10 Italicised while for heading it increases to 12. For clarity and ease of understanding, the document is organized using headings and subheadings for each section.Every requirement statement in the SRS is given one of three priority levels (high, medium, or low) to reflect how important it is to the system's overall operation. Unless otherwise stated, it is presumed that detailed requirements inherit the priority level of higher-level needs. Bold text is used to draw attention to essential details and significant information.

2. Overall System Description

2.1. Project Background

The “Shopping Flow” project is a new, self-contained product designed to cater to the growing trend of online shopping. It is not a follow-on member of a product family, nor is it a replacement for certain existing systems. Instead, it is an innovative solution that aims to provide a secure and efficient platform for online shopping.

The product is a standalone application.The major components of the overall system include the User Interface (UI), Product Catalog, Shopping Cart, and Admin Interface. The UI allows users to interact with the application, the Product Catalog lists the items available for purchase, the Shopping Cart handles user orders, and the Admin Interface enables the management of product listings and inventory.

The UI interacts with the Product Catalog to display items to the user.The Shopping Cart is connected to both the UI (for adding/removing items) and the Product Catalog (for fetching item details).The Admin Interface interacts with the Product Catalog for managing product listings and inventory.

2.2. Project Scope

The “ShopFlow” project is a shopping application that integrates various product categories and a Python-based GUI. The application targets a diverse audience of online shoppers seeking to purchase a variety of products, including accessories, clothes, electronics, and medical care items.

2.3. Not In Scope

The current "ShopFlow" project does not cover the following features: sophisticated analytics and reporting; AI-powered suggestions; broad social media integration; internationalization and localization capabilities; sophisticated inventory management beyond basic functions; integration with third-party logistics providers; offline functionality; extensive customization options; and platform scalability beyond initial requirements. Without going into great detail or requiring a great deal of interface with other systems, the project's main goal is to create a feature-rich online shopping platform that is useful for both administrators and customers.

2.4. Project Objectives

The "ShopFlow" project's objectives are to create a reliable and user-friendly online shopping platform that takes advantage of new opportunities and successfully tackles issues that are common in the contemporary online retail landscape. The project intends to reduce overhead expenses and improve operational efficiency by streamlining administrative activities for businesses, such as effective product management and order processing, by developing a holistic solution. In addition, the platform aims to give users a personalized and enjoyable shopping experience with easy-to-use navigation, customized product recommendations, and quick checkout procedures. By means of these initiatives, "ShopFlow" hopes to promote enhanced user contentment, encourage patronage, and eventually accelerate company expansion in the cutthroat online market.

2.5. Stakeholders

The "ShopFlow" system's stakeholders include a range of business user classes as well as technical staff that work in software development and administration. Customers who use the platform to explore and buy items, administrators in charge of handling orders and products, and other organizations like partners or suppliers in the supply chain are examples of business stakeholders. Technical stakeholders also include developers who are in charge of designing, coding, and testing software; project managers who

supervise the process and guarantee that it aligns with business goals; quality assurance testers who confirm the functionality and performance of the system and system administrators who handle deployment and maintenance duties. The development, implementation, and successful operation of the "ShopFlow" platform are contingent upon the effective collaboration and communication among various parties.

2.6. Operating Environment

The "ShopFlow" program is designed to run on desktop PCs exclusively in a Windows environment. It will work with Windows 10 and subsequent iterations. The program will be available offline and won't require internet access for fundamental functions. The program will employ file-based storage for data management because it lacks a database component. Data like product details, customer accounts, and purchase histories will be stored directly on the user's local file system. This method makes the program architecture simpler and less dependent, making it easier for Windows users to install and utilize.

