



**Noakhali Science and Technology University**



**INSTITUTE OF INFORMATION TECHNOLOGY**

**Software Requirement Specification  
for**

**ZenithZone**

*“Join the ZenithZone, where every purchase is a journey  
to the top of satisfaction!”*



# **Software Requirement Specification**

for

# **ZenithZone**

## **Team Members:**

### **Rajib Kumar Dhar**

ID- MUH2125005M

Year- 03 Term- 01 Session- 2020-21

✉: rajib2516@student.nstu.edu.bd

### **Nafisa Binte Jashim**

ID- BKH2125014F

Year- 03 Term- 01 Session- 2020-21

✉: nafisa2516@student.nstu.edu.bd

### **Nadia Jahan**

ID- BFH2125017F

Year- 03 Term- 01 Session- 2020-21

✉: nadia2516@student.nstu.edu.bd

### **Sidratul Muntaha**

ID- BKH2125026F

Year- 03 Term- 01 Session- 2020-21

✉: nisha2516@student.nstu.edu.bd

## **Supervised By:**

### **Nazmun Nahar**

#### **Lecturer,**

Institute of Information Technology

Noakhali Science & Technology University

✉: nazmun@nstu.edu.bd

## Table of Contents

<b>1.</b>	<b>Introduction:</b>	<b>1</b>
<b>1.1</b>	<b>Problem Statement:</b>	<b>1</b>
<b>1.2</b>	<b>Purpose:</b>	<b>1</b>
<b>1.3</b>	<b>Project Scope:</b>	<b>1</b>
<b>1.4</b>	<b>Overview:</b>	<b>2</b>
<b>2.</b>	<b>Stakeholder and Characteristics.....</b>	<b>2</b>
<b>2.1</b>	<b>Stakeholder Identification.....</b>	<b>2</b>
<b>2.2</b>	<b>Stakeholder Categories.....</b>	<b>3</b>
<b>2.3</b>	<b>Stakeholder Analysis .....</b>	<b>4</b>
<b>3.</b>	<b>Requirement Specification .....</b>	<b>6</b>
<b>3.1</b>	<b>Functional Requirement.....</b>	<b>6</b>
<b>3.1.1</b>	<b>Signup.....</b>	<b>6</b>
<b>3.1.2</b>	<b>Login.....</b>	<b>6</b>
<b>3.1.3</b>	<b>Profile Management.....</b>	<b>7</b>
<b>3.1.4</b>	<b>Search Items .....</b>	<b>7</b>
<b>3.1.5</b>	<b>View Items .....</b>	<b>7</b>
<b>3.1.6</b>	<b>Add to Cart.....</b>	<b>8</b>
<b>3.1.7</b>	<b>Wish list.....</b>	<b>8</b>
<b>3.1.8</b>	<b>Personalized Product Recommendations.....</b>	<b>8</b>
<b>3.1.9</b>	<b>Auto Complete Search .....</b>	<b>9</b>
<b>3.1.10</b>	<b>Voice Search .....</b>	<b>9</b>
<b>3.1.11</b>	<b>Art and Craft Exhibition.....</b>	<b>9</b>
<b>3.1.12</b>	<b>Second-hand Sale .....</b>	<b>10</b>
<b>3.1.13</b>	<b>Checkout and Payment.....</b>	<b>10</b>
<b>3.1.14</b>	<b>Order Tracking .....</b>	<b>10</b>
<b>3.1.15</b>	<b>Cancel Order .....</b>	<b>11</b>
<b>3.1.16</b>	<b>Rate Items .....</b>	<b>11</b>
<b>3.1.17</b>	<b>Review Item .....</b>	<b>11</b>
<b>3.1.18</b>	<b>View sales .....</b>	<b>12</b>
<b>3.1.19</b>	<b>Delivered Items.....</b>	<b>12</b>
<b>3.1.20</b>	<b>Upload Listings.....</b>	<b>12</b>
<b>3.1.21</b>	<b>Reminder System .....</b>	<b>13</b>
<b>3.1.22</b>	<b>User Notifications .....</b>	<b>13</b>
<b>3.1.23</b>	<b>Gift Vouchers .....</b>	<b>13</b>
<b>3.2</b>	<b>Data requirements: .....</b>	<b>14</b>

3.2.1	User Data .....	14
3.2.2	Seller/Artisan/Craftsman Data .....	14
3.2.3	Product Data.....	14
3.2.4	Order Data.....	15
3.2.5	Cart and Wishlist Data.....	15
3.2.6	Search and Recommendation Data .....	15
3.2.7	Review and Rating Data.....	16
3.2.8	Notification Data .....	16
3.2.9	Transaction and Wallet Data .....	16
3.2.10	System and Analytics Data.....	17
3.3	Performance requirements: .....	17
3.3.1	Load Handling.....	17
3.3.2	Scalability.....	17
3.3.3	Response Efficiency.....	18
3.3.4	Transaction Processing.....	18
3.3.5	Uptime and Reliability.....	18
3.3.6	Data Throughput.....	18
3.3.7	Search Performance.....	19
3.3.8	Image and Media Loading .....	19
3.3.9	Database Performance.....	19
3.3.10	Page Rendering .....	19
3.3.11	Resource Utilization.....	20
3.3.12	Content Delivery .....	20
3.4	Usability requirements: .....	20
3.4.1	Intuitive Navigation .....	20
3.4.2	Accessibility .....	21
3.4.3	Mobile Responsiveness .....	21
3.4.4	Consistent User Interface (UI) Design.....	21
3.4.5	Search Functionality .....	21
3.4.6	Clear Feedback and Error Messages .....	22
3.4.7	Simplified Checkout Process.....	22
3.4.8	Personalization .....	22
3.4.9	User Support and Help.....	22
3.4.10	Secure Login and Account Management .....	23
3.5	Security requirements: .....	23
3.5.1	Secure Authentication.....	23
3.5.2	Access Control.....	23
3.5.3	Secure Payment Processing with SSLCOMMERZ .....	24

3.5.4	Regular Security Audits and Penetration Testing .....	24
3.5.5	Secure Development Practices .....	24
4.	Requirement Engineering Process for ZenithZone .....	24
4.1	Requirement Elicitation Techniques.....	24
4.1.1	Hold Interviews:.....	25
4.1.2	Surveys and Questionnaires:.....	25
4.1.3	Observation: .....	25
4.1.4	Document Analysis: .....	25
4.2	Sample of Requirement Collection.....	25
4.2.1	Requirement Collection 01.....	25
4.2.2	Requirement Collection 02 .....	26
4.2.3	Requirement Collection 03.....	27
4.3	Requirement Validation .....	27
4.3.1	Review the Requirements:.....	27
4.3.2	Simulate the Requirements: .....	28
5.	Design and Implementation Constraints: .....	28
5.1	User Interface Technology .....	28
5.1.1	Programming Language: .....	28
5.1.2	JavaScript and jQuery Library:.....	28
5.1.3	CSS Framework:.....	28
5.2	Implemented Tools and Platforms .....	28
5.2.1	Code Editor: .....	28
5.2.2	Web Development Framework:.....	28
5.2.2	Front-end Library: .....	28
5.2.3	Back-end Language: .....	29
5.2.4	Database Management System: .....	29
5.2.5	Version Control System: .....	29
5.2.6	Hosting Platform:.....	29
5.2.7	Accessibility and Testing Tools:.....	29
5.2.8	Project Management Tools: .....	29
5.3	Web Server .....	29
5.4	Database Server .....	29
6.	Use Case Diagram .....	30
7.	Use Case Description .....	31
8.	Activity Diagram.....	57
9.	Sequence Diagram .....	75
10.	Swimlane Diagram.....	94
11.	State Diagram.....	112

12. Glossary .....	118
13. References.....	118
14. Appendix.....	119
14.1 Prioritization of requirements .....	119
14.1.1 Three-level scale of requirement prioritized. ....	119
14.1.2 Prioritization of Requirements of ZenithZone .....	120
14.2 Traceability Matrix.....	122

## List of Tables

Table 01 Requirement Collection 01.....	25
Table 02 Requirement Collection 02.....	26
Table 03 Requirement Collection 03.....	27
Table 04 Registration .....	31
Table 05 Log in .....	32
Table 06 Profile Management .....	34
Table 07 Search Product .....	35
Table 08 Personalized Recommendation .....	37
Table 09 Add to Cart.....	38
Table 10 Wish List .....	39
Table 11 View Wish List .....	41
Table 12 Second Hand Product .....	42
Table 13 Art/Craft .....	43
Table 14 Bid on Art/Craft .....	44
Table 15 Profit Margin .....	45
Table 16 Wallet System .....	47
Table 17 Gift Voucher .....	48
Table 18 Delivery Options .....	49
Table 19 Checkout and Payment .....	51
Table 20 Notification .....	52
Table 21 Track Order .....	54
Table 22 Review .....	55

## List of Figures

Figure. 01- Stakeholder Matrix: Power/interest grid .....	4
Figure. 02: Stakeholder Power/Interest for ZenithZone System .....	4
Figure. 03: Use Case Diagram of ZenithZone .....	30
Figure. 04: Activity Diagram for Registration Request .....	57
Figure. 05: Activity Diagram for Log in .....	58
Figure. 06: Activity Diagram for Profile Management .....	59
Figure. 07: Activity Diagram for Personalize Recommendation .....	60
Figure. 08: Activity Diagram for Search Product .....	61
Figure. 09: Activity Diagram for Add to Cart .....	62
Figure. 10: Activity Diagram for Add to wish list .....	63
Figure. 11: Activity Diagram for View Wish List.....	63
Figure. 12: Activity Diagram for Second Hand Product .....	64
Figure. 13: Activity Diagram for Add Art/Craft .....	65
Figure. 14: Activity Diagram for Bid on Art/Craft .....	66
Figure. 15: Activity Diagram for Profit Margin .....	67
Figure. 16: Activity Diagram for Wallet.....	68
Figure. 17: Activity Diagram for Gift Voucher .....	69
Figure. 18: Activity Diagram for Delivery Options .....	70
Figure. 19: Activity Diagram for Checkout and Payment .....	71
Figure. 20: Activity Diagram for Notification.....	72
Figure. 21: Activity Diagram for Track Order .....	73
Figure. 22: Activity Diagram for Review .....	74
Figure. 23: Sequence Diagram for Registration Request.....	75
Figure. 24: Sequence Diagram for Login .....	76
Figure. 25: Sequence Diagram for Profile Management .....	77
Figure. 26: Sequence Diagram for Search Product .....	78
Figure. 27: Sequence Diagram for Personalized Recommendation .....	79
Figure. 28: Sequence Diagram for Add to Cart .....	80
Figure. 29: Sequence Diagram for Wish List .....	81
Figure. 30: Sequence Diagram for View Wish List .....	82
Figure. 31: Sequence Diagram for Second Hand Product .....	83
Figure. 32: Sequence Diagram for Add/Craft .....	84
Figure. 33: Sequence Diagram for Bid on Art/Craft .....	85
Figure. 34: Sequence Diagram for Profit Margin .....	86
Figure. 35: Sequence Diagram for Wallet System .....	87
Figure. 36: Sequence Diagram for Gift Voucher .....	88

<b>Figure. 37: Sequence Diagram for Delivery Options .....</b>	<b>89</b>
<b>Figure. 38: Sequence Diagram for Payment .....</b>	<b>90</b>
<b>Figure. 39: Sequence Diagram for Notification .....</b>	<b>91</b>
<b>Figure. 40: Sequence Diagram for Tracks Order .....</b>	<b>92</b>
<b>Figure. 41: Sequence Diagram for Review .....</b>	<b>93</b>
<b>Figure. 42: Swimlane Diagram for Registration Request .....</b>	<b>94</b>
<b>Figure. 43: Swimlane Diagram for Login .....</b>	<b>95</b>
<b>Figure. 44: Swimlane Diagram for Profile Management .....</b>	<b>96</b>
<b>Figure. 45: Swimlane Diagram for Search Product .....</b>	<b>97</b>
<b>Figure. 46: Swimlane Diagram for Personalize Recommendation .....</b>	<b>98</b>
<b>Figure. 47: Swimlane Diagram for Add to Cart .....</b>	<b>99</b>
<b>Figure. 48: Swimlane Diagram for Wish List .....</b>	<b>100</b>
<b>Figure. 49: Swimlane Diagram for View Wish List .....</b>	<b>101</b>
<b>Figure. 50: Swimlane Diagram for Second Hand Product .....</b>	<b>101</b>
<b>Figure. 51: Swimlane Diagram for Art/Craft .....</b>	<b>102</b>
<b>Figure. 52: Swimlane Diagram for Bid on Art/Craft .....</b>	<b>103</b>
<b>Figure. 53: Swimlane Diagram for Profit Margin .....</b>	<b>104</b>
<b>Figure. 54: Swimlane Diagram for Wallet .....</b>	<b>105</b>
<b>Figure. 55: Swimlane Diagram for Gift Voucher .....</b>	<b>106</b>
<b>Figure. 56: Swimlane Diagram for Delivery Options .....</b>	<b>107</b>
<b>Figure. 57: Swimlane Diagram for Checkout and Payment .....</b>	<b>108</b>
<b>Figure. 58: Swimlane Diagram for Notification .....</b>	<b>109</b>
<b>Figure. 59: Swimlane Diagram for Tracks Order .....</b>	<b>110</b>
<b>Figure. 60: Swimlane Diagram for Review .....</b>	<b>111</b>
<b>Figure. 61: State Diagram for Sign up .....</b>	<b>112</b>
<b>Figure. 62: State Diagram for Login .....</b>	<b>112</b>
<b>Figure. 63: Swimlane Diagram for Search Product .....</b>	<b>113</b>
<b>Figure. 64: State Diagram for Add to Cart .....</b>	<b>113</b>
<b>Figure. 65: State Diagram for Art/Craft .....</b>	<b>114</b>
<b>Figure. 66: State Diagram for Second Hand Product .....</b>	<b>114</b>
<b>Figure. 67: State Diagram for Checkout and Payment .....</b>	<b>115</b>
<b>Figure. 68: State Diagram for Wallet .....</b>	<b>115</b>
<b>Figure. 69: State Diagram for Delivery .....</b>	<b>116</b>
<b>Figure. 70: State Diagram for Gift Voucher .....</b>	<b>116</b>
<b>Figure. 71: State Diagram for Reminder .....</b>	<b>117</b>
<b>Figure. 72: State Diagram for Review .....</b>	<b>117</b>

## **1. Introduction:**

### **1.1 Problem Statement:**

Step into a world where shopping is not just a transaction but an elevated experience of convenience and satisfaction. ZenithZone beckons you with a symphony of innovative features, meticulously crafted to redefine your online shopping journey. From the seamless efficiency of Auto Search and the hands-free allure of Voice Search to the personalized magic of tailored product recommendations and the organizational bliss of the Wish List and Add to Cart functionalities, ZenithZone transcends traditional e-commerce boundaries. With curated exhibitions for artisans and a vibrant marketplace for second-hand treasures, ZenithZone is more than just a platform; it's a destination where every purchase becomes a journey to the zenith of satisfaction.

### **1.2 Purpose:**

The purpose of the ZenithZone project is to make online shopping better by creating a platform with advanced features that meet user needs. Using the latest technology and focusing on user-friendly design, ZenithZone aims to solve common problems customers face when shopping online. Specifically, the project aims to:

- Build an easy-to-use, feature-rich platform with personalized product suggestions, voice search, wish lists, and quick search options.
- Provide a smooth and enjoyable shopping experience that makes it simple to find, choose, and buy products.
- Create a lively community of shoppers, artisans, and second-hand sellers by offering special exhibitions, auctions, and a marketplace for used items.
- Increase customer satisfaction with special benefits like gift vouchers, various delivery options, and secure payment methods.
- Continuously improve the platform based on user feedback, new technology, and market trends.

By achieving these goals, ZenithZone aims to become a trusted and essential online shopping destination for customers, artisans, and sellers, enhancing the overall shopping experience for everyone involved.

### **1.3 Project Scope:**

#### **1.3.1 E-commerce Functionality:**

The primary scope of ZenithZone is to operate as an e-commerce platform, facilitating online transactions between buyers and sellers. It offers a wide range of products, including both new and second-hand items, across different categories such as fashion, electronics, art, and crafts. Users can browse, search, purchase, and review products seamlessly through the platform.

#### **1.3.2 Advanced Features:**

ZenithZone distinguishes itself by offering advanced features such as personalized product recommendations, wish lists, voice search, and intuitive search options. These features enhance the user experience, improve engagement, and differentiate the platform from competitors.

### **1.3.3 Platform Development:**

Design, develop, and deploy a user-friendly e-commerce website with responsive design optimized for desktop and mobile devices.

### **1.3.4 Scalability and Performance:**

Design the website infrastructure to handle high traffic loads and ensure optimal performance during peak times.

### **1.3.5 Compliance and Legal Considerations:**

Ensure compliance with e-commerce regulations, data protection laws, and other legal requirements relevant to the operation of the website.

### **1.3.6 Support and Maintenance:**

Offer ongoing technical support and maintenance services to address any issues, update features, and keep the website secure and up-to-date.

## **1.4 Overview:**

The ZenithZone project aims to transform the online shopping experience by creating a user-friendly platform that seamlessly integrates advanced features to meet diverse user needs. Key features include personalized product recommendations, efficient search options, a wish list for future purchases, a marketplace for artisans and second-hand sellers, flexible delivery options, a secure payment gateway, and exclusive benefits such as gift vouchers. By focusing on these aspects, ZenithZone seeks to simplify and enhance the process of browsing, selecting, and purchasing products, fostering a vibrant community of customers, artisans, and sellers, and ultimately providing a satisfying and engaging shopping experience.

## **2. Stakeholder and Characteristics**

Stakeholders are individuals, groups, or entities that have a vested interest or concern in the success, outcomes, or activities of a particular project, organization, or system. In the context of the ZenithZone project, stakeholders would include anyone who is directly or indirectly affected by or can affect the project.

### **2.1 Stakeholder Identification**

- **Customers:** Customers are the primary users of the ZenithZone platform who visit the website to explore, search for, and purchase products. They seek convenience, a wide range of options, personalized recommendations, and a seamless shopping experience.
- **Artisans and Craftsmen:** These are individuals who create unique, handmade products and use the ZenithZone platform to showcase and sell their items. They may offer handcrafted goods such as jewelry, artwork, pottery, textiles, and more. Artisans and craftsmen aim to showcase their creativity, reach a broader audience, and generate sales through the platform.
- **Sellers:** Sellers are individuals or businesses who use the ZenithZone platform to sell pre-owned or gently used items. These may include clothing, electronics, furniture, collectibles,

and other second-hand goods. Second-hand sellers aim to declutter, recycle, and find new homes for their pre-owned items while potentially earning some revenue.

- **Admin:** The admin team oversees the operation and management of the ZenithZone platform. They are responsible for maintaining the website, managing product listings, handling transactions, ensuring compliance with regulations, and providing support to both customers and sellers.
- **Shipping Partners:** Shipping partners collaborate with ZenithZone to provide delivery services to customers. They ensure timely and reliable delivery of products, contributing to customer satisfaction and loyalty.
- **Payment Gateway Providers:** Payment gateway providers facilitate secure online transactions on ZenithZone. They ensure that payments are processed smoothly and securely, enhancing trust and confidence among customers and sellers.

## 2.2 Stakeholder Categories

The Stakeholders can be divided into two groups: primary and secondary stakeholders.

**Primary Stakeholders:** Those directly affected by the system and have a major influence.

**Secondary Stakeholders:** Those indirectly affected, or who have a lesser level of influence.

### **Primary Stakeholders:**

- Customer
- Artisans and Craftsmen
- Seller
- Admin

These stakeholders are considered primary because their direct involvement and influence are crucial for the success, functionality, and sustainability of the ZenithZone platform. Their needs, expectations, and interactions directly shape the platform's development and performance.

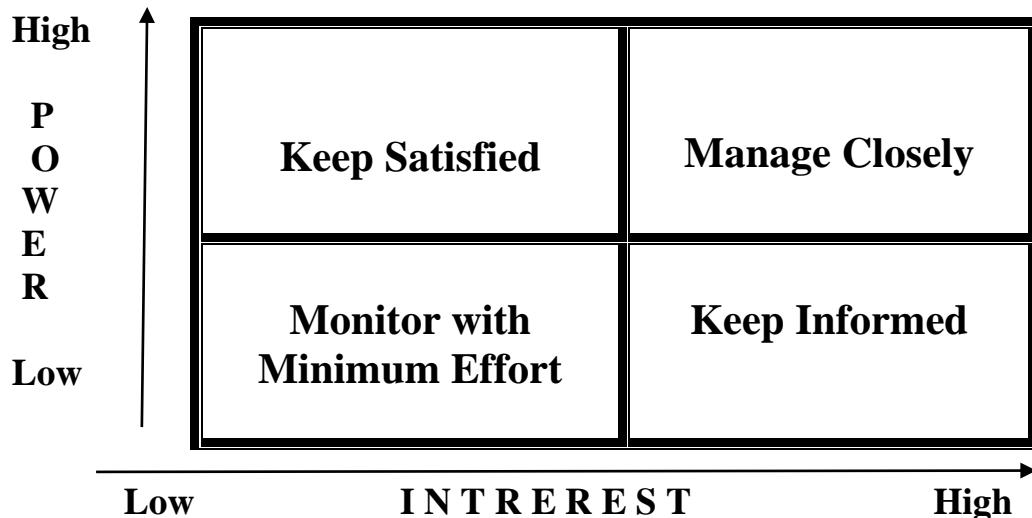
### **Secondary Stakeholders:**

- Shipping Partners
- Payment Gateway Providers

Both shipping partners and payment gateway providers play important roles in supporting the operations of ZenithZone, their impact on the platform's core functions and user experience is considered secondary in comparison to primary stakeholders.

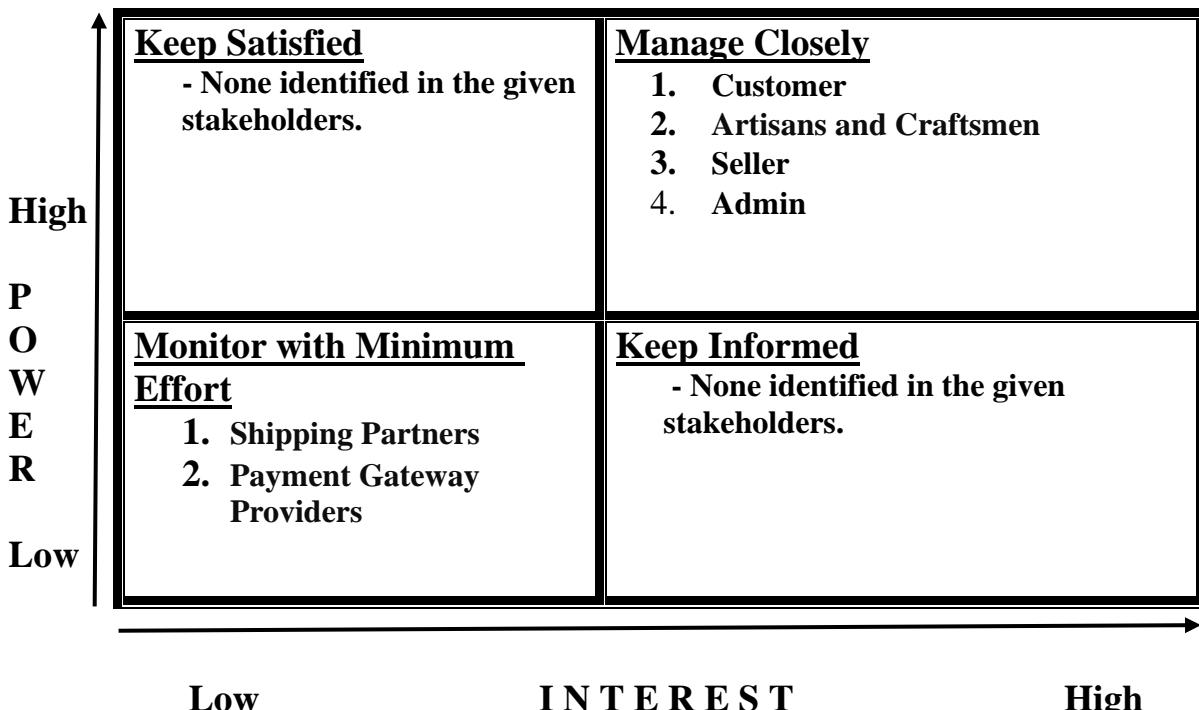
### 2.3 Stakeholder Analysis

The power/interest grid (Figure 1) aids in categorizing stakeholders based on authority (power) and concern about project outcomes (interest). This common technique guides communication strategies. While alternatives like power/influence grid exist, we'll focus on the power/interest grid.



*Figure.1- Stakeholder Matrix: Power/interest grid and the Strategies to Deal with stakeholders.*

This matrix classifies stakeholders into four quadrants based on power and interest. Power signifies authority, while interest reflects concern. Thorough research helps prioritize stakeholders. Strategies for engagement, tailored to power and interest levels, are crucial, as depicted in Figure 1.



*Figure.2- Stakeholder Power/Interest for ZenithZone System*

Based on their level of influence and interest in the ZenithZone project, stakeholders can be categorized into the following groups:

**High Power / High Interest:**

- **Customers:** As primary users, their authority is evident in driving the platform's success. While their interest is high, individual customer engagement may vary. Managing them closely ensures their satisfaction and loyalty through personalized experiences and support.
- **Artisans and Craftsmen:** While they hold significant power and interest in contributing unique handmade products, their engagement may vary. Managing them closely ensures their needs are met, fostering continued participation and satisfaction.
- **Sellers:** Depending on their level of engagement, sellers may have varying degrees of interest. Managing them closely ensures their needs are addressed, fostering a positive seller experience and platform contribution.
- **Admin Team:** While they have high power due to their role in managing the platform, their interest may vary depending on operational needs. Managing them closely ensures smooth platform operation and effective support for primary stakeholders.

**Low Power/Low Interest:**

- **Shipping Partners:** While they play a crucial role in the platform's logistics, their authority and interest may be lower compared to primary stakeholders. Monitoring them with minimum effort ensures timely delivery without excessive resource allocation.
- **Payment Gateway Providers:** Their involvement is vital for secure transactions, but their interest in platform details may be minimal. Monitoring with minimum effort ensures smooth payment processing without unnecessary engagement.

### **3. Requirement Specification**

#### **3.1 Functional Requirement**

Functional requirements are those that serve as examples for the system's internal operation, its description, and an explanation of each subsystem. It comprises of the task that the system should complete, the associated processes, the data that the system should store, and the user interfaces.

##### **3.1.1 Signup**

<b>FR-1</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should have the ability to register accounts with ZenithZone.		
<b>Description</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should be able to register accounts on ZenithZone. The required information for sign-up includes name, email, password, address, phone number, and user type (customer, seller, artisan, or craftsman). Sellers, artisans, and craftsmen must also provide business details and upload their National ID (NID) card for verification. Users must verify their mobile number through an OTP (One-Time Password) sent via SMS to complete the registration process.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen	<b>Priority</b>	High

##### **3.1.2 Login**

<b>FR-2</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should have the ability to log into their accounts on ZenithZone using their registered email/username and password.		
<b>Description</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should be able to log into their ZenithZone accounts using their registered email/username and password combination. In case they forget their password, a "Forgot Password" functionality should be available, allowing users to reset their password by providing their registered email address. Upon submission, a password reset link should be sent to the user's email for verification and password reset.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin	<b>Priority</b>	High

### 3.1.3 Profile Management

<b>FR-3</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should have the ability to manage their profiles on ZenithZone.		
<b>Description</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should be able to view and update their profile information on ZenithZone. This includes updating personal details such as name, email, phone number, address, and profile picture. Sellers, artisans, and craftsmen should also be able to update their business details and upload new verification documents if needed. Additionally, users should be able to change their password, manage their account settings, and view their transaction history.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen	<b>Priority</b>	Low

### 3.1.4 Search Items

<b>FR-4</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should have the ability to search for items on ZenithZone.		
<b>Description</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should be able to search for items on ZenithZone using a search bar. The search functionality should support various filters such as category, price range, location, seller/artist name, and product ratings. The search bar should also support Auto Complete, providing suggestions as users type their queries.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen	<b>Priority</b>	Medium

### 3.1.5 View Items

<b>FR-5</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should have the ability to view item details on ZenithZone.		
<b>Description</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should be able to view detailed information about items listed on ZenithZone. This includes item images, descriptions, price, seller/artist details, ratings, and reviews. The item detail page should also display related items and recommendations based on user preferences and browsing history. Additionally, users should be able to add items to their wish list or cart from the item detail page.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen	<b>Priority</b>	Medium

### 3.1.6 Add to Cart

<b>FR-6</b>	Users (Customers) should have the ability to add items to their cart on ZenithZone.		
<b>Description</b>	Users (Customers) should be able to add items they wish to purchase to a virtual shopping cart. This feature should allow users to add multiple items from different sellers to their cart, update item quantities, remove items, and save their cart for future use. The cart should provide a summary of the items, including images, names, prices, and total cost.		
<b>Stakeholders</b>	Customers	<b>Priority</b>	Medium

### 3.1.7 Wish list

<b>FR-7</b>	Users (Customers) should have the ability to add items to a wish list on ZenithZone.		
<b>Description</b>	Users (Customers) should be able to add items they are interested in to a wish list. This feature allows users to save products for future reference, monitor price changes, and share their wish list with others. Users should be able to add, view, and remove items from their wish list, as well as receive notifications for price drops or stock updates.		
<b>Stakeholders</b>	Customers	<b>Priority</b>	Low

### 3.1.8 Personalized Product Recommendations

<b>FR-8</b>	Customers should receive personalized product recommendations on ZenithZone.		
<b>Description</b>	The platform employs sophisticated algorithms to examine user data, encompassing factors such as gender, age, hobbies, and browsing patterns. This comprehensive analysis enables the platform to deliver personalized product suggestions, enriching the shopping journey for every individual user. Personalized recommendations should appear on the homepage, product pages, and other relevant sections of the site.		
<b>Stakeholders</b>	Customers	<b>Priority</b>	Medium

### 3.1.9 Auto Complete Search

<b>FR-9</b>	Customers should have the ability to use the Auto Complete Search feature on ZenithZone.		
<b>Description</b>	The Auto Complete Search feature enhances browsing efficiency by dynamically suggesting relevant products as customers type their search queries into the search bar. This functionality minimizes the need for manual input and accelerates the search process, ensuring a seamless and efficient browsing experience.		
<b>Stakeholders</b>	Customers, Admin	<b>Priority</b>	Medium

### 3.1.10 Voice Search

<b>FR-10</b>	Customers should have the ability to use the Voice Search feature on ZenithZone.		
<b>Description</b>	The Voice Search feature offers customers a convenient and hands-free method of searching for products on ZenithZone. Utilizing their device's microphone, customers can simply speak their search queries aloud, eliminating the need for typing. This feature caters to users who prefer voice-based interactions or may have limited dexterity, enhancing accessibility and inclusivity.		
<b>Stakeholders</b>	Customers	<b>Priority</b>	Medium

### 3.1.11 Art and Craft Exhibition

<b>FR-11</b>	Artisans and Craftsmen should have the ability to showcase their works in the Art and Craft Exhibition on ZenithZone.		
<b>Description</b>	The Art and Craft Exhibition feature allows artisans and craftsmen to submit their works to be showcased on the platform. They can set a minimum price for each item, and the platform organizes a bidding process for a specific time period. Customers can bid on the showcased artworks, and when an artwork is successfully sold, the platform takes a percentage of the selling price as a commission. The rest is given to the respective artisan or craftsman who created the artwork.		
<b>Stakeholders</b>	Artisans, Craftsmen, Customers, Admin	<b>Priority</b>	High

### **3.1.12 Second-hand Sale**

<b>FR-12</b>	Sellers should have the ability to list and sell second-hand items on ZenithZone.		
<b>Description</b>	The Second-hand Sale feature allows users to register as sellers and showcase their old or used products on the platform. Sellers can set prices for each item, and customers can browse and purchase these products directly through the platform. When a product is sold, the platform takes a percentage of the selling price as a commission, and the remaining amount is given to the seller.		
<b>Stakeholders</b>	Sellers, Customers, Admin	<b>Priority</b>	High

### **3.1.13 Checkout and Payment**

<b>FR-13</b>	Customers should have the ability to checkout and make payments for their purchases on ZenithZone.		
<b>Description</b>	The Checkout and Payment feature allows customers to finalize their purchases by reviewing their order details, selecting delivery options, and making secure payments. The platform supports multiple payment methods and ensures a seamless and secure transaction process.		
<b>Stakeholders</b>	Customers	<b>Priority</b>	High

### **3.1.14 Order Tracking**

<b>FR-14</b>	Customers should have the ability to track the status of their orders on ZenithZone.		
<b>Description</b>	The Order Tracking feature allows customers to monitor the progress of their orders from the moment they are placed until they are delivered. The platform provides real-time updates on order status, including processing, shipping, and delivery stages.		
<b>Stakeholders</b>	Customers	<b>Priority</b>	Medium

### 3.1.15 Cancel Order

<b>FR-15</b>	Customers should have the ability to cancel their orders on ZenithZone.		
<b>Description</b>	The Cancel Order feature allows customers to cancel their orders before they are shipped. This functionality ensures flexibility and customer satisfaction by allowing users to change their minds and modify their purchases.		
<b>Stakeholders</b>	Customers	<b>Priority</b>	Low

### 3.1.16 Rate Items

<b>FR-16</b>	Customers should have the ability to rate products they have purchased on ZenithZone.		
<b>Description</b>	The Rate Item feature allows customers to provide feedback on products they have purchased by assigning a rating. This feature helps other customers make informed purchasing decisions and allows sellers to improve their products based on customer feedback.		
<b>Stakeholders</b>	Customers, Sellers	<b>Priority</b>	Medium

### 3.1.17 Review Item

<b>FR-17</b>	Customers should have the ability to write reviews for products they have purchased on ZenithZone.		
<b>Description</b>	The Review Item feature allows customers to provide detailed feedback on products they have purchased by writing reviews. This feature helps other customers make informed purchasing decisions and allows sellers to improve their products based on customer insights.		
<b>Stakeholders</b>	Customers, Sellers, Admin	<b>Priority</b>	Medium

### 3.1.18 View sales

<b>FR-1</b>	Sellers should have the ability to view their sales on ZenithZone.		
<b>Description</b>	The View Sales feature allows sellers to track and manage their sales on ZenithZone. This functionality provides insights into their sales performance, enabling them to analyze sales data, track revenue, and manage inventory effectively.		
<b>Stakeholders</b>	Sellers, Admin	<b>Priority</b>	Medium

### 3.1.19 Delivered Items

<b>FR-1</b>	Sellers should have the ability to view and manage their delivered items on ZenithZone.		
<b>Description</b>	The Delivered Items feature allows sellers to track and manage items that have been delivered to customers. This functionality provides insights into successful deliveries, helps manage customer feedback, and enables better post-sale service.		
<b>Stakeholders</b>	Sellers, Admin	<b>Priority</b>	Medium

### 3.1.20 Upload Listings

<b>FR-1</b>	Sellers should have the ability to upload and manage their product listings on ZenithZone.		
<b>Description</b>	The Upload Listings feature allows sellers to add new products to the platform, including detailed product information, images, and pricing. This functionality helps sellers manage their inventory and keep their product offerings up to date.		
<b>Stakeholders</b>	Sellers, Admin	<b>Priority</b>	High

### 3.1.21 Reminder System

<b>FR-1</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should receive reminders for various activities and promotions on ZenithZone.		
<b>Description</b>	The Reminder System feature ensures that users are reminded about important activities, such as items left in the cart, upcoming promotions, and personalized recommendations. Reminders are sent via email or SMS to keep users engaged and informed.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, and Craftsmen	<b>Priority</b>	Medium

### 3.1.22 User Notifications

<b>FR-1</b>	Users (Customers, Sellers, Artisans, and Craftsmen) should receive notifications for various activities and updates on ZenithZone.		
<b>Description</b>	The Users (Customers, Sellers, Artisans, and Craftsmen) notifications feature ensures that users are informed about important activities, such as order updates, new messages, promotional offers, and system alerts. Notifications can be delivered via email, SMS, or in-app alerts.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, and Craftsmen, Admin	<b>Priority</b>	Medium

### 3.1.23 Gift Vouchers

<b>FR-1</b>	Users should have the ability to create, purchase, redeem, and manage gift vouchers on ZenithZone.		
<b>Description</b>	The Gift Vouchers feature enables users to purchase, gift, and redeem vouchers for discounts or promotions on ZenithZone. This feature also allows for the management of voucher balances and tracking of voucher usage.		
<b>Stakeholders</b>	Customers, Admin	<b>Priority</b>	Low

### **3.2 Data requirements:**

Data requirements refer to the specific types, formats, sources, and quality standards of data needed to support a project, system, or analysis effectively. It involves identifying and defining the data necessary to achieve the objectives of the project or system.

#### **3.2.1 User Data**

<b>DR-1</b>	Management of user information.		
<b>Description</b>	ZenithZone should efficiently manage user information, including registration details (name, email, password, phone number), login credentials, personal information (address, gender, age), and communication preferences. This data is crucial for creating personalized shopping experiences and facilitating communication.		
<b>Stakeholders</b>	Users (Customers, Sellers, Artisans, Craftsmen, Admin).	<b>Priority</b>	High

#### **3.2.2 Seller/Artisan/Craftsman Data**

<b>DR-2</b>	Management of seller and artisan information.		
<b>Description</b>	The platform should manage detailed profiles of sellers, artisans, and craftsmen, including identity verification (NID, business licenses), product listings, sales history, and performance metrics. This data ensures the authenticity of sellers and helps in managing the marketplace effectively.		
<b>Stakeholders</b>	Sellers, Artisans, Craftsmen, Admin.	<b>Priority</b>	High

#### **3.2.3 Product Data**

<b>DR-3</b>	Comprehensive product information management.		
<b>Description</b>	ZenithZone should maintain detailed product information, including product descriptions, images, prices, categories, stock levels, and seller information. Accurate and comprehensive product data is essential for customer decision-making and inventory management.		
<b>Stakeholders</b>	Sellers, Admin, Customers.	<b>Priority</b>	High

### 3.2.4 Order Data

<b>DR-4</b>	Efficient management of order information.		
<b>Description</b>	The system should track orders from placement to delivery, including order details (products, quantities, prices), payment status, delivery status, and order history. This data is vital for order fulfillment, customer service, and logistical operations.		
<b>Stakeholders</b>	Customers, Sellers, Admin.	<b>Priority</b>	High

### 3.2.5 Cart and Wishlist Data

<b>DR-5</b>	Management of cart and wishlist information.		
<b>Description</b>	ZenithZone should manage items added to user carts and wishlists, ensuring this data is up-to-date and accessible. This includes tracking product availability, pricing changes, and user preferences, which help in improving the shopping experience and conversion rates.		
<b>Stakeholders</b>	Customers, Admin.	<b>Priority</b>	Medium

### 3.2.6 Search and Recommendation Data

<b>DR-6</b>	Data management for search and recommendations.		
<b>Description</b>	The platform should collect and analyze search queries, browsing patterns, purchase history, and user demographics to provide personalized product recommendations. This data enhances user engagement and increases sales by tailoring the shopping experience to individual preferences.		
<b>Stakeholders</b>	Customers, Admin.	<b>Priority</b>	Medium

### 3.2.7 Review and Rating Data

<b>DR-7</b>	Management of reviews and ratings.		
<b>Description</b>	The system should collect, store, and display user reviews and ratings for products and sellers. This includes managing the authenticity of reviews, moderating content, and aggregating ratings to provide transparent feedback for future customers.		
<b>Stakeholders</b>	Customers, Sellers, Admin.	<b>Priority</b>	Medium

### 3.2.8 Notification Data

<b>DR-8</b>	Management of user notifications.		
<b>Description</b>	ZenithZone should manage notification data, including promotional messages, order updates, and reminders tailored to user preferences. This involves tracking user interaction with notifications and optimizing communication strategies to enhance engagement.		
<b>Stakeholders</b>	Customers, Sellers, Artisan, Craftsman, Admin.	<b>Priority</b>	Medium

### 3.2.9 Transaction and Wallet Data

<b>DR-9</b>	Efficient management of transactions and wallet balances.		
<b>Description</b>	The platform should handle transaction details, payment processing, and wallet balance management securely and accurately. This includes managing various payment methods, ensuring secure transactions, and maintaining user wallet balances for future purchases.		
<b>Stakeholders</b>	Customers, Admin.	<b>Priority</b>	High

### 3.2.10 System and Analytics Data

<b>DR-10</b>	Data collection for system performance and analytics.		
<b>Description</b>	ZenithZone should collect data on system performance (load times, error rates), user behavior (page views, session duration), and sales analytics (conversion rates, revenue). This data helps in monitoring the health of the platform and making data-driven business decisions.		
<b>Stakeholders</b>	Admin, Developers.	<b>Priority</b>	Medium

## 3.3 Performance requirements:

Performance requirements specify the performance characteristics that a system, product, or service must meet to satisfy user needs and expectations effectively. These requirements define the levels of performance, capacity, speed, responsiveness, and efficiency that the system should achieve under various conditions and usage scenarios. Performance requirements are essential for ensuring that the system performs optimally and delivers a satisfactory user experience.

### 3.3.1 Load Handling

<b>PR-1</b>	The platform should handle user interactions smoothly without any noticeable delays.		
<b>Description</b>	Ensure that all pages and features load quickly to provide a smooth user experience. The system should be capable of handling a high volume of concurrent users without experiencing slowdowns, ensuring that the site remains responsive even during peak usage times.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen ,Admin Team	<b>Priority</b>	High

### 3.3.2 Scalability

<b>PR-2</b>	The system should support a high number of concurrent users without performance degradation.		
<b>Description</b>	Ensure the platform can handle a large number of simultaneous users, especially during peak shopping times or promotional events. The architecture should be designed to scale horizontally and vertically to accommodate growth in user base and transaction volume.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen ,Admin Team	<b>Priority</b>	High

### 3.3.3 Response Efficiency

<b>PR-3</b>	User interactions (e.g., adding items to the cart, wishlisting, searching for products) should be processed efficiently.		
<b>Description</b>	Ensure that user actions are processed quickly to maintain a responsive user interface. Actions such as clicking buttons, submitting forms, and navigating between pages should be executed with minimal delay, providing a smooth and engaging user experience.		
<b>Stakeholders</b>	Customers, Sellers, Admin Team	<b>Priority</b>	High

### 3.3.4 Transaction Processing

<b>PR-4</b>	The platform should process transactions efficiently and without errors.		
<b>Description</b>	Ensure that payment and order placement transactions are processed accurately and swiftly. The payment gateway should be reliable, supporting multiple payment methods and providing instant confirmation to users upon successful transactions.		
<b>Stakeholders</b>	Customers, Sellers, Payment Gateway Providers	<b>Priority</b>	High

### 3.3.5 Uptime and Reliability

<b>PR-5</b>	The platform should be highly available and reliable.		
<b>Description</b>	Ensure the platform minimizes downtime to enhance user trust and reliability. Implement redundancy, failover mechanisms, and regular maintenance schedules to keep the platform operational at all times.		
<b>Stakeholders</b>	Customers, Sellers, Admin Team	<b>Priority</b>	High

### 3.3.6 Data Throughput

<b>PR-6</b>	The system should handle a large volume of transactions in a short period.		
<b>Description</b>	Ensure that the platform can process multiple transactions simultaneously without performance degradation. Optimize database queries and server resources to handle high transaction volumes efficiently.		
<b>Stakeholders</b>	Customers, Sellers, Admin Team, Payment Gateway Providers	<b>Priority</b>	Medium

### 3.3.7 Search Performance

<b>PR-7</b>	Search results should be displayed quickly after a query is submitted.		
<b>Description</b>	Ensure that the search functionality is efficient, providing quick access to desired products. Optimize search algorithms and indexing techniques to deliver relevant results promptly, enhancing the user's ability to find products quickly.		
<b>Stakeholders</b>	Customers, Sellers, Admin Team	<b>Priority</b>	High

### 3.3.8 Image and Media Loading

<b>PR-8</b>	Images and media content should load promptly.		
<b>Description</b>	Ensure that product images and other media content load quickly to enhance the visual experience. Utilize image compression, lazy loading, and content delivery networks (CDNs) to reduce load times and improve user engagement.		
<b>Stakeholders</b>	Customers, Sellers, Admin Team	<b>Priority</b>	Medium

### 3.3.9 Database Performance

<b>PR-9</b>	The database should respond to queries efficiently.		
<b>Description</b>	Ensure that database operations are optimized for quick access to data. Implement indexing, caching, and query optimization techniques to ensure rapid retrieval and update of information, supporting a smooth user experience.		
<b>Stakeholders</b>	Admin Team, Developers	<b>Priority</b>	High

### 3.3.10 Page Rendering

<b>PR-10</b>	All web pages should render completely and smoothly.		
<b>Description</b>	Ensure that all pages of the platform render quickly and completely, providing a smooth browsing experience. Optimize front-end performance by minimizing JavaScript, CSS, and HTML, and ensure compatibility across different browsers and devices.		
<b>Stakeholders</b>	Customers, Sellers, Admin Team, Developers	<b>Priority</b>	High

### 3.3.11 Resource Utilization

<b>PR-11</b>	The platform should maintain efficient CPU and memory utilization.		
<b>Description</b>	Ensure that the system resources are efficiently used and avoid bottlenecks during high traffic periods. Monitor and optimize server performance to prevent overuse of CPU, memory, and other resources, maintaining system stability and performance.		
<b>Stakeholders</b>	Admin Team, Developers	<b>Priority</b>	Medium

### 3.3.12 Content Delivery

<b>PR-12</b>	Use a Content Delivery Network (CDN) to deliver static content efficiently.		
<b>Description</b>	Ensure that static content (e.g., images, stylesheets, scripts) is delivered quickly to users by leveraging CDN services. CDNs help distribute the load and reduce latency, ensuring that content is delivered from servers closest to the user's geographical location.		
<b>Stakeholders</b>	Customers, Sellers, Admin Team	<b>Priority</b>	Medium

## 3.4 Usability requirements:

Usability requirements define the characteristics of a system or product that contribute to its ease of use, user satisfaction, and effectiveness in achieving user goals. These requirements focus on ensuring that the system is intuitive, user-friendly, and accessible to its intended users, thereby enhancing the overall user experience.

