



**Daffodil**  
*International*  
**University**

## Ecommerce with Inventory Management & Accounting

**Submitted By**

**Erfan Ahmed**

183-35-2610

Dept. of Software Engineering

**Supervised By**

**Khalid Been Budruzzaman Biplob**

Lecturer (Senior Scale)

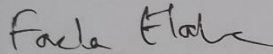
Faculty of Science and Information Technology

Department of Software Engineering

## APPROVAL

This thesis titled on **Ecommerce with Inventory Management & Accounting**, submitted by **Erfan Ahmed (ID: 183-35-2610)** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering and approval as to its style and contents.

## BOARD OF EXAMINERS



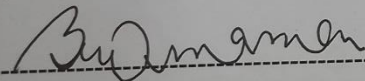
**Chairman**

-----  
**Dr Md. Fazla Elahe**  
**Assistant Professor & Associate Head**  
Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University



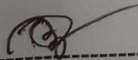
**Internal Examiner 1**

-----  
**A.H.M Shahariar Parvez**  
**Associate Professor**  
Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University



**Internal Examiner 2**

-----  
**Khalid Been Budruzzaman Biplob**  
**Lecturer (Senor Scale)**  
Department of Software Engineering  
Faculty of Science and Information Technology  
Daffodil International University



**External Examiner**

-----  
**Md Tanvir Quader**  
Senior Software Engineer  
Solutions Architect, a2i

# ACKNOWLEDGEMENT

First of all, I feel fortunate that I made it to the last semester. Alhamdulillah, praise be to Allah for the innumerable blessings. I have gained a great deal of knowledge about software development and computer science through my university's skilled lecturers since I started my studies. Teachers provide morals, etiquette, and knowledge of software in addition to other relevant subjects. I must always be true to myself and be grateful to my parents for giving me this chance. I owe a debt of gratitude to Khalid Been Md. Badruzzaman Biplob for all of his help and assistance in seeing the project through to its completion, including his direction and ongoing oversight. My supervisor helps me to see this project (the "Learning Tree") through to completion. I would like to express my gratitude to my fellow students for their assistance with creating the project and to those who volunteered their skills to support me.

<b>CHAPTER 1</b> .....	4	Add/Delete Invoice .....	18
Admin .....	6	Add/Delete Journal .....	21
Background .....	5	Add/Delete Order .....	17
Benefits & Beneficiaries .....	5	Add/Delete Payment.....	22
Focused Modules .....	7	Add/Delete Purchase .....	18
Modules of Ecommerce with Inventory Man agement & Accounting.....	7	Add/Delete Receipt.....	23
Objectives .....	7	Add/Delete Return.....	19
Project Overview.....	4	Add/Delete Warranty.....	19
Project Purpose.....	5	Add/Delete/Edit Item.....	17
Project Schedule .....	7	Block Diagram .....	58
Proposed System Model .....	6	Checkout .....	24
Stakeholders .....	6	Claim Return .....	25
Users .....	6	Claim Warranty .....	25
<b>CHAPTER 2</b> .....	8	Class Diagram.....	60
Accounts Manage .....	9, 10	Development Model .....	13
Add to Cart.....	10	ER Diagram.....	59
Availability.....	12	Generate Report .....	23
Checkout .....	11	Login .....	16
Claim Return .....	11	Register .....	16
Claim Warranty .....	11	Sequence Diagram .....	42
Functional Requirements (FR).....	8, 11	Use Case Diagram .....	14
Generate Reports.....	10	<b>CHAPTER 4</b> .....	62
Invoice Manage.....	9	Feature Testing.....	62
Journal Manage .....	10	Testing Strategies .....	62
Maintainability.....	12	<b>CHAPTER 5</b> .....	72
Order Manage.....	8	After sales Area .....	77
Payment Manage .....	10	Clients Area .....	75
Performance .....	11, 12	Dashboard.....	74
Product Manage.....	8	Journal Area .....	78
Purchase Manage.....	9	Journal wise Report .....	79
Receipt Manage .....	10	Order Place .....	72
Registration.....	8	Orders Area .....	75
Reliability .....	12	Product Area .....	76
Return Manage .....	9	Profit & Loss Report .....	78
Security .....	12	Purchase Area .....	76
Warranty Manage .....	9	Register & Login.....	73
<b>CHAPTER 3</b> .....	13	Sale Area .....	77
Activity Diagram.....	27	<b>CHAPTER 6</b> .....	80
Add to Cart.....	24	Limitations .....	80
Add/Delete Account.....	20	Obstacle and Achievements.....	80
		Project Summary.....	80

# CHAPTER 1

## Introduction

### 1.1 Project Overview

Nowadays, every owner of a physical business creates an online website to offer their goods to consumers. Bangladesh is home to a large number of e-commerce websites; however, each owner is liable for maintaining their own website along with their accounting and inventory management system. Owners must engage a worker to synchronize the data on their website with their inventory and accounting software, which is a major inconvenience. Updating the pricing and number of products on a website and in inventory software on a daily basis is a major annoy. Additionally, an electronic store must uphold warranties, RMAs, and daily consumer input. It is a laborious and extremely slow procedure. If the warranty system has an online portal option that synchronizes automatically with their inventory and accounting software.