2.7. System Constraints

The "ShopFlow" system operates within various constraints imposed by the external environment:

- *Software constraints: The software must adhere to programming languages and frameworks compatible with Windows operating systems, limiting development options to those supported by the platform.*
- *Hardware constraints: The system's performance may be constrained by the hardware specifications of the target Windows computers, necessitating optimization for lower-end hardware configurations.*
- *Cultural constraints: The software should support multiple languages or be easily adaptable to different language requirements to accommodate users from diverse cultural backgrounds.*
- *Legal constraints: The system must comply with relevant laws and regulations governing e-commerce, data privacy, and intellectual property rights, ensuring legal operation and protection for both businesses and customers.*
- *Environmental constraints: The software should function effectively in various environmental conditions, including both quiet office environments and potentially noisy retail spaces, necessitating consideration for accessibility and usability in different contexts.*
- *User constraints: The project's user interface should be intuitive and user-friendly, accommodating users of varying technical proficiency and potentially including graphical controls for ease of use.*
- *Off-the-shelf components constraints: If third-party components or libraries are used in the project, their constraints, such as licensing restrictions or compatibility issues, must be considered and addressed to ensure smooth integration and operation within the system.*

2.8. Assumptions & Dependencies

Assumptions:

- *Users have basic knowledge of using computers.*
- *All inputs provided by the user are valid and accurate.*
- *All necessary hardware and software resources required for the deployment of the software*
- *The software system will be available on time.*

Dependencies:

- *The software system is dependent on the accuracy and completeness of the data entered by the users.*
- *The software system is dependent on the compatible operating system.*
- *The software system is dependent on the availability of third-party services, such as cloud hosting and email services, for account creation and password recovery*

3. External Interface Requirements

3.1. Hardware Interfaces

The shopFlow software product will require a standard desktop or laptop computer with a keyboard and mouse. There are no special hardware requirements beyond this. The system will read and write data to the file system for data storage and retrieval, eliminating the need for a database system.

3.2. Software Interfaces

Web Browsers:

Compatibility with popular web browsers such as Google Chrome, Firefox will be ensured to allow users seamless access to the shopFlow platform. The purpose of this connection is to provide users with a user-friendly interface to browse products, manage their shopping carts, and place orders.

File System:

shopFlow will read and write data to the file system for data storage and retrieval. This eliminates the need for a traditional database system and simplifies data management. Data items exchanged will include product information, customer details, order history, and other relevant data.

Payment Gateway:

Integration with a payment gateway service enables secure online payments, involving data exchange for transaction details, payment authorization, and payment confirmation messages.

3.3. Communications Interfaces

Protocol Adherence:

ShopFlow will use industry-standard protocols like HTTP/HTTPS for secure data transfer and may utilize FTP for specific file operations.

Error Handling:

Robust error-handling mechanisms will manage communication failures or data transmission errors, ensuring a seamless user experience with proper error logging and reporting.

Email Integration:

Email communication will be integrated into shopFlow for sending registration confirmations, event updates, and notifications to users.

4. Functional Requirement

4.1.1 Admin Login

4.1.1.1 Description and Priority*The admin login function is of high priority as it forms the backbone of the application. It allows the admin to log into their account.*

4.1.1.2 Stimulus/Response Sequences

- *Stimulus: Admin logs into their account.*
- *Response: The system authenticates the admin and provides access to the admin dashboard.*

4.1.1.3 Functional Requirements

REQ-1: The system must authenticate the admin's credentials during login.

REQ-1.1: The system must provide error messages for invalid login attempts.

4.1.2 Add Items

4.1.2.1 Description and Priority

The add items function is of high priority as it allows the admin to manage the product catalog.

4.1.2.2 Stimulus/Response Sequences

- Stimulus: Admin adds a new item to the product catalog.

- Response: The system updates the product catalog with the new item.

4.1.2.3 Functional Requirements

REQ-2: The system must allow the admin to add new items to the product catalog.

REQ-2.1: The system must validate the information of the new item being added.

REQ-2.2: The system must provide a confirmation message when a new item is successfully added

4.1.3: Remove Items

4.1.3.1 Description and Priority

The remove items function is of high priority as it allows the admin to manage the product catalog.

4.1.3.2 Stimulus/Response Sequences

- Stimulus: Admin removes an item from the product catalog.

- Response: The system updates the product catalog by removing the specified item.

4.1.3.3 Functional Requirements

REQ-3: The system must allow the admin to remove items from the product catalog.

REQ-3.1: The system must validate the admin's decision to remove an item.

REQ-3.2: The system must provide a confirmation message when an item is successfully removed.

4.1.4 Customer Login/Sign Up

4.1.4.1 Description and Priority

The customer login/sign up function is of high priority as it directly impacts the user experience and functionality of the application.