### 3.4.1 Intuitive Navigation

<b>UR-1</b>	The platform should provide intuitive and easy-to-use navigation.		
<b>Description</b>	Ensure that users can easily find their way around the platform. Menus, links, and buttons should be clearly labeled and logically organized. Navigation should be consistent across all pages, enabling users to find products, manage their accounts, and access support without confusion.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team	<b>Priority</b>	High

### **3.4.2 Accessibility**

<b>UR-2</b>	The platform should be accessible to users with disabilities.		
<b>Description</b>	Ensure compliance with accessibility standards (e.g., WCAG 2.1) to make the platform usable by people with various disabilities. This includes providing text alternatives for non-text content, ensuring sufficient color contrast, and enabling keyboard navigation.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team	<b>Priority</b>	High

### **3.4.3 Mobile Responsiveness**

<b>UR-3</b>	The platform should be fully functional and aesthetically pleasing on mobile devices.		
<b>Description</b>	Ensure that the website is responsive, providing a seamless user experience across different devices, including smartphones, tablets, and desktops. Elements should resize and adjust appropriately to different screen sizes.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team	<b>Priority</b>	High

### **3.4.4 Consistent User Interface (UI) Design**

<b>UR-4</b>	The platform should have a consistent and visually appealing user interface.		
<b>Description</b>	Ensure that the UI design is consistent throughout the platform, with a uniform color scheme, typography, and layout. Consistency helps users feel familiar with the site, reducing the learning curve and enhancing usability.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team, Developer	<b>Priority</b>	Medium

### **3.4.5 Search Functionality**

<b>UR-5</b>	The platform should provide an efficient and user-friendly search function.		
<b>Description</b>	Ensure that users can easily find products using the search bar. Implement features such as autocomplete, filters, and sorting options to enhance the search experience.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team	<b>Priority</b>	High

### 3.4.6 Clear Feedback and Error Messages

<b>UR-6</b>	The platform should provide clear feedback and error messages.		
<b>Description</b>	Ensure that users receive immediate and understandable feedback on their actions, such as adding items to the cart or submitting forms. Error messages should be clear and provide guidance on how to correct issues.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team	<b>Priority</b>	High

### 3.4.7 Simplified Checkout Process

<b>UR-7</b>	The platform should offer a simplified and streamlined checkout process.		
<b>Description</b>	Ensure that the checkout process is straightforward, minimizing the number of steps required to complete a purchase. Provide a progress indicator and allow users to review their order before finalizing it.		
<b>Stakeholders</b>	Customers, Admin Team	<b>Priority</b>	High

### 3.4.8 Personalization

<b>UR-8</b>	The platform should provide personalized user experiences.		
<b>Description</b>	Ensure that users receive personalized recommendations and content based on their preferences and browsing history. Personalization enhances user engagement and satisfaction.		
<b>Stakeholders</b>	Customers	<b>Priority</b>	Medium

### 3.4.9 User Support and Help

<b>UR-9</b>	The platform should provide accessible user support and help resources.		
<b>Description</b>	Ensure that users can easily access help and support resources, such as FAQs, live chat, and customer service contact information. Providing support options helps users resolve issues quickly and efficiently.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team	<b>Priority</b>	Low

### 3.4.10 Secure Login and Account Management

<b>UR-10</b>	The platform should provide secure and user-friendly login and account management features.		
<b>Description</b>	Ensure that users can easily log in and manage their accounts, including updating personal information, viewing order history, and managing payment methods. Security measures should be in place to protect user data.		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team	<b>Priority</b>	High

## 3.5 Security requirements:

Security requirements outline the measures and safeguards necessary to protect a system, application, or data from unauthorized access, breaches, theft, damage, or disruption. These requirements ensure that the system maintains confidentiality, integrity, and availability of information, mitigates risks, and complies with relevant security standards and regulations. Security requirements are essential for safeguarding sensitive data, maintaining user trust, and preventing security incidents or breaches.

### 3.5.1 Secure Authentication

<b>SR-1</b>	Implement secure authentication mechanisms for all users.		
<b>Description</b>	Require strong passwords for user accounts and support multi-factor authentication (MFA) to enhance security. Passwords should be stored using secure hashing algorithms (e.g., bcrypt).		
<b>Stakeholders</b>	Customers, Sellers, Artisans, Craftsmen, Admin Team	<b>Priority</b>	High

### 3.5.2 Access Control

<b>SR-2</b>	Implement role-based access control (RBAC).		
<b>Description</b>	Define and enforce roles and permissions for different types of users (e.g., customers, sellers, admin). Ensure that users can only access resources and perform actions that are permitted by their roles.		
<b>Stakeholders</b>	Admin Team	<b>Priority</b>	High

### **3.5.3 Secure Payment Processing with SSLCOMMERZ**

<b>SR-3</b>	Ensure secure payment processing with SSLCOMMERZ.		
<b>Description</b>	Integrate the SSLCOMMERZ payment gateway for processing transactions. Ensure compliance with PCI DSS standards. Do not store sensitive payment information on the platform; instead, rely on SSLCOMMERZ for secure payment processing.		
<b>Stakeholders</b>	Customers, Sellers, IT Security Team	<b>Priority</b>	High

### **3.5.4 Regular Security Audits and Penetration Testing**

<b>SR-4</b>	Conduct regular security audits and penetration testing.		
<b>Description</b>	Perform regular security assessments, including vulnerability scans and penetration tests, to identify and address security weaknesses. Engage third-party security experts for independent audits.		
<b>Stakeholders</b>	Admin Team, IT Security Team	<b>Priority</b>	Medium

### **3.5.5 Secure Development Practices**

<b>SR-5</b>	Adopt secure software development practices.		
<b>Description</b>	Follow secure coding standards and practices throughout the software development lifecycle. Conduct code reviews and use automated tools to identify security vulnerabilities in the code.		
<b>Stakeholders</b>	Development Team	<b>Priority</b>	High

## **4. Requirement Engineering Process for ZenithZone**

Requirements engineering (RE) is a structured process for figuring out what a software system needs to do. It involves tasks to understand, document, and manage what different people want from the software. This ensures that the final product meets everyone's needs and expectations.

### **4.1 Requirement Elicitation Techniques**

Requirement elicitation techniques for ZenithZone involve various methods to gather, identify, and prioritize the needs and expectations of stakeholders. Given the diverse stakeholder groups involved, it's essential to employ techniques that capture a comprehensive understanding of their requirements. Here are some requirement elicitation techniques suitable for ZenithZone: This process ensures that ZenithZone meets the diverse needs and requirements of its users effectively.

#### **4.1.1 Hold Interviews:**

Conducting one-on-one or group interviews with stakeholders such as customers, artisans, sellers, administrators, and support teams. These interviews can uncover their needs, preferences, pain points, and expectations regarding the platform's features, functionality, and usability.

#### **4.1.2 Surveys and Questionnaires:**

Distributing surveys or questionnaires to a wider audience of stakeholders to gather quantitative and qualitative feedback on their requirements. Surveys can help identify common themes, preferences, and priorities across different stakeholder groups.

#### **4.1.3 Observation:**

Observing users' interactions with similar e-commerce platforms, including their browsing behavior, purchasing patterns, and usability issues. Direct observation can provide valuable insights into users' needs and behaviors in real-world scenarios.

#### **4.1.4 Document Analysis:**

Reviewing existing documents, such as business requirements documents, user stories, use cases, and market research reports, to extract relevant requirements and insights into stakeholder needs and expectations.

### **4.2 Sample of Requirement Collection**

#### **4.2.1 Requirement Collection 01**

*Table 1: Requirement Collection 01*

Requirement Elicitation Techniques	Interviews Field Observation Surveys/Questionnaires
Collect Form	ZenithZone Development Team Customers (Artisans, Sellers, and Customers) Admin Team
Findings:	<p><b>User Account Management:</b></p> <ul style="list-style-type: none"> <li>a) User Registration: Intuitive and user-friendly registration process for artisans, sellers, and customers.</li> <li>b) User Login: Secure and seamless login functionality for accessing the platform.</li> <li>c) Account Management: Comprehensive account management features for users to update their profiles, manage orders, and track shipments.</li> </ul> <p><b>Product Search Feature:</b></p> <ul style="list-style-type: none"> <li>a) User-friendly Search Feature: Intuitive search functionality for customers to find products easily based on keywords, categories, or filters.</li> <li>b) Advanced Search Options: Advanced search options such as filters by price range, product type, seller location, and customer ratings for tailored searches.</li> </ul>
Advised Topics for Implementation	Optimize registration for simplicity, offer a comprehensive search system for swift product discovery, and integrate interactive communication tools to enhance engagement on the ZenithZone platform.

#### 4.2.2 Requirement Collection 02

**Table 2: Requirement Collection 02**

Requirement Elicitation Techniques	Surveys/Questionnaires Document Analysis
Collect Form	Artisans and Craftsmen Sellers Customers
Findings:	<p><b>Product Listing and Management:</b></p> <p>a) Product Upload: Easy and intuitive process for artisans and sellers to upload product listings, including images, descriptions, pricing, and inventory management.</p> <p>b) Product Categorization: Flexible categorization options allowing artisans and sellers to classify products into relevant categories and subcategories for better organization and discoverability.</p> <p>c) Product Management: Tools and features for managing product listings, updating inventory, modifying product details, and tracking sales and orders.</p> <p><b>Payment and Transaction Handling:</b></p> <p>a) Secure Payment Gateway: Integration with secure payment gateways to facilitate safe and reliable online transactions, including support for various payment methods such as credit/debit cards, digital wallets, and bank transfers.</p> <p>b) Order Processing: Efficient order processing system with features for order confirmation, payment verification, order fulfillment, and shipping arrangements.</p> <p>c) Transaction History: Accessible transaction history for users to view past orders, payments, and invoices for transparency and record-keeping purposes</p>
Advised Topics for Implementation	Create efficient product management tools, secure payment gateways, and feedback channels to streamline processes, ensure safe transactions, and enhance customer engagement, satisfaction, and loyalty on the platform.

### 4.2.3 Requirement Collection 03

**Table 3: Requirement Collection 03**

Requirement Elicitation Techniques	Interviews Focus Groups
Collect Form	Customers (Buyers) Sellers
Findings:	<p><b>Review and Rating System:</b></p> <p>a) Customer Feedback on Products: Customers desire the ability to leave detailed reviews and ratings for products they have purchased, including the option to rate aspects such as product quality, packaging, delivery speed, and seller communication.</p> <p>b) Seller Rating and Reputation: Customers value transparency and accountability from sellers, and therefore appreciate the opportunity to view seller ratings and reviews from other buyers before making a purchase.</p> <p>c) Overall Shopping Experience: Customers want the ability to rate and provide feedback on their overall shopping experience on ZenithZone, including factors such as website usability, checkout process, and customer service interactions.</p>
Advised Topics for Implementation	Implement a robust review and rating system, accessible customer support channels, and proactive feedback collection mechanisms to foster customer engagement, satisfaction, and loyalty on the ZenithZone platform.

## 4.3 Requirement Validation

In the ZenithZone project, requirement validation ensures that the specified requirements are accurate, complete, and aligned with the platform's goals and stakeholder expectations. During the initial stages, some requirements may appear clear and sufficient, but gaps and ambiguities can surface during implementation. The validation process aims to address these issues to ensure ZenithZone meets the standards and expectations set by stakeholders.

### 4.3.1 Review the Requirements:

- **Conduct a Rigorous Review:** Perform a thorough review of the collected requirements to ensure they are accurate, consistent, and aligned with ZenithZone's objectives.
- **Engage Stakeholders:** Involve diverse stakeholders, including customers, artisans, craftsmen, sellers, administrators, and developers, in the review process to gather comprehensive perspectives.
- **Document Discrepancies:** Identify and document any discrepancies, conflicts, or ambiguities found during the review. Collaborate with stakeholders to resolve these issues and refine the requirements.

#### **4.3.2 Simulate the Requirements:**

- **Prototyping Tools:** Utilize prototyping tools to simulate the proposed features and functionalities of the ZenithZone platform.
- **Test Usability and Functionality:** Conduct usability and functionality tests on the prototype to identify any gaps or inconsistencies in the requirements.
- **Gather Stakeholder Feedback:** Collect feedback from stakeholders through user testing sessions. Use this feedback to iterate on the prototype and make necessary adjustments to ensure it meets the stakeholders' needs and expectations.

### **5. Design and Implementation Constraints:**

Design and implementation constraints are critical considerations that shape the development and implementation process of the ZenithZone project. These constraints encompass the tools, technologies, and platforms utilized to ensure the success of the project.

#### **5.1 User Interface Technology**

User interface (UI) technology plays a crucial role in the design and functionality of the ZenithZone platform, ensuring a seamless and intuitive user experience. The following technologies are utilized:

##### **5.1.1 Programming Language:**

**JavaScript:** JavaScript serves as the primary language for implementing dynamic and interactive features on the front-end of the ZenithZone platform. It facilitates client-side behavior and enhances user interaction with the application.

##### **5.1.2 JavaScript and jQuery Library:**

These libraries are employed to add client-side functionality and interactivity to the UI components of the ZenithZone platform. They simplify JavaScript programming and offer a wide range of pre-built UI components and widgets.

##### **5.1.3 CSS Framework:**

CSS frameworks like Bootstrap are utilized for styling and layout design, ensuring consistency and responsiveness across different devices and screen sizes. Bootstrap provides a robust set of UI components and utilities that streamline front-end development.

### **5.2 Implemented Tools and Platforms**

To implement the ZenithZone project effectively, various tools and platforms are employed to streamline development and deployment processes:

##### **5.2.1 Code Editor:**

Code editors such as Visual Studio Code or Atom are utilized for writing and editing the codebase of the ZenithZone platform, providing essential features for code development and debugging.

##### **5.2.2 Web Development Framework:**

A web development framework like AngularJS, React, or Vue.js is selected to build the front-end of the ZenithZone platform. These frameworks offer pre-built components and tools that expedite development and enhance maintainability.

##### **5.2.3 Front-end Library:**

Front-end libraries such as Bootstrap or Foundation are used for UI component design and layout, ensuring a visually appealing and user-friendly interface for the ZenithZone platform.

**5.2.4 Back-end Language:**

A back-end language, such as JavaScript (Node.js) or Python (Django), is chosen to handle server-side processing and database management for the ZenithZone platform.

**5.2.5 Database Management System:**

A database management system (DBMS) like MySQL or MongoDB is employed to store and manage user data, product listings, and other information relevant to the ZenithZone platform.

**5.2.6 Version Control System:**

A version control system like Git is utilized to manage the source code, track changes, and facilitate collaboration among developers working on the ZenithZone project.

**5.2.7 Hosting Platform:**

A hosting platform such as Heroku, AWS (Amazon Web Services), or GCP (Google Cloud Platform) is selected to deploy and host the ZenithZone platform, making it accessible to users on the internet.

**5.2.8 Accessibility and Testing Tools:**

The ZenithZone platform is tested using tools like Lighthouse, Pa11y, and aXe to ensure accessibility compliance and usability for users with diverse abilities.

**5.2.9 Project Management Tools:**

Project management tools like Trello, Asana, or Jira are utilized to organize tasks, track progress, and facilitate communication within the development team.

**5.3 Web Server**

The Apache HTTP server is chosen as the web server to serve files and respond to user requests for the ZenithZone platform. Apache is renowned for its stability, security, and scalability, making it an ideal choice for hosting web applications.

**5.4 Database Server**

MySQL is selected as the database server to store and manage all relevant information for the ZenithZone platform. MySQL is preferred for its security features, reporting capabilities, fault tolerance, and performance diagnostics, ensuring robust data management for the platform.

## 6. Use Case Diagram

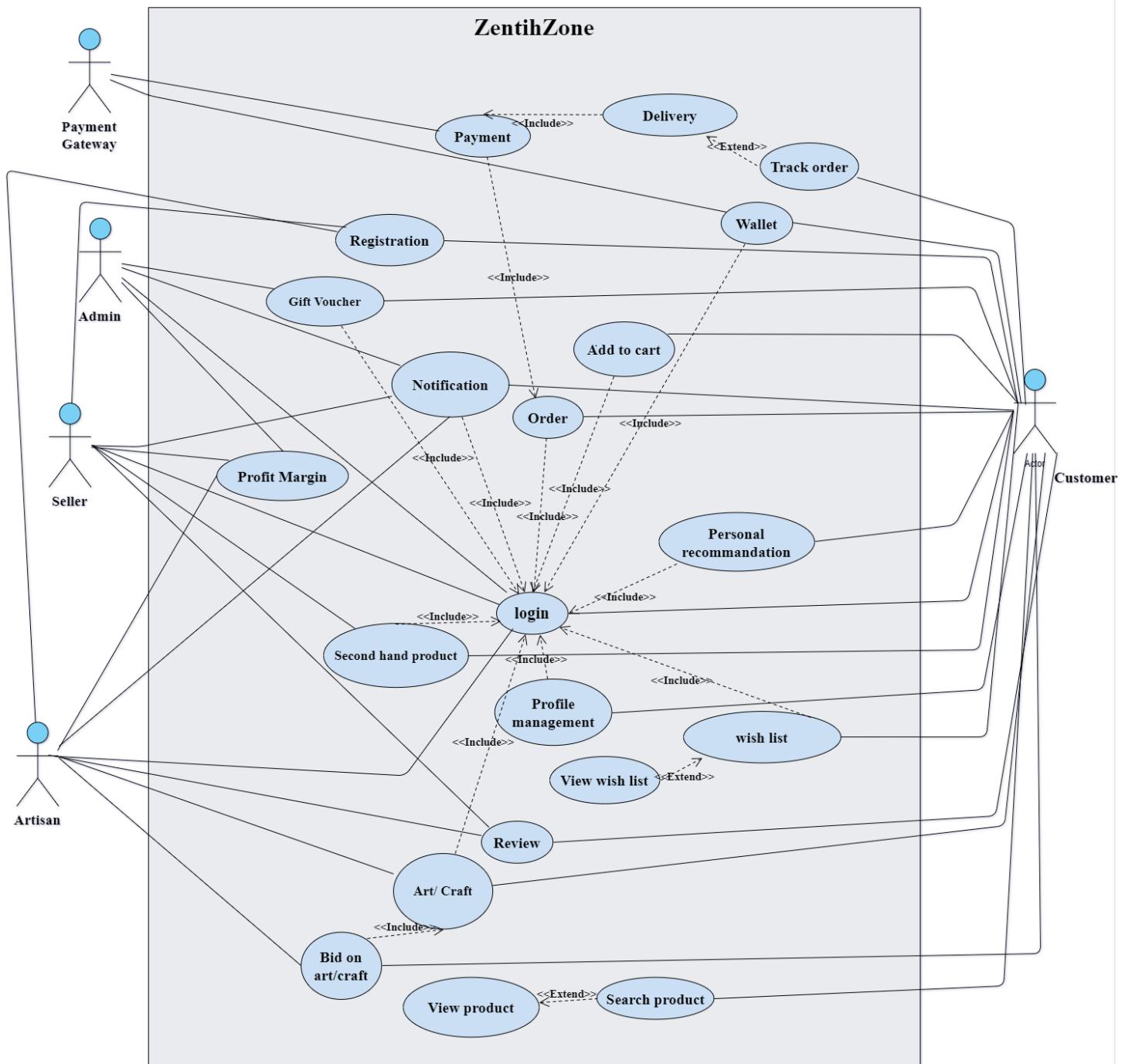


Figure 3: Use Case Diagram of ZenithZone

## 7. Use Case Description

**Table 4: Registration**

<b>Use Case No</b>	<b>01</b>	
<b>Use Case Name</b>	Registration	
<b>Goal</b> <a longer statement of the goal in context if needed>	Register a user (customer, seller, artisan) into the system.	
<b>Preconditions</b> <what we expect is already the state of the world>	Must have valid information	
<b>Success End Condition</b> <the state of the world upon successful completion>	Registered into the system successfully.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	Registration failed.	
<b>Primary Actors:</b>	User (customer, seller, artisan)	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	The user (customer, seller, artisan) clicks the “Registration” button.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (customer, seller, artisan) accesses the registration page.
	<b>2</b>	The system presents the registration form to the user (customer, seller, artisan).
	<b>3</b>	The user fills in the required registration details based on their chosen registration type <ul style="list-style-type: none"> <li>• <b>Customer:</b> Name, email, mobile number, password.</li> <li>• <b>Seller:</b> Name, email, mobile number, password, National Identification Number (NID).</li> <li>• <b>Artisan:</b> Name, email, mobile number, password, National Identification Number (NID)</li> </ul>
	<b>4</b>	The user (customer, seller, artisan) submits the registration form
	<b>5</b>	<ul style="list-style-type: none"> <li>• The system verifies the submitted information by</li> </ul>

		<ul style="list-style-type: none"> <li>sending an OTP to the user's mobile number</li> <li>The user enters the received OTP into the system</li> <li>The system verifies the OTP.</li> <li>If the OTP is valid, and if the user is a seller or artisan, the system verifies the NID.</li> </ul>
	<b>6</b>	If the information is valid, the system creates a new account for the user.
	<b>7</b>	The system notifies the user (customer, seller, artisan) of successful registration.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>6a</b>	If the user (customer, seller, artisan) submits incomplete or invalid information, the system prompts the passenger to correct the errors.
	<b>6a1</b>	The user (customer, seller, artisan) corrects the information and resubmits the form.
	<b>6a2</b>	The system re-verifies the submitted information.
	<b>6a3</b>	If the information is now valid, the system proceeds with creating the account as described in the main flow.
<b>Quality Requirement</b>	<b>1</b>	The system must encrypt sensitive information (e.g., passwords) during transmission and storage to prevent unauthorized access.
	<b>2</b>	The registration form should have clear labels and instructions to guide users (customer, seller, artisan) through the process.
	<b>3</b>	The registration process should respond promptly to user input, with minimal latency.

*Table 5: Log in*

<b>Use Case No</b>	<b>02</b>
<b>Use Case Name</b>	<b>Log in</b>
<b>Goal</b> <a longer statement of the goal in context if needed>	Accessing the ZenithZone account
<b>Preconditions</b> <what we expect is already the state of the world>	The user (customer, seller, artisan) must have a registered account on ZenithZone
<b>Success End Condition</b> <the state of the world upon successful completion>	Successfully log in into the system.
<b>Failed End Condition</b> <the state of the world if goal abandoned>	Log in failed

<b>Primary Actors:</b>	User (customer, seller, artisan)	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	The user (customer, seller, artisan) clicks the “Log in” button.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	Steps	Action
	<b>1</b>	The user (customer, seller, artisan) navigates to the login page.
	<b>2</b>	The system presents the login form
	<b>3</b>	The user (customer, seller, artisan) enters their registered email and password
	<b>4</b>	The user (customer, seller, artisan) submits the registration form.
	<b>5</b>	The system verifies the entered credentials.
	<b>6</b>	The system grants access to the user (customer, seller, artisan) specific features of the system.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	Steps	Action
	<b>5a</b>	If the user (customer, seller, artisan) enters incorrect credentials, the system prompts them to re-enter the correct email and password combination.
	<b>5a1</b>	The user (customer, seller, artisan) corrects the credentials and resubmits the form.
	<b>5a2</b>	The system re-verifies the entered credentials.
	<b>5a3</b>	If the credentials are now valid, the system proceeds with granting access as described in the main flow.
<b>Quality Requirement</b>	<b>1</b>	The system must encrypt sensitive information (e.g., passwords) during transmission and storage to prevent unauthorized access.
	<b>2</b>	The registration form should have clear labels and instructions to guide users (customer, seller, artisan) through the process.
	<b>3</b>	The registration process should respond promptly to user (customer, seller, artisan) input, with minimal latency.

**Table 6: Profile management**

<b>Use Case No</b>	<b>03</b>	
<b>Use Case Name</b>	<b>Profile management</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	The goal of the Profile Management use case is to enable users to efficiently manage and maintain their personal information and account settings on ZenithZone.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user must be logged into their ZenithZone account.	
<b>Success End Condition</b> <the state of the world upon successful completion>	<ul style="list-style-type: none"> <li>· The user's profile information is successfully updated and saved in the system.</li> <li>· The user receives a confirmation message indicating that their changes have been applied</li> </ul>	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	<ul style="list-style-type: none"> <li>• The system fails to update the profile information due to invalid input, system errors, or network issues.</li> <li>• The user receives an error message explaining the issue and, if possible, steps to resolve or try again.</li> </ul>	
<b>Primary Actors:</b> <b>Secondary Actor:</b>	<ul style="list-style-type: none"> <li>• User (Customer)</li> <li>• System (ZenithZone platform)</li> </ul>	
<b>Trigger</b> <the action upon the system that starts the case>	Triggered by unusual account activity, prompting the user to review and update their profile information to secure their account.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	<ul style="list-style-type: none"> <li>• The user logs into their ZenithZone account.</li> <li>• The user navigates to the "Profile" or "Account Settings" section.</li> </ul>
	<b>2</b>	The system displays the user's current profile information, including personal details, contact information, and address.
	<b>3</b>	<ul style="list-style-type: none"> <li>• The user selects the information they wish to update, such as name, email address, phone number, password, or shipping address.</li> <li>• The user enters the updated information</li> </ul>
	<b>4</b>	<ul style="list-style-type: none"> <li>• The system validates the updated information to ensure it meets the required format and criteria.</li> <li>• If the information is valid, the system saves the changes to the database.</li> <li>• The system confirms the successful update to the user.</li> </ul>
<b>Alternative Flows</b>	<b>Steps</b>	<b>Action</b>

<a: condition causing branching> <a1: action or name of sub use case>	<b>3a</b>	The user enters invalid information (e.g., incorrect email format, weak password).
	<b>3a1</b>	The system detects the invalid input and displays an error message indicating the specific issue.
	<b>3a2</b>	The user is prompted to correct the information and resubmit.
	<b>3b</b>	A network or server issue prevents the system from saving the updated profile information.
	<b>3b1</b>	The system displays an error message indicating a network or server issue.
	<b>3b2</b>	The user is advised to try again later or contact customer support if the issue persists.
<b>Quality Requirement</b>	<b>1</b>	The profile information must be accurately saved and retrieved.
	<b>2</b>	Profile management actions must be protected against unauthorized access.
	<b>3</b>	The system should reliably update profile information without frequent errors or downtime

*Table 7: Search Product*

<b>Use Case No</b>	<b>04</b>	
<b>Use Case Name</b>	<b>Search Product</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Enable users (customer) to efficiently and accurately find the products they are looking for on the ZenithZone platform	
<b>Preconditions</b> <what we expect is already the state of the world>	The user (customer) must have access to the ZenithZone platform	
<b>Success End Condition</b> <the state of the world upon successful completion>	The system displays relevant product listings based on the user's (customer) search query.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	System failed to display relevant products.	
<b>Primary Actors:</b>	Customer	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	The user (customer) types a search query into the search bar and clicks on the search icon/button. or The user(customer) clicks on the microphone icon and speaks a search query into their device's microphone.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (customer) navigates to the search bar on the ZenithZone homepage

	<b>2</b>	The user (customer) types a keyword into the search bar or speaks the search query into their device's microphone
	<b>3</b>	The system processes the entered keyword.
	<b>4</b>	As the user (customer) types, the system displays auto-suggestions and relevant products based on the entered query
	<b>5</b>	The user (customer) either selects one of the auto-suggestions or completes the keyword/phrase and submits the search query.
	<b>6</b>	The system searches the product database for items matching the query
	<b>7</b>	The system displays a list of relevant product listings based on the search query
	<b>8</b>	The user (customer) browses through the displayed product listings
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>1a</b>	The user (customer) types a keyword into the search bar or speaks the search query into their device's microphone
	<b>1a1</b>	The system searches the product database for items matching the query
	<b>1a2</b>	If no matching products are found, the system displays a "No results found" message.
	<b>1a3</b>	The system provides alternative suggestions or related categories to assist the user in refining their search
	<b>1a4</b>	The system displays a list of relevant product listings based on the refined search query.
	<b>1a5</b>	The user (customer) browses through the displayed product listings
<b>Quality Requirement</b>	<b>1</b>	The search results must be highly relevant to the user's query, incorporating factors such as product names, descriptions, categories, and tags.
	<b>2</b>	The search bar should be easily accessible and user-friendly, with clear instructions or placeholders indicating where users (customer) can enter their queries.
	<b>3</b>	The system should filter out inappropriate or harmful content from the search results to ensure a safe user (customer) experience

**Table 8: Personalized Recommendations**

<b>Use Case No</b>	<b>05</b>	
<b>Use Case Name</b>	<b>Personalized Recommendations</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Allows the system to provide tailored product suggestions to users based on their hobby, gender, preferences, and past interactions on ZenithZone	
<b>Preconditions</b> <what we expect is already the state of the world>	User (customer) must have a registered account on ZenithZone and user (customer) must have interacted with the platform (e.g., browsing, searching, purchasing products).	
<b>Success End Condition</b> <the state of the world upon successful completion>	User (customer) receives a list of personalized product recommendations.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	User (customer) failed to receive a list of personalized product recommendations.	
<b>Primary Actors:</b>	User (customer)	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	The user (customer) logs into their ZenithZone account. This action prompts the system to start analyzing the user's past behavior and preferences to generate personalized recommendations.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (customer) logs into their ZenithZone account
	<b>2</b>	The system analyzes the user's past behavior, including browsing history, search queries, previous purchases, and wish list items.
	<b>3</b>	The system displays a list of personalized product recommendations on the user's(customer) homepage
	<b>4</b>	The user (customer) views the personalized product recommendations and can click on any product to see more details.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>1a</b>	The user (customer) logs into their ZenithZone account
	<b>1a1</b>	The system displays a list of personalized product recommendations on the user's(customer) homepage
	<b>1a2</b>	The user (customer) indicates that they are not interested in some of the recommended products (e.g., by clicking a "Not Interested" button)
	<b>1a3</b>	The system adjusts future recommendations by excluding similar products and learning from the user's (customer) feedback.

	<b>1a4</b>	The user (customer) views the new personalized product recommendations and can click on any product to see more details.
<b>Quality Requirement</b>	<b>1</b>	Recommendations must be highly relevant to the user's (customer) interests, preferences, and past behavior on the platform.
	<b>2</b>	Recommendations should be presented in a user-friendly manner, with clear and appealing visuals, product details, and easy access to more information or purchase options

*Table 9: Add to Cart*

<b>Use Case No</b>	<b>06</b>	
<b>Use Case Name</b>	<b>Add to cart</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Enable users (customer) to select and temporarily store products they intend to purchase in a virtual shopping cart, allowing for easy review and modification before proceeding to checkout.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user (customer) must be logged into their ZenithZone account	
<b>Success End Condition</b> <the state of the world upon successful completion>	The selected product is successfully added to the user's (customer) virtual shopping cart.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The selected product is not added to the user's (customer) shopping cart due to various potential issues.	
<b>Primary Actors:</b>	User (customer)	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	The user (customer) selects a product they are interested in and clicks the "Add to Cart" button on the product detail page.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (customer) navigates the ZenithZone platform, browsing through various products.
	<b>2</b>	The user (customer) selects a product they are interested in purchasing.
	<b>3</b>	The user (customer) selects any required options for the product (e.g., size, color, quantity)
	<b>4</b>	The user (customer) clicks the "Add to Cart" button on the product page.

	<b>5</b>	<b>5.1)</b> The system checks if the selected product and options are available in the desired quantity. <b>5.2)</b> If the product is available, the system proceeds to the next step <b>5.3)</b> If the product is unavailable, the system displays an appropriate message to the user.
	<b>6</b>	The system adds the selected product to the user's (customer) virtual shopping cart
	<b>7</b>	The system displays a confirmation message to the user, indicating that the product has been successfully added to the cart.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>5a</b>	The user (customer) selects a product and attempts to add it to the cart.
	<b>5a1</b>	If the product is unavailable, the system displays an out-of-stock message to the user (customer) and suggests similar available products.
	<b>5a2</b>	The user (customer) selects an alternative product or adjusts the options to find an available product.
	<b>5a3</b>	The system adds the newly selected product to the cart if available
	<b>5a4</b>	The system displays a confirmation message to the user, indicating that the alternative product has been successfully added to the cart.
<b>Quality Requirement</b>	<b>1</b>	The "Add to Cart" button should be clearly visible and accessible on product pages.
	<b>2</b>	The system should provide immediate feedback (e.g., confirmation message, cart update) when a product is added to the cart.
	<b>3</b>	Products added to the cart should remain in the cart until the user removes them or completes the purchase.

*Table 10: Wish list*

<b>Use Case No</b>	<b>07</b>
<b>Use Case Name</b>	<b>Wish list</b>
<b>Goal</b> <a longer statement of the goal in context if needed>	Enable users (customer) to create a personalized list of products they are interested in, allowing them to easily track and access these products for future consideration or purchase.
<b>Preconditions</b> <what we expect is already the state of the world>	The user (customer) must be logged into their ZenithZone account
<b>Success End Condition</b> <the state of the world upon successful completion>	The selected product is successfully added to the user's (customer) wish list

<b>Failed End Condition</b> <the state of the world if goal abandoned>	The selected product is failed to added to the user's (customer) wish list	
<b>Primary Actors:</b>	User (customer)	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	User (customer) clicks the "Add to Wish List" button on a product page.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	Steps	Action
	<b>1</b>	The user (customer) must be logged into their ZenithZone account
	<b>2</b>	The user (customer) navigates the ZenithZone platform, browsing various products.
	<b>3</b>	The user (customer) selects a product they are interested in
	<b>4</b>	The user (customer) clicks the "Add to Wish List" button on the product page
	<b>5</b>	The system verifies that the user (customer) is logged in. If the user (customer) is not logged in, the system prompts them to log in
	<b>6</b>	The system adds the selected product to the user's (customer) wish list
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	Steps	Action
	<b>3a</b>	The user (customer) selects a product they are interested in
	<b>3a1</b>	The system checks if the product is already in the user's (customer) wish list.
	<b>3a2</b>	If the product is already in the wish list, the system displays a message indicating that the product is already saved in the wish list.
<b>Quality Requirement</b>	<b>1</b>	The "Add to Wish List" button should be clearly visible and accessible on product pages.
	<b>2</b>	The system should provide immediate feedback (e.g., confirmation message, cart update) when a product is added to the wish list.
	<b>3</b>	Products added to the wish list should remain there until the user (customer) removes them or completes the purchase.

Table 11: View Wish List

<b>Use Case No</b>	<b>08</b>	
<b>Use Case Name</b>	<b>View Wish List</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Allow users to access and review the products they have previously added to their wish list.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user (customer) must be logged into their ZenithZone account	
<b>Success End Condition</b> <the state of the world upon successful completion>	The system successfully retrieves and displays the user's (customer) wish list, showing all the products the user has added, along with relevant details (e.g., product name, image, price, availability).	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The system failed to display the user's (customer) wish list, showing all the products the user has added, along with relevant details (e.g., product name, image, price, availability).	
<b>Primary Actors:</b>	User (customer)	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	User (customer) selects the "Wish List" option from the website's navigation menu	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (customer) must be logged into their ZenithZone account.
	<b>2</b>	The user (customer) navigates to the wish list section of their account.
	<b>3</b>	The system retrieves the user's (customer) wish list from the database.
	<b>4</b>	The user (customer) reviews the items in their wish list, with options to view product details, remove items, or move items to the shopping cart .
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>2a</b>	The user (customer) navigates to the wish list section of their account.
	<b>2a1</b>	The system retrieves the user's (customer) wish list from the database.
	<b>2a2</b>	The system finds that there are no items in the wish list
	<b>2a3</b>	The system displays a message indicating that the wish list is empty and suggests browsing products to add to the wish list.
<b>Quality Requirement</b>	<b>1</b>	The wish list section should be easily accessible and user-friendly, with a clear layout and navigation options.

	<b>2</b>	The system should provide immediate feedback when the wish list is displayed or if there are any issues.
	<b>3</b>	The wish list should reflect real-time updates, showing the most current product details (e.g., price, availability).

**Table 12: Second Hand Product**

<b>Use Case No</b>	<b>09</b>	
<b>Use Case Name</b>	<b>Secondhand product</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Enable users (sellers) to list and sell their pre-owned items, providing a marketplace for customers to discover and purchase second-hand products at affordable prices.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user (seller) must be logged into their ZenithZone account	
<b>Success End Condition</b> <the state of the world upon successful completion>	The selected second-hand product is successfully listed by the seller and made available for customers to browse and purchase.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The selected second-hand product failed to be listed by the seller and is not available for customers to browse and purchase.	
<b>Primary Actors:</b>	User (Seller, Customer)	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	User (customer) clicks the "Purchase Second hand" button on a product page.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (Seller) must be logged into their ZenithZone account.
	<b>2</b>	The user (Seller) finalizes their products that they are interested to sell.
	<b>3</b>	The user (Seller) communicates with ZenithZone to sell it.
	<b>4</b>	The user (customer) clicks the "Purchase Second Hand" button on the product page.
	<b>5</b>	The system verifies that the user (customer) is logged in. If the user (customer) is not logged in, the system prompts them to log in
	<b>6</b>	The system communicates with customer to sell the second-hand product.
	<b>7</b>	The system displays a confirmation message to the user (customer) that they are confirmed to buy the product.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>4a</b>	The status of sold product has not been updated.
	<b>4a1</b>	The system shows invalid.

<b>Quality Requirement</b>	<b>1</b>	The "Purchase Second Hand" button should be clearly visible and accessible.
	<b>2</b>	The system should provide immediate feedback (e.g., confirmation message) when the product is sold or purchase.

*Table I3: Art/Craft*

<b>Use Case No</b>	<b>10</b>	
<b>Use Case Name</b>	<b>Art/Craft</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	The goal of the Art and Craft use case is to provide a platform for artisans and craftsmen to showcase and sell their artworks and handcrafted items to a broader audience.	
<b>Preconditions</b> <what we expect is already the state of the world>	<ul style="list-style-type: none"> <li>• The artisan/craftsman must be registered and logged into their ZenithZone account.</li> <li>• The artwork to be submitted must meet the platform's guidelines and standards.</li> </ul>	
<b>Success End Condition</b> <the state of the world upon successful completion>	<ul style="list-style-type: none"> <li>• The artwork is sold to the highest bidder.</li> <li>• The artisan receives the payment minus the platform's commission.</li> <li>• The customer receives the purchased artwork.</li> </ul>	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	<ul style="list-style-type: none"> <li>• No bids meet the minimum price set by the artisan.</li> <li>• The artwork is not sold, and the artisan is notified to either relist the item or adjust the price/bidding parameters.</li> </ul>	
<b>Primary Actors:</b>	User (Artisan/Craftsman)	
<b>Secondary Actor:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	The Art and Craft use case is triggered when an artisan or craftsman submits a new piece of artwork or a craft item to be showcased on ZenithZone's platform.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	1	<ul style="list-style-type: none"> <li>• The artisan logs into their ZenithZone account.</li> <li>• The artisan navigates to the submission page and uploads details and images of their artwork.</li> </ul>
	2	<ul style="list-style-type: none"> <li>• The system reviews the submission to ensure it meets the platform's guidelines and standards.</li> <li>• If approved, the artwork is listed for exhibition with a set minimum price.</li> </ul>
	3	<ul style="list-style-type: none"> <li>• Customers browse the art and craft exhibition section.</li> <li>• Interested customers place bids on the artwork.</li> </ul>

		during the specified bidding period.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>1a</b>	The system finds the artwork submission does not meet guidelines.
	<b>1a1</b>	The system notifies the artisan of the issues and suggests corrections
	<b>1a2</b>	The artisan corrects the submission and resubmits it for approval
<b>Quality Requirement</b>	<b>1</b>	The system must accurately track and record bids, ensuring the highest bid is correctly identified and processed.
	<b>2</b>	All transactions and data related to artworks, bids, and payments must be securely handled to protect users and artisans.
	<b>3</b>	The system should reliably manage the exhibition and bidding process without downtime or errors.

*Table 14: Bid on Art/Craft.*

<b>Use Case No</b>	<b>11</b>	
<b>Use Case Name</b>	<b>Bid on Art/Craft.</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Enable user(customer) to participate in the bidding process for showcased artworks and crafts, providing a platform for artisans and craftsmen to sell their works through an organized auction process.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user(customer) must be logged into their ZenithZone account.	
<b>Success End Condition</b> <the state of the world upon successful completion>	The user's (customer's) bid is successfully placed, and they are notified if they have won the auction at the end of the bidding period.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The user's (customer's) bid fails to be placed, and they are notified of the error.	
<b>Primary Actors:</b>	User (Customer, Artisans/Craftsman)	
<b>Secondary Actors:</b>	System	
<b>Trigger</b> <the action upon the system that starts the case>	User (customer) clicks the "Place Bid" button on an artwork's auction page.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (Customer) must be logged into their ZenithZone account.
	<b>2</b>	The user (customer) navigates to the artwork or craft they are interested in bidding on.

	<b>3</b>	The user (customer) clicks the "Place Bid" button on the artwork's auction page.
	<b>4</b>	The system prompts the user (customer) to enter their bid amount.
	<b>5</b>	The user (customer) enters their bid amount and confirms the bid.
	<b>6</b>	The system verifies that the bid amount is higher than the current highest bid and within the auction rules.
	<b>7</b>	The system will set a specific time for bidding.
	<b>8</b>	The system notifies the user (customer) that their bid has been successfully placed.
	<b>9</b>	At the end of the auction period, the system notifies the winning bidder (customer) and the artisan/craftsman of the auction result.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>3a</b>	The user (customer) decides not to place a bid and leaves the auction page.
	<b>3a1</b>	The system records no action and the auction continues.
	<b>6a</b>	The bid amount entered is not higher than the current highest bid.
	<b>6a1</b>	The system displays an error message prompting the user (customer) to enter a higher bid.
<b>Quality Requirement</b>	<b>1</b>	The "Place Bid" button should be clearly visible and accessible.
	<b>2</b>	The system should provide immediate feedback (e.g., confirmation message) when a bid is placed successfully or if an error occurs.
	<b>3</b>	The auction status and highest bid should be updated in real-time to ensure all users have the latest information.