In order to tackle this, I made an e-commerce platform that combines inventory management and accounting. This allows website owners to utilize inventory & accounting management software to handle many aspects of their items, such as price, quantity, description, profit, loss, and warranty, all at once. Additionally, they are not need to buy inventory websites and software separately. It will be a full-featured ERP program including accounting, e-commerce, and inventory management.

## **1.2 Project Purpose**

"Ecommerce with Inventory Management & Accounting" aims to provide consumers with an intuitive and user-friendly platform by digitizing the outdated system through the use of full-featured computer software and information technology. Every user has access to the environment needed to use this system. As previously said, "Ecommerce with Inventory Management & Accounting" is user-friendly, safe, dependable, and error-free. It helps consumers to make better use of graphs and charts in reporting.

### **Functionalities provided by “Ecommerce with Inventory Management & Accounting”**

- An ERP and Ecommerce platform
- Fresh User Interface with better experience
- Easy to manage accounting, sales, inventory and after sales support

#### **1.2.1 Background**

My goal is to develop a user-friendly environment that will assist users to the extent that they are satisfied, based on Business Platform systems. Here, the consumer can purchase the goods that the merchant chose to make available to them. It is easy to use and all in one.

#### **1.2.2 Benefits & Beneficiaries**

This system would be helpful for one platform Enterprise Resource Planning:

- This software benefits a company to manager their website, sales, purchase, inventory and after sales support from one software.
- This system keeps records transaction.
- Simple access form for authorized users from anywhere.
- It will save expenses, maintenance cost and time.

## 1.3 Stakeholders

### 1.3.1 Admin

The whole system is managed by the admin. A new account with account information can be added by the admin. The account could be deleted by the admin. In this arrangement, he also authorizes the user's diary. All account information can also be updated by the admin. He is able to critically evaluate the system's development.

### 1.3.2 Users

Users who have registered are able to use the system and transact in goods. Users have the choice to tweak their data. Both password resets and admin collaboration are available to users.

## 1.4 Proposed System Model

In order to achieve the planned framework and behavior of my system, this model has been assembled by myself. I made this model so I could investigate and control the system's architecture.



Fig 1.4: System Model

#### Admin

- Add/Delete/Edit Item
- Add/Delete Order
- Add/Delete Purchase
- Add/Delete Invoice
- Add/Delete Warranty
- Add/Delete Return

#### Accounts

- Add/Delete Accounts
- Add/Delete Journal
- Add/Delete Payment
- Add/Delete Receipt
- Generate Reports

#### Customer

- Add to Cart
- Checkout
- Claim Warranty
- Claim Return

## **1.5 Modules of Ecommerce with Inventory Management & Accounting**

- Admin Module: Utilized for overall system and website management.
- Accounts Module: Transaction for managing accounting and reports
- Customer Module: Used for customer product purchase and order management.

### **1.5.1 Focused Modules**

- Admin: - The interested real time first created a profile with their password and email address. He is able to supervise every aspect of the system.
- Accounts: - The interested real time first established their password and email address. He can then govern the financial statements and reports.
- Customer: - The interested human being first enrolled by entering their password and email address. He is then able to make an internet-based order.

## **1.6 Project Schedule**

I must set an agenda in order to finish this task on time, recognizing the limited time. It also refers to interacting all the required duties in a timely way.

## **1.7 Ambition**

The main aim of the project is to create an atmosphere that is accessible before the system can provide an internet platform for businesses. The main function of the system is to serve as a platform for remote business. From the comfort of their own residences, users may find items that are available and buy or sell them.