4.1.4.2 Stimulus/Response Sequences

- Stimulus: Customer creates a new account or logs into their existing account.

- Response: The system authenticates the customer and provides access to the shopping interface.

4.1.4.3 Functional Requirements

REQ-4: The system must authenticate the customer's credentials during login or account creation.

REQ-4.1: The system must provide error messages for invalid login or sign up attempts.

REQ-4.2: The system must provide a confirmation message when a new account is successfully created.

REQ-4.3: The system must reject login requests if the password is entered incorrectly three times.

4.1.5 Add to Cart

4.1.5.1 Description and Priority

The add to cart function is of high priority as it directly impacts the user experience and functionality of the application.

4.1.5.2 Stimulus/Response Sequences

- Stimulus: Customer adds an item to their shopping cart.

- Response: The system updates the shopping cart with the new item.

4.1.5.3 Functional Requirements

REQ-5: The system must allow the customer to add items to their shopping cart.

REQ-5.1: The system must update the shopping cart in real-time as items are added.

REQ-5.2: The system must provide a confirmation message when a new item is successfully added to the cart.

4.1.6 Remove from Cart

4.1.6.1 Description and Priority

The remove from cart function is of high priority as it directly impacts the user experience and functionality of the application.

4.1.6.2 Stimulus/Response Sequences

- Stimulus: Customer removes an item from their shopping cart.

- Response: The system updates the shopping cart by removing the specified item.

4.1.6.3 Functional Requirements

REQ-6: The system must allow the customer to remove items from their shopping cart.

REQ-6.1: The system must update the shopping cart in real-time as items are removed.

REQ-6.2: The system must provide a confirmation message when an item is successfully removed from the cart.

4.1.7: Place Order

4.1.7.1 Description and Priority

The place order function is of high priority as it directly impacts the user experience and functionality of the application.

4.1.7.2 Stimulus/Response Sequences

- Stimulus: Customer finalizes and places their order.

- Response: The system processes the order and provides a confirmation to the customer.