**Table 15: Profit Margin**

<b>Use Case No</b>	<b>12</b>
<b>Use Case Name</b>	<b>Profit Margin</b>
<b>Goal</b> <a longer statement of the goal in context if needed>	The primary goal of the "Process Commission" use case is to ensure the accurate and efficient calculation, deduction, and recording of commission fees from sellers' transactions.
<b>Preconditions</b> <what we expect is already the state of the world>	<ul style="list-style-type: none"> <li>• A product sale transaction must be completed.</li> <li>• The seller's account must be verified and active.</li> </ul>

<b>Success End Condition</b> <the state of the world upon successful completion>	<ul style="list-style-type: none"> <li>The commission is accurately calculated and deducted from the seller's earnings.</li> </ul>	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	<ul style="list-style-type: none"> <li>The commission is not correctly calculated or deducted, resulting in either overcharging or undercharging the seller.</li> <li>The transaction details are not recorded correctly in the platform's financial records, leading to discrepancies.</li> <li>The seller does not receive a notification, causing confusion or lack of transparency.</li> </ul>	
<b>Primary Actors:</b> <b>Secondary Actor:</b>	System Seller	
<b>Trigger</b> <the action upon the system that starts the case>	The Process Commission use case is triggered when a transaction involving the sale of a product (whether new, second-hand, or artisanal) is completed on the ZenithZone platform.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	Steps	Action
	1	The system detects that a sale transaction has been successfully completed.
	2	The system calculates the commission based on the predefined percentage of the selling price.
	3	The system deducts the calculated commission from the seller's earnings.
	4	The system records the transaction details, including the commission amount, in the platform's financial records.
	5	The system notifies the seller of the transaction completion and the commission deducted.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	Steps	Action
	3a	The system encounters an error in automatically deducting the commission due to a calculation or system issue
	3a1	The system detects that a sale transaction has been successfully completed.
	3a2	The system calculates the commission based on the predefined percentage of the selling price.
<b>Quality Requirement</b>	1	<ul style="list-style-type: none"> <li>The commission must be accurately calculated based on the predefined percentage for each transaction.</li> <li>All calculations should be free from errors, ensuring sellers are neither overcharged nor undercharged.</li> </ul>
	2	<ul style="list-style-type: none"> <li>The commission processing system should be dependable, consistently performing calculations and deductions without failure.</li> <li>The system should handle peak loads efficiently, ensuring that all transactions are</li> </ul>

		processed in a timely manner.
	3	<ul style="list-style-type: none"> <li>The system must ensure the security of financial data during commission processing.</li> <li>Sensitive information related to transactions and earnings must be protected against unauthorized access and breaches</li> </ul>

**Table 16: Wallet System**

<b>Use Case No</b>	13	
<b>Use Case Name</b>	<b>Wallet System</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Enable users (customers) to deposit funds into their ZenithZone account, creating a wallet balance for convenient and quick checkout, streamlining transactions and offering flexibility in managing their purchases.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user(customer) must be logged into their ZenithZone account.	
<b>Success End Condition</b> <the state of the world upon successful completion>	The user's (customer's) funds are successfully deposited into their wallet, and the balance is updated for use in future transactions.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The user's (customer's) funds fail to be deposited into their wallet, and an error message is displayed.	
<b>Primary Actors:</b>	User (Customer)	
<b>Secondary Actors:</b>	System, Payment gateway.	
<b>Trigger</b> <the action upon the system that starts the case>	User (customer) clicks the "Add Funds" button in their wallet section.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	1	The user (Customer) must be logged into their ZenithZone account.
	2	The user (customer) navigates to the wallet section.
	3	The user (customer) clicks the "Add Funds" button.
	4	The system prompts the user (customer) to enter the amount to be added.
	5	The user (customer) enters the amount and selects a payment method.
	6	The system processes the payment through the selected payment gateway.
	7	The payment gateway verifies the transaction and confirms the transfer of funds.
	8	The system updates the user's wallet balance with the added funds.

	<b>9</b>	The system notifies the user (customer) that the funds have been successfully added to their wallet.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>3a</b>	The user (customer) decides not to add funds and leaves the wallet section.
	<b>3a1</b>	The system records no action and the wallet balance remains unchanged.
	<b>6a</b>	The payment fails due to insufficient funds or other payment issues.
	<b>6a1</b>	The system displays an error message and prompts the user (customer) to retry or select a different payment method.
<b>Quality Requirement</b>	<b>1</b>	The "Add Funds" button should be clearly visible and accessible.
	<b>2</b>	The wallet balance should be updated in real-time to reflect the added funds accurately.

**Table 17: Gift Voucher**

<b>Use Case No</b>	<b>14</b>	
<b>Use Case Name</b>	<b>Gift Voucher</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Allow user(customer) to use gift vouchers for discounts or special offers, making them happier and encouraging them to keep shopping on ZenithZone with exclusive benefits and savings.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user(customer) must be logged into their ZenithZone account. The gift voucher must be valid and applicable to the user's purchase.	
<b>Success End Condition</b> <the state of the world upon successful completion>	The user's (customer's) gift voucher is successfully applied.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The user's (customer's) gift voucher fails to be applied, and an error message is displayed.	
<b>Primary Actors:</b>	User (Customer)	
<b>Secondary Actors:</b>	System.	
<b>Trigger</b> <the action upon the system that starts the case>	User (customer) enters the gift voucher code during the checkout process.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (Customer) must be logged into their ZenithZone account.
	<b>2</b>	The user (customer) adds items to their cart and proceeds to checkout.
	<b>3</b>	The user (customer) enters the gift voucher code in the designated field.

	<b>4</b>	The system validates the gift voucher code for authenticity and applicability.
	<b>5</b>	The system notifies the user (customer) that the gift voucher has been successfully applied.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>3a</b>	The user (customer) decides not to enter a gift voucher code and proceeds with the checkout.
	<b>3a1</b>	The system processes the order without applying any discount or promotion.
	<b>4a</b>	The gift voucher code entered is invalid or expired.
	<b>4a1</b>	The system displays an error message prompting the user (customer) to enter a valid gift voucher code.
<b>Quality Requirement</b>	<b>1</b>	The gift voucher code entry field should be clearly visible and accessible during the checkout process.
	<b>2</b>	The system should provide immediate feedback (e.g., confirmation or error message) when the gift voucher code is entered.

*Table 18: Delivery Options*

<b>Use Case No</b>	<b>15</b>	
<b>Use Case Name</b>	<b>Delivery Options</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Allow users (customers) to select their preferred delivery option, either Express Delivery for urgent needs or Regular Delivery for cost savings, ensuring flexibility and satisfaction with the shipping process.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user (customer) must be logged into their ZenithZone account. Items must be added to the cart and ready for checkout.	
<b>Success End Condition</b> <the state of the world upon successful completion>	The user's (customer's) chosen delivery option is successfully applied to their order.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The user's (customer's) delivery option fails to be applied, and an error message is displayed	
<b>Primary Actors:</b>	User (Customer)	
<b>Secondary Actors:</b>	System.	
<b>Trigger</b> <the action upon the system that starts the case>	User (customer) clicks the "Choose Delivery Option" button in the checkout section.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (Customer) must be logged into their ZenithZone account.
	<b>2</b>	The user (customer) adds items to their cart

		and proceeds to checkout.
	<b>3</b>	The user (customer) is presented with delivery options (Express Delivery and Regular Delivery).
	<b>4</b>	The user (customer) selects their preferred delivery option.
	<b>5</b>	The system calculates the shipping cost based on the selected delivery option.
	<b>6</b>	The system updates the order total to include the shipping cost.
	<b>7</b>	The user (customer) reviews the order summary, including the selected delivery option and shipping cost.
	<b>8</b>	The user (customer) confirms and places the order.
	<b>9</b>	The system notifies the user (customer) that the order, including the selected delivery option, has been successfully placed.
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>3a</b>	The user (customer) decides not to select a delivery option and abandons the checkout process.
	<b>3a1</b>	The system records no action, and the order remains pending without a selected delivery option.
	<b>4a</b>	The user (customer) selects an unavailable delivery option.
	<b>4a1</b>	The system displays an error message prompting the user (customer) to select a different delivery option.
<b>Quality Requirement</b>	<b>1</b>	The delivery options should be clearly visible and accessible during the checkout process.
	<b>2</b>	The shipping cost should be accurately calculated and updated in real-time based on the selected delivery option.

Table 19: Checkout and Payment

<b>Use Case No</b>	<b>16</b>	
<b>Use Case Name</b>	<b>Checkout and payment</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	Enable users (customers) to complete their purchases safely and conveniently using various payment methods, ensuring a smooth and secure transaction process.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user(customer) must be logged into their ZenithZone account.	
<b>Success End Condition</b> <the state of the world upon successful completion>	The user's (customer's) payment is successfully processed, and the order is confirmed.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The user's (customer's) payment fails to be processed, and an error message is displayed.	
<b>Primary Actors:</b>	User (Customer)	
<b>Secondary Actors:</b>	Payment Gateway System.	
<b>Trigger</b> <the action upon the system that starts the case>	User (customer) clicks the "Proceed to Payment" button during the checkout process.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	The user (Customer) must be logged into their ZenithZone account.
	<b>2</b>	The user (customer) adds items to their cart and proceeds to checkout.
	<b>3</b>	The user (customer) reviews the order summary and clicks the "Proceed to Payment" button.
	<b>4</b>	The system presents the user (customer) with various payment options (e.g., Nagad, bKash, Rocket, Visa, Mastercard).
	<b>5</b>	The user (customer) selects a payment method and enters the necessary payment details.
	<b>6</b>	The system securely transmits the payment information to the selected payment gateway.
	<b>7</b>	The payment gateway processes the payment and communicates with the respective bank or mobile banking provider for authorization.
	<b>8</b>	The payment gateway confirms the transaction status (approved or declined).
	<b>9</b>	The system updates the order status based on the payment confirmation and displays a success message to the user (customer).
	<b>10</b>	The system sends an order confirmation email to the user (customer).
<b>Alternative Flows</b>	<b>Steps</b>	<b>Action</b>

<a: condition causing branching> <a1: action or name of sub use case>	<b>3a</b>	The user (customer) decides not to proceed with the payment and abandons the checkout process.
	<b>3a1</b>	The system records no action, and the order remains pending without payment.
	<b>5a</b>	The user (customer) selects a payment method but enters incorrect or incomplete payment details.
	<b>5a1</b>	The system displays an error message and prompts the user (customer) to re-enter the payment details.
	<b>8a</b>	The payment is declined due to insufficient funds or other issues.
	<b>8a1</b>	The system displays an error message and prompts the user (customer) to select a different payment method or retry the payment.
<b>Quality Requirement</b>	<b>1</b>	The "Proceed to Payment" button should be clearly visible and accessible.
	<b>2</b>	The system should provide immediate feedback (e.g., confirmation or error message) when payment information is entered and processed.
	<b>3</b>	Payment information should be securely transmitted and handled to ensure user privacy and security.
	<b>4</b>	The order status should be updated in real-time based on payment confirmation to reflect the accurate transaction state.

*Table 20: Notification*

<b>Use Case No</b>	<b>17</b>
<b>Use Case Name</b>	<b>Notification</b>
<b>Goal</b> <a longer statement of the goal in context if needed>	Keep customer informed & engaged by providing updates timely
<b>Preconditions</b> <what we expect is already the state of the world>	User must have a registered account
<b>Success End Condition</b> <the state of the world upon successful completion>	Gets notification timely
<b>Failed End Condition</b> <the state of the world if goal abandoned>	Did not get notification

<b>Primary Actors:</b> <b>Secondary Actor:</b>	User (Customer, Seller, Artisan) System	
<b>Trigger</b> <the action upon the system that starts the case>	The users (customer, seller, artisan) status updates for orders.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>
	<b>1</b>	System identifies an event that triggers a notification (such as a new promotion, order status update)
	<b>2</b>	System generates a new notification message according to the event
	<b>3</b>	The system sends the notification to the customers registered email address or phone number.
	<b>4</b>	The customer receives the notification on their device
	<b>5</b>	The customer reads the notification & takes the suggested actions
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>Steps</b>	<b>Action</b>
	<b>3a</b>	Notification cannot be delivered due to invalid contact information.
	<b>3a1</b>	System attempts to send the notification but fails due to invalid email address or phone number.
	<b>3a2</b>	System logs the failure and may send an alert to the customer's account dashboard or prompt the customer to update their contact details the next time they log in.
	<b>3a3</b>	Customer updates their contact information in their account settings
	<b>3a4</b>	System retries sending the notification, if applicable
<b>Quality Requirement</b>	<b>1</b>	Notifications should be delivered consistently without failures.
	<b>2</b>	Notifications should be personalized and relevant to the user's interests and activities on the Zenithzone
	<b>3</b>	The content of notifications should be accurate and free from errors

**Table 21: Tracks order**

<b>Use Case No</b>	<b>18</b>		
<b>Use Case Name</b>	<b>Tracks order</b>		
<b>Goal</b> <a longer statement of the goal in context if needed>	The main goal of track order use case is to track a customers order whether as the order is coming on its way		
<b>Preconditions</b> <what we expect is already the state of the world>	<ul style="list-style-type: none"> <li>The customer must have placed an order on ZenithZone.</li> <li>The customer must be logged into their ZenithZone account</li> </ul>		
<b>Success End Condition</b> <the state of the world upon successful completion>	<ul style="list-style-type: none"> <li>The customer successfully views the current status, shipment details, and estimated delivery date of their selected order.</li> <li>The information is accurate and up-to-date.</li> </ul>		
<b>Failed End Condition</b> <the state of the world if goal abandoned>	<ul style="list-style-type: none"> <li>The customer is unable to view the order status due to various reasons such as system errors, network issues, or the order not being found.</li> <li>The system displays an appropriate error message explaining the issue and, if possible, provides steps to resolve or seek further assistance.</li> </ul>		
<b>Primary Actors:</b> <b>Secondary Actor:</b>	User (Customer) System (ZenithZone platform)		
<b>Trigger</b> <the action upon the system that starts the case>	When a user tracks an order		
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up after>	<b>Steps</b>	<b>Action</b>	
	<b>1</b>	<ul style="list-style-type: none"> <li>The customer logs into their ZenithZone account.</li> <li>The customer navigates to the "My Orders" section.</li> </ul>	
	<b>2</b>	<ul style="list-style-type: none"> <li>The system displays a list of the customer's recent orders.</li> <li>The customer selects the order they want to track</li> </ul>	
	<b>3</b>	<ul style="list-style-type: none"> <li>The system retrieves the latest tracking information from the delivery service.</li> <li>The system displays the current status of the order, including shipment progress, current location, and estimated delivery date.</li> </ul>	
		<b>4</b> The customer views the tracking details and confirms the current status of their order.	

Alternative Flows <a: condition causing branching> <a1: action or name of sub use case>	Steps	Action
	<b>2a</b>	The system fails to retrieve tracking information from the delivery service.
	<b>2a1</b>	The system displays a message indicating that tracking information is currently unavailable.
	<b>2a2</b>	The customer is provided with options to try again later or contact customer support
<b>Quality Requirement</b>	<b>1</b>	The tracking information must be accurate and up-to-date.
	<b>2</b>	The system should consistently provide tracking information without errors.
	<b>3</b>	The system must ensure that only authenticated users can access tracking information for their orders.
	<b>4</b>	The tracking information must be retrieved and displayed promptly.

Table 22: Review

<b>Use Case No</b>	<b>19</b>	
<b>Use Case Name</b>	<b>Review</b>	
<b>Goal</b> <a longer statement of the goal in context if needed>	The goal of the Review use case is to facilitate customers in providing feedback on products showcased on ZenithZone, enabling them to rate and review products to share their experiences and opinions with others.	
<b>Preconditions</b> <what we expect is already the state of the world>	The user must be logged into their ZenithZone account. The user must have purchased the product they wish to review.	
<b>Success End Condition</b> <the state of the world upon successful completion>	The user's (customer's) review and rating are successfully submitted and recorded for the respective product.	
<b>Failed End Condition</b> <the state of the world if goal abandoned>	The user's (customer's) review or rating fails to be submitted, and they are notified of the error.	
<b>Primary Actors:</b> <b>Secondary Actor:</b>	User (Customer) System	
<b>Trigger</b> <the action upon the system that starts the case>	The Review use case is triggered when a user (customer) navigates to a product page and clicks on the "Write a Review" or "Rate this Product" option.	
<b>Main Success Flows</b> <the steps of the scenario from trigger to goal delivery and any clean up	<b>Steps</b>	<b>Action</b>
	1	The user (Customer) must be logged into their ZenithZone account.

after>	<b>2</b>	The user (customer) navigates to the product page of the item they wish to review.
	<b>3</b>	The user (customer) clicks on the "Write a Review" or "Rate this Product" option.
	<b>4</b>	The system prompts the user (customer) to provide their rating and write their review.
	<b>5</b>	The user (customer) enters their rating (if applicable) and writes their review in the provided text box.
	<b>6</b>	The user (customer) submits their review and rating.
	<b>7</b>	The system verifies the submitted data and records the review and rating for the respective product.
	<b>Steps</b>	<b>Action</b>
<b>Alternative Flows</b> <a: condition causing branching> <a1: action or name of sub use case>	<b>5a</b>	The user (customer) decides not to write a review or rate the product.
	<b>7a</b>	The system records no action, and the user continues browsing the product page.
<b>Quality Requirement</b>	<b>1</b>	The "Write a Review" or "Rate this Product" option should be clearly visible and easily accessible on the product page.
	<b>2</b>	The system should provide immediate feedback (e.g., confirmation message) when a review and rating are successfully submitted or if an error occurs.
	<b>3</b>	The review and rating should be accurately recorded and displayed on the product page in real-time to provide transparency and aid other customers in making informed decisions.

## 8. Activity Diagram

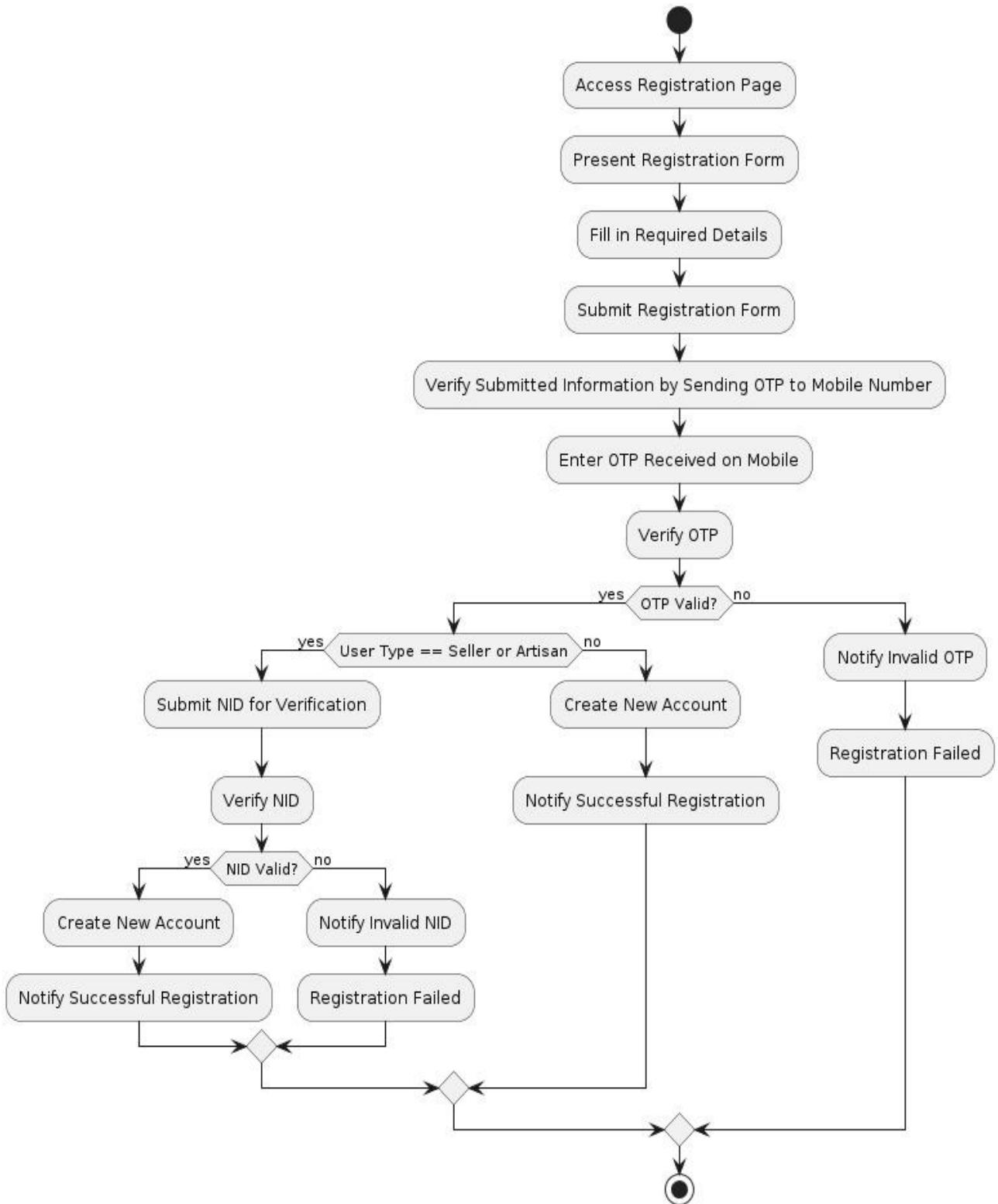


Figure 4: Activity Diagram for Registration Request

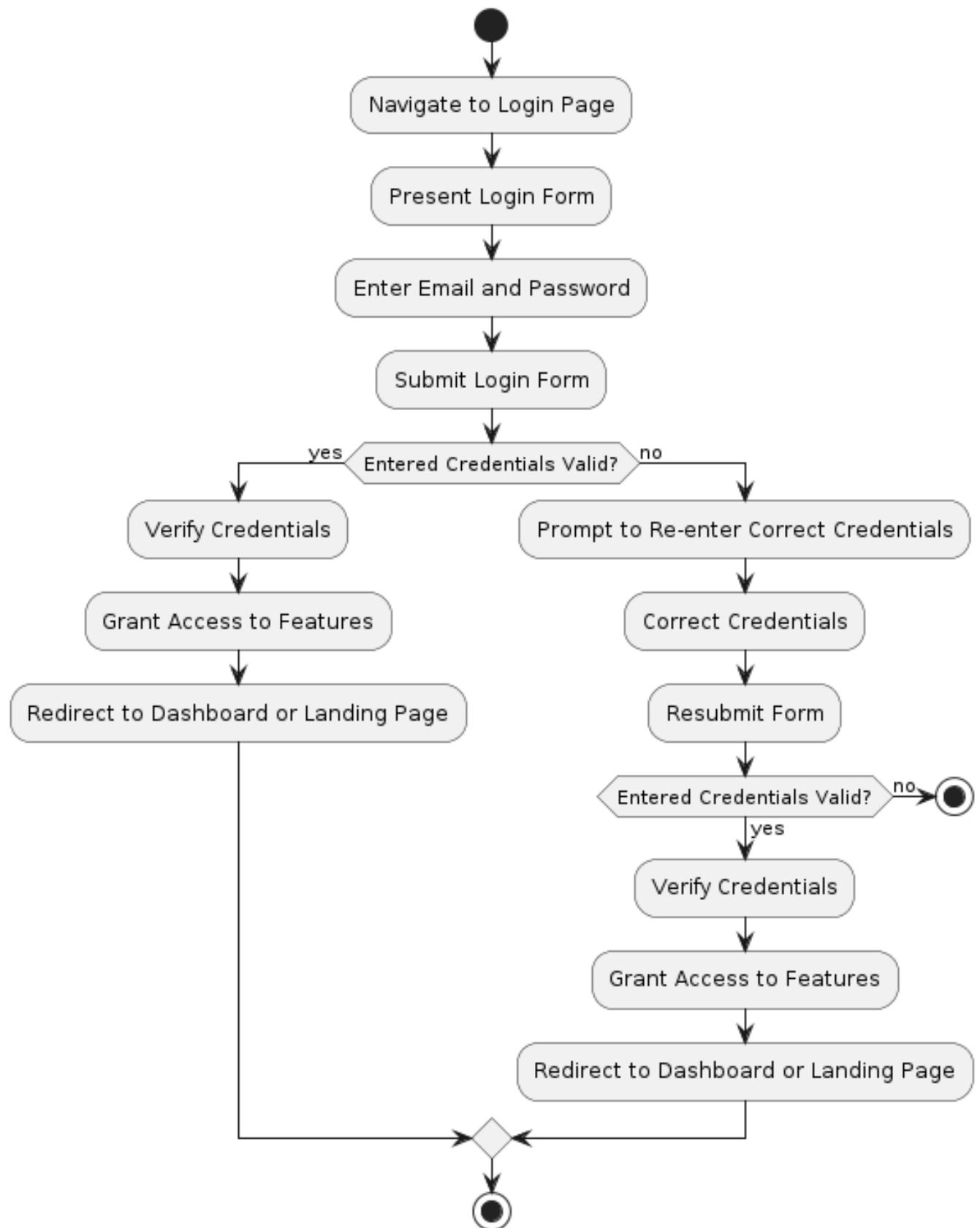


Figure 5: Activity Diagram for Login

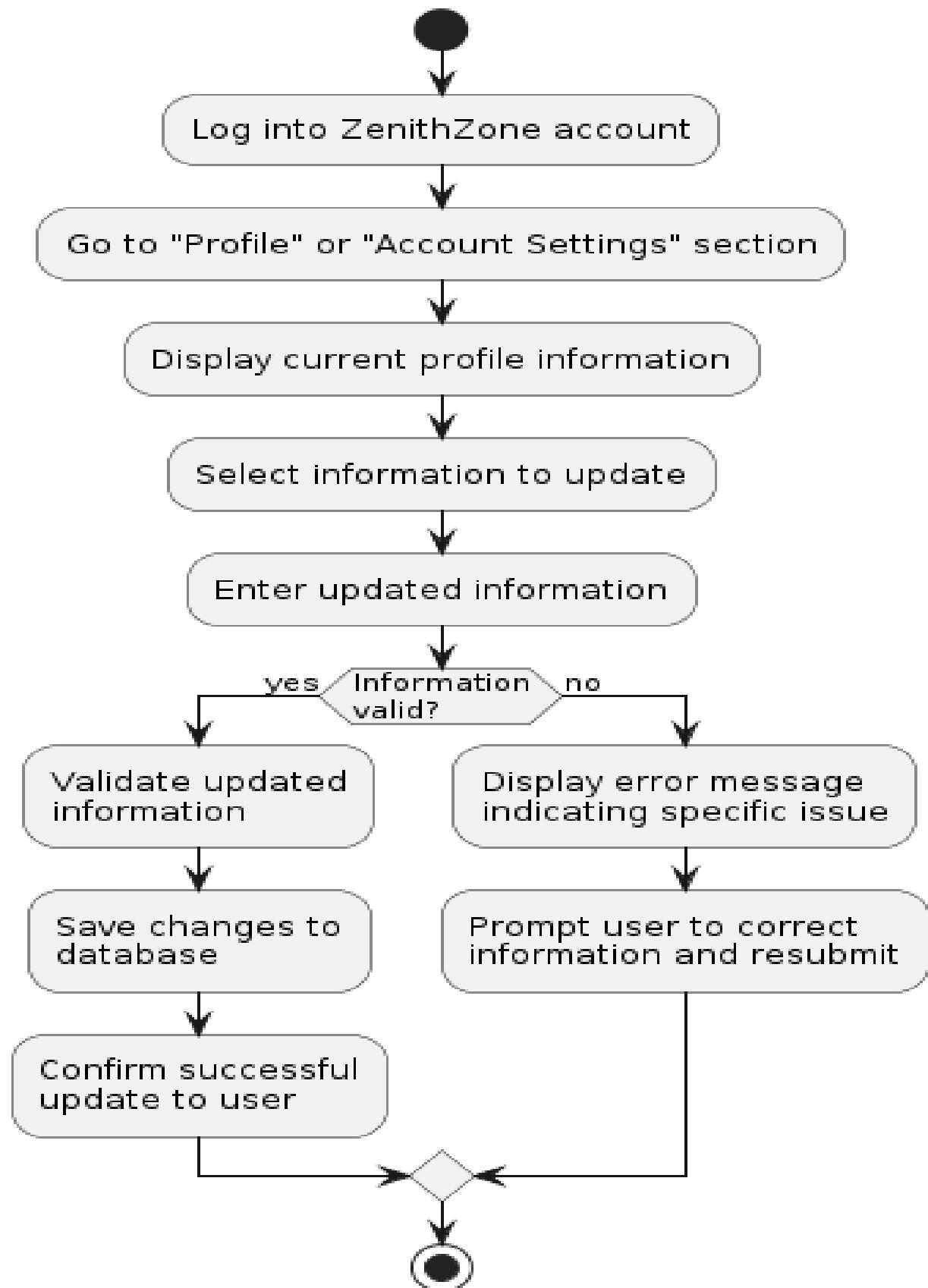


Figure 6: Activity Diagram for Profile Management

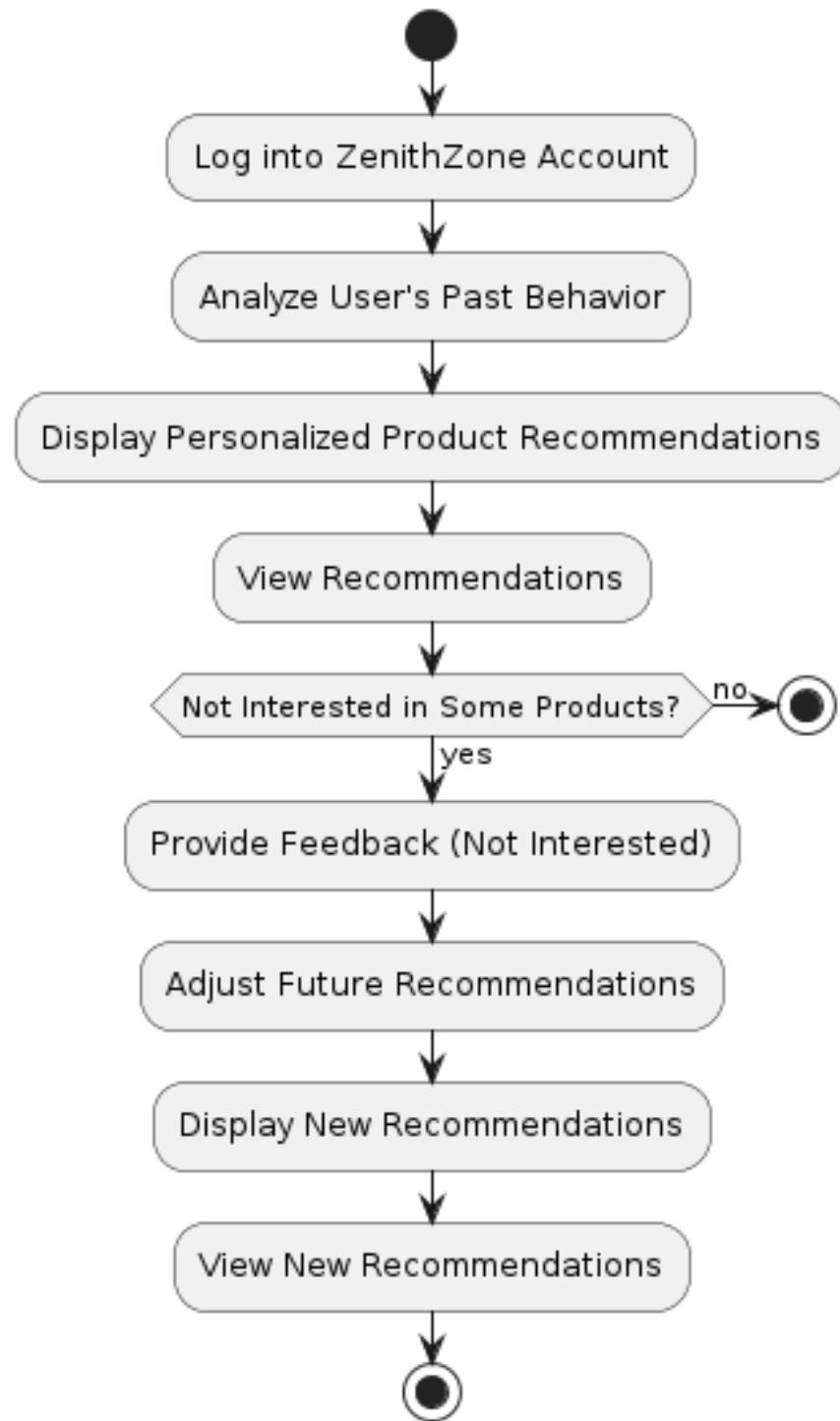
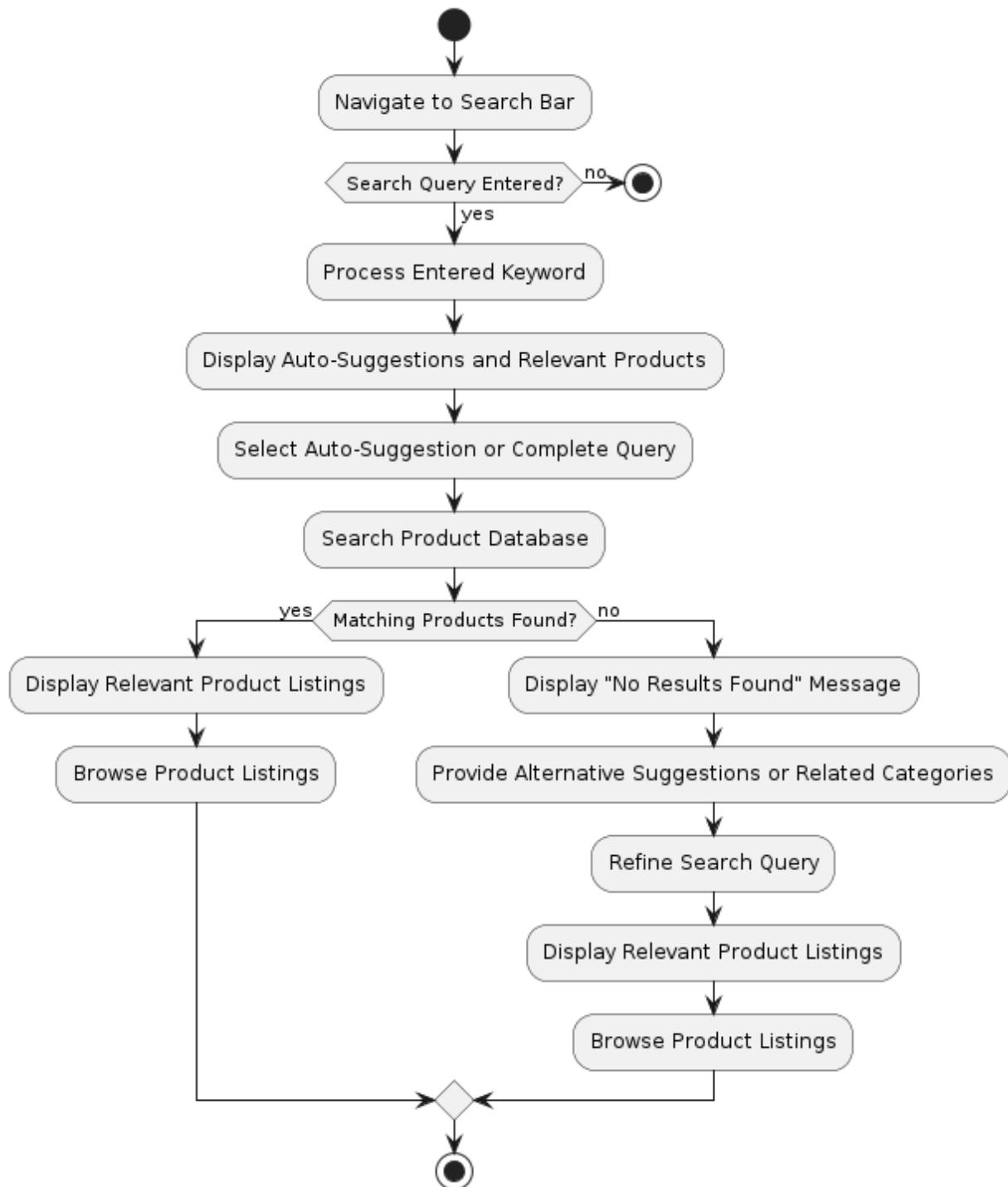


Figure 7: Activity Diagram for Personalize Recommendation



**Figure 8: Activity Diagram for Search Product**

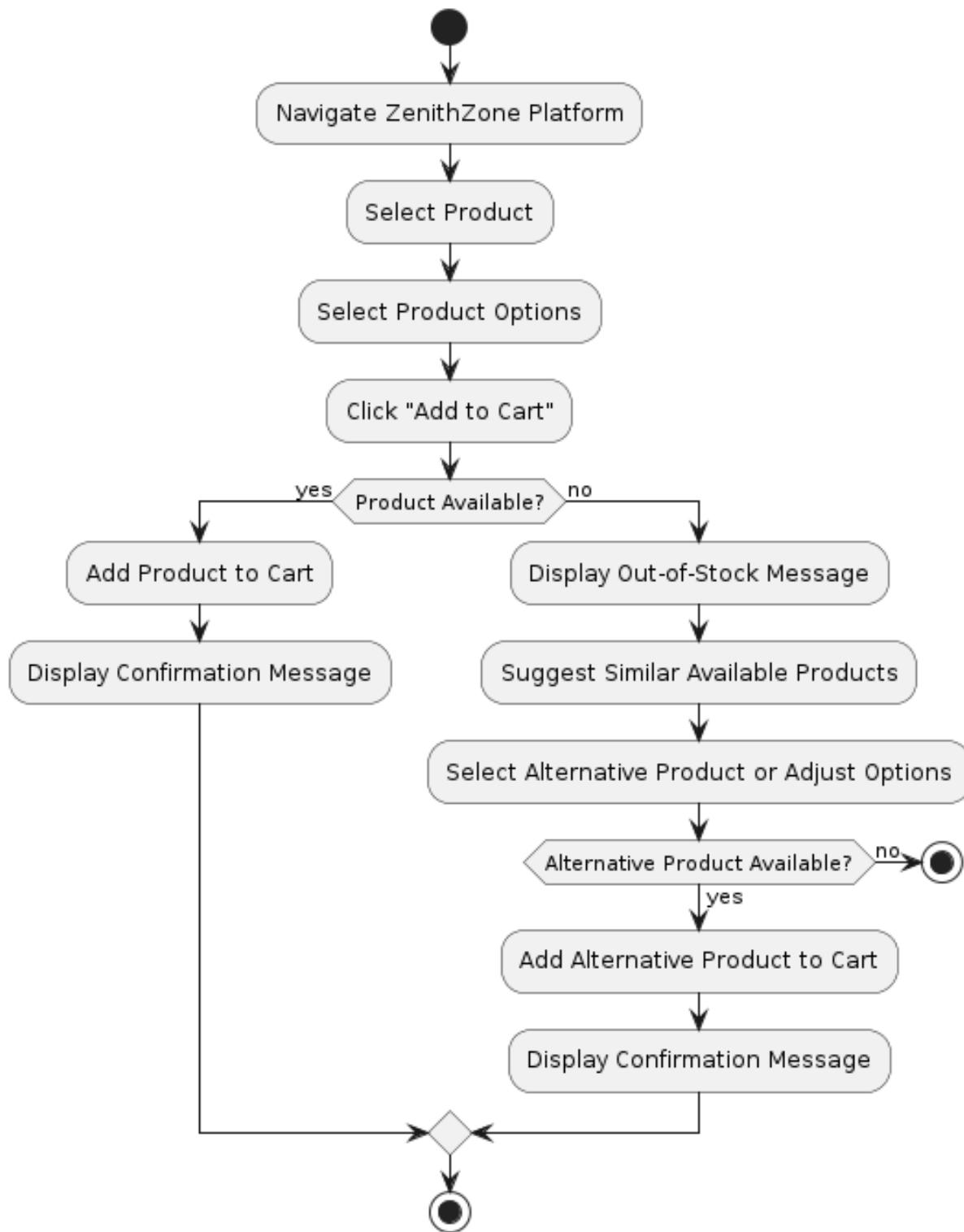
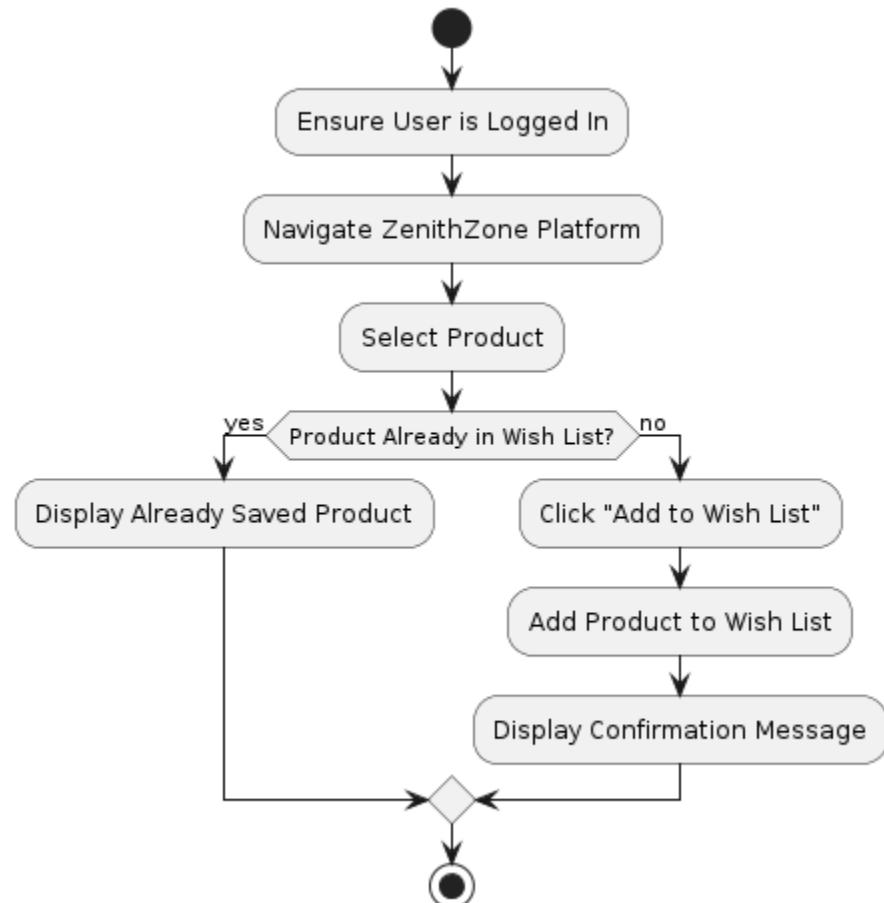
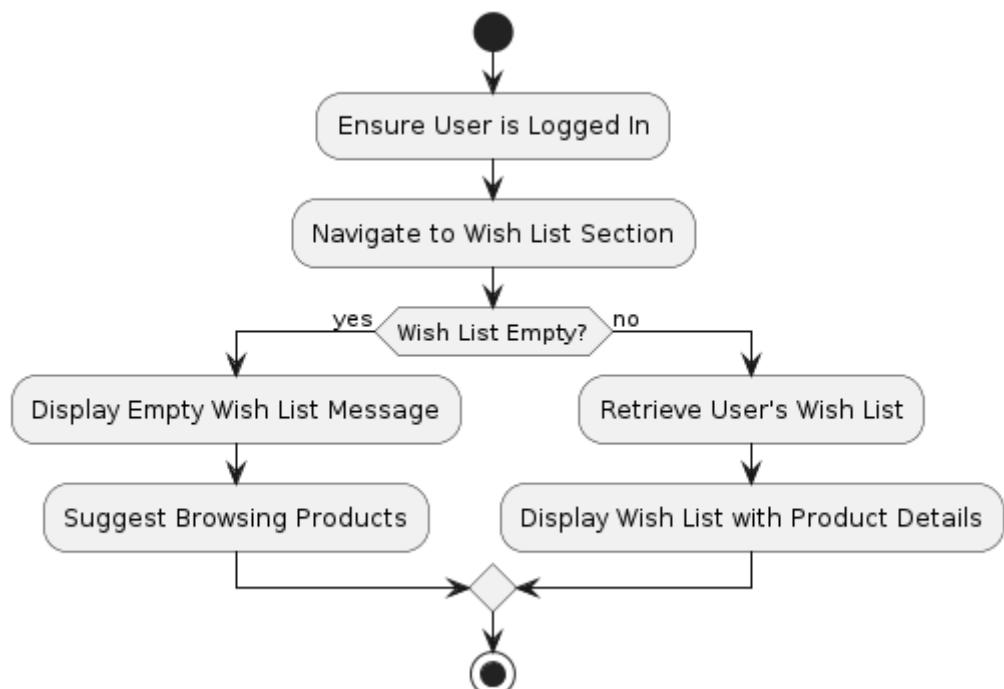


Figure 9: Activity Diagram for Add to Cart



**Figure 10: Activity Diagram for Add to wish list**



**Figure 11: Activity Diagram for view wish list**

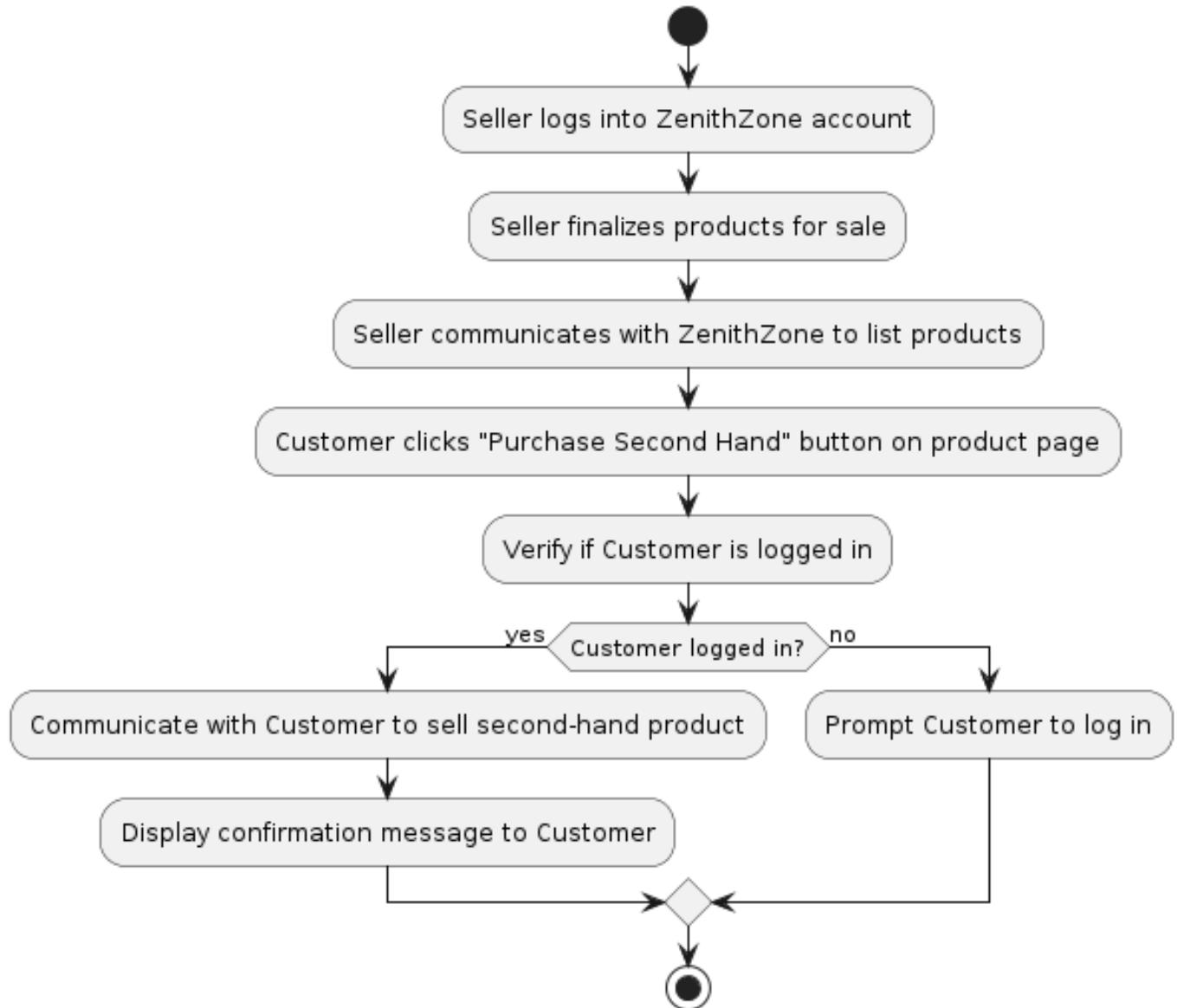
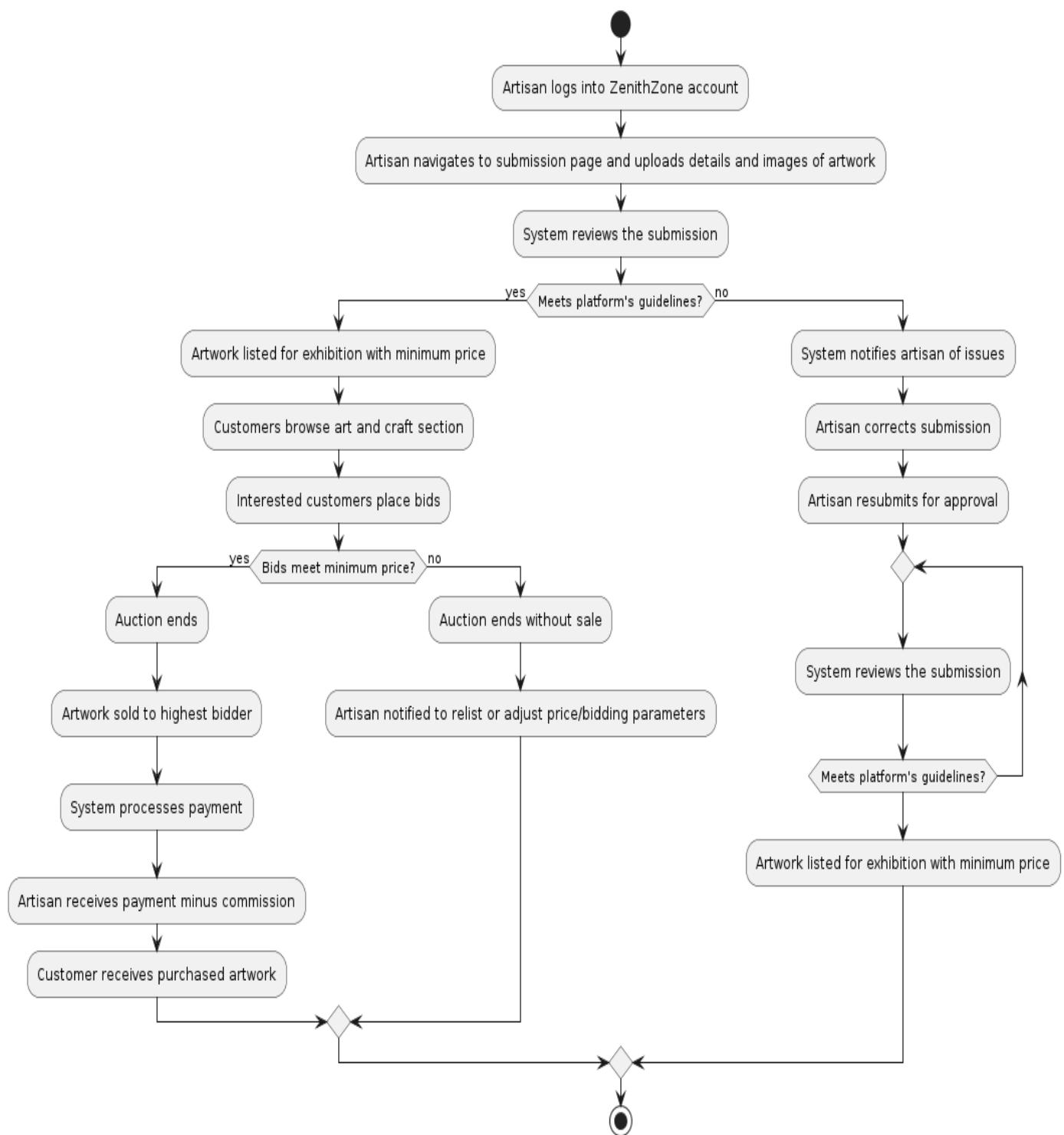
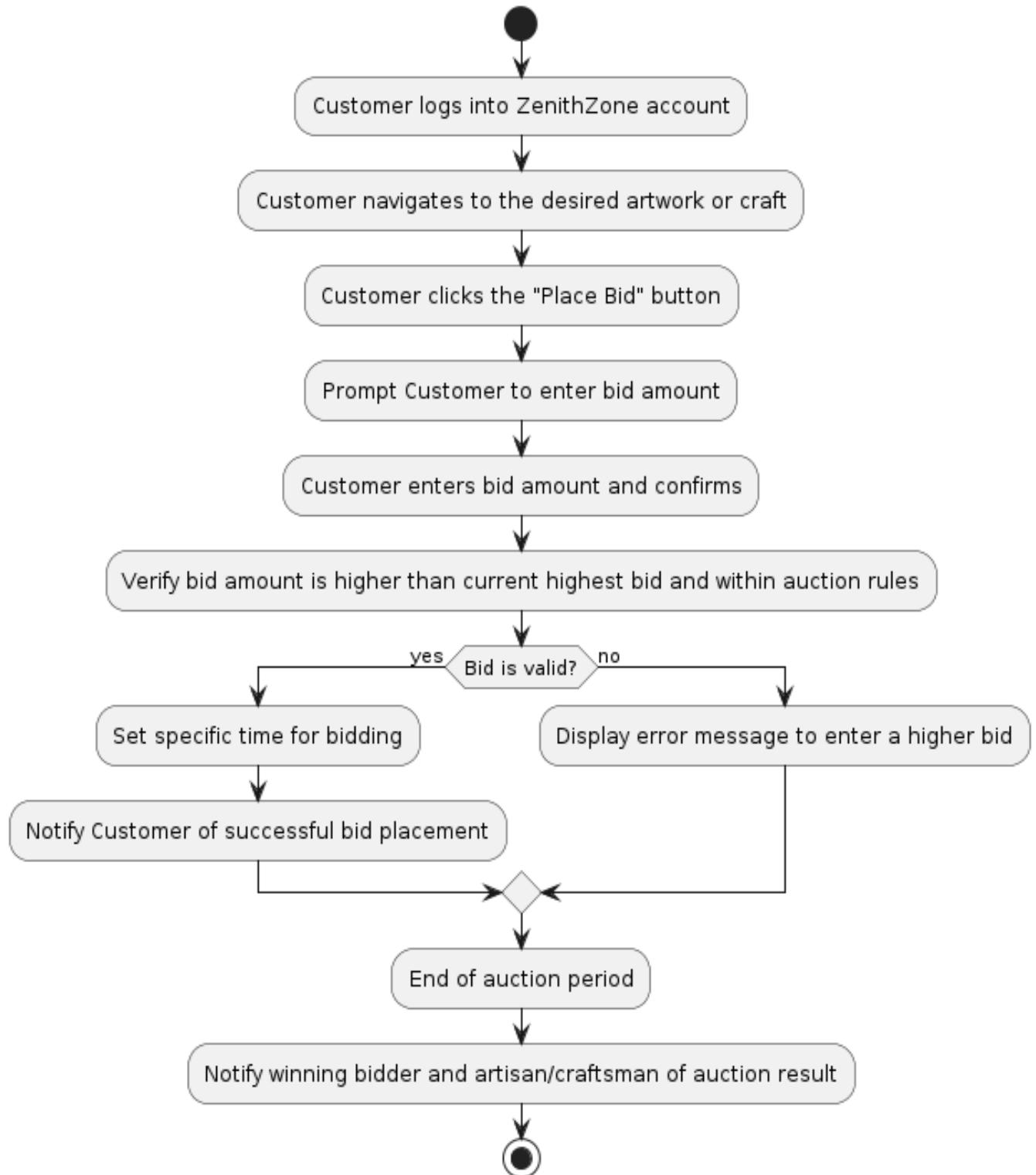


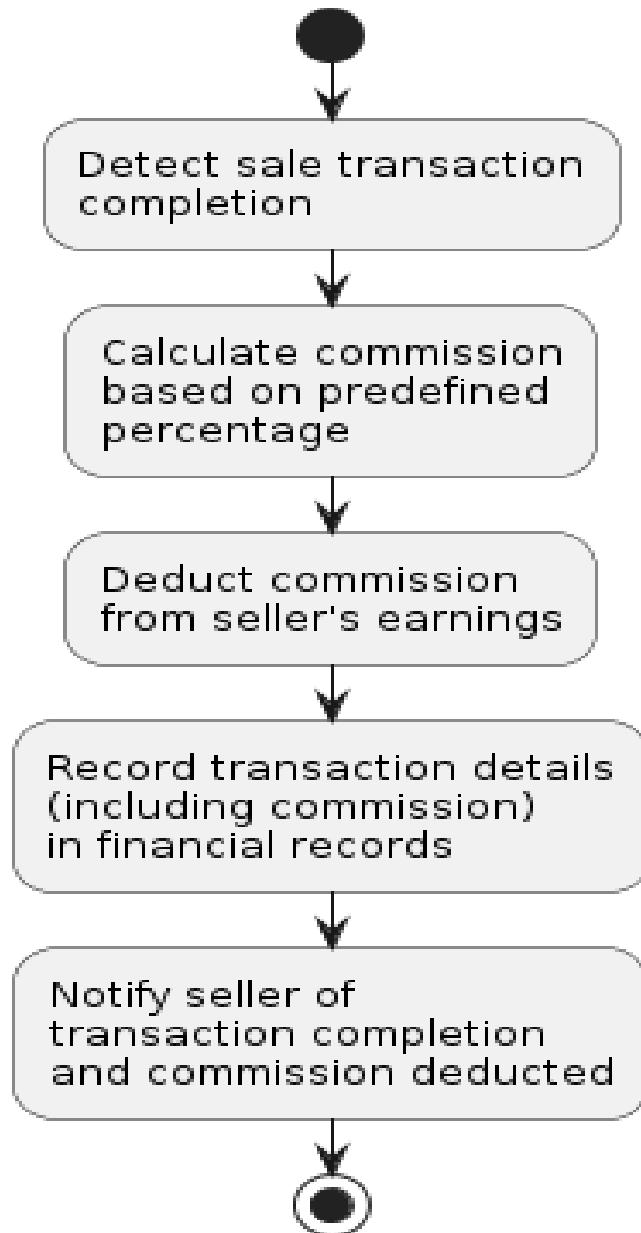
Figure 12: Activity Diagram for Add Second Hand product



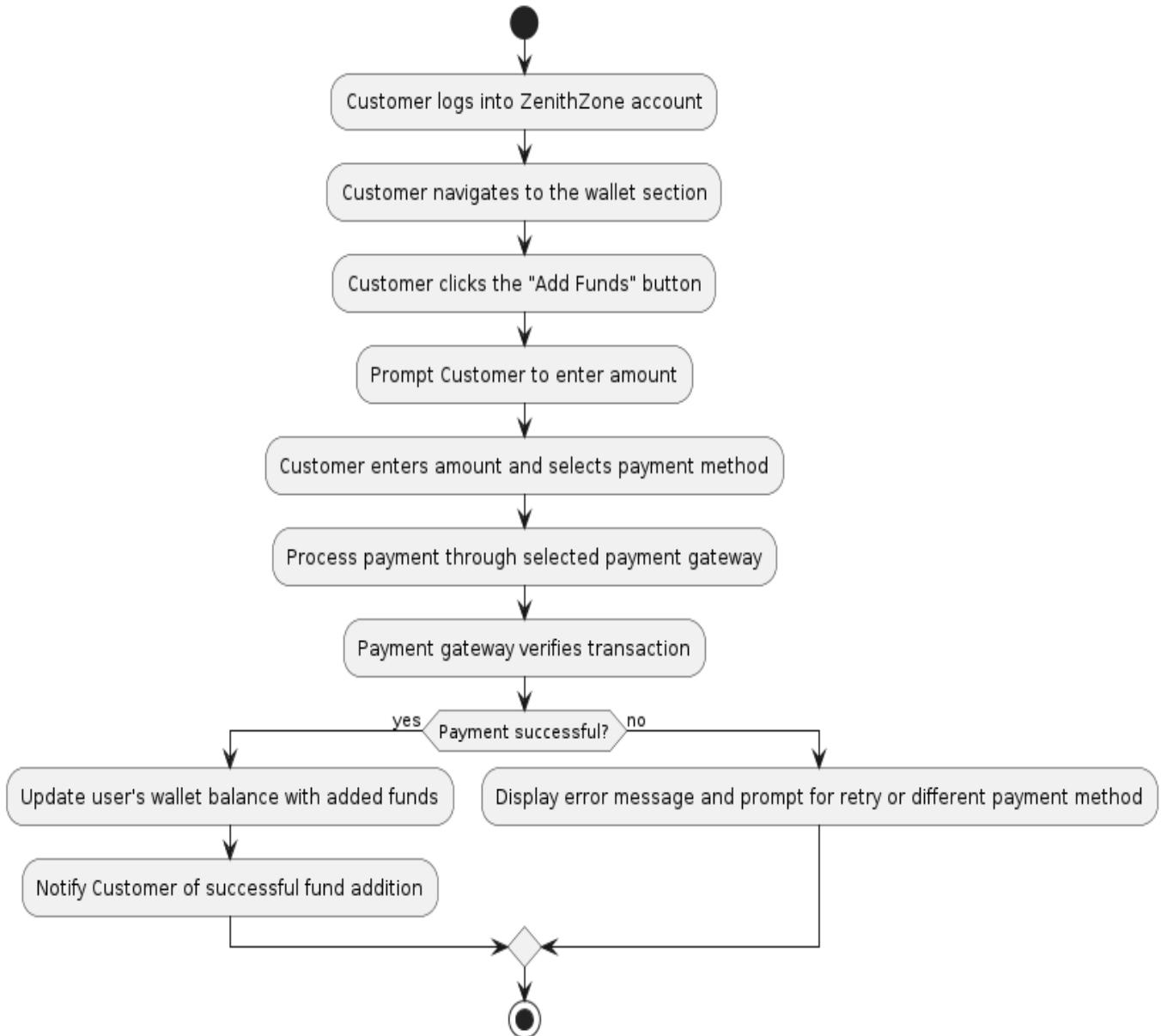
**Figure 13: Activity Diagram for Add Art/Craft**



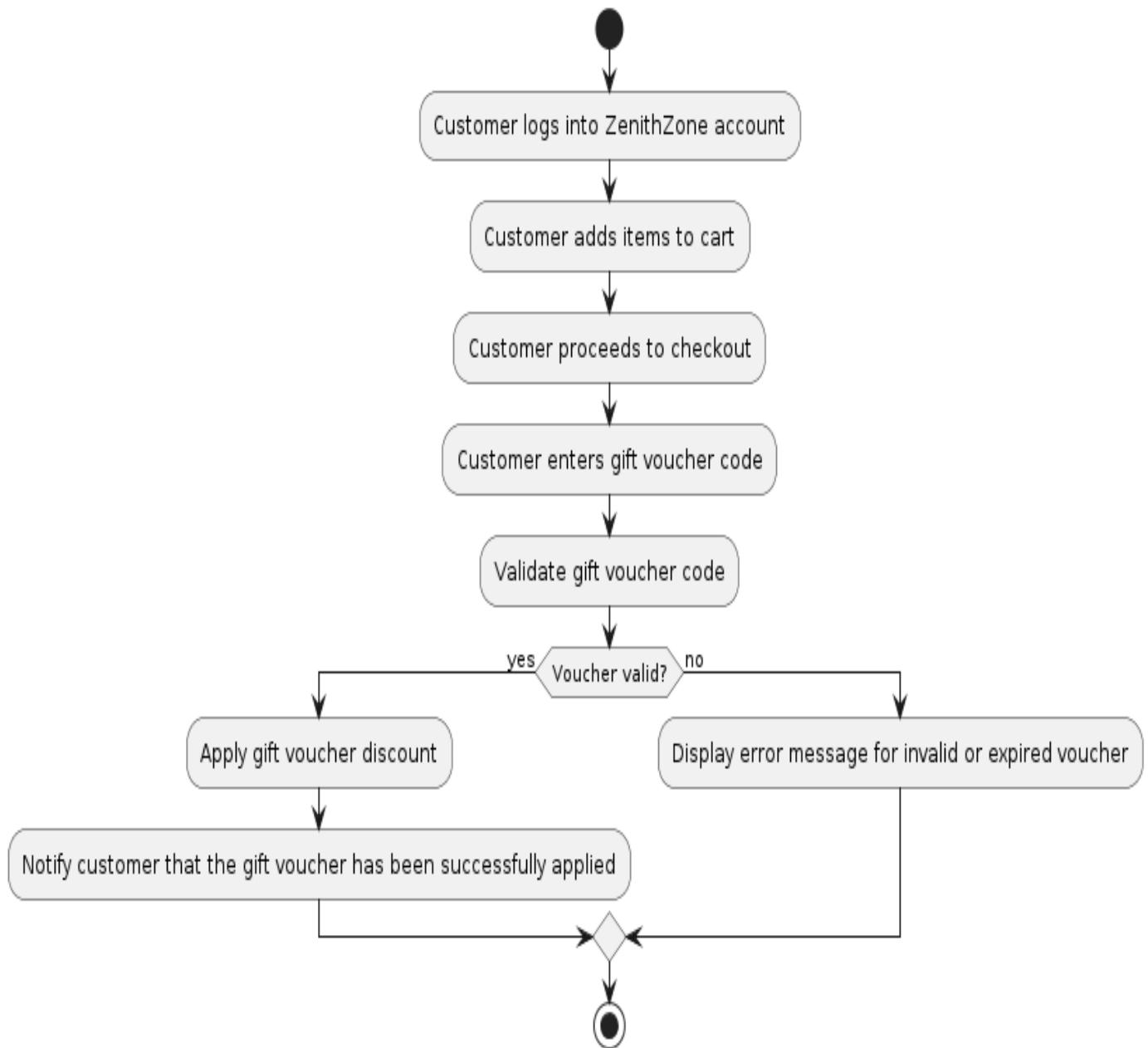
*Figure 14: Activity Diagram for Bid on Art/Craft*



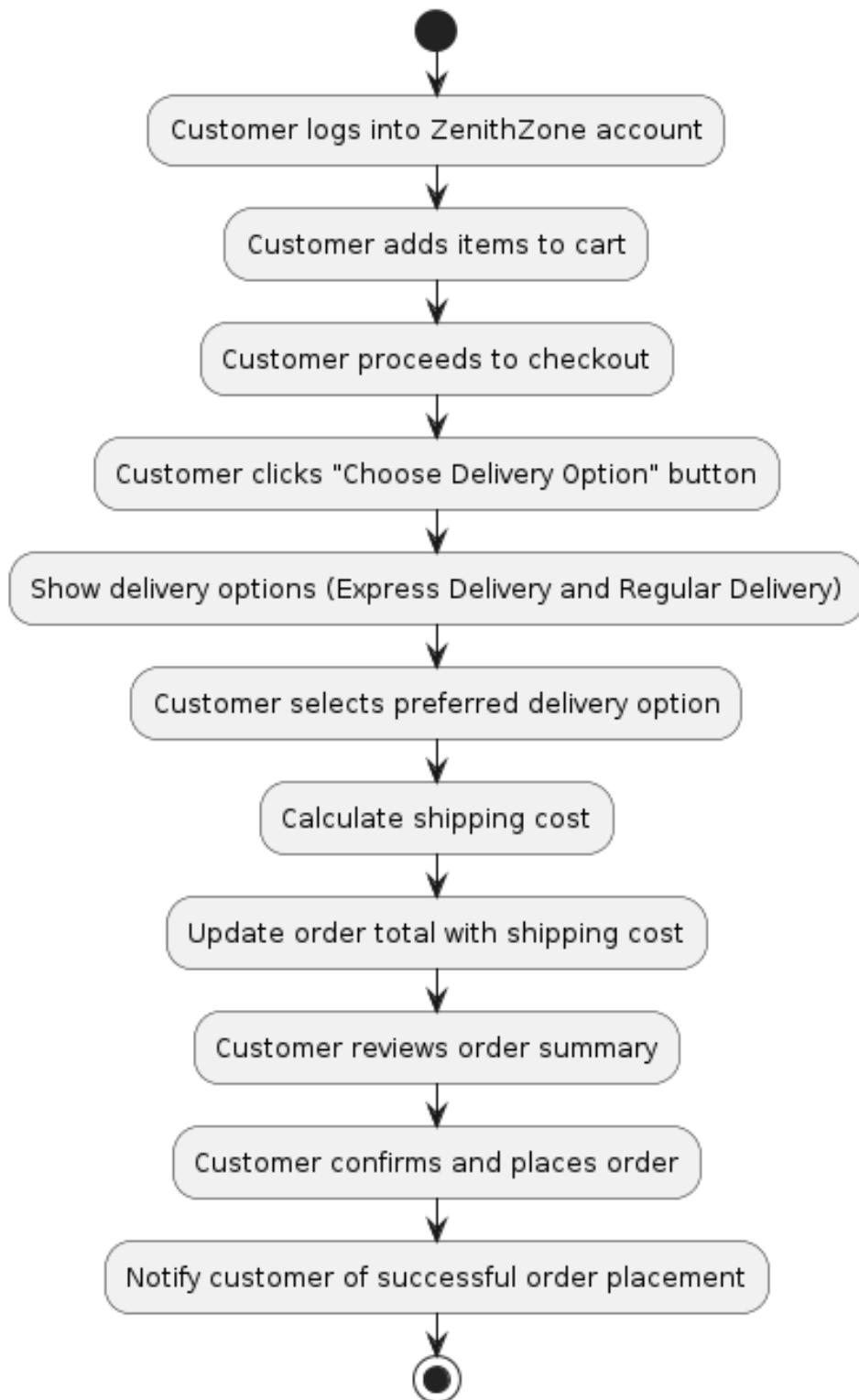
*Figure 15: Activity Diagram for Profit Margin*



**Figure 16: Activity Diagram for Wallet**



**Figure 17: Activity Diagram for Gift Voucher**



*Figure 18: Activity Diagram for Delivery Options*

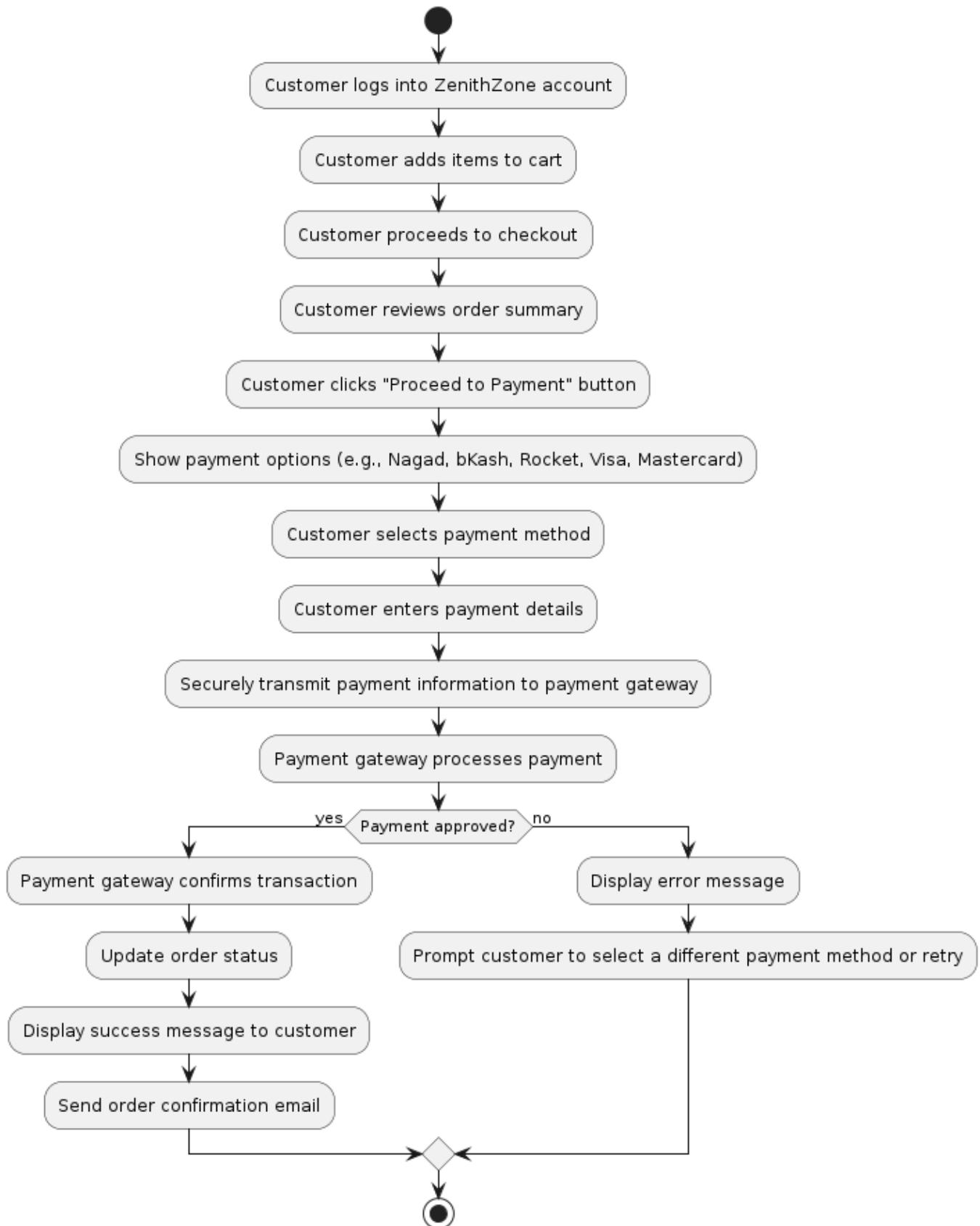
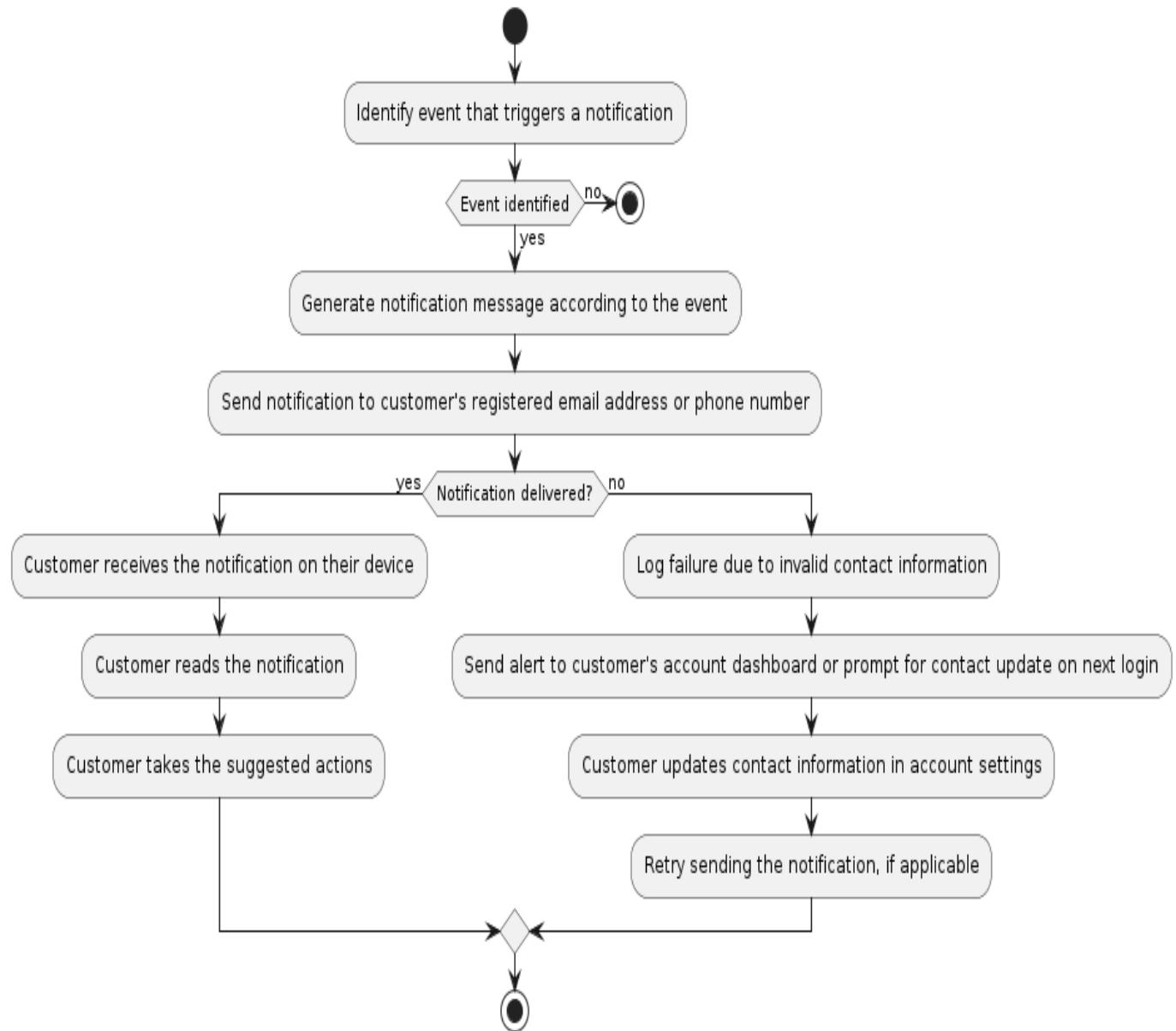


Figure 19: Activity Diagram for Checkout and Payment



**Figure 20: Activity Diagram for Notification**

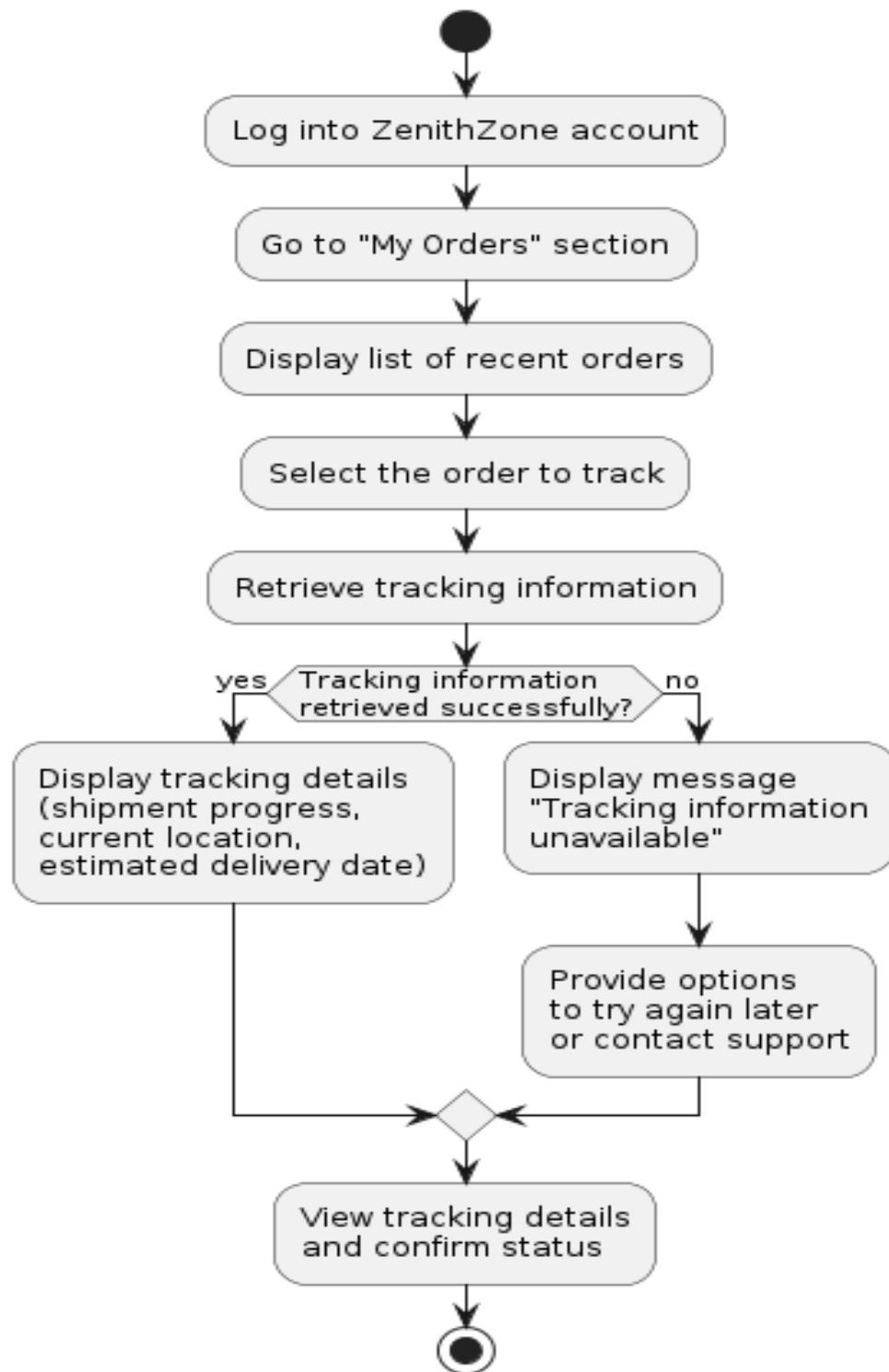
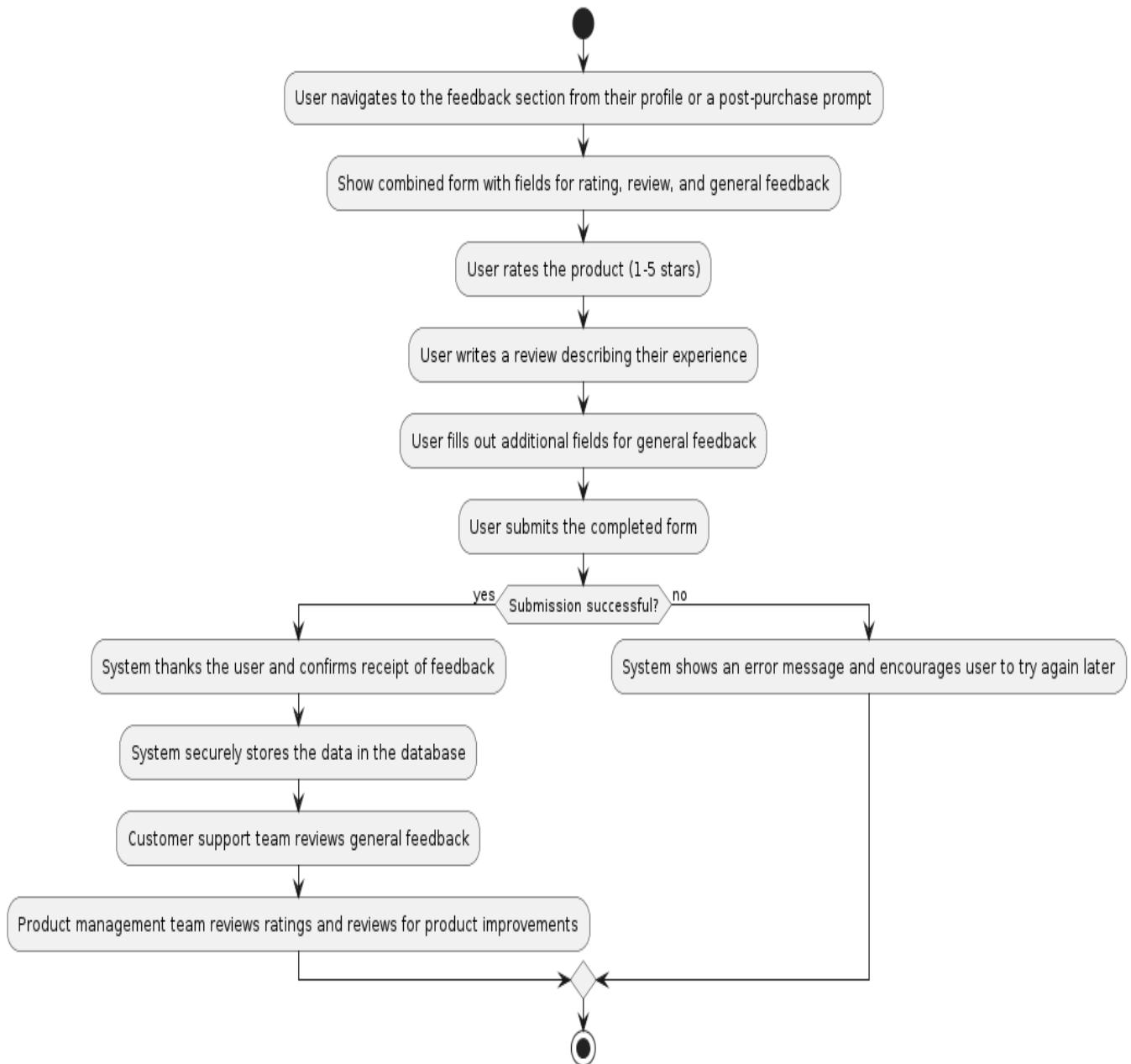
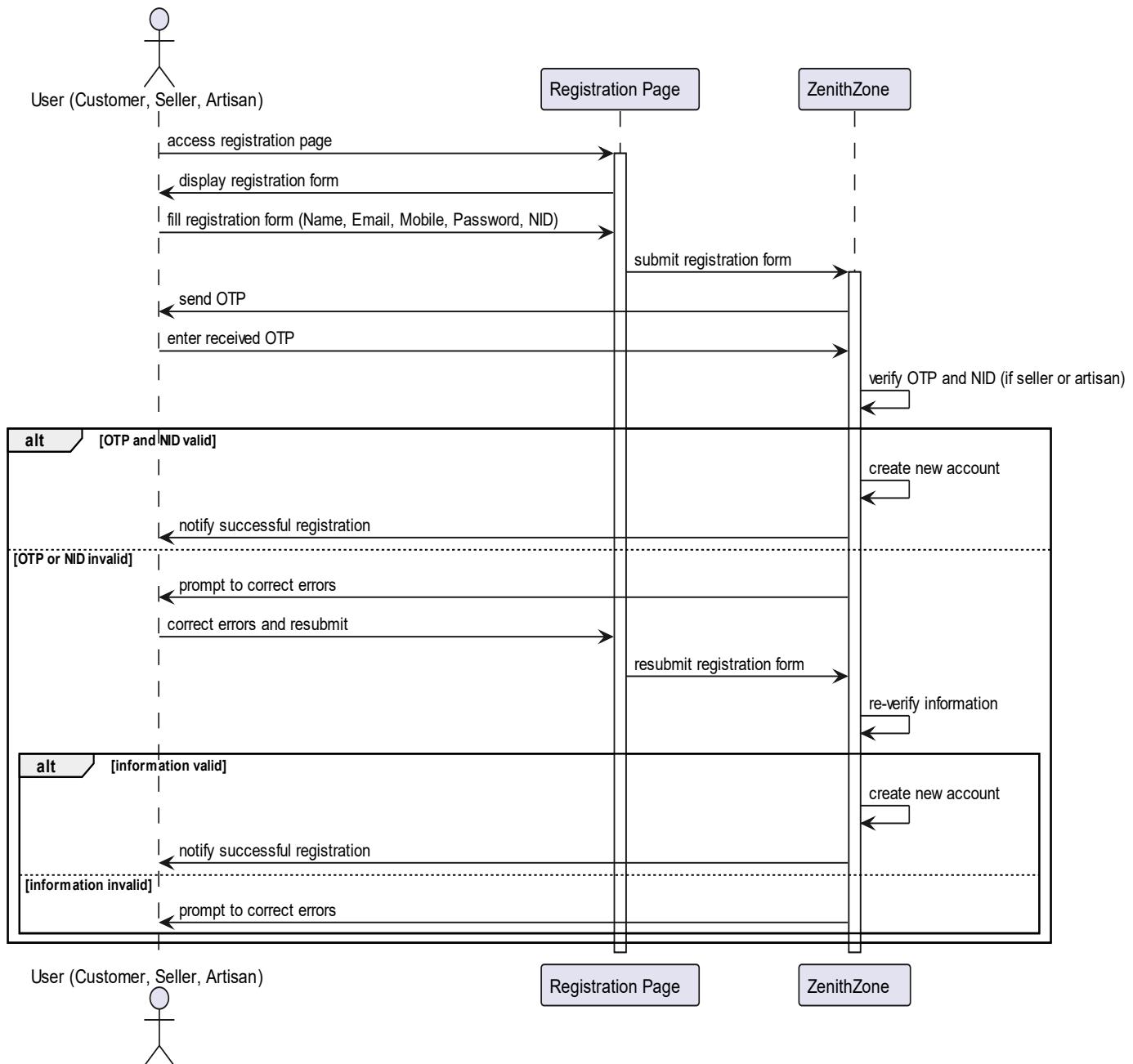