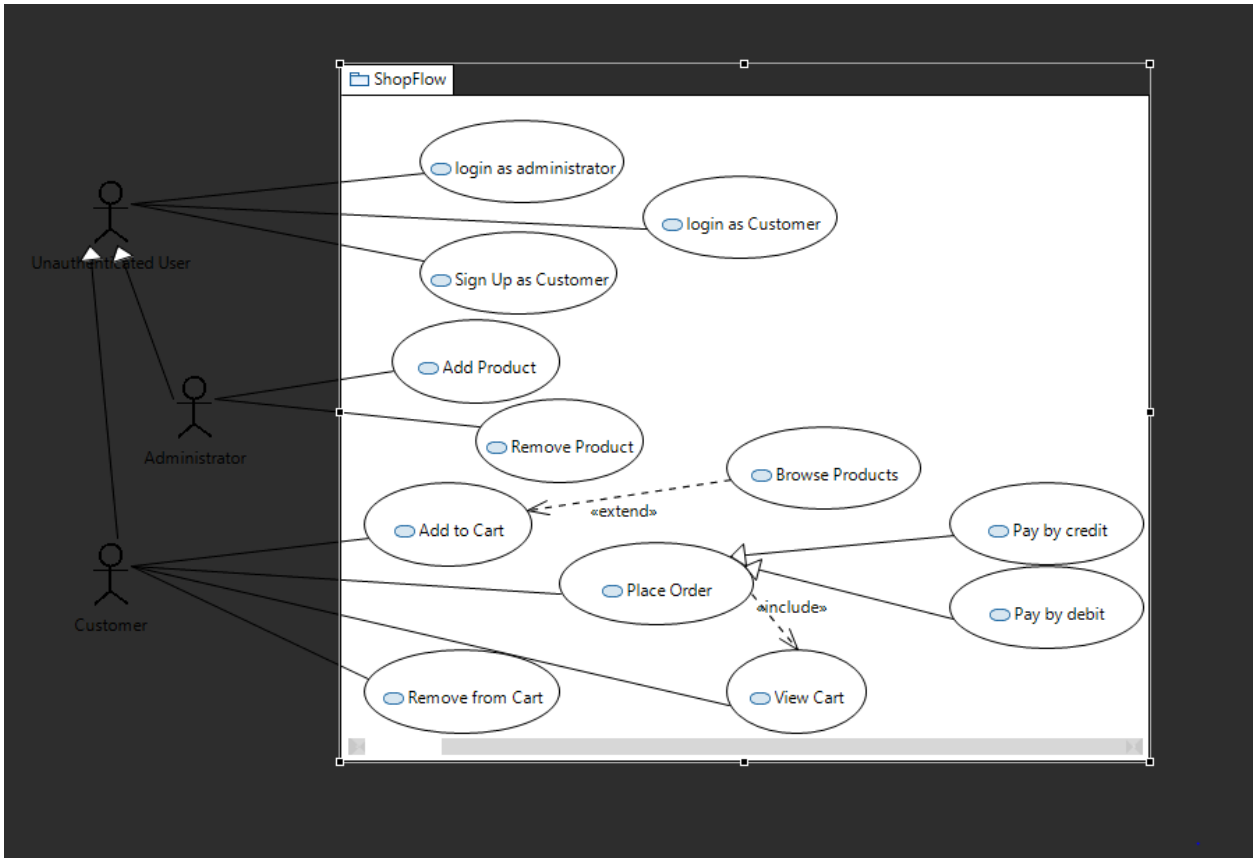
4.1.7.3 Functional Requirements

REQ-7: The system must process the customer's order when they choose to finalize and place their order.

REQ-7.1: The system must provide a summary of the order for review before finalization.

REQ-7.2: The system must provide a confirmation message and receipt when an order is successfully placed.

4.2 Use Cases



4.2.1 Admin Login

The admin login function is of high priority as it forms the backbone of the application. It allows the admin to log into their account.

UC001: Admin Login		
Use case Id:	UC001	
Actors: Admin, System		
Feature: Admin Account Management		
Pre-condition:	The admin has access to the application.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The admin opens the application.	The system displays the admin login page.
2.	*The admin enters their credentials (username, password).*	*The system validates the input. If valid, the system grants access to the admin dashboard.*
Alternate Scenarios:.		

2a: *If the admin enters incorrect credentials, the system displays an error message.* ***2b:*** *If the admin forgets their password, they can reset it via email.*

Post Conditions	
Step#	***Description***
1.	The admin is logged in.
2.	*Session tokens are generated.*
Use Case Cross referenced	-

4.2.2 Add Items

The add items function is of high priority as it allows the admin to manage the product catalog.

UC002: Add Items		
Use case Id:	UC002	
Actors:	Admin, System	
Feature:	Product Catalog Management	
Pre-condition:	The admin is logged in	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The admin opens the admin dashboard.	The system displays the product catalog management section.
2.	*The admin selects "Add Item."*	*System opens forms with fields for new item.*
3.	*The admin enters details for the new item (name, description, price, etc.).*	*The system validates the input.*
Alternate Scenarios:		
3a: *If the admin enters invalid information, the system displays an error message.* ***3b:*** *If the admin cancels adding an item, the system returns to the admin dashboard.*		

Post Conditions	
Step#	***Description***
1.	The product catalog is updated with the new item.
Use Case Cross referenced	-

4.2.3 Remove Item

The remove items function is of high priority as it allows the admin to manage the product catalog.

UC003: Remove Item		
Use case Id:	UC003	
Actors: *Admin, System*		
Feature: *Product Catalog Management*		
Pre-condition:	*The admin is logged in.*	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The admin opens the admin dashboard.	The system displays the product catalog management section.
2.	*The admin selects "Remove Item."*	*System displays all Items to select*
3.	*The admin specifies the item to be removed.*	*The system validates the admin's decision.*
Alternate Scenarios:		
3a: *If the admin cancels removing an item, the system returns to the admin dashboard.* ***3b:*** *If the admin selects items that do not exist then show an error message and allow selection again.*		
Post Conditions		
Step#	***Description***	
1.	The specified item is removed from the product catalog.	
Use Case Cross referenced	-	

4.2.4 Customer SignUp

The customer sign up function is of high priority as it directly impacts the user experience, security, privacy of user and functionality of the application.