Figure 21: Activity Diagram for Track Order



**Figure 22: Activity Diagram for Review**

## 9. Sequence Diagram



*Figure 23: Sequence Diagram for Registration Request*

## Software Requirements Specification for ZenithZone

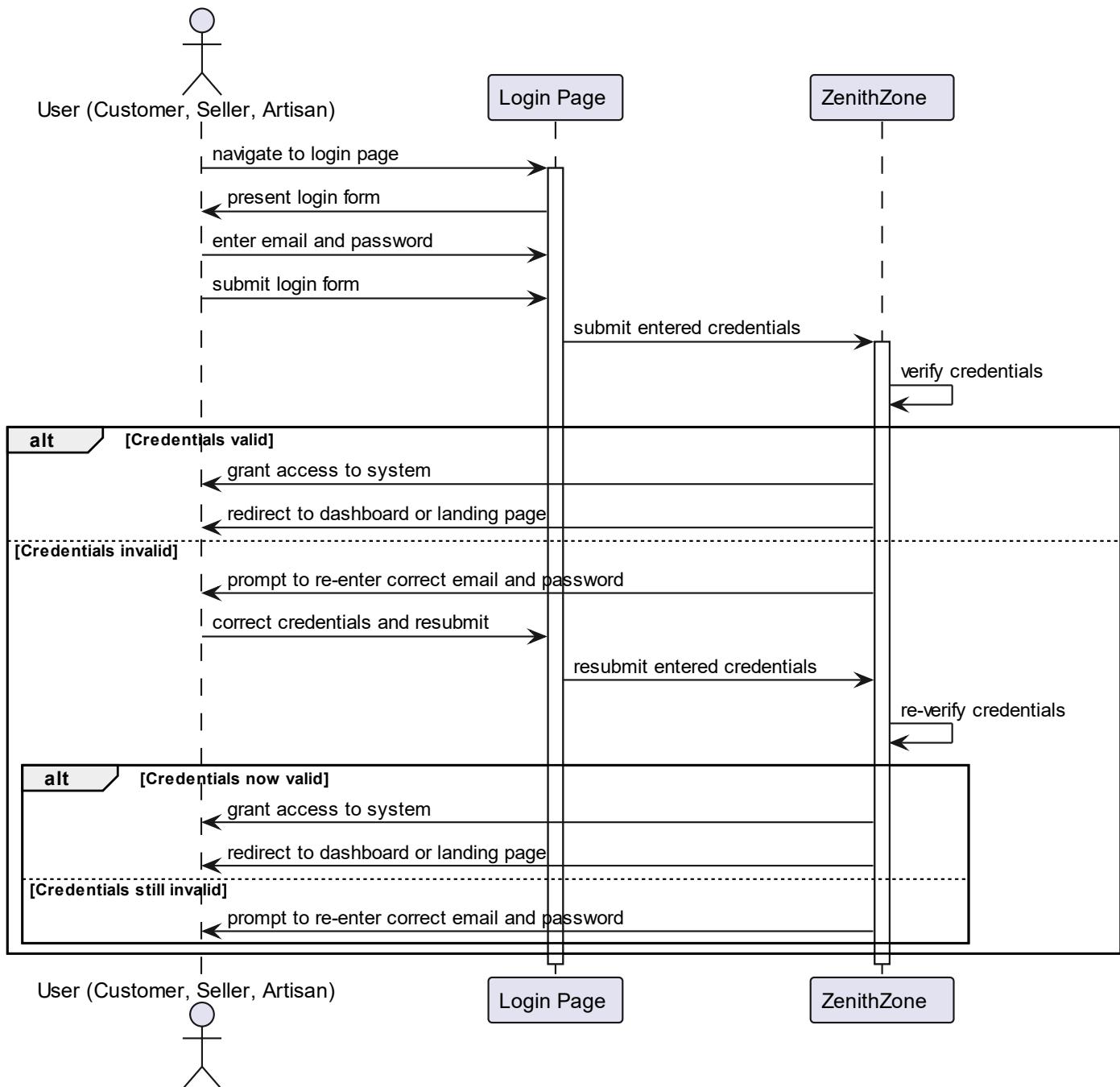


Figure 24: Sequence Diagram for Login

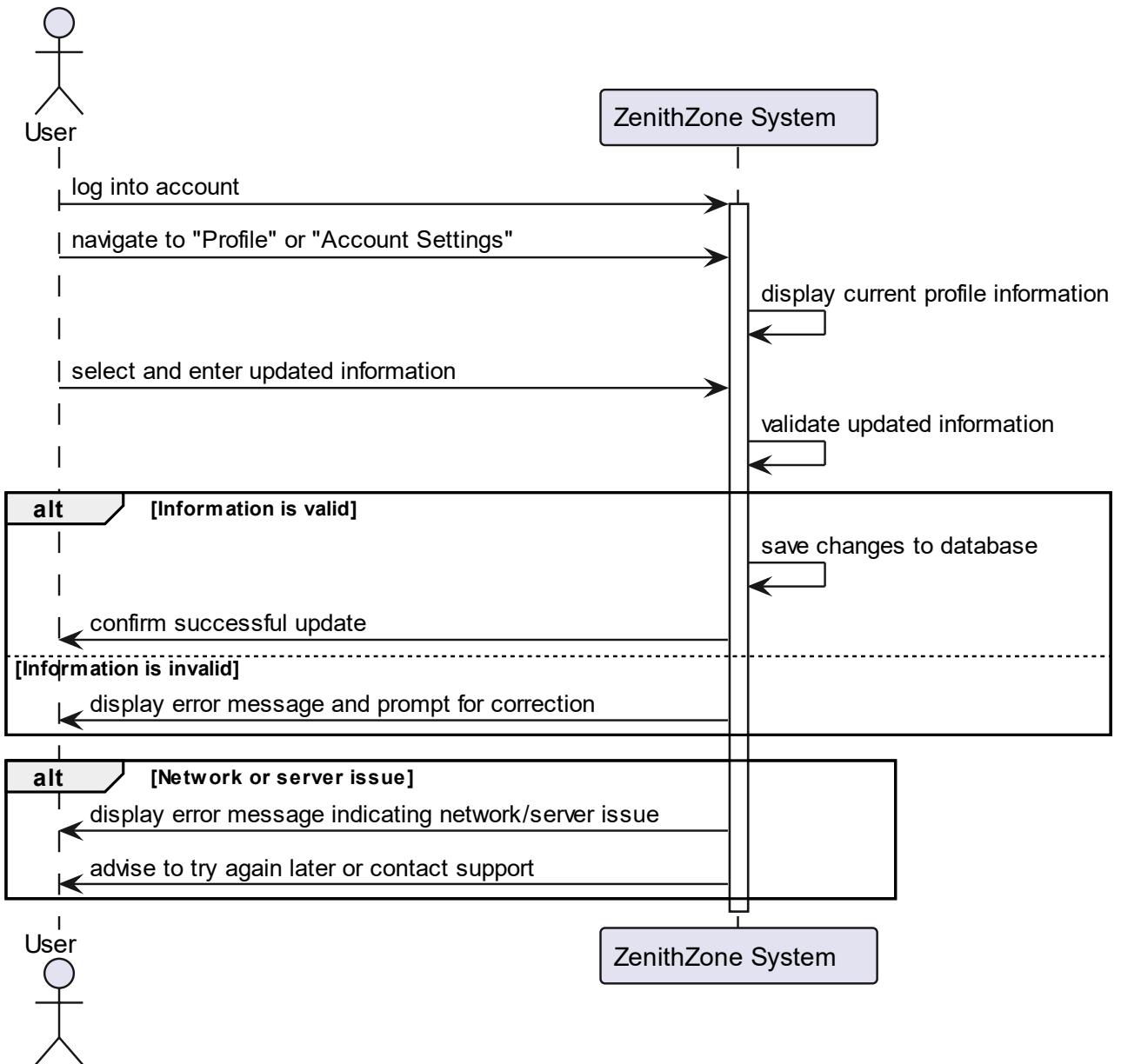


Figure 25: Sequence Diagram for Profile Management

## Software Requirements Specification for ZenithZone

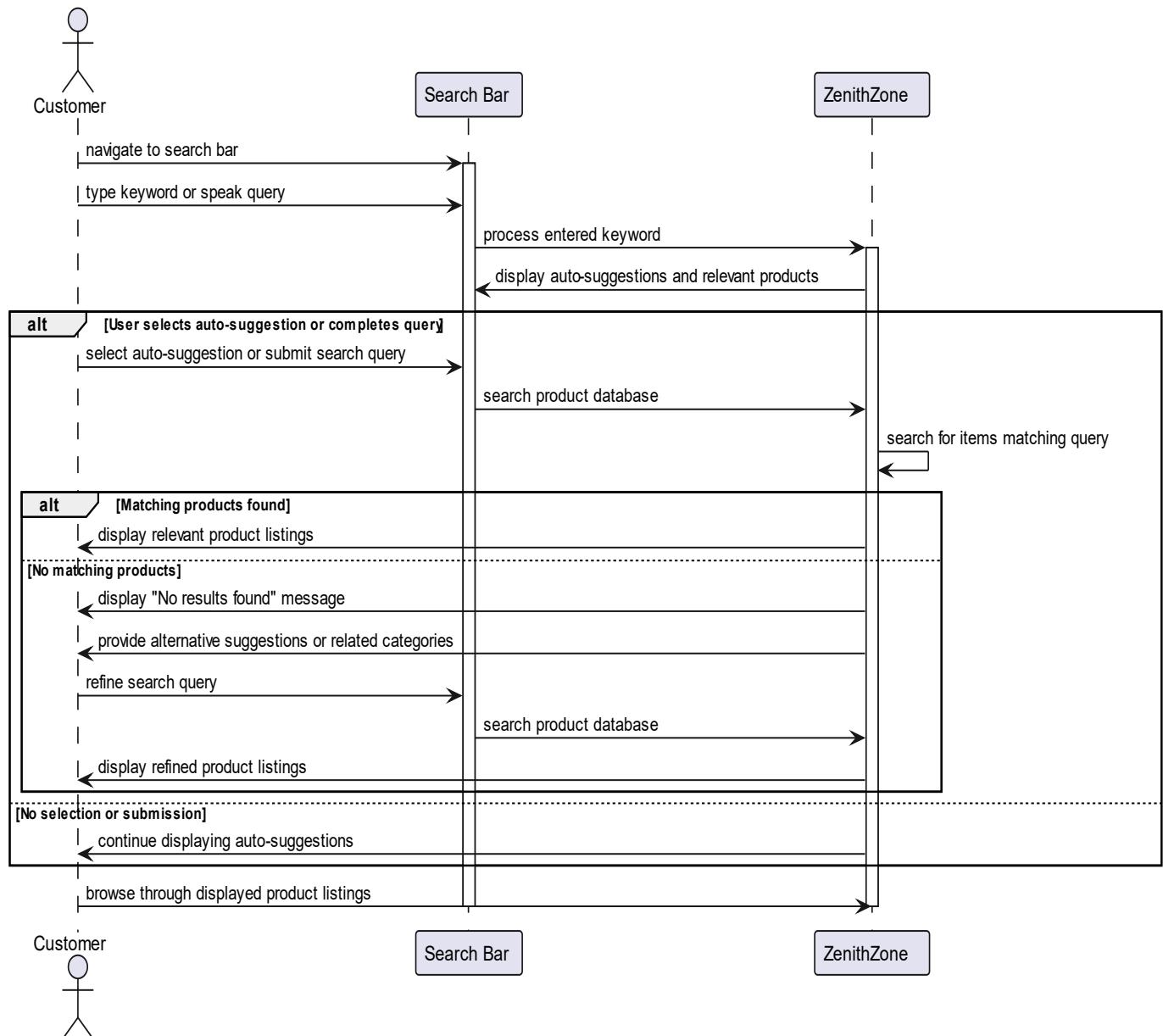


Figure 26: Sequence Diagram for Search Product

## Software Requirements Specification for ZenithZone

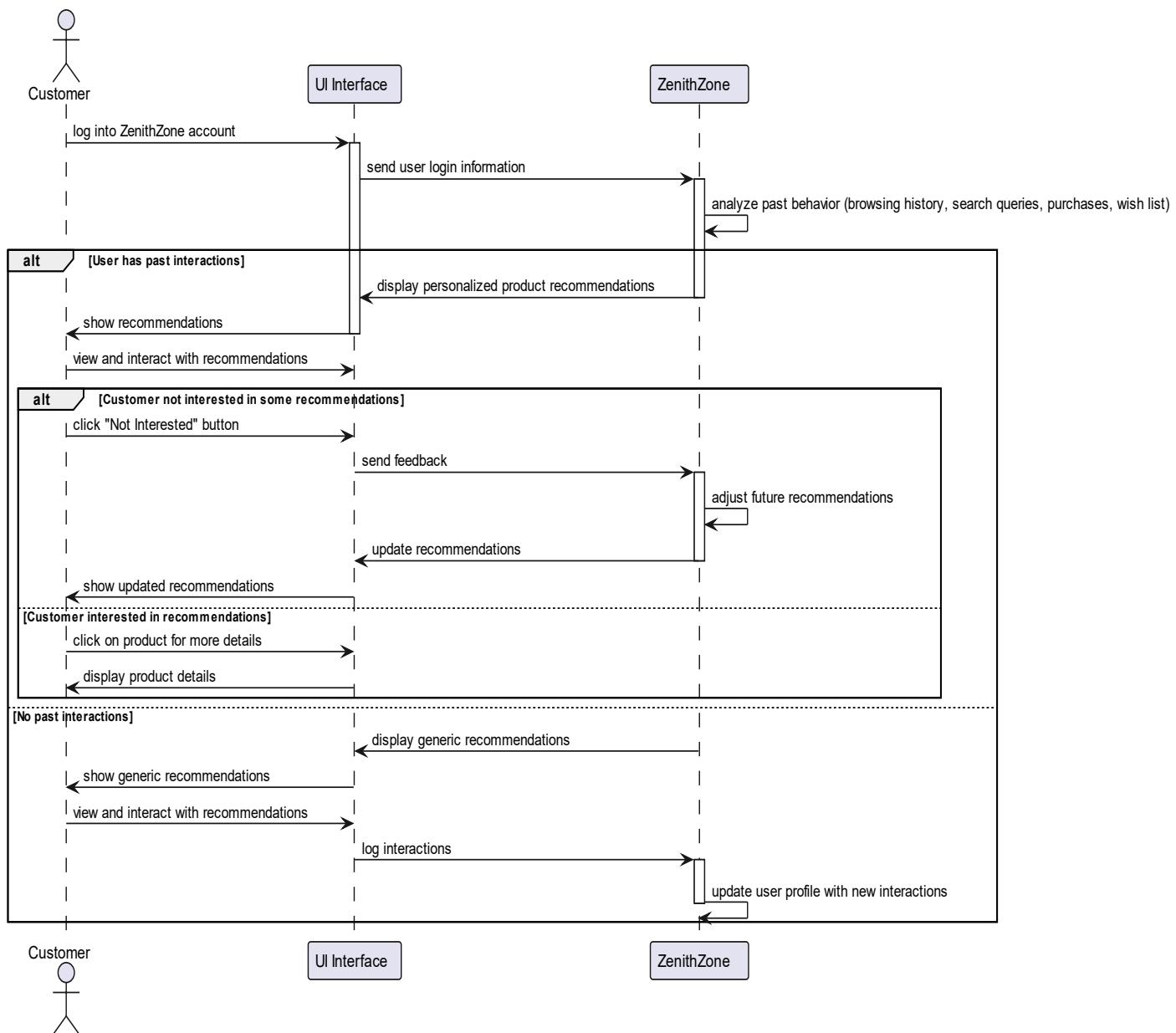


Figure 27: Sequence Diagram for Personalized Recommendation

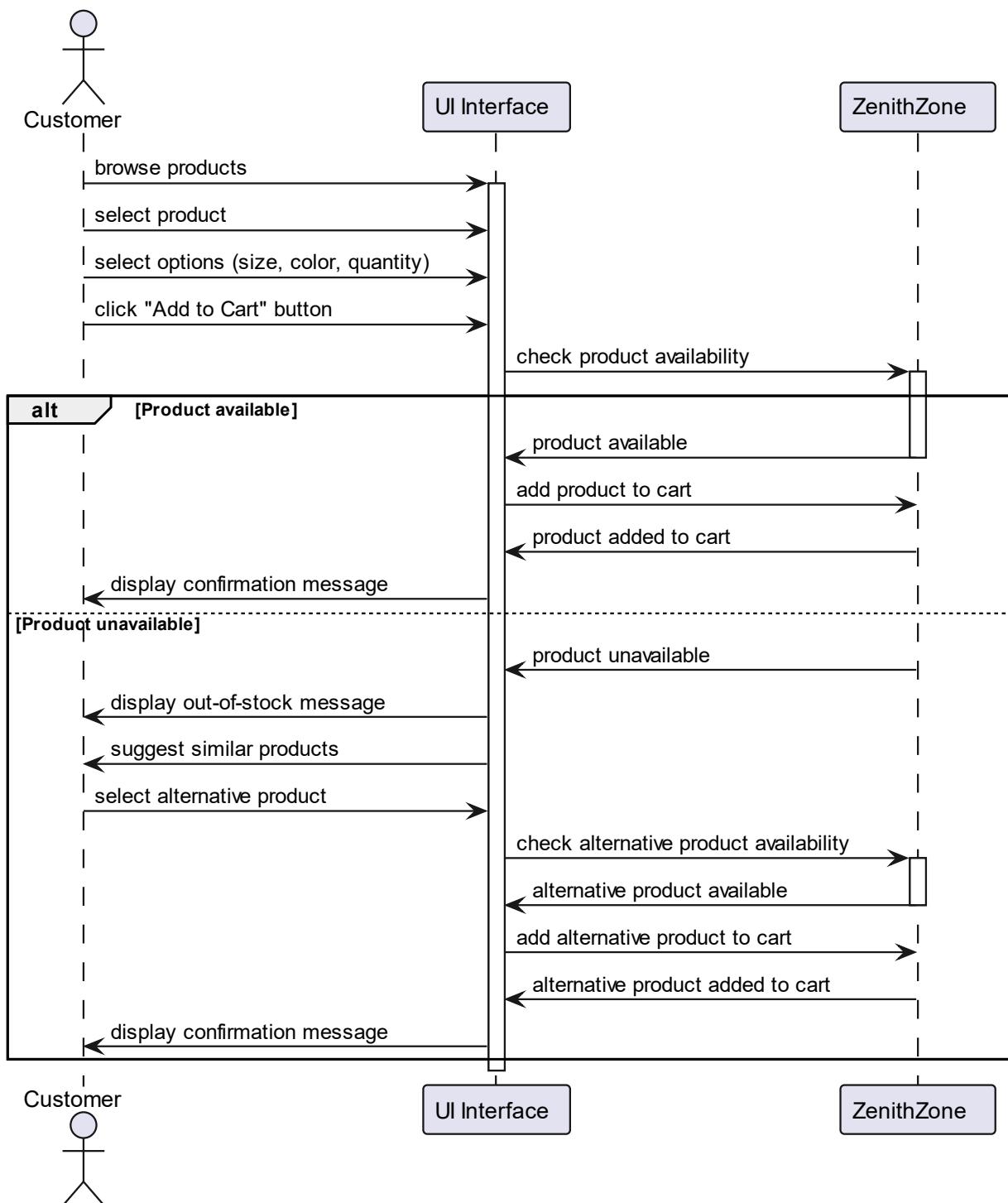


Figure 28: Sequence Diagram for Add to Cart

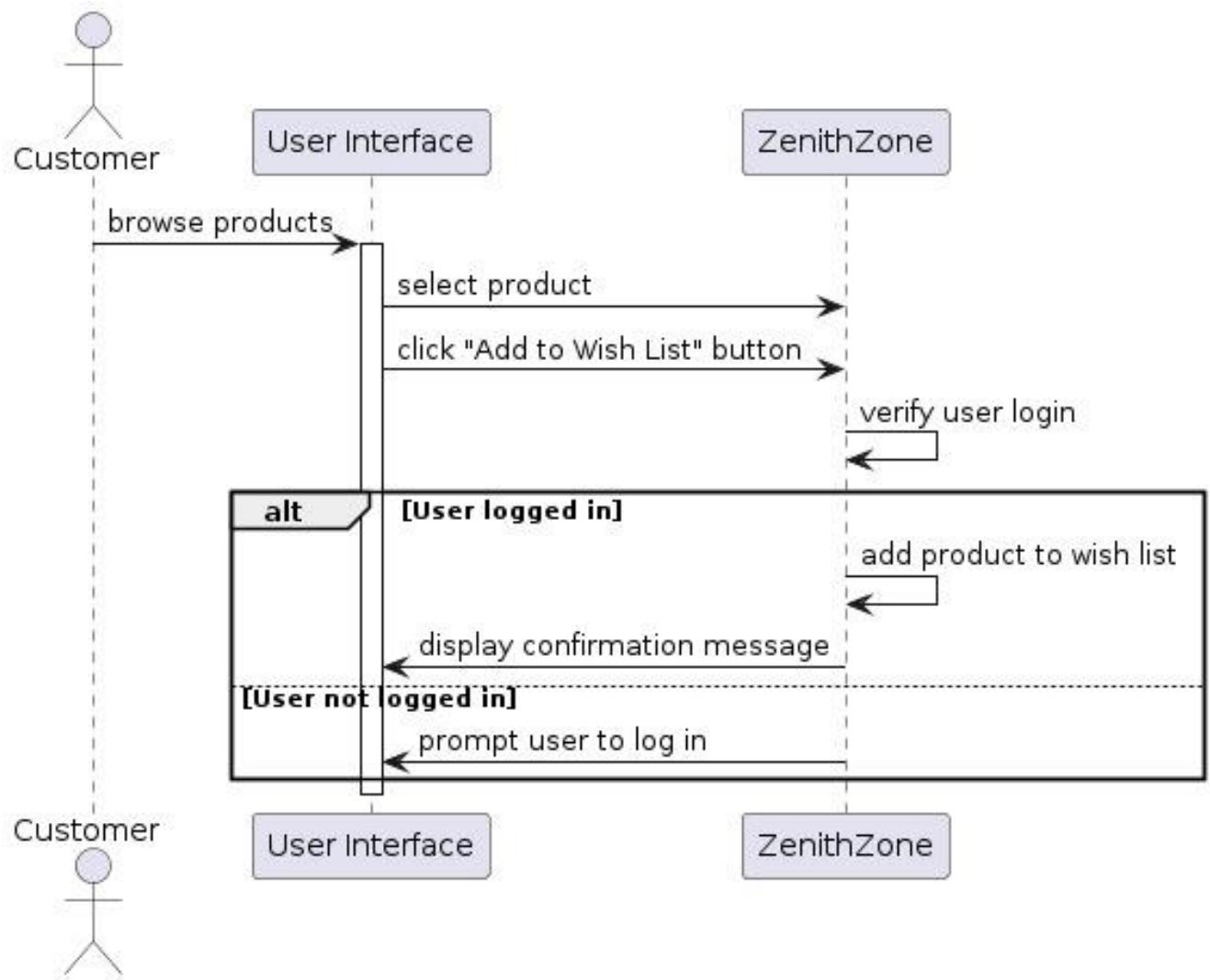


Figure 29: Sequence Diagram for Wish List

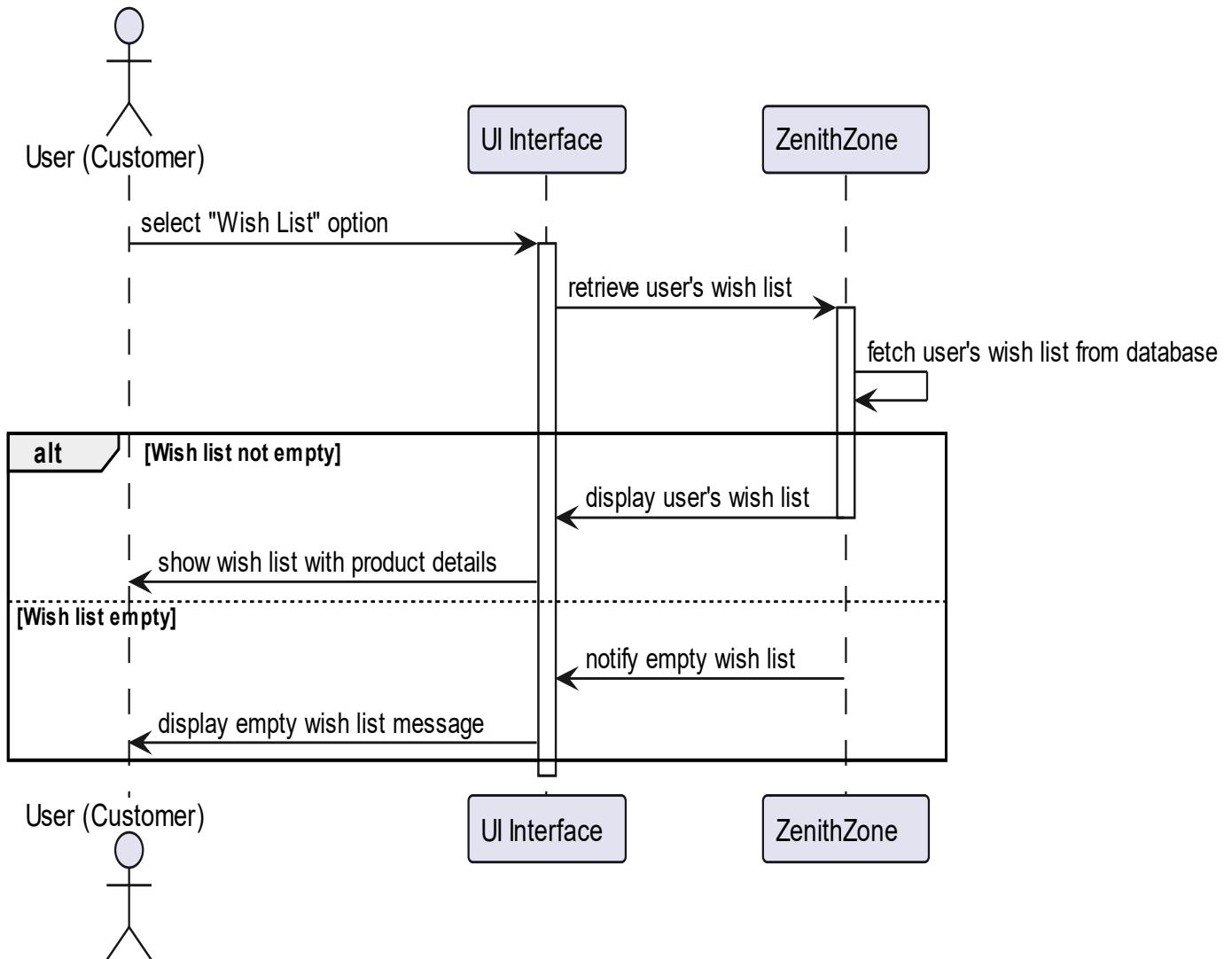


Figure 30: Sequence Diagram for View Wish List

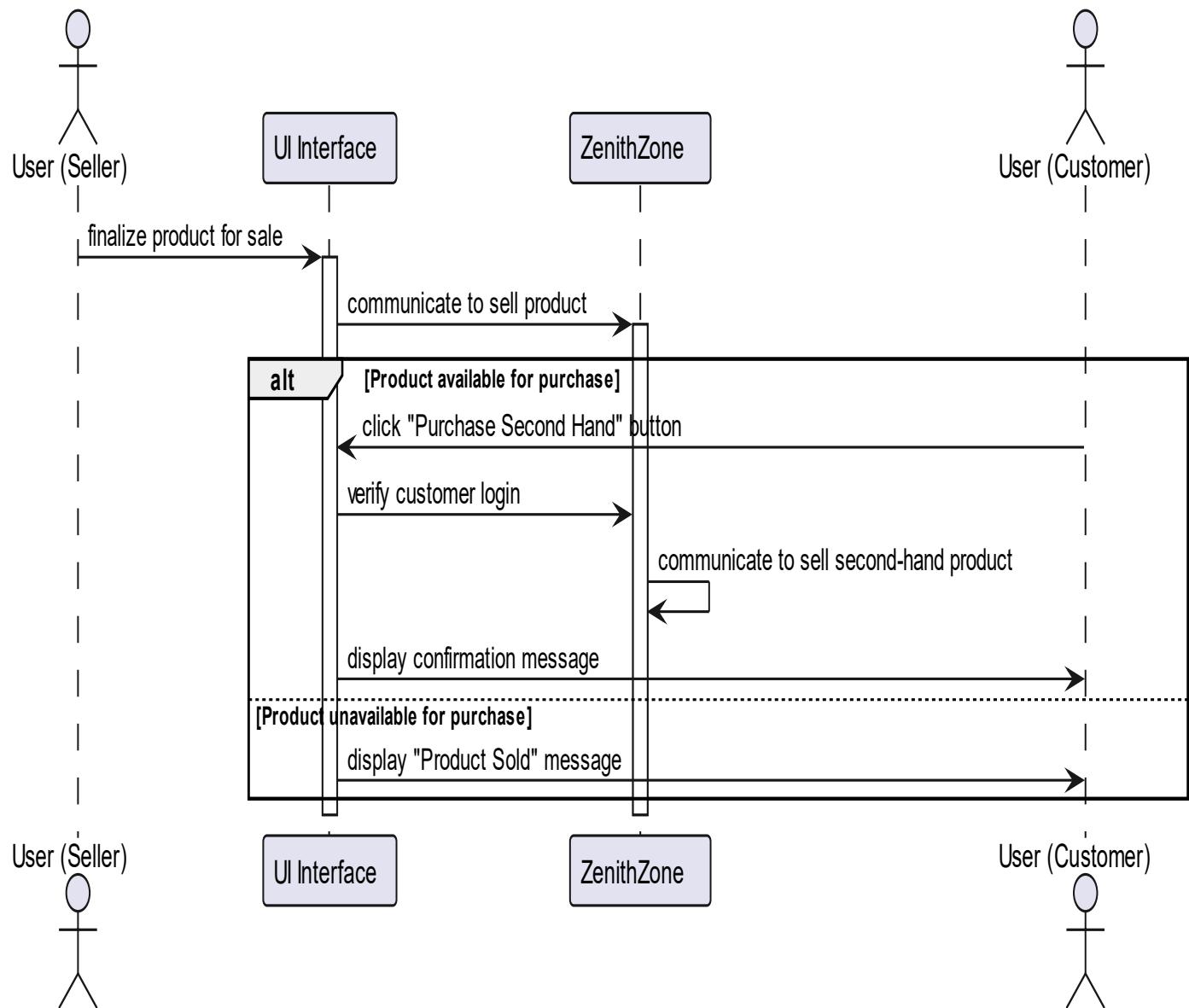


Figure 31: Sequence Diagram for Second Hand Product

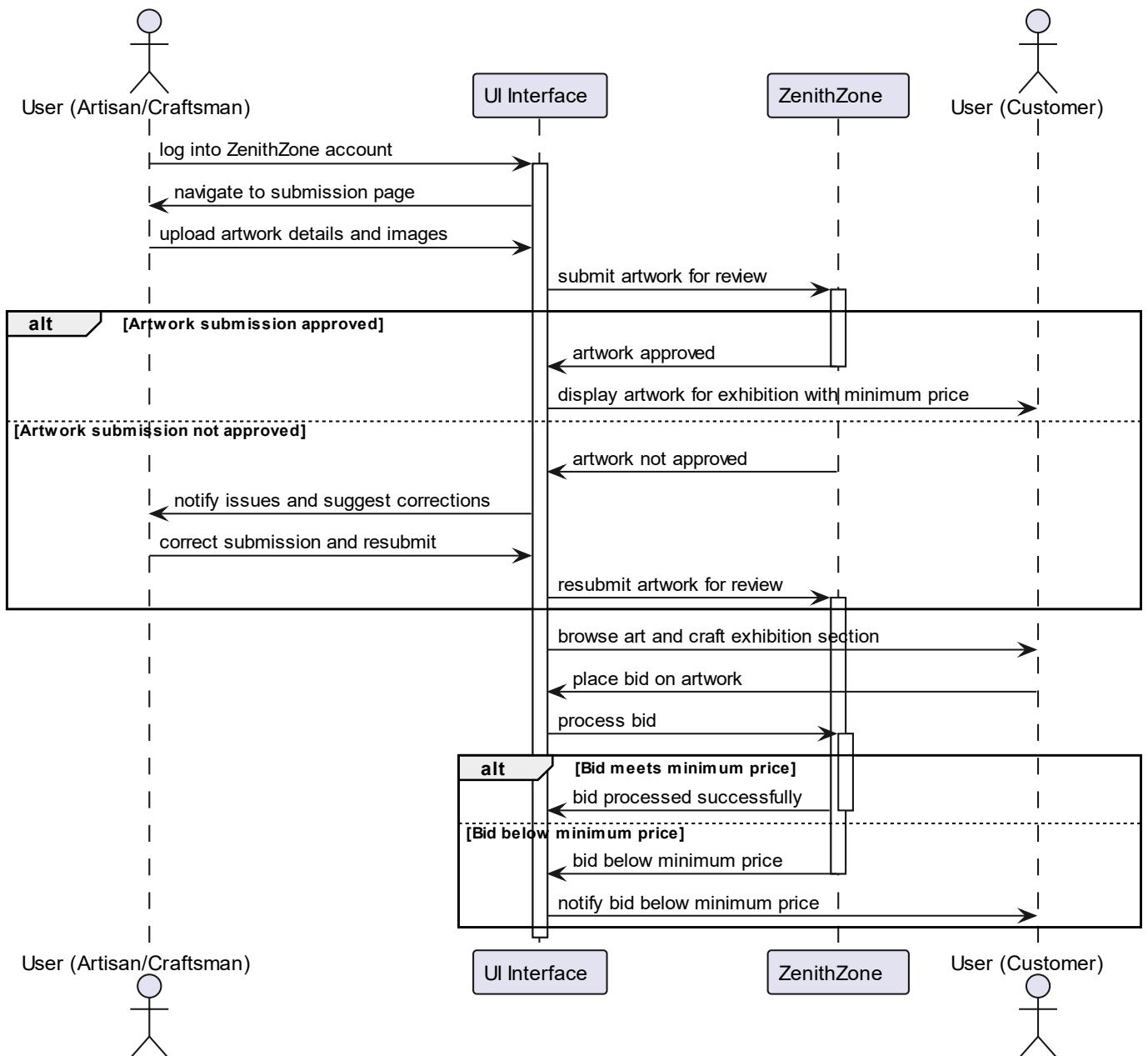


Figure 32: Sequence Diagram for Add/Craft

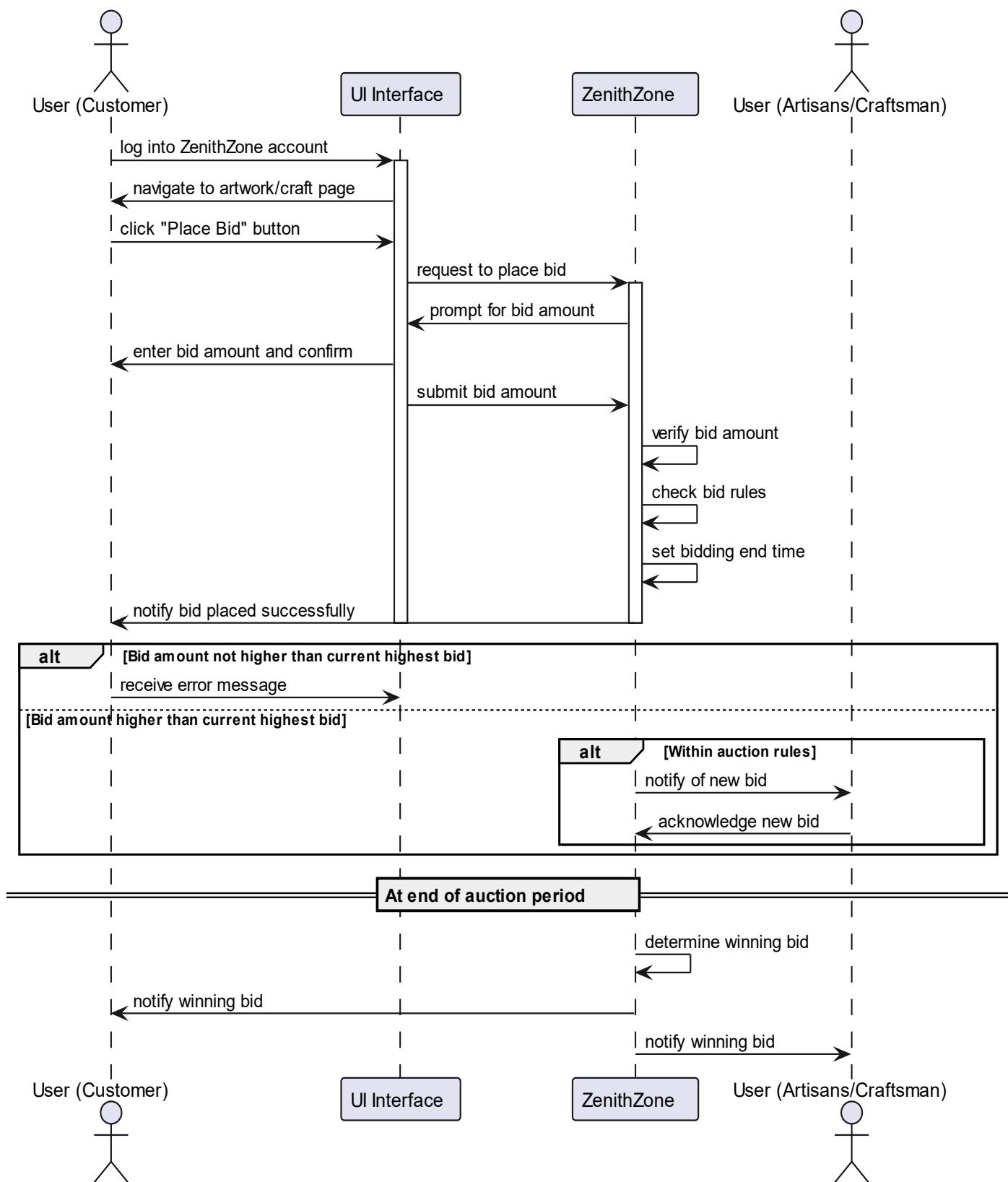


Figure 33: Sequence Diagram for Bid on Add/Craft

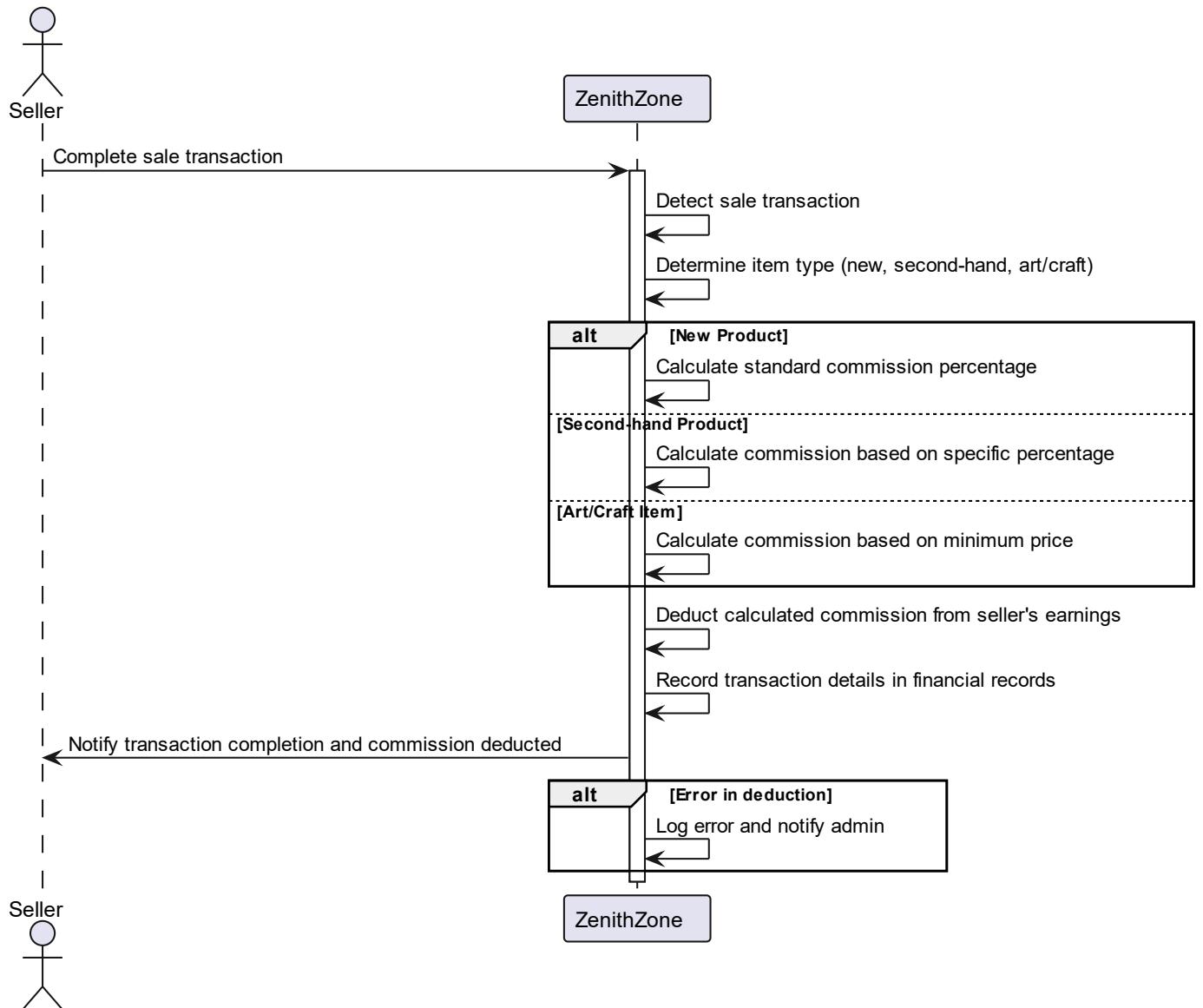


Figure 34: Sequence Diagram for Profit Margin

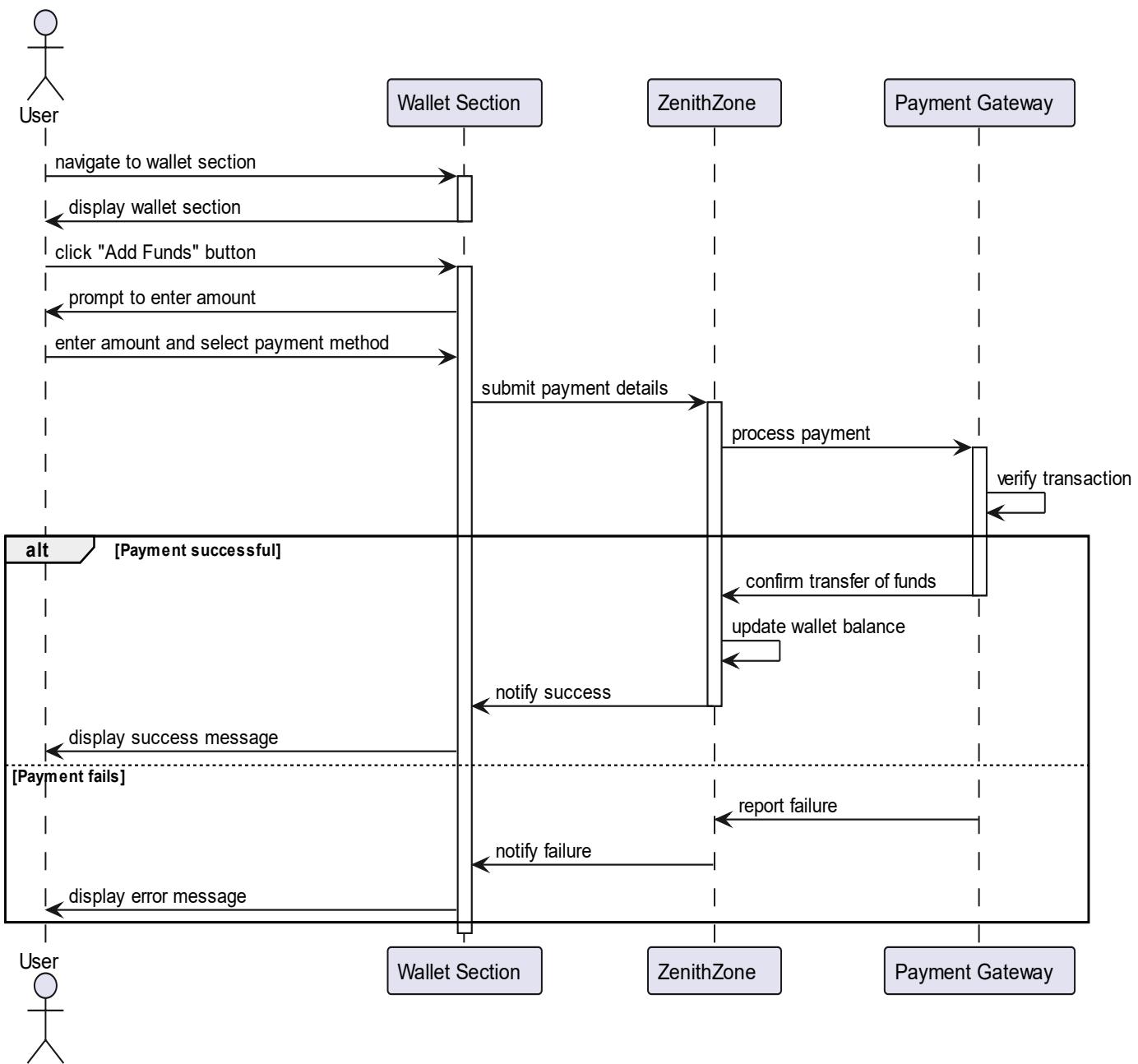


Figure 35: Sequence Diagram for Wallet system

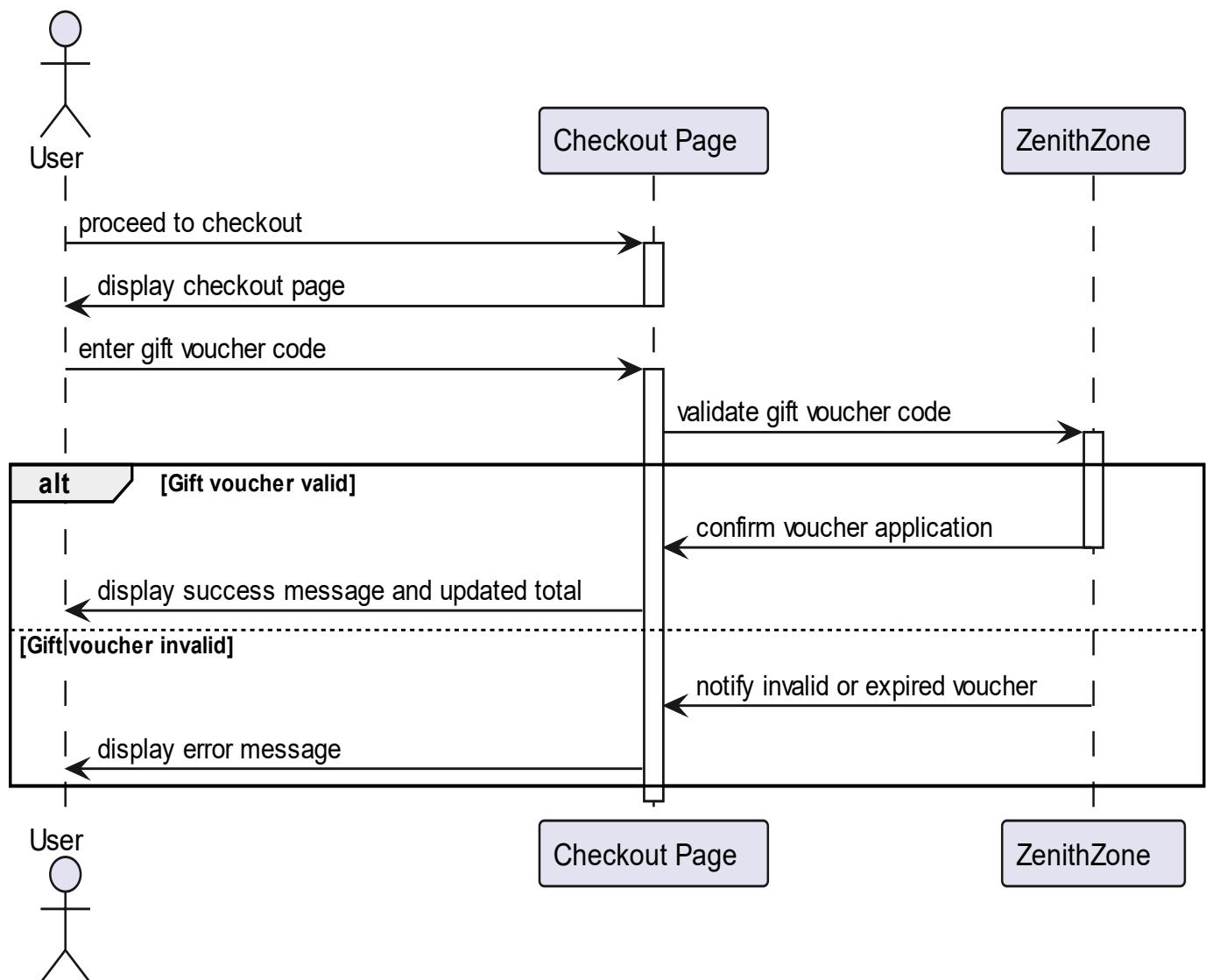


Figure 36: Sequence Diagram for Gift Voucher

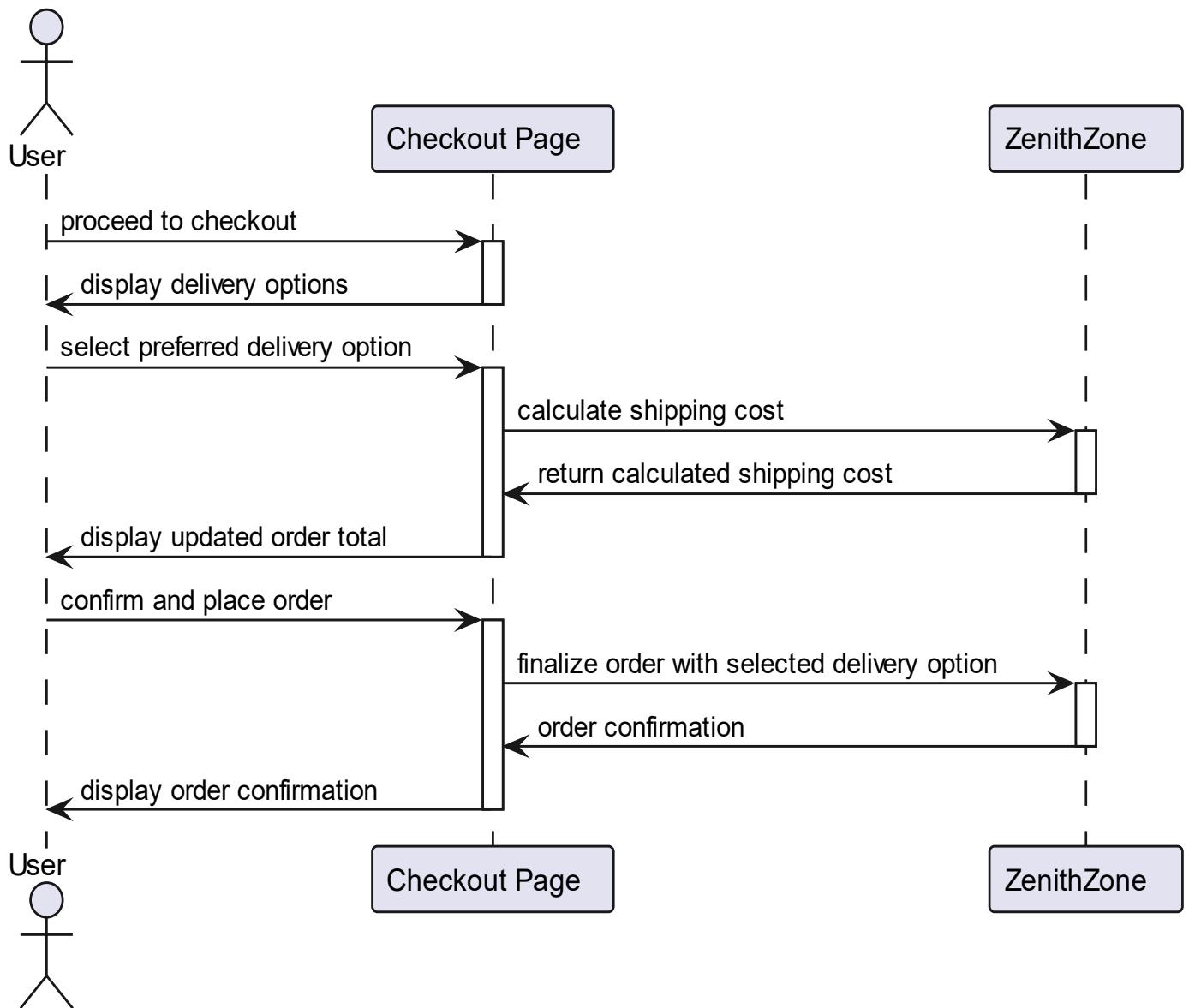
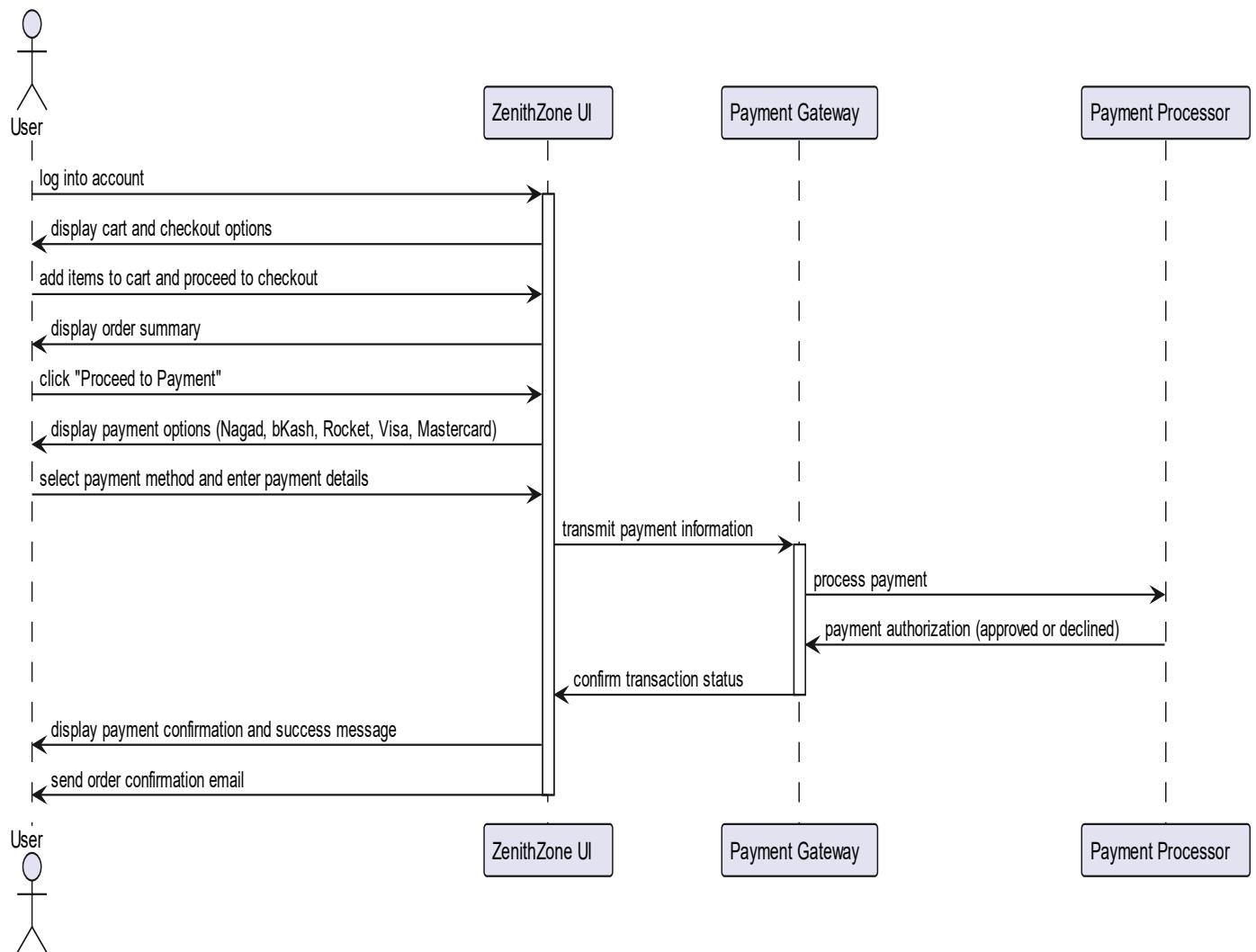


Figure 37: Sequence Diagram for Delivery options



**Figure 38: Sequence Diagram for Payment**

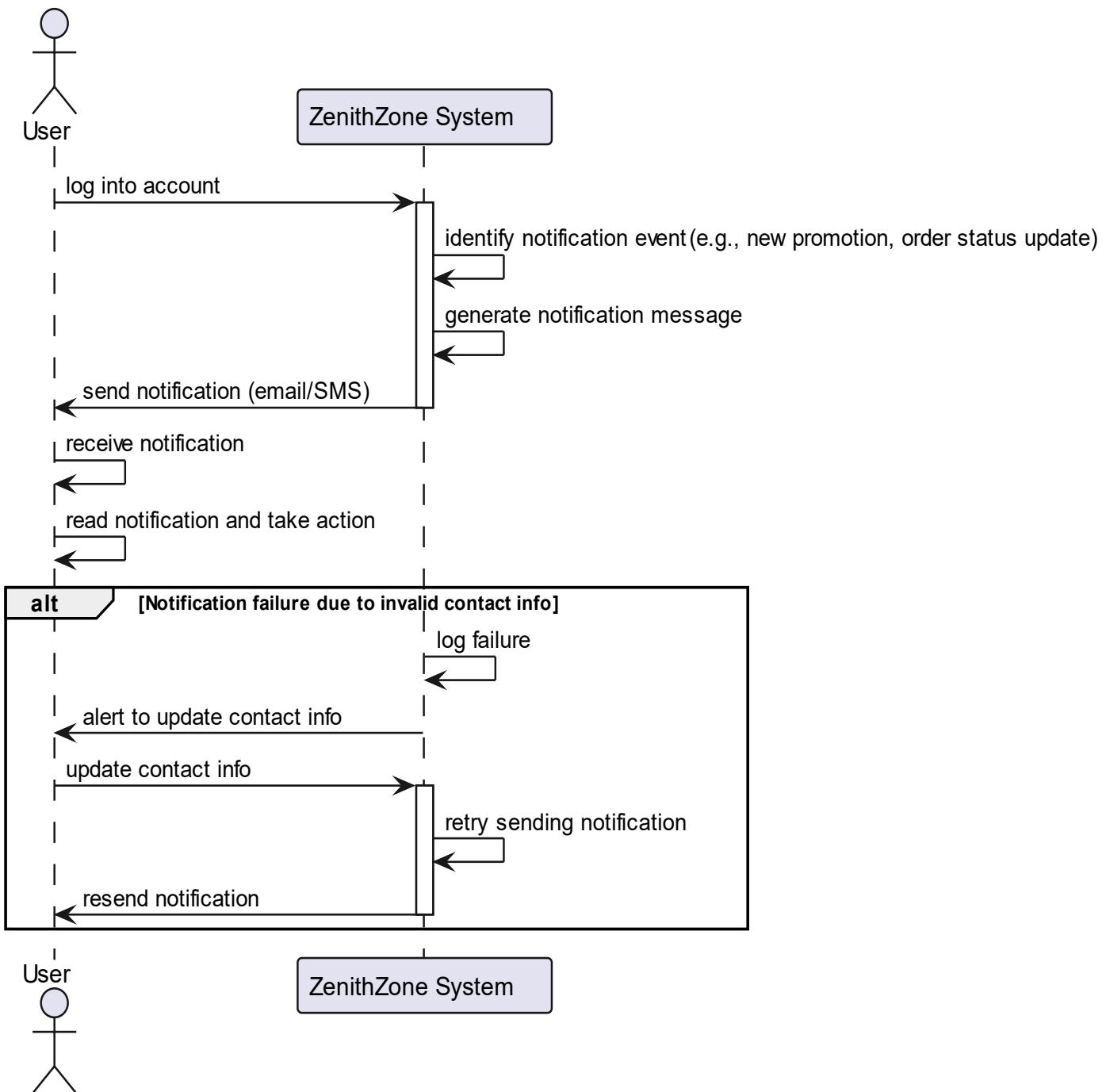
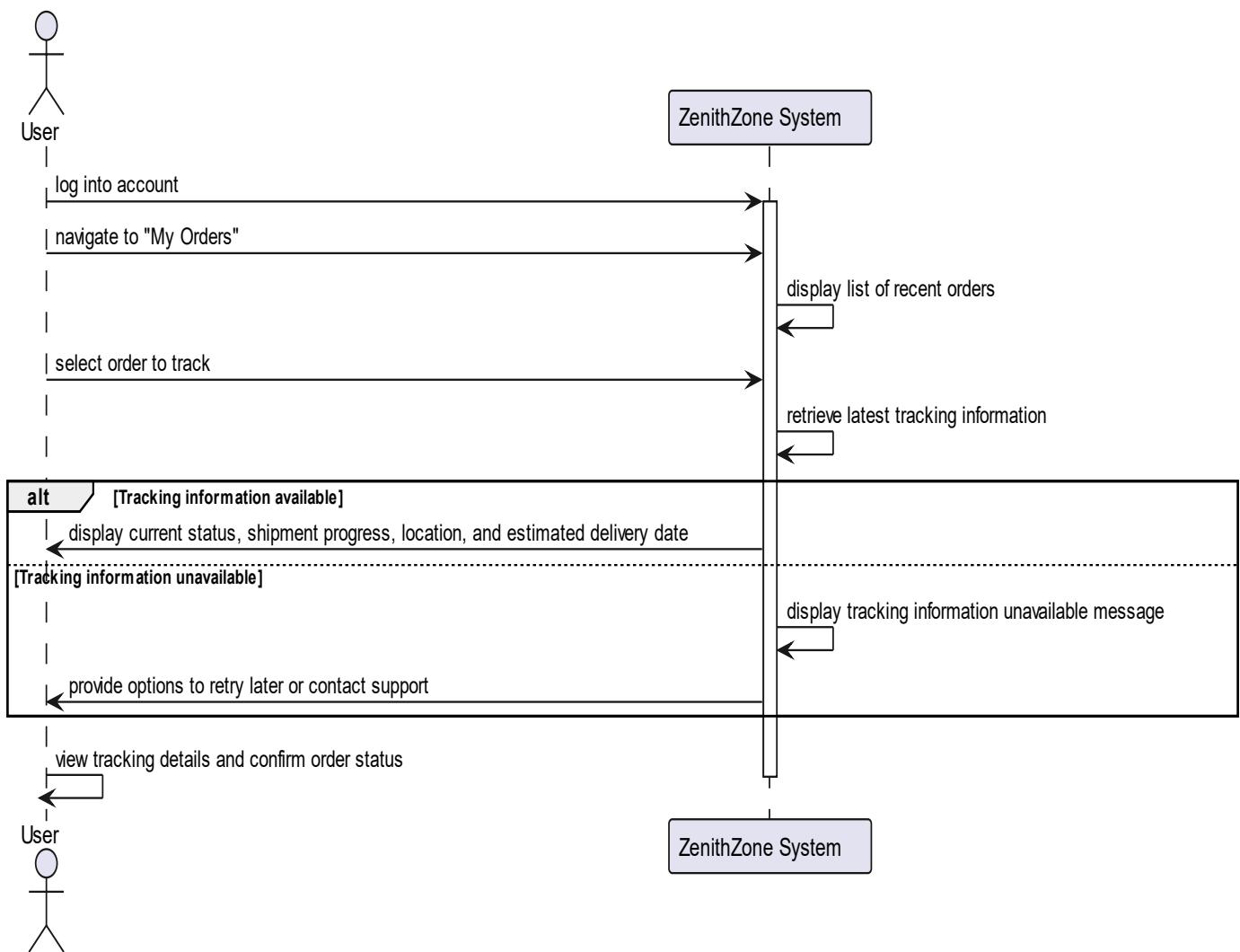
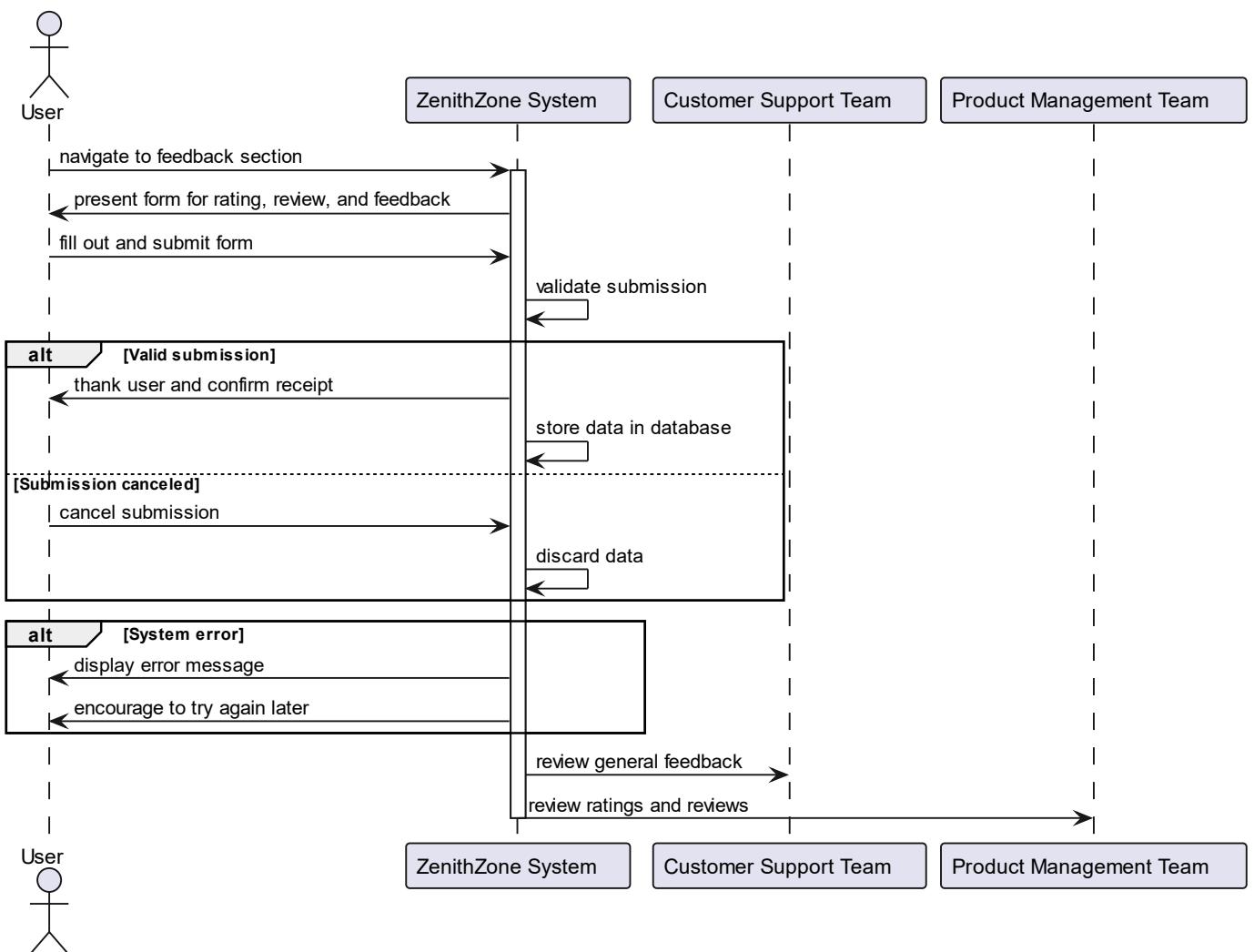


Figure 39: Sequence Diagram for Notification



**Figure 40: Sequence Diagram for Tracks Order**



**Figure 41: Sequence Diagram for Review**

## 10. Swimlane Diagram

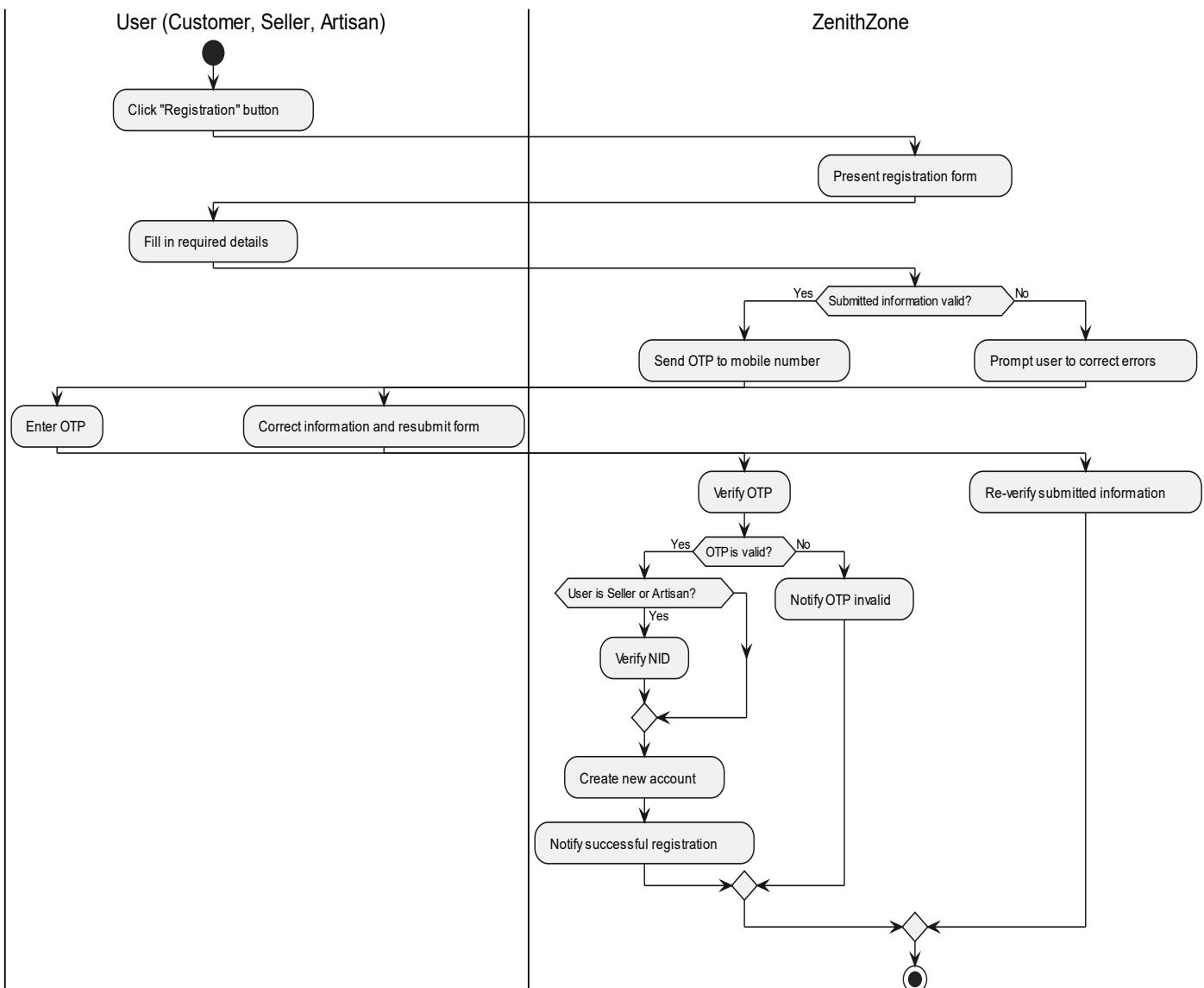
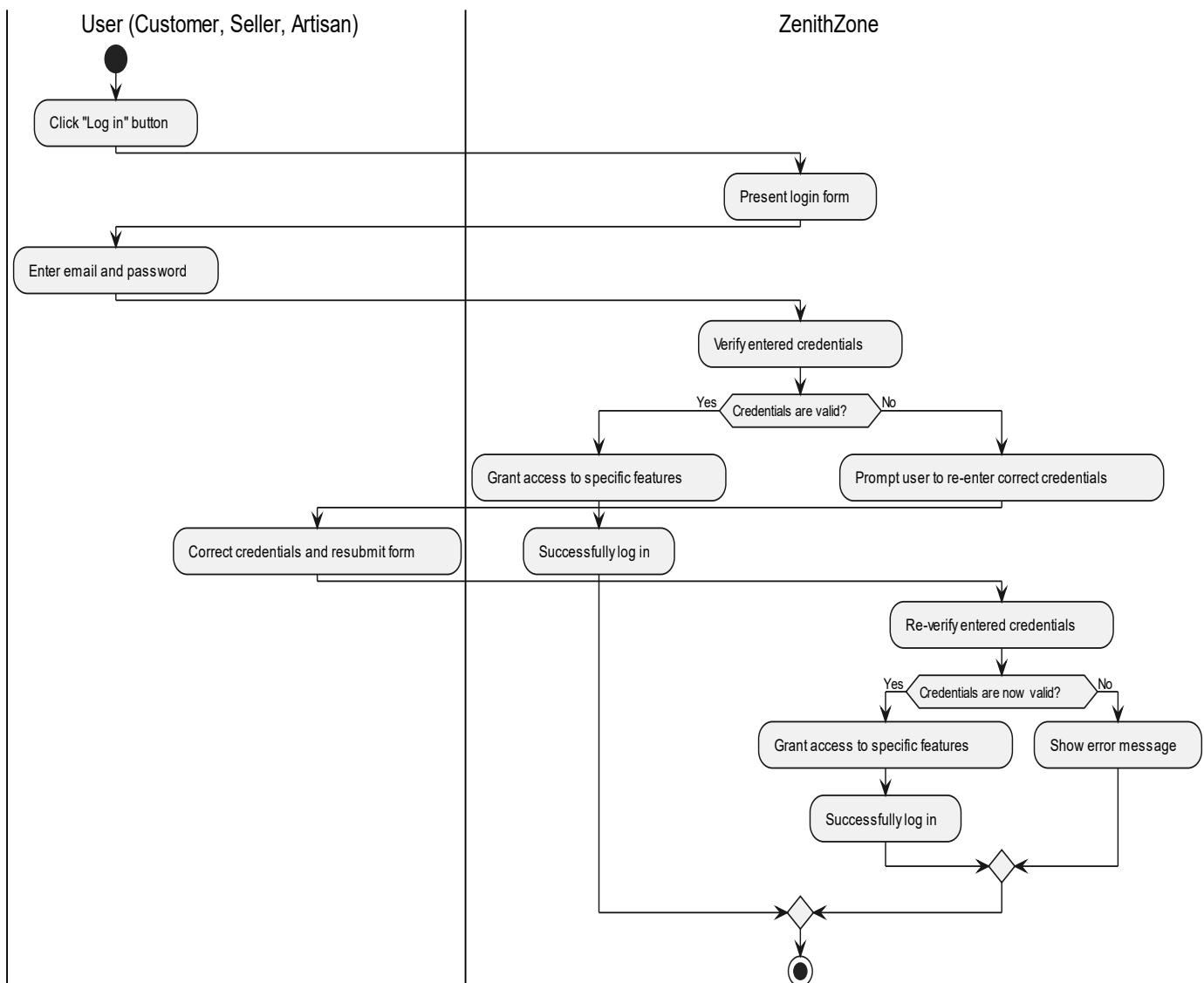


Figure 42: Swimlane Diagram for Registration Request



**Figure 43: Swimlane Diagram for Login**

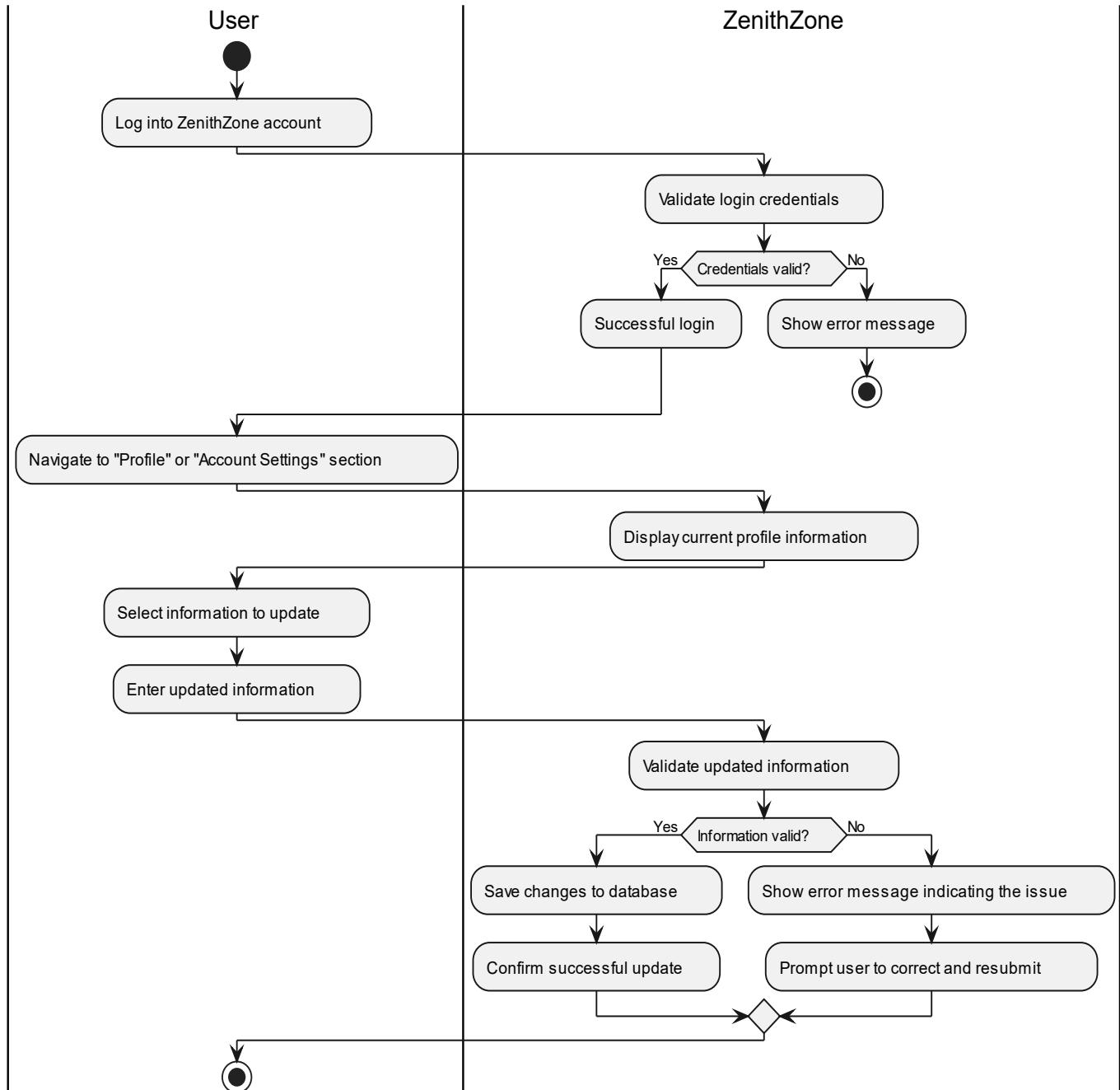
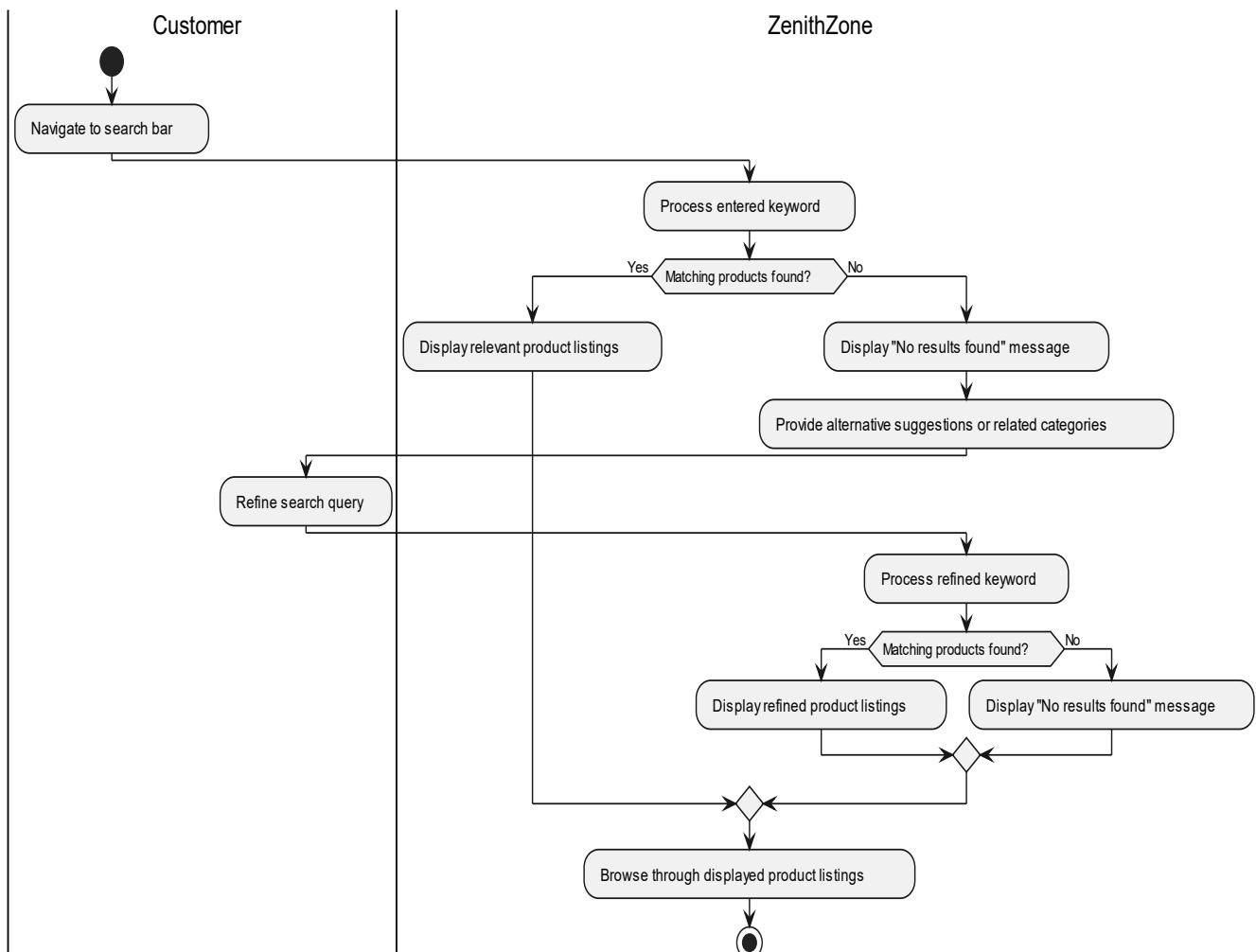
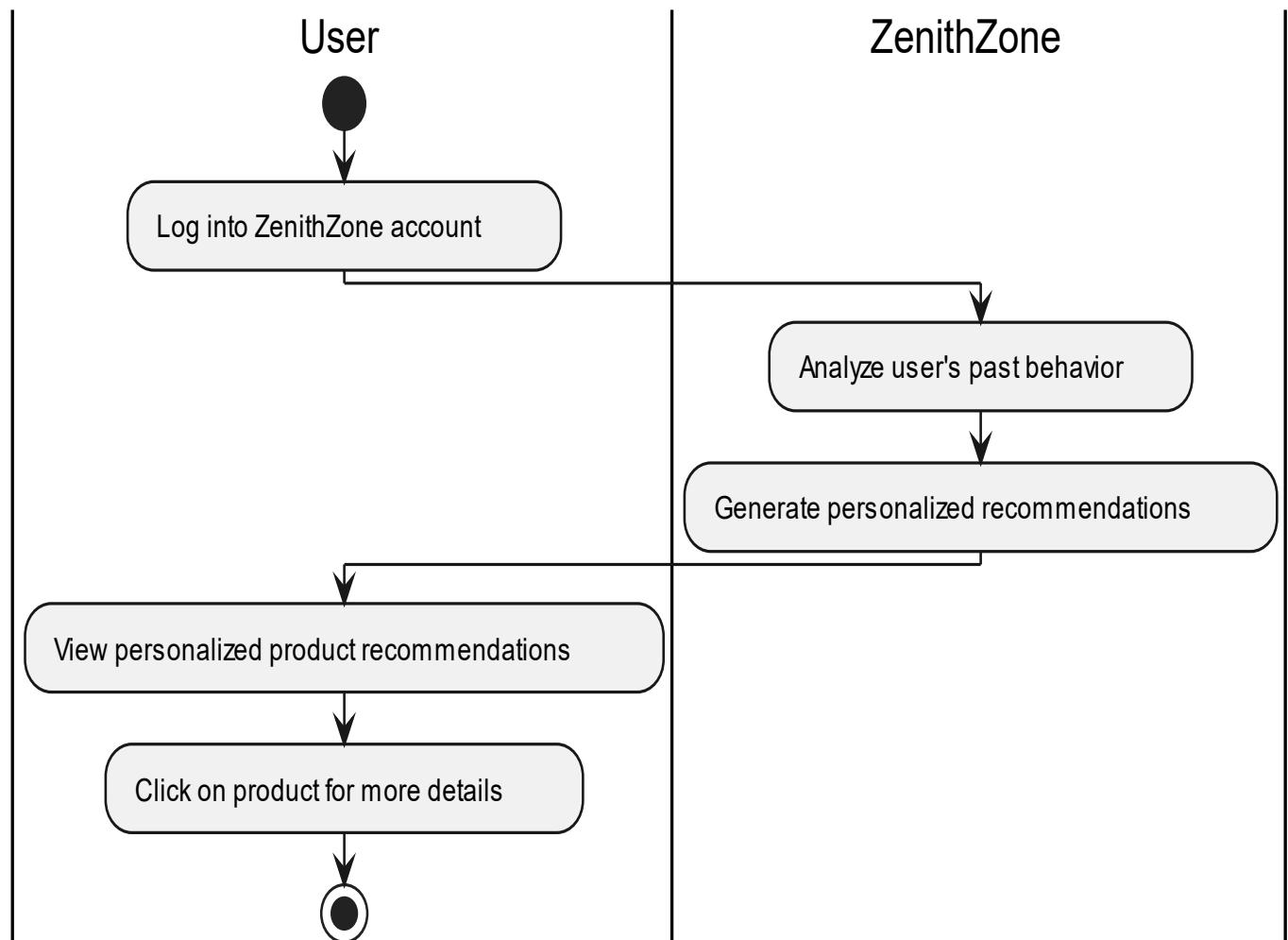