UC004: Customer SignUp		
Use case Id:	UC004	
Actors: *Customer, System*		
Feature: *User Account Management*		
Pre-condition:	*The customer has access to the application*	
Scenarios		
Step#	***Action***	***Software Reaction***

1.	The customer opens the application.	System displays the Sign Up/ Login page
2.	*Customer chooses sign up*	*System displays Sign Up form*
3.	*The customer enters their details (name, email, password).*	*If entries are valid, the system validates by sending confirmation email.*
4.	*Enter code received from email*	*System validates code*
Alternate Scenarios:		
3a: *If the customer enters an invalid email address, the system displays an error message.* ***2a:*** *If the customer's chosen password does not meet security requirements, the system displays an error message.* ***2b:*** *If the customer cancels sign-up, the system returns to the login page.* ***4a:*** *If confirmation email is not acted upon or if code entered is incorrect, system returns to main page* ***3a:*** *If the user already has an account, route the user to login.*		
Post Conditions		
Step#	***Description***	
1.	A new customer account is created.	
2.	The customer receives a confirmation email.	
Use Case Cross referenced	UC005: Customer Login	

4.2.5 Customer Login

The customer login function is of high priority as it directly impacts the user experience, security, privacy of user and functionality of the application

UC005: Customer Login		
Use case Id:	UC004	
Actors: Customer, System		
Feature: User Account Management		
Pre-condition:	The customer has a registered account.	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The customer opens the application.	The system displays the login page.
2.	*The customer enters their credentials (email, password).*	*The system validates the input.*
Alternate Scenarios:		
2a: *If the customer enters incorrect login credentials, the system displays an error message* ***2b:*** *If the customer forgets their password, they can reset it via email.* ***1a:*** *If customer doesn't have account, redirect to Sign Up page*		
Post Conditions		
Step#	***Description***	

1.	*The customer is redirected to the shopping interface.*
2.	The system grants access to the shopping interface.
Use Case Cross referenced	UC004: Customer SignUp

4.2.6 Add to Cart

[Use Case Diagram]
The add to cart function is of high priority as it directly impacts the user experience and functionality of the application.

UC006: Add to Cart		
Use case Id:	UC006	
Actors: Customer, System		
Feature: *Shopping Cart Management*		
Pre-condition:	*The customer is logged in.* *The customer is browsing the product catalog.*	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The customer selects an item to add to their cart.	The system displays the item details and quantity selection.
2.	*The customer specifies the quantity and clicks "Add to Cart."*	*If valid, the system adds the item to the customer's shopping cart.*
Alternate Scenarios:		
1a: *If the customer cancels adding the item to the cart, the system returns to the product details page.* ***2a:*** *If the customer enters an invalid quantity, the system displays an error message.*		
Post Conditions		
Step#	***Description***	
1.	*The item is added to the customer's shopping cart.*	
2.	The system updates the cart total and displays a confirmation message	
Use Case Cross referenced	-	

4.2.7 Remove from Cart

[Use Case Diagram]
The remove from cart function is of high priority as it directly impacts the user experience and functionality of the application.

UC007: Remove from cart

Use case Id:	UC007	
Actors: Customer, System		
Feature: Shopping Cart Management		
Pre-condition:	*The customer is logged in.*	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The customer clicks on Remove from Cart..	System displays cart
2.	*Customer chooses the item to remove.*	*System asks for quantity to be removed.*
3.	*Customer selects quantity to be removed*	*System validates and removes item, updates cart and total.*
Alternate Scenarios:		
1a: *If a customer cancels, then return.* ***2a:*** *If the item identifier does not exist, then allow the customer to enter a different identifier.* ***3a:*** If the quantity entered is greater than that present in cart, show error message.		
Post Conditions		
Step#	***Description***	
1.	*The system displays a confirmation message that the item has been removed*	
2.	The system updates the cart by removing the specified item.	
3.	*The system recalculates the cart total.*	
Use Case Cross referenced	<Related use cases, which use or are used by this use case>	

4.2.8 Place Order

The place order function is of high priority as it directly impacts the user experience and functionality of the application.