Figure 44: Swimlane Diagram for Profile Management



**Figure 45: Swimlane Diagram for Search Product**



*Figure 46: Swimlane Diagram for Personalize Recommendation*

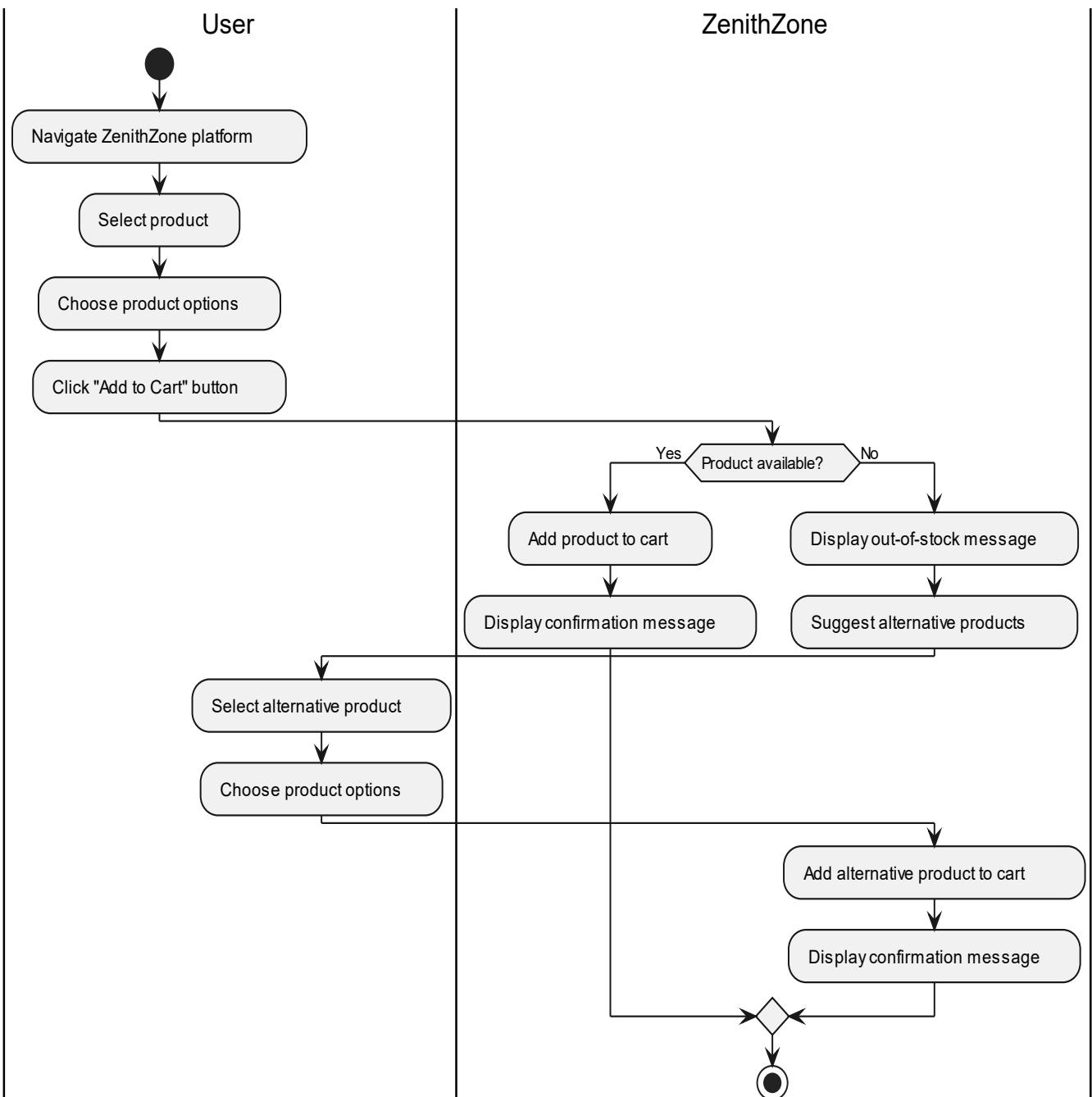
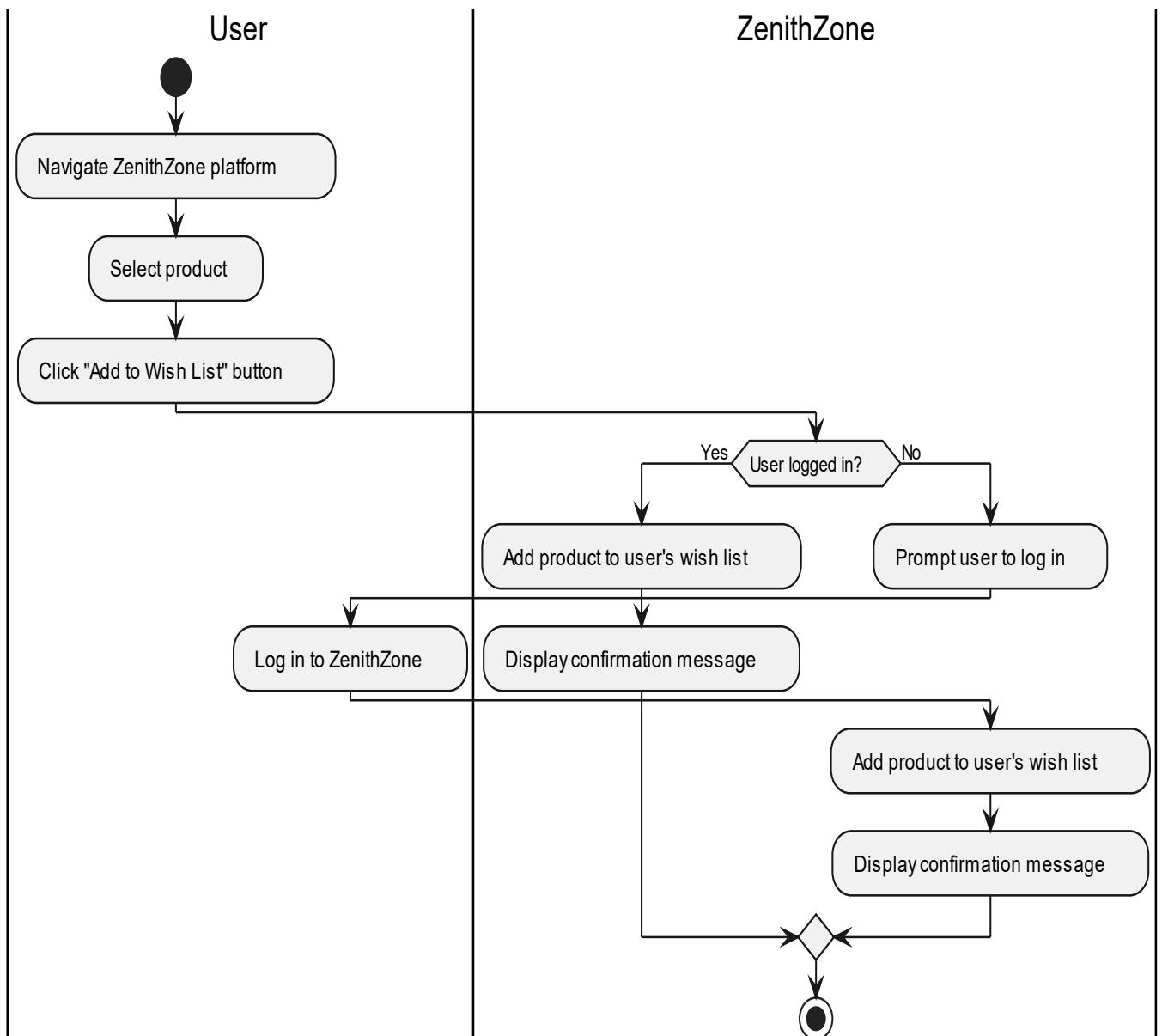
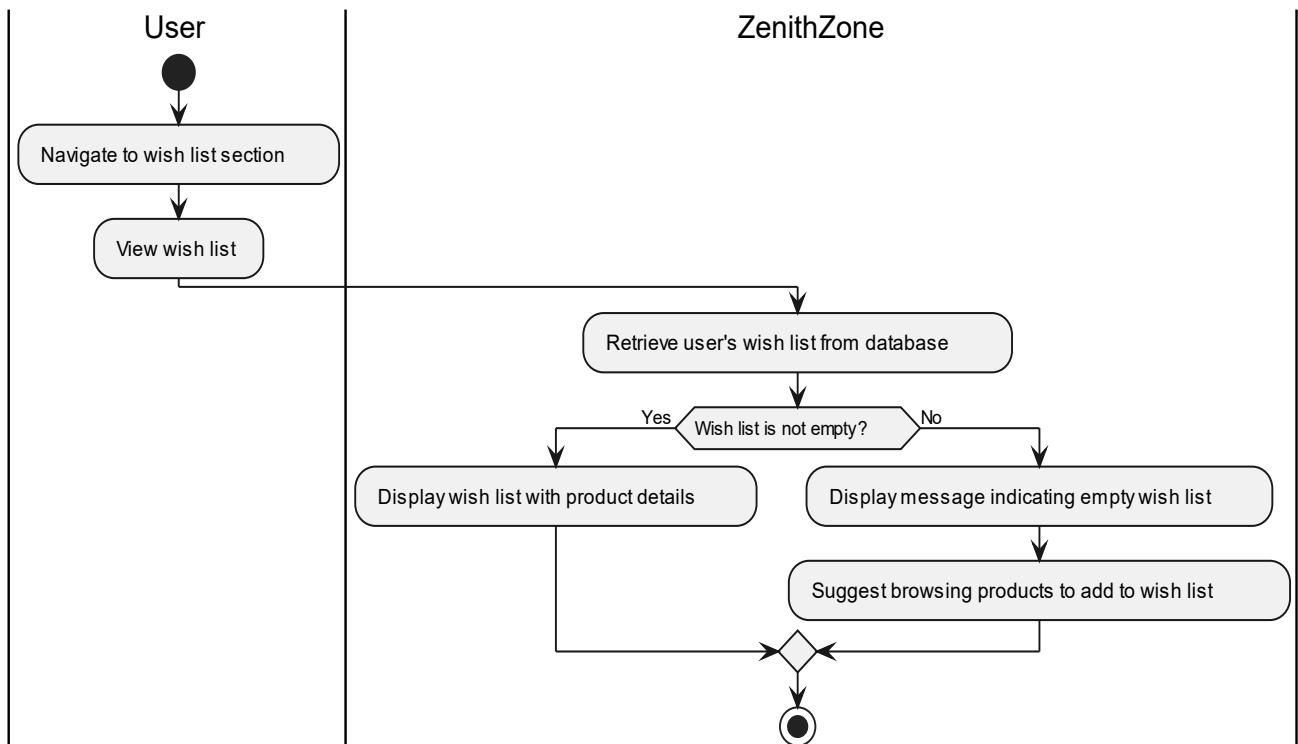


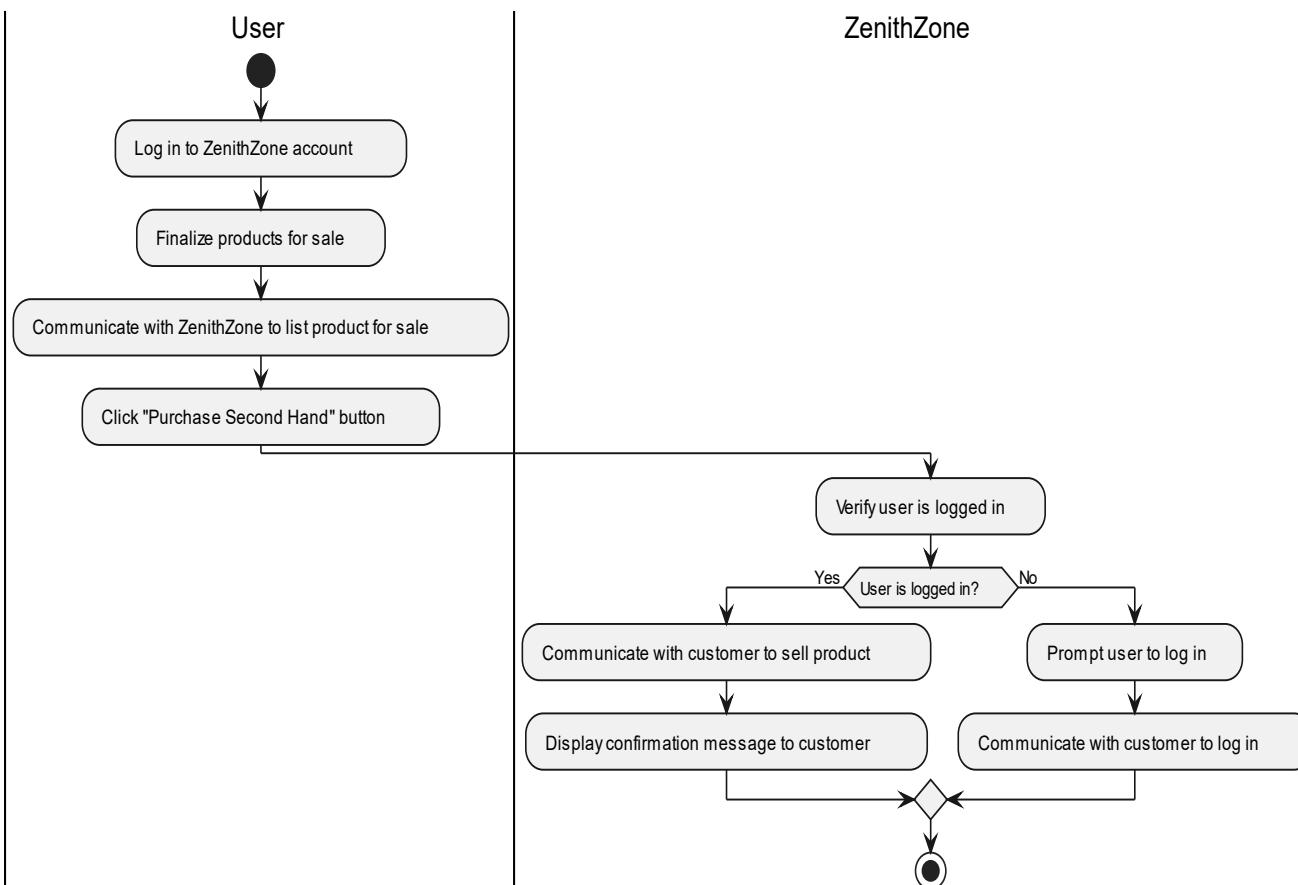
Figure 47: Swimlane Diagram for Add to Cart



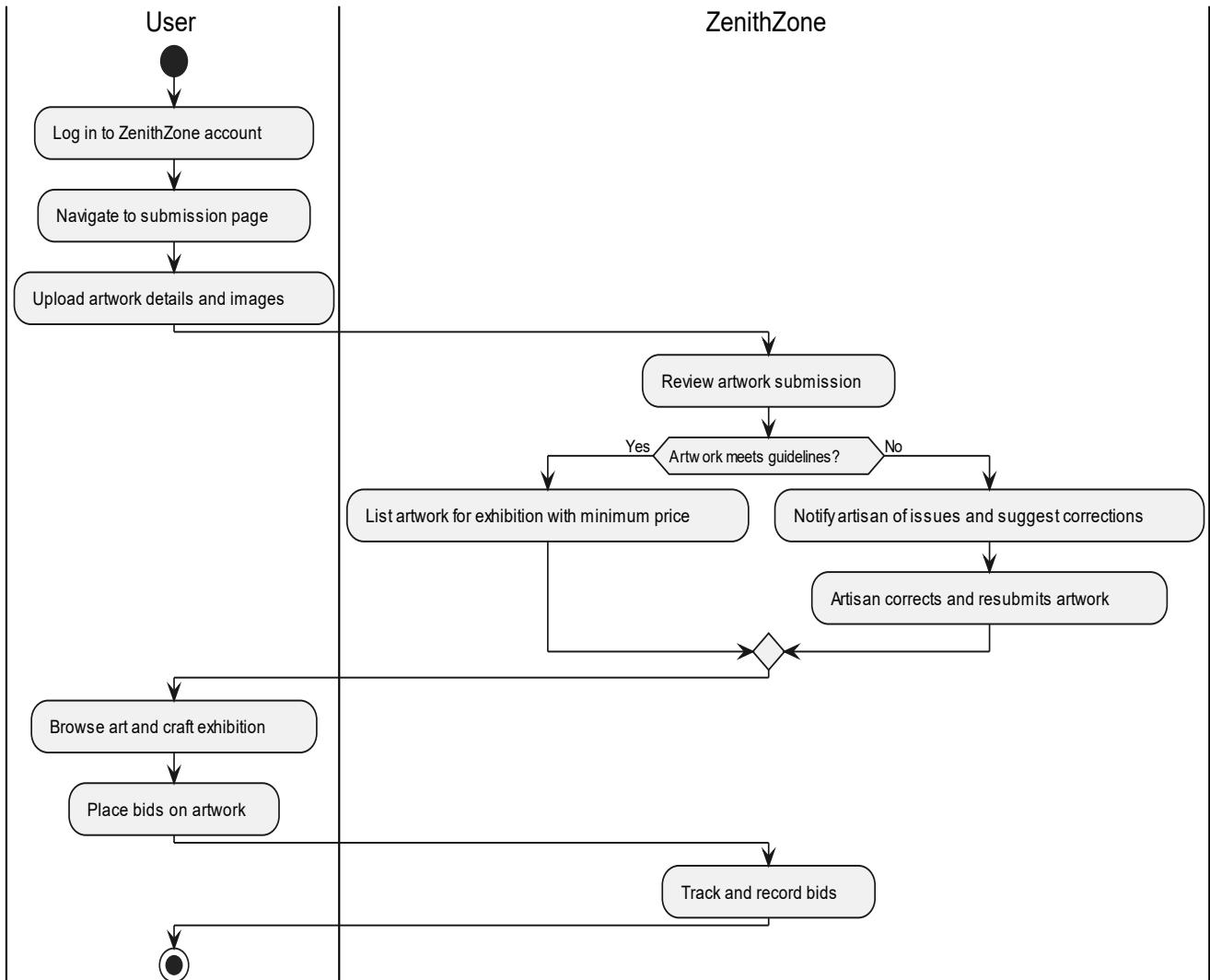
**Figure 48: Swimlane Diagram for Wish List**



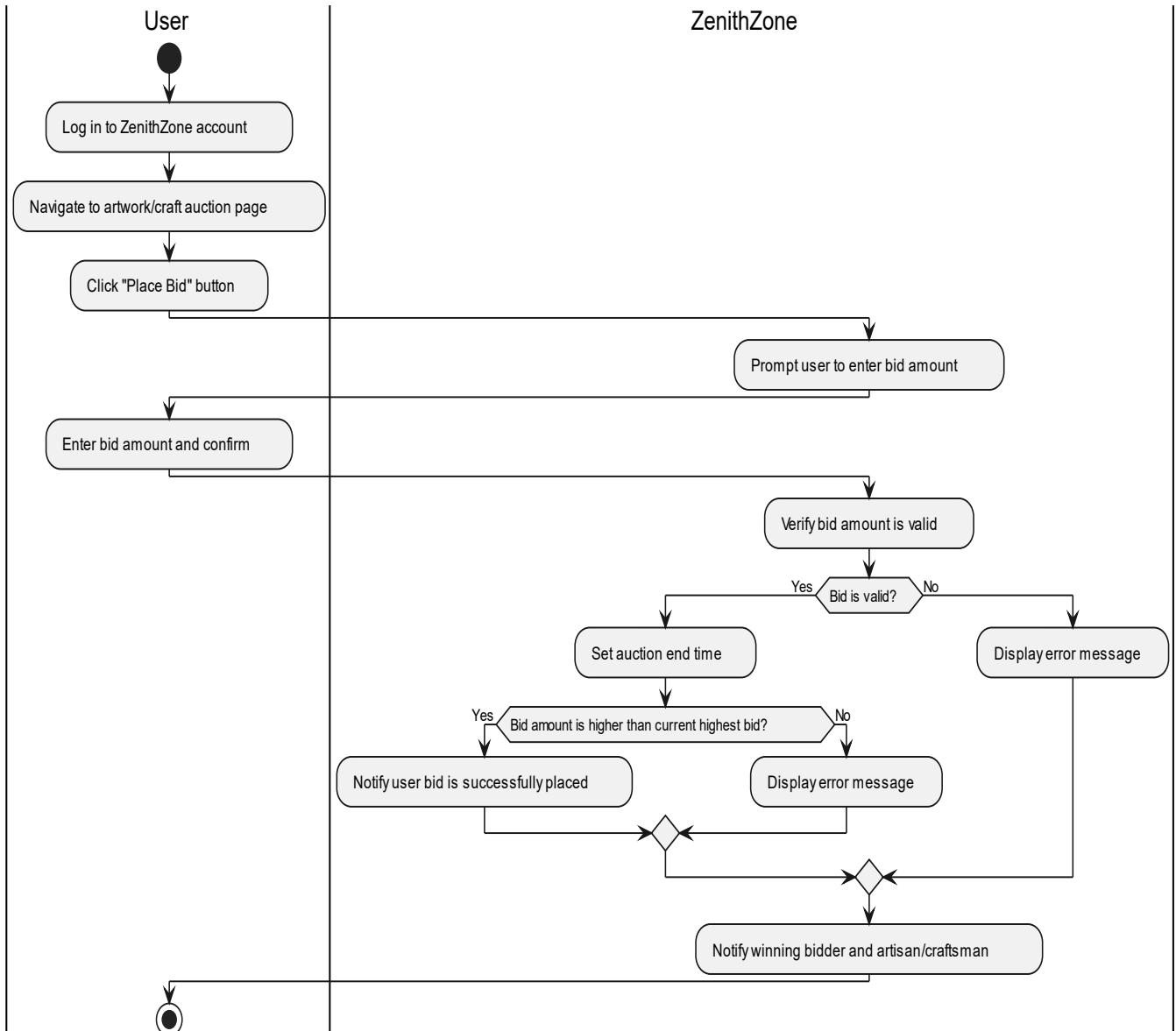
**Figure 49: Swimlane Diagram for View Wish List**



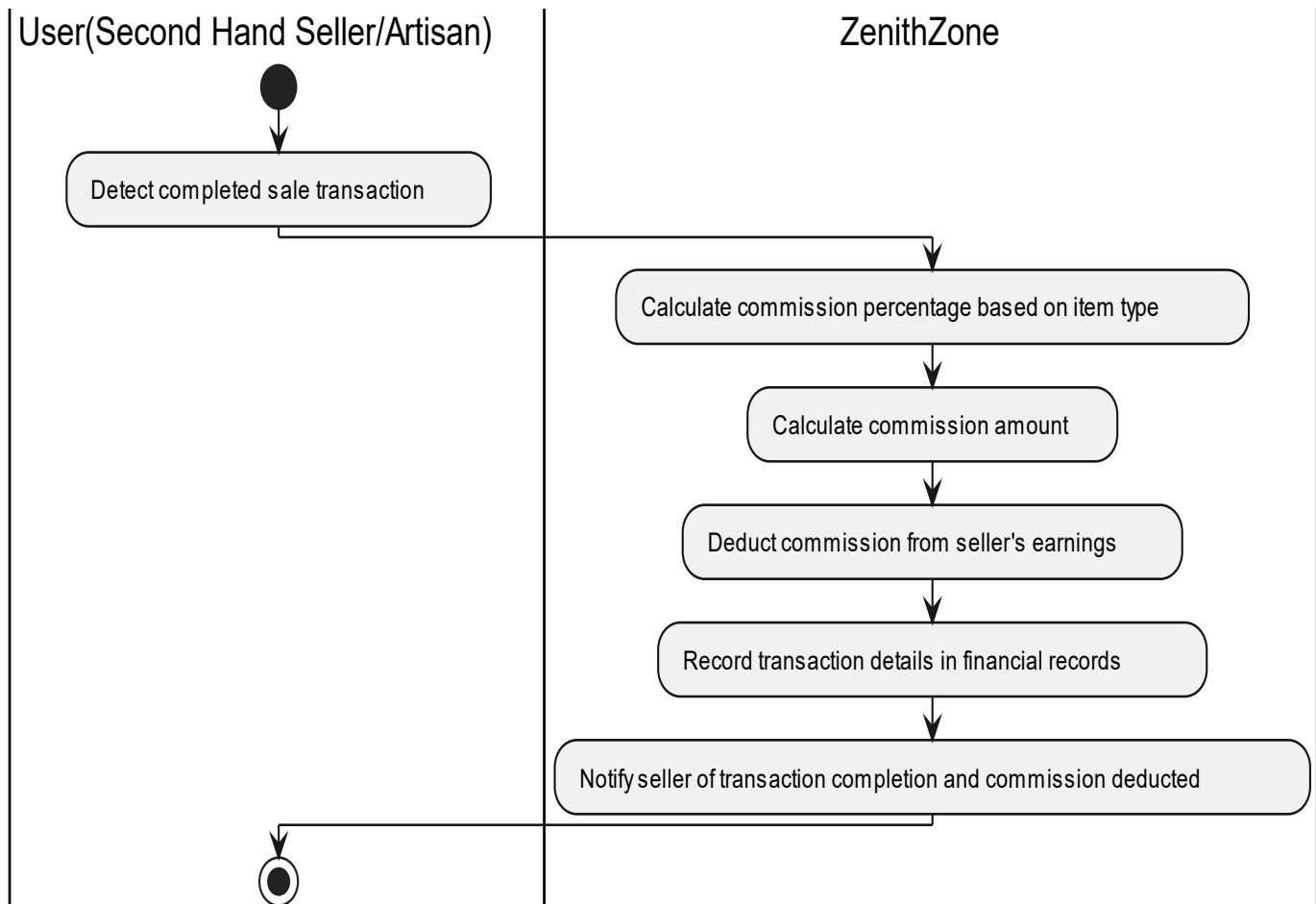
**Figure 50: Swimlane Diagram for Second Hand Product**



**Figure 51: Swimlane Diagram for Art/Craft**



**Figure 52: Swimlane Diagram for Bid on Art/Craft**



*Figure 53: Swimlane Diagram for Profit Margin*

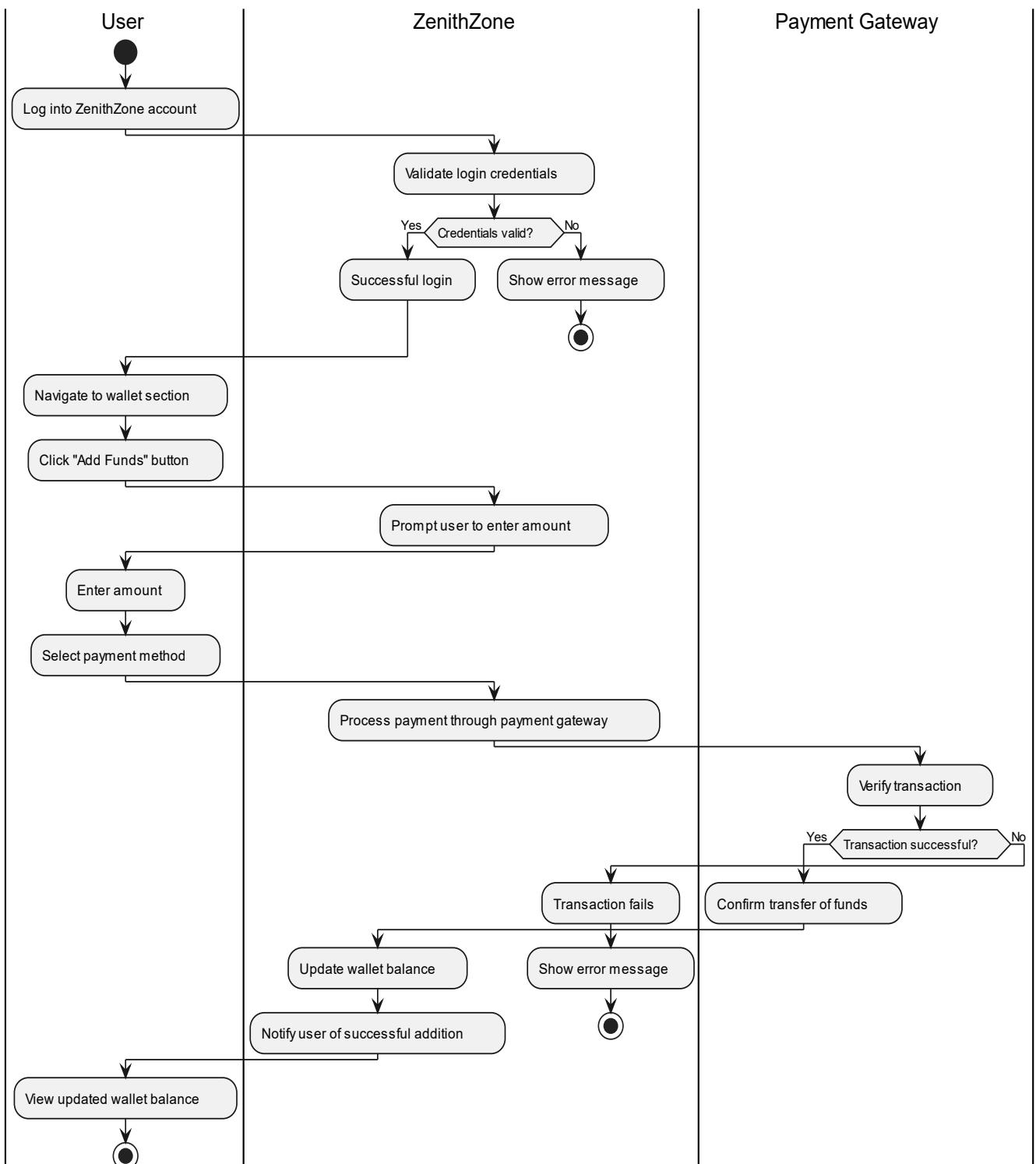


Figure 54: Swimlane Diagram for Wallet System

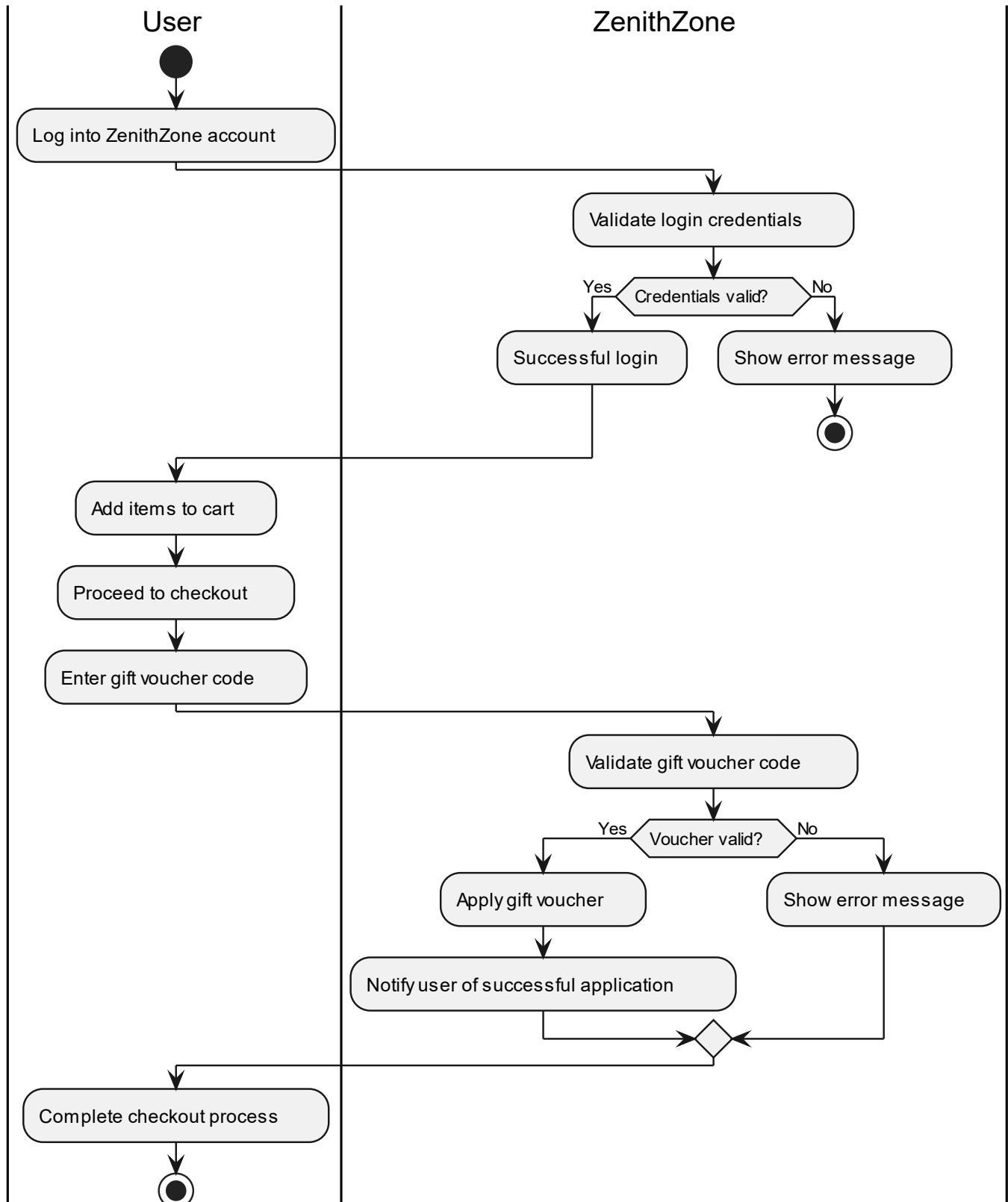
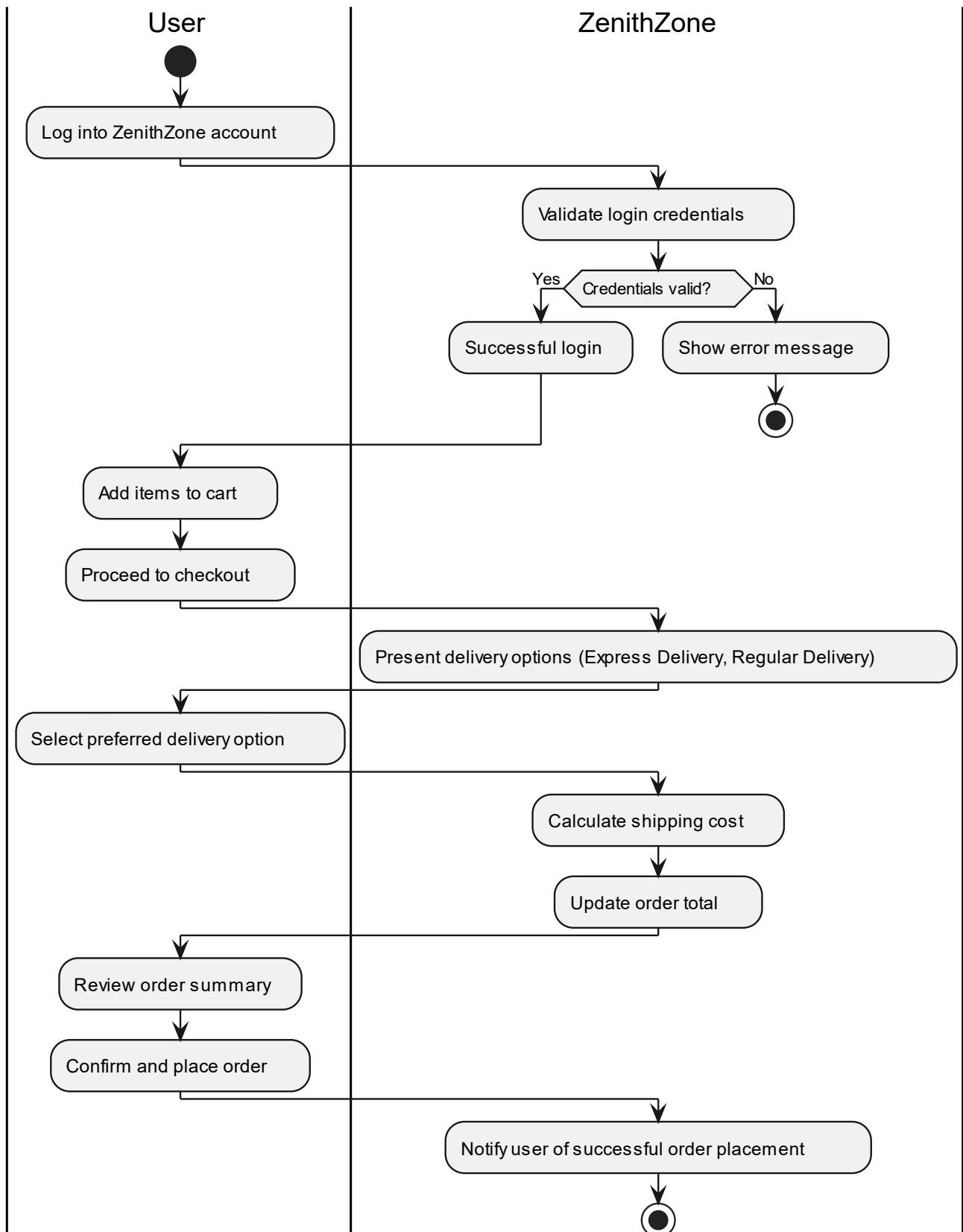
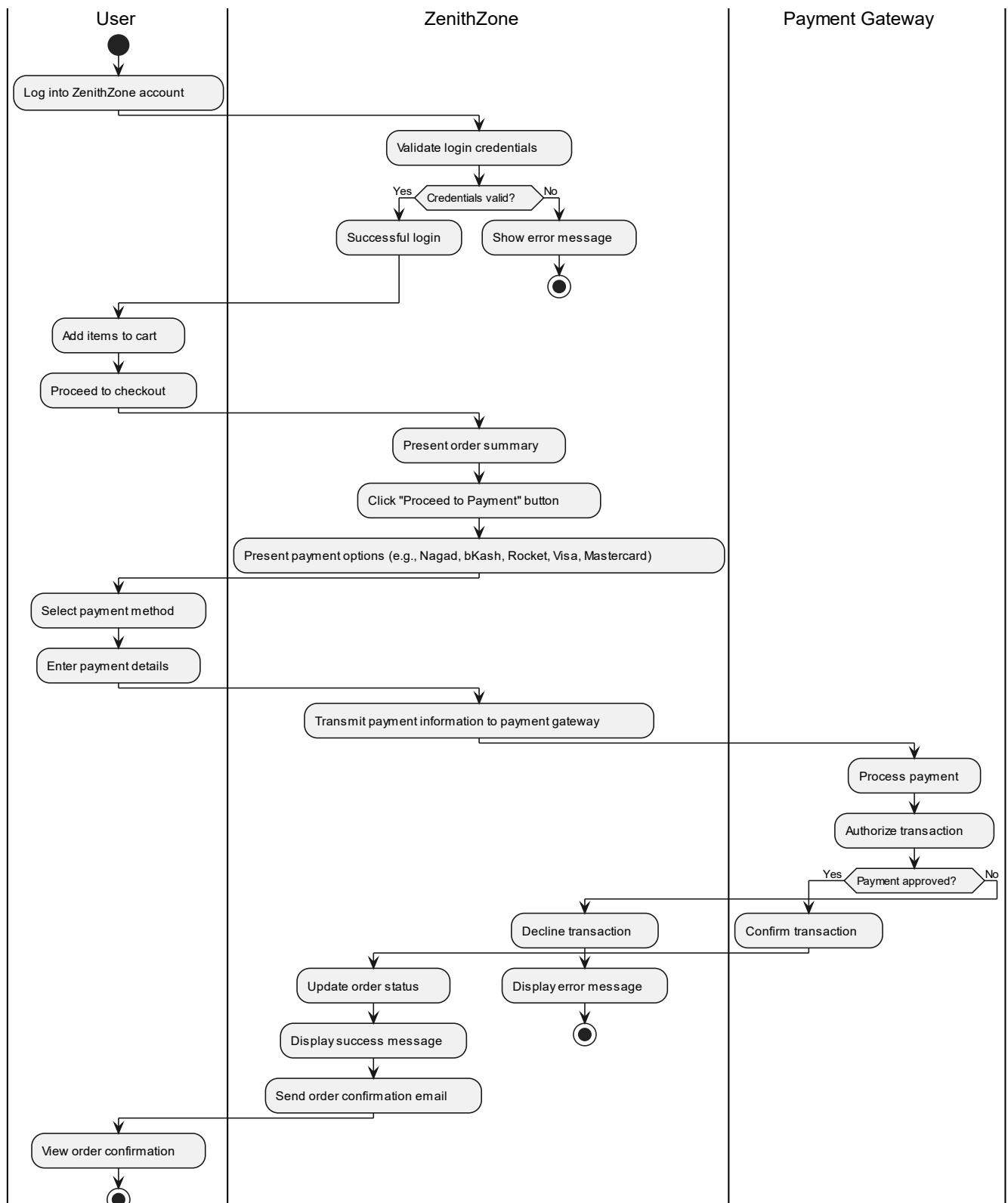