UC008: Place Order		
Use case Id:	UC008	
Actors: Customer, System		
Feature: *Order Management*		
Pre-condition:	*The customer is logged in.* *The customer has items in their shopping cart.*	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The customer selects “Place Order.”	The system displays the cart information along with total cost and asks for confirmation to proceed.
2.	*Customer selects confirm*	*System shows options for payment options*
3.	*Customer selects appropriate payment method*	*System validates payment.*
Alternate Scenarios:		

2a: *If the customer cancels placing the order, the system returns to the shopping cart.*

Post Conditions	
Step#	***Description***
1.	The order is successfully placed.
2.	*The customer's account balance is updated.*
3.	*The inventory is adjusted for the purchased items.*
4.	*An order confirmation number is generated.*
Use Case Cross referenced	-

4.2.9 View Cart

This allows user to see the cart and all items and quantity present

UC009: View Cart		
Use case Id:	UC009	
Actors:	Customer, System	
Feature:	*Cart Management*	
Pre-condition:	*The customer is logged in.*	
Scenarios		
Step#	***Action***	***Software Reaction***
1.	The customer selects "View Cart."	The system retrieves the list of items in the cart. The system displays the cart contents (item names, quantities, prices).
Alternate Scenarios:		
1a: *If the cart is empty, the system displays a message indicating that the cart is empty.* .		
Post Conditions		
Step#	***Description***	
1.	The cart contents remain unchanged.	
2.	*The customer's returns to main menu*	
Use Case Cross referenced	UC008	

5. Non-functional Requirements

5.1. Performance Requirements

Speed: *The ShopFlow system must deliver fast response times for crucial functionalities like user authentication, product management (adding and removing items), and order processing. Users should experience minimal delay when interacting with the platform, ensuring a seamless shopping experience.*

Precision: *Accuracy is vital for functionalities such as order processing, billing calculations, and inventory management in ShopFlow. The system must ensure precise calculations and data handling to prevent errors in transactions and discrepancies in inventory levels or billing information.*

Concurrency: *ShopFlow should support multiple concurrent users accessing and performing transactions simultaneously without compromising performance. This capability is essential during peak usage periods to maintain responsiveness and accommodate the influx of users without slowdowns or delays.*

Capacity: *The ShopFlow system should be capable of handling a significant volume of data, transactions, and concurrent users without experiencing performance degradation. Scalability is crucial to accommodate future growth in user base and transaction volume, ensuring the system can handle increased workload demands.*

Safety: *Data security and user privacy are paramount for ShopFlow. The system must employ robust security measures, including encryption for data transmission and storage, secure authentication mechanisms, and access controls to safeguard sensitive information and prevent unauthorized access.*

Reliability: *The ShopFlow system must operate reliably under normal and peak load conditions, minimizing downtime and ensuring continuous availability. It should be resilient to failures, with mechanisms for fault tolerance, error recovery, and disaster recovery to maintain uninterrupted service and preserve user trust.*

5.2. Safety Requirements

Data Safety:

To ensure the safety of user accounts and sensitive data within ShopFlow, robust authentication and authorization processes will be implemented. This will restrict system access to authorized users only. Additionally, critical information such as customer details, financial records, and transactional data will be encrypted to prevent unauthorized access or data breaches. Regular backup and recovery options will also be integrated to safeguard against data loss and ensure business continuity in the event of system failures.

User Safety:

In order to enhance user experience and prevent errors, ShopFlow will provide error alerts and clear instructions throughout the platform. Users will also have the capability to report bugs, issues, or security

concerns to system administrators. Furthermore, the system will log user actions to monitor usage and identify any potential unauthorized access or malicious activity, ensuring user safety and system integrity.

System Safety:

To maintain system availability and resilience, shopFlow will implement a comprehensive backup and recovery strategy to mitigate the impact of failures or disasters. Additionally, measures will be in place to prevent system overload and performance deterioration, including load balancing mechanisms and capacity management protocols. These efforts will ensure optimal system performance and minimize the risk of breakdowns or disruptions.

5.3. Security Requirements

User Authentication and Authorization:

In shopFlow, to mitigate unauthorized access, users will be required to authenticate themselves using strong passwords or two-factor authentication. This will enhance the security of the system by ensuring that only authorized individuals can access sensitive information and functionalities.

Data Security and Privacy:

To uphold data security and privacy, shopFlow will implement mechanisms for tracking and auditing user access to sensitive data, such as customer information and transaction records. Regular backups and disaster recovery testing will be conducted to ensure the availability and integrity of data, reducing the risk of data loss or corruption.

5.4. User Documentation

User Manual: *A comprehensive user manual providing detailed instructions on how to use the shopFlow platform effectively. It will cover topics such as user registration, navigation, product browsing, shopping cart management, order processing, and account settings.*

Online Help: *An online help system integrated into the shopFlow platform, accessible through the user interface. It will provide context-sensitive assistance and guidance to users as they navigate different sections of the application.*

Tutorials: *Step-by-step tutorials or video guides demonstrating common tasks and workflows within the shopFlow platform. These tutorials will help users learn how to perform specific actions, such as adding items to the cart, placing orders, or managing account settings.*

FAQ Section: *A frequently asked questions (FAQ) section addressing common queries and troubleshooting tips for users. It will provide answers to common questions related to account management, payment processing, order tracking, and other relevant topics.*

Contact Information: *Contact details for customer support or technical assistance, including email addresses, phone numbers, and helpdesk portal links. Users can reach out for assistance in case they encounter issues or have questions about using the shopFlow platform.*

6. References

User Interface style guide:
"Huzawar UI Style Guide",author "Munawar Shereen", Version 1.0, Date October 2, 2023 ,https://www.shopflowdev.com/ui-style-guide .

Contract:
"ShopFlow Service Agreement ",author "Huzaifa Faran " education , Version 1.0, Date September 7, 2023 https://www.shopflowdev.com/service-agereement .

Standards:
"ISO/IEC 9126 Software Engineering - Product quality ", author "ISO and IEC ", Version 2001, Date October 15,2001, https://www.iso.org/standard/22749.html .

System Requirements Specifications:
"ShopFlow System Requirement",author "Faraz Iqbal", Version 2.0, Date October 2, 2023 ,https://www.shopflow.com/system-requirement .

Use Case Documentation:
"ShopFlow Use Cases",author "Huzaifa Faran ", Version 2.0, Date October 23, 2023 ,https://www.shopflowdevl.com/use-cases .

Version and Scope Documentation:
"ShopFlow Version and Scope ",author "Huzaifa Faran ", Version 2.0, Date November 1, 2023, https://www.shopflowdevl.com/version-scope .

The links that were used when the project was being developed are as follows:

https://www.google.com/url?sa=t&source=web&rct=j&url=https://krazytech.com/projects/sample-software-requirements-specificationsrs-report-airline-database&ved=2ahUKEwjqrcGCqeH-AhVLy7sIHSTTANYQFnoECA4QAQ&usg=AOvVaw22CzBwewIGeRwWNEqReDMv

https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.cse.msu.edu/~cse435/Handouts/SRSExample-webapp.doc&ved=2ahUKEwjqrcGCqeH-AhVLy7sIHSTTANYQFnoECAwQAQ&usg=AOvVaw249wPeutmbLpdGGSWXPd77

https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.geeksforgeeks.org/software-requirement-specification-srs-format/amp/&ved=2ahUKEwjqrcGCqeH-AhVLy7sIHSTTANYQFnoECD4QAQ&usg=AOvVaw06tjxud9Q-fHOAwGbnbGdf

https://www.google.com/url?sa=t&source=web&rct=j&url=https://www.reqview.com/doc/iso-iec-ieee-29148-srs-example/&ved=2ahUKEwiEx4-hqeH-AhVn_rsIHdIaDugQFnoECCoQAQ&usg=AOvVaw0C3mot5kYXhvl1kGQqEq4z

7. Appendices

Error Handling Strategy:

Define a detailed error handling strategy for shopFlow, including error codes, messages, and how errors will be communicated to users. This will ensure efficient troubleshooting and user guidance in case of errors.

User Roles and Permissions:

Clearly define user roles (e.g., admin, regular user) and their corresponding permissions within shopFlow. This will establish access control measures and ensure proper segregation of duties among users.

Third-Party Integrations:

List and specify any third-party services or APIs to be integrated into shopFlow, outlining the scope and functionalities of each integration. This will ensure seamless interoperability with external systems and enhance the overall functionality of the platform.

Performance Metrics:

Determine specific performance metrics to be measured for shopFlow, such as response time, throughput, and error rates. Establish acceptable thresholds for each metric to ensure optimal system responsiveness and user experience.