Figure 55: Swimlane Diagram for Gift Voucher



*Figure 56: Swimlane Diagram for Delivery Options*



**Figure 57: Swimlane Diagram for Checkout and Payment**

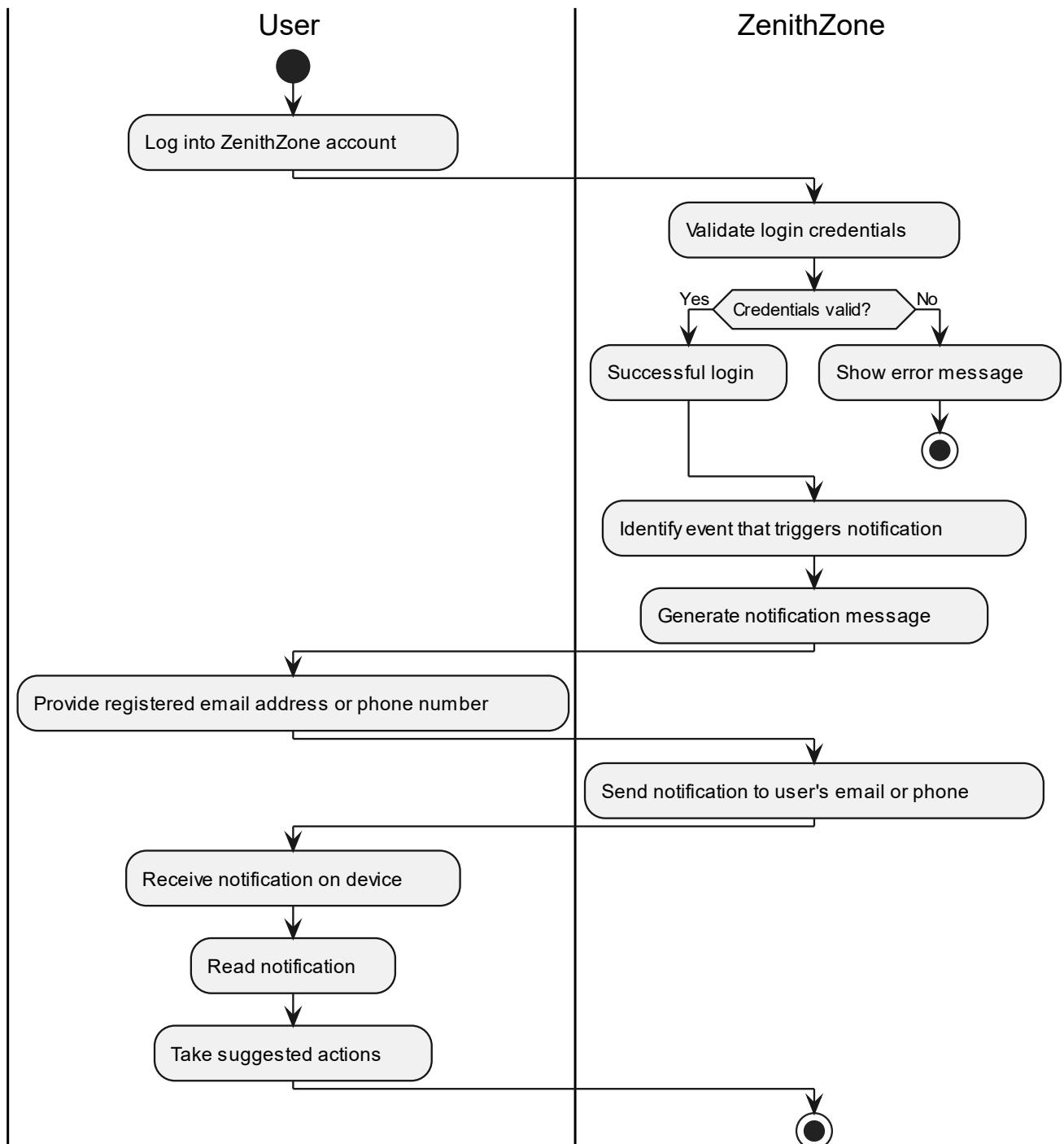


Figure 58: Swimlane Diagram for Notification

## Software Requirements Specification for ZenithZone

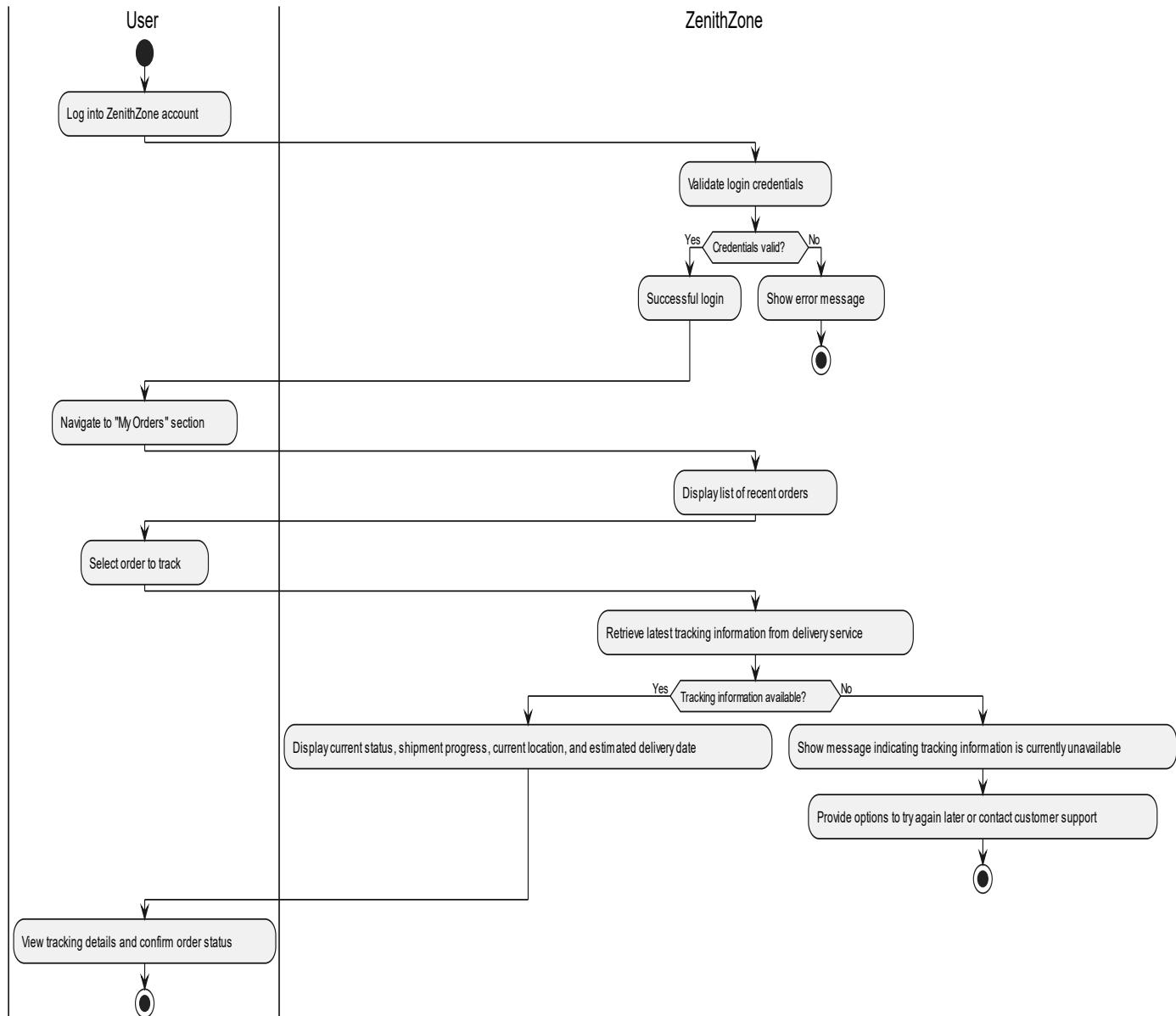


Figure 59: Swimlane Diagram for Tracks Order

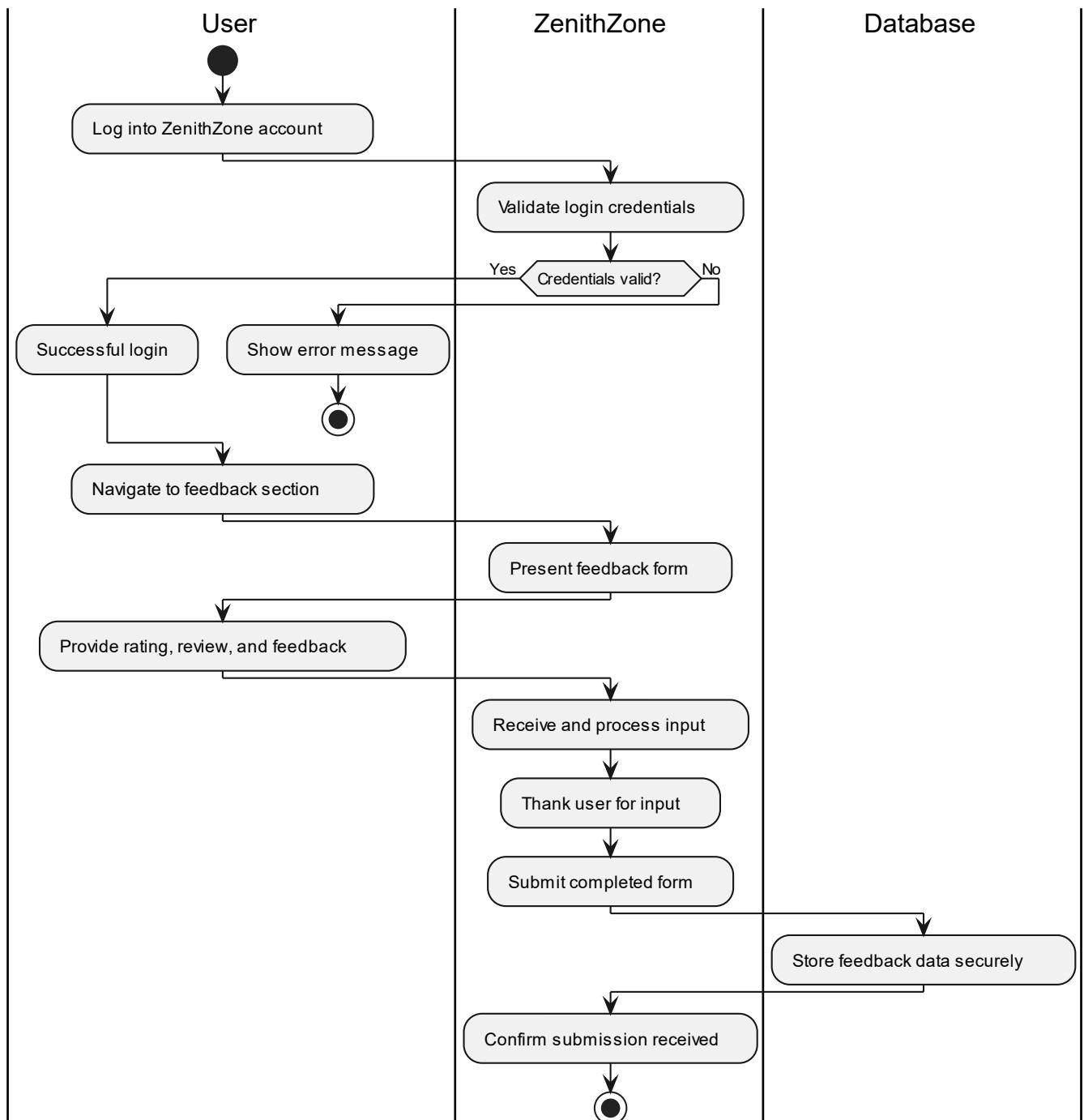
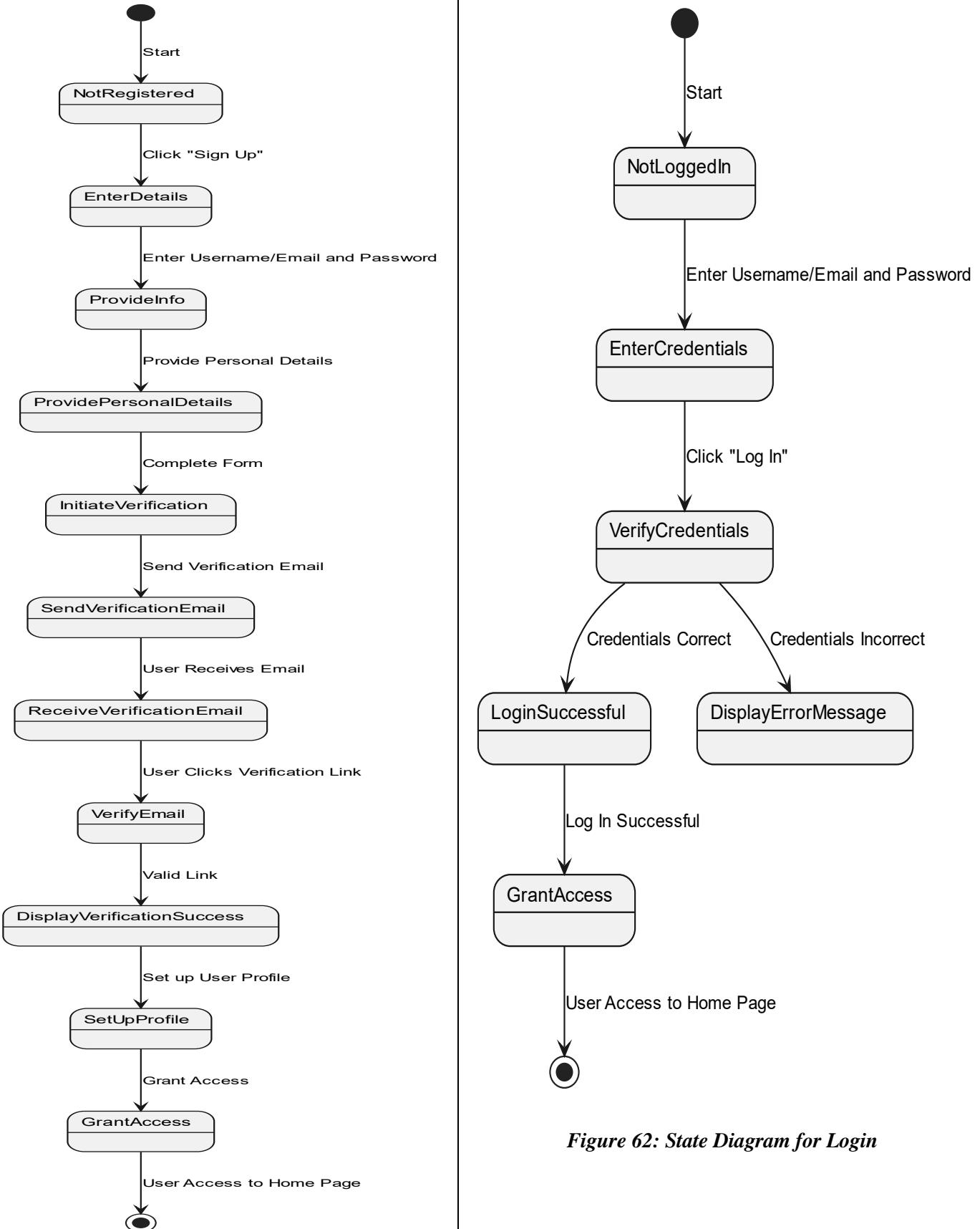


Figure 60: Swimlane Diagram for Review

## 11. State Diagram



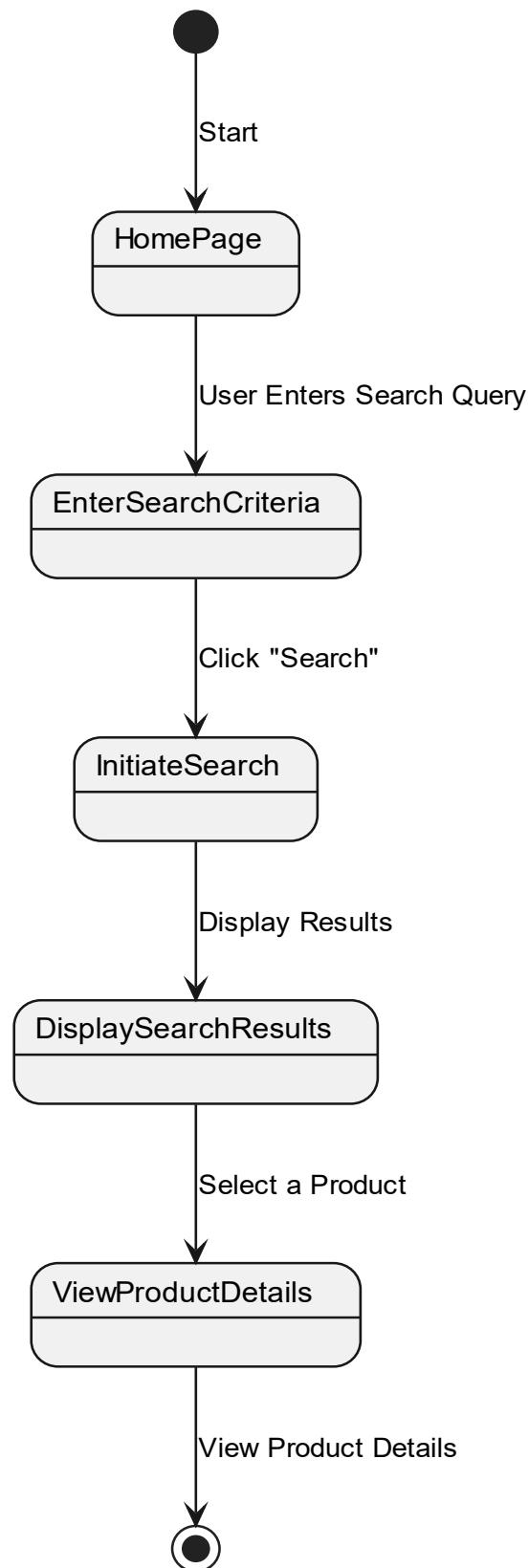


Figure 63: State Diagram for Search Product

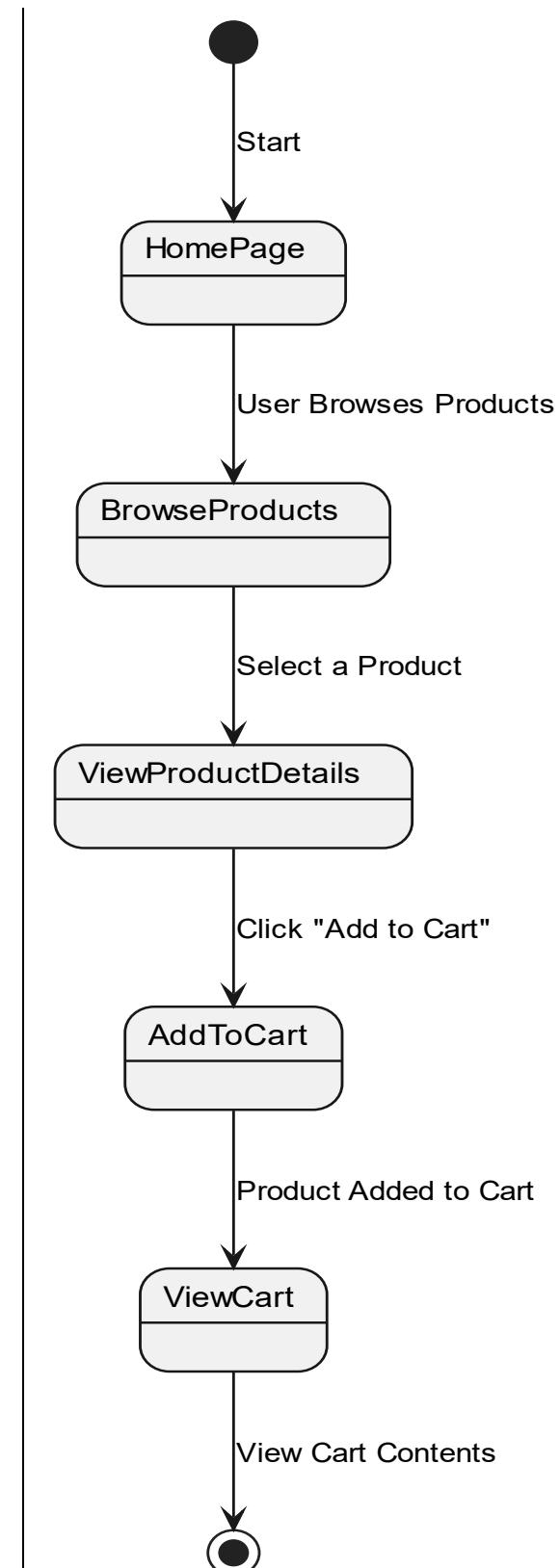


Figure 64: State Diagram for Add to Cart

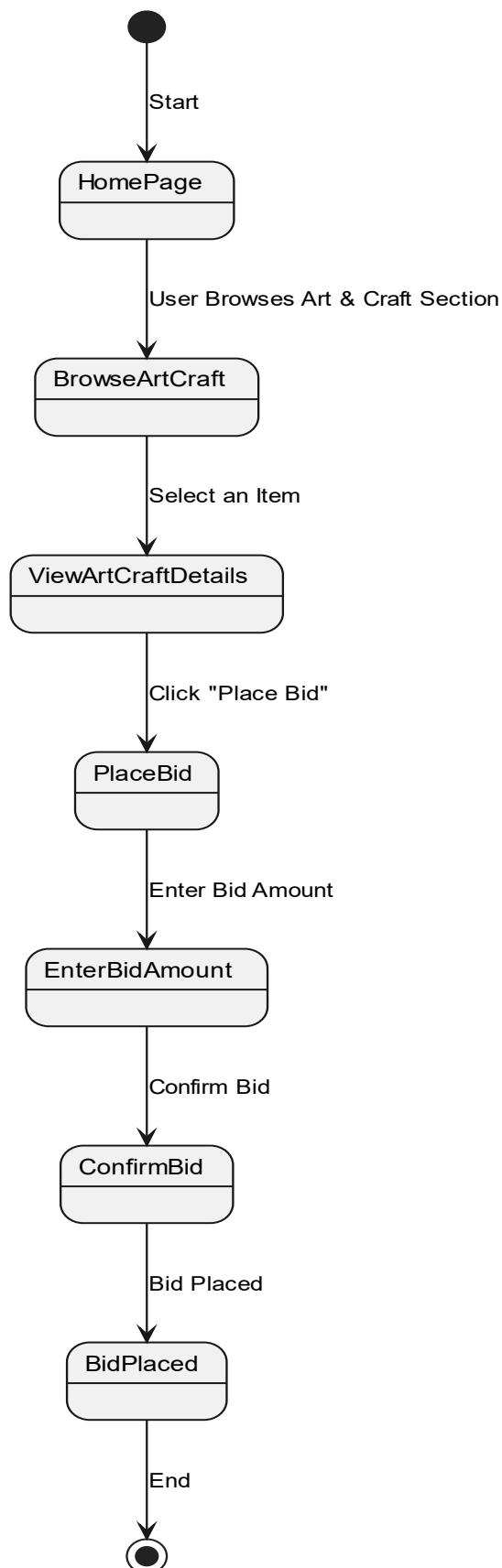


Figure 65: State Diagram for Art/Craft

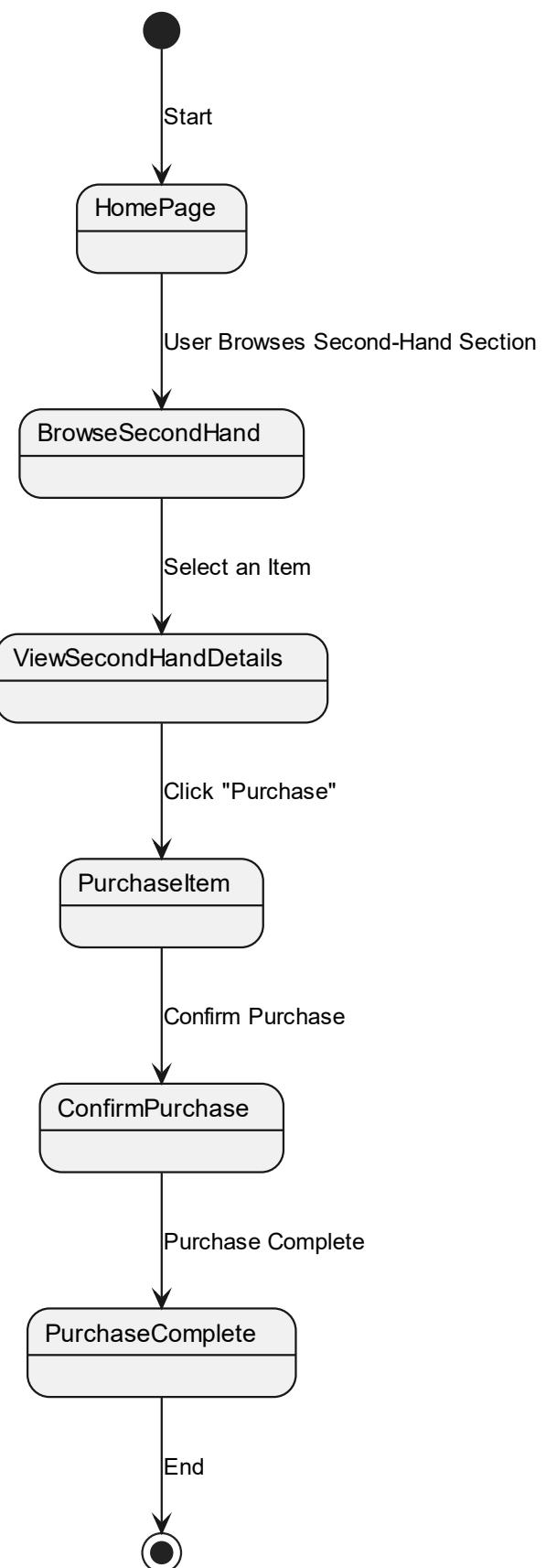
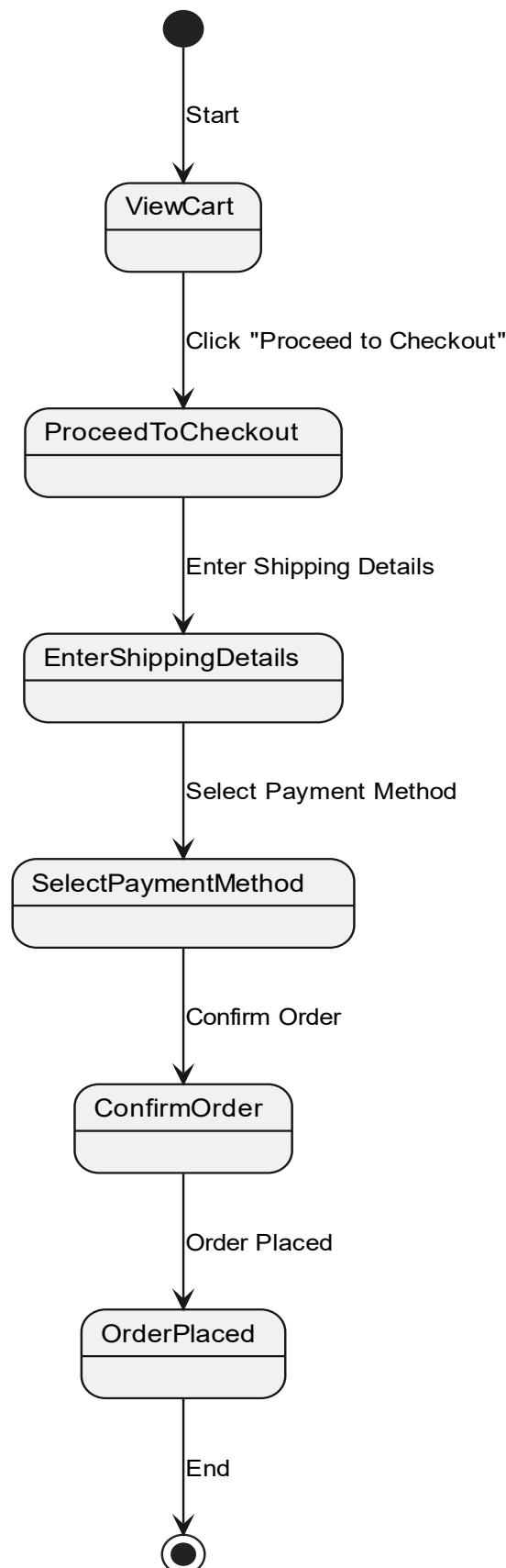
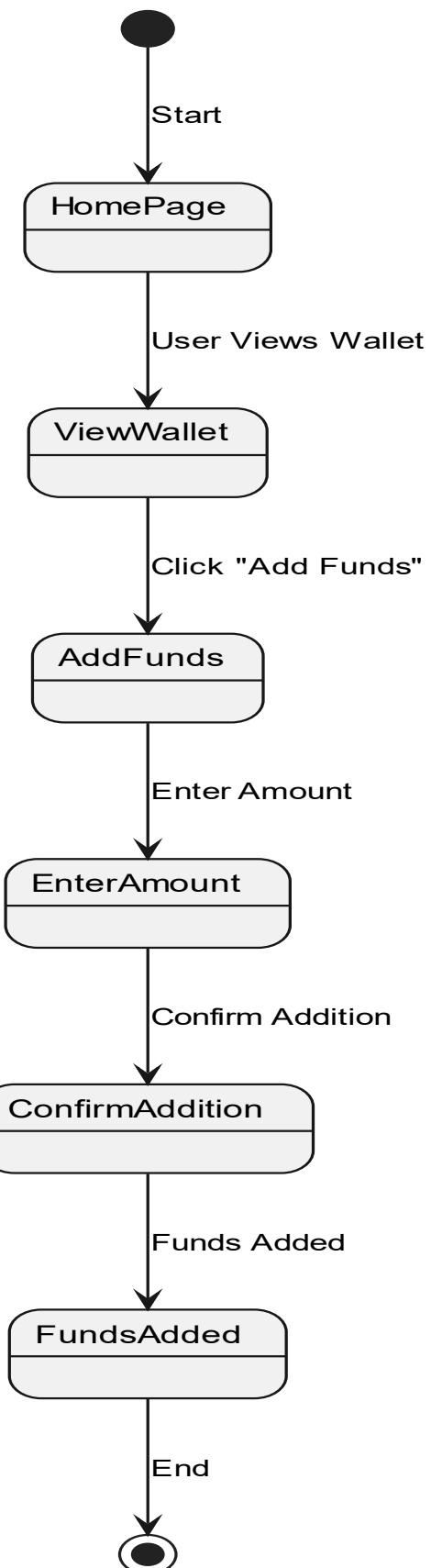


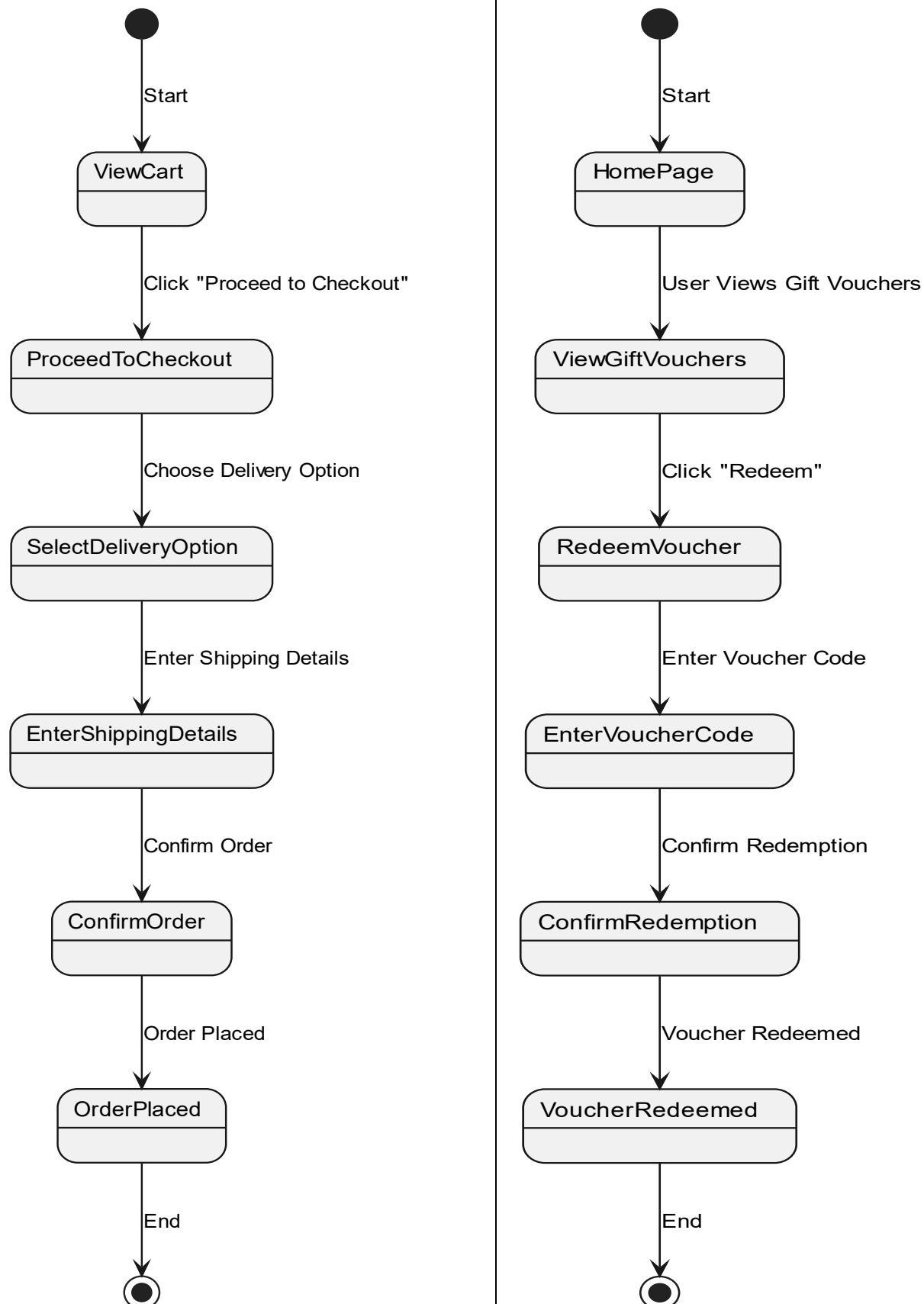
Figure 66: State Diagram for Second Hand Product

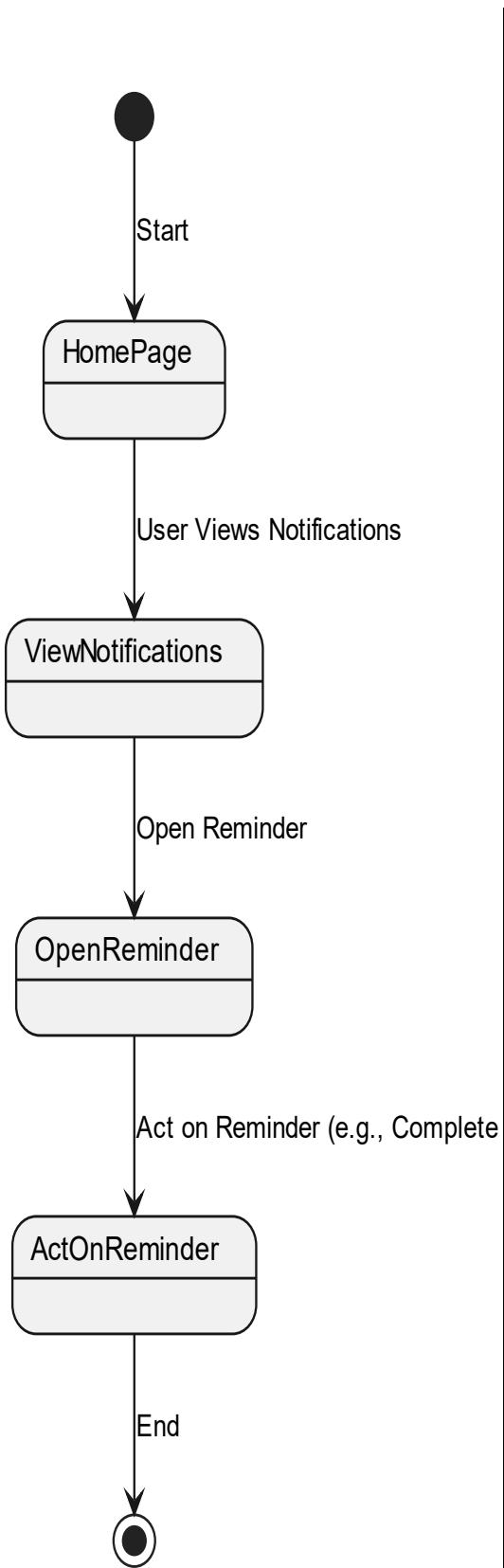


**Figure 67: State Diagram for Checkout and Payment**

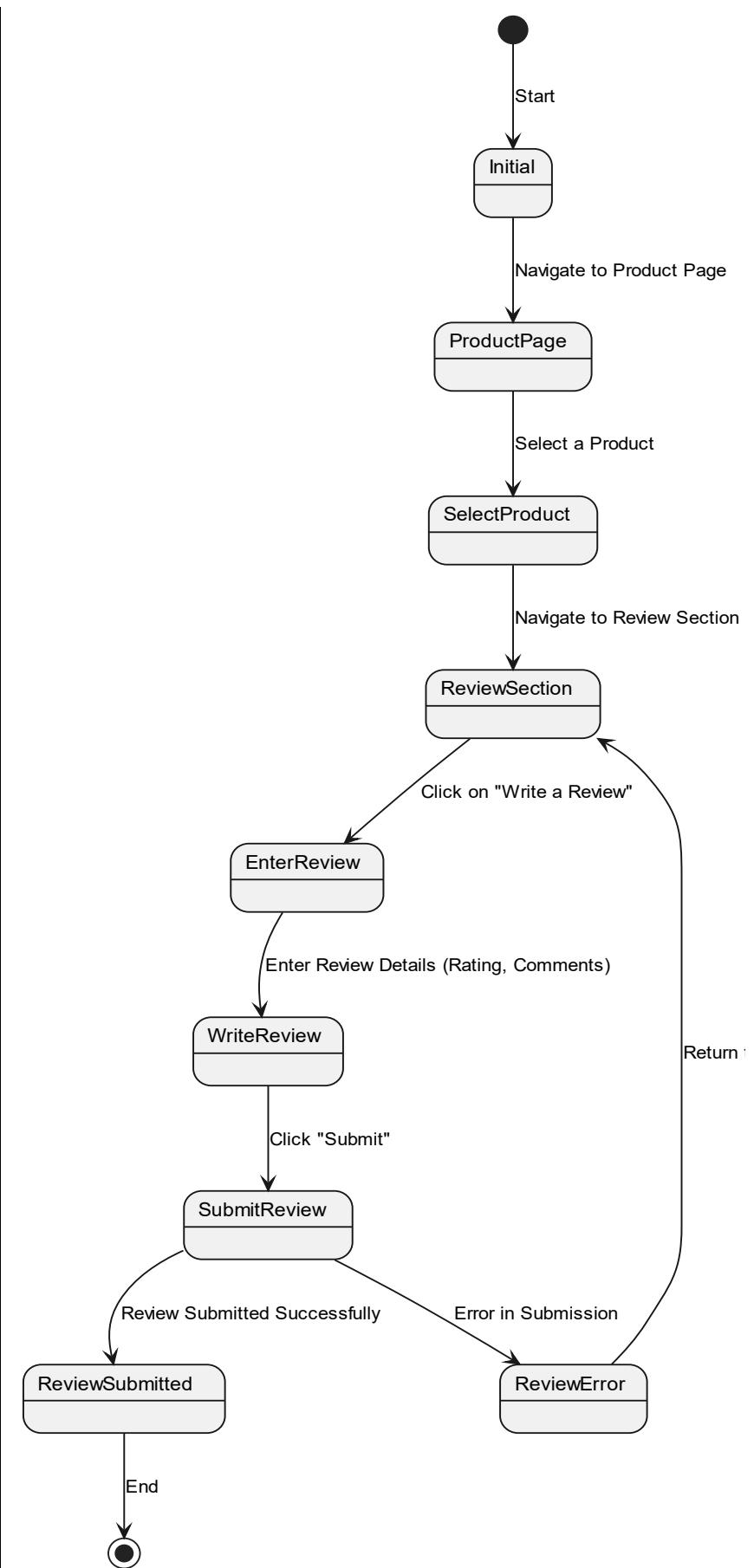


**Figure 68: State Diagram for Wallet**





**Figure 71: State Diagram for Reminder**



**Figure 72: State Diagram for Review**

## 12. Glossary

This section provides definitions for all document names, acronyms, and abbreviations. The application domain's terms and concepts are defined.

GUI	Graphical User Interface
API	Application Programming Interface
SRS	Software Requirement Specification
UI	User Interface
SDLC	Software Development Life Cycle
MB	Megabytes
XML	Extensible Markup Language
RESTful	Representational State Transfer
HTML	Hyper Text Markup Language

## 13. References

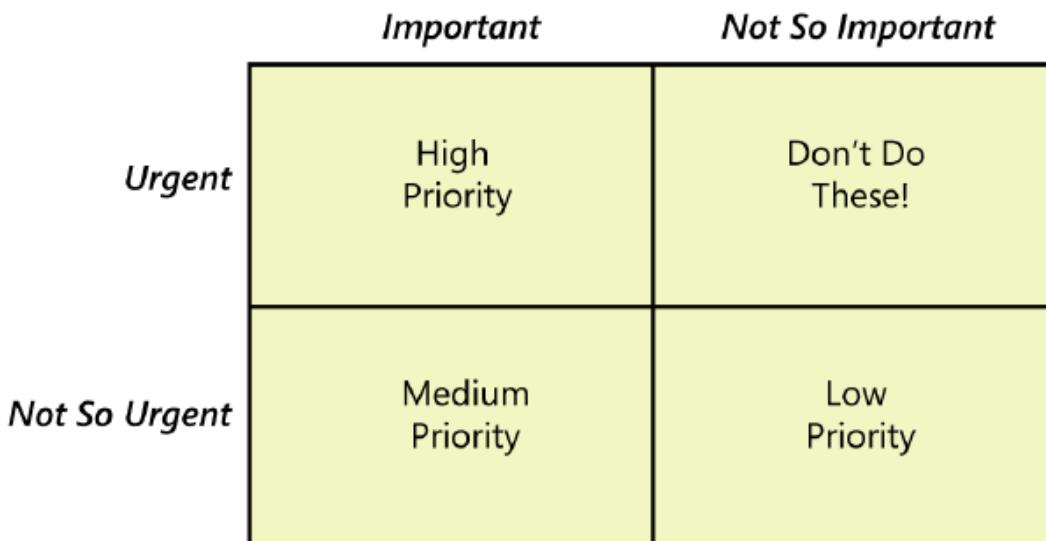
1. IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998.
2. Software Engineering 9<sup>th</sup> Edition by Ian Sommerville.
3. Requirements Engineering Fundamentals by Klaus Pohl
4. Database System Concepts 6<sup>th</sup> Edition by Abraham Silberschatz.
5. Database Design Using Entity-Relationship Diagrams by Sikha Saha Bagui And Richard Walsh Earp

## 14. Appendix

### 14.1 Prioritization of requirements

#### 14.1.1 Three-level scale of requirement prioritized.

In our prioritization approach, we employ a three-level scale to categorize requirements based on their importance to the success of the project:



- High Priority:** These requirements are fundamental to the project's success and are crucial for achieving the project's objectives. They typically pertain to core functionalities that are essential for the product or service. Failing to implement these requirements could lead to significant negative consequences for the project.
- Medium Priority:** Requirements in this category are important for the project's success but may not be critical. While they contribute to enhancing the user experience or providing additional functionalities, their absence would not necessarily lead to project failure.
- Low Priority:** These requirements are desirable to have but are not essential for the project's success. They may offer supplementary features or improvements to the user experience, but their absence would not significantly impact the project's overall success.
- Don't Do This:** This category includes items that are less important but still require attention. They should be addressed after completing higher-priority medium priority items. Items on the right side of the dividing line within this category hold a higher priority and should be considered for implementation once higher-priority tasks are completed.

This prioritization scale serves as a guide for teams to make informed decisions about resource allocation and project focus. It helps in managing trade-offs and addressing competing demands effectively throughout the project lifecycle.

### **14.1.2 Prioritization of Requirements of ZenithZone**

#### **➤ Functional Requirements Prioritization:**

##### **1. High Priority:**

- FR-1: Sign Up
- FR-2: Login
- FR-11: Art and Craft Exhibition
- FR-12: Second Hand Sale
- FR-13: Checkout and Payment
- FR-20: Upload Listing

##### **2. Medium Priority:**

- FR-4: Search Items
- FR-5: View Items
- FR-6: Add to Cart
- FR-8: Personalized Product Recommendation
- FR-8: Auto Complete Search
- FR-10: Voice Search
- FR-14: Order Tracking
- FR-16: Rate Items
- FR-17: Review Item
- FR-18: View Sales
- FR-19: Delivered Items
- FR-21: Reminder System
- FR-22: User Notifications
- 

##### **3. Low Priority:**

- FR-3: Profile Management
- FR-7: Wish list
- FR-15: Cancel Order
- FR-15: Gift Voucher

#### **➤ Data requirements prioritization**

##### **1. High Priority:**

- DR-1: User Data
- DR-2: Seller/Artisan/Craftsman Data
- DR-3: Product Data
- DR-4: Order Data
- DR-9: Transaction and Wallet Data

##### **2. Medium Priority:**

- DR-5: Cart and Wishlist Data
- DR-6: Search and Recommendation Data
- DR-7: Review and Rating Data
- DR-8: Notification Data
- DR-10: System and Analytics Data

➤ **Performance requirements prioritization**

**1. High Priority:**

- PR-1: Load Handling
- PR-2: Scalability
- PR-3: Response Efficiency
- PR-4: Transaction Processing
- PR-5: Uptime and Reliability
- PR-7: Search Performance
- PR-9: Database Performance
- PR-10: Page Rendering

**2. Medium Priority:**

- PR-6: Data Throughput
- PR-8: Image and Media Loading
- PR-11: Resource Utilization
- PR-12: Content Delivery

➤ **Usability requirements prioritization**

**1. High Priority:**

- UR-1: Intuitive Navigation
- UR-2: Accessibility
- UR-3: Mobile Responsiveness
- UR-5: Search Functionality
- UR-6: Clear Feedback and Error Messages
- UR-7: Simplified Checkout Process
- UR-10: Secure Login and Account Management

**2. Medium Priority:**

- UR-4: Consistent User Interface (UI) Design
- UR-8: Personalization

**3. Low Priority:**

- UR-9: User Support and Help

➤ **Security requirements prioritization:**

**1. High Priority:**

- SR-1: Secure Authentication
- SR-2: Access Control
- SR-3: Secure Payment Processing with SSLCOMMERZ
- SR-5: Secure Development Practices

**2. Medium Priority:**

- SR-4: Regular Security Audits and Penetration Testing

## **14.2 Traceability Matrix**

**Use cases (UC):**

Sure, here are the remaining use cases in UC format:

UC1: Registration

UC2: Login

UC3: Profile management

UC4: Search product

UC5: Personal recommendation

UC6: View product

UC7: Add to cart

UC8: Wish list

UC9: View wish list

UC10: Wallet

UC11: Second hand product

UC12: Art/ Craft

UC13: Bid on art/craft

UC14: Profit Margin

UC15: Order

UC16: Payment

UC17: Delivery

UC18: Track order

UC19: Review

UC20: Notification

UC21: Gift Voucher

**Functional requirements (FR):**

FR1: Signup

FR2: Login

FR3: Profile Management

FR4: Search Items

FR5: View Items

FR6: Add to Cart

FR7: Wish list

FR8: Personalized Product Recommendations

FR9: Auto Complete Search

FR10: Voice Search

FR11: Art and Craft Exhibition

FR12: Second-hand Sale

FR13: Checkout and Payment

FR14: Order Tracking

FR15: Cancel Order

FR16: Rate Items

FR17: Review Item

FR18: View sales

FR19: Delivered Items

FR20: Upload Listings

FR21: Reminder System

FR22: User Notifications

FR23: Gift Vouchers

**Table 23: Traceability Matrix**

FR/UC	UC 1	UC 2	UC 3	UC 4	UC 5	UC 6	UC 7	UC 8	UC 9	UC 10	UC 11	UC 12	UC 13	UC 14	UC 15	UC 16	UC 17	UC 18	UC 19	UC 20	UC 21
<b>FR1</b>	✓																				
<b>FR2</b>		✓																			
<b>FR3</b>		✓	✓																		
<b>FR4</b>				✓	✓																
<b>FR5</b>						✓	✓														
<b>FR6</b>							✓										✓				
<b>FR7</b>								✓	✓												
<b>FR8</b>					✓	✓															
<b>FR9</b>				✓																	
<b>FR10</b>				✓						✓											
<b>FR11</b>											✓	✓	✓								
<b>FR12</b>											✓										
<b>FR13</b>					✓									✓	✓						
<b>FR14</b>														✓	✓	✓					
<b>FR15</b>														✓	✓						
<b>FR16</b>																		✓	✓		
<b>FR17</b>																		✓	✓		
<b>FR18</b>																✓	✓				
<b>FR19</b>															✓	✓	✓				
<b>FR20</b>															✓	✓				✓	
<b>FR21</b>																		✓	✓		
<b>FR22</b>																		✓	✓		
<b>FR23</b>																					✓