# CHAPTER 2

## Engineering Prerequisites

### 2.1 Essential Functions (FR):

The operational demands associated with a system—also termed a behavior specification—define its function. The functional must have can encompass calculation, data processing, business activities, user interaction, or any other specialized consist of that stands out the stated objective of a system. I'm currently displaying the functional requirements related to this project:

#### 2.1.1 Enrolling in System

<b>FR 1</b>	Signing upward
The Overview	Each actor needs to register for the purpose of to invoke the interface.
The attendees	Admin, Accountant, Customer

#### 2.1.2 Sign-in and Sign-out

<b>FR 2</b>	Sign-in and Sign-out
The Overview	Login and Logout should be another condition for using this application
The attendees	Admin, Accountant, Customer

#### 2.1.3 Product Manage

<b>FR 3</b>	Product Manage
The Overview	Admin can add/edit/delete product
The attendees	Admin

#### 2.1.4 Order Manage

<b>FR 4</b>	Order Manage
-------------	--------------

The Overview	Admin can add/delete order
The attendees	Admin

### 2.1.5 Purchase Manage

<b>FR 5</b>	Purchase Manage
The Overview	Admin can add/delete purchase
The attendees	Admin

### 2.1.6 Invoice Manage

<b>FR 6</b>	Invoice Manage
The Overview	Admin can add/delete invoice
The attendees	Admin

### 2.1.7 Warranty Manage

<b>FR 7</b>	Warranty Manage
The Overview	Admin can add/delete warranty
The attendees	Admin

### 2.1.8 Return Manage

<b>FR 8</b>	Return Manage
The Overview	Admin can add/delete return
The attendees	Admin

### 2.1.9 Accounts Manage

<b>FR 9</b>	Accounts Manage
The Overview	Accountant can add/delete order
The attendees	Accountant

### 2.1.10 Accounts Manage

<b>FR 10</b>	Accounts Manage
The Overview	Accountant can add/delete accounts
The attendees	Accountant

### 2.1.11 Journal Manage

<b>FR 11</b>	Journal Manage
The Overview	Accountant can add/delete journal
The attendees	Accountant

### 2.1.12 Payment Manage

<b>FR 12</b>	Payment Manage
The Overview	Accountant can add/delete payment
The attendees	Accountant

### 2.1.13 Receipt Manage

<b>FR 13</b>	Receipt Manage
The Overview	Accountant can add/delete receipt
The attendees	Accountant

### 2.1.14 Distribute Reports

<b>FR 14</b>	Distribute Reports
The Overview	Accountant can generate reports
The attendees	Accountant

### 2.1.15 Add to Cart

<b>FR 15</b>	Add to cart
The Overview	Customer can add to cart item
The attendees	Customer

### 2.1.16 Checkout

<b>FR 16</b>	Checkout
The Overview	Customer can checkout item
The attendees	Customer

### 2.1.17 Claim Warranty

<b>FR 17</b>	Claim Warranty
The Overview	Customer can claim warranty
The attendees	Customer

### 2.1.18 Claim Return

<b>FR 15</b>	Claim Return
The Overview	Customer can claim return
The attendees	Customer

## 2.2 Functional Requirements (FR):

### 2.2.1 Performance:

<b>NonFR 1</b>	The system will function flawlessly to deliver all services.
The Overview	When the supervisor hunts for a specific task, the result needs to the floor more quickly.

The attendees	Controller as Administrator
---------------	-----------------------------

### 2.2.2 Performance:

<b>NonFR 2</b>	Depending on storage, the system can record an infinite amount of data.
The Overview	The data shall be retained within the database for five years.
The attendees	Controller

### 2.2.3 Trustworthiness:

<b>NonFR 3</b>	The system must operate precisely in order to perform its many tasks.
The Overview	The necessary tasks should be completed by the system quickly; however, accomplishment of some crucial acts must be verified. Regular modifications to the system are essential.
The attendees	Controller as Administrator

### 2.2.4 Security:

<b>NonFR 4</b>	The database has to be protected, and certain cryptographic techniques should be used.
The Overview	Data must be shielded from outside threats. Encryption is an excellent way to protect data. Every request should have been independently confirmed.
The attendees	Controller as Administrator

### 2.2.5 Maintainability:

<b>NonFR 5</b>	All choices and saved profiles can be returned to their default values by the administrator.
The Overview	The administrator can simply update all of their data and maintain every profile in a certain location.
The attendees	Controller as Administrator

### 2.2.6 Availability:

<b>NonFR 6</b>	The system needs to be accessible. 24 x 7
The Overview	The administrator should have round-the-clock access to the program.

The attendees	Controller as Administrator
---------------	-----------------------------

# CHAPTER 3

## Design Specification plus System Evaluation

### 3.1 Model of Evolution:

Since I am a long-term developer and a novice in development, I decided to finish my project using the Agile Model. The agile method is more flexible for manufacturing than any other traditional paradigm since it anticipates change. Little modifications to my system won't require me to drastically alter my strict timetable. I found that testing at every stage was crucial to making sure my system was operating flawlessly. The agile paradigm allows me to test each unit as the system is being developed. The agile framework empowers me to design a system that is readily modified, reduces risk through frequent testing, produces a higher-quality output, and can be completed abruptly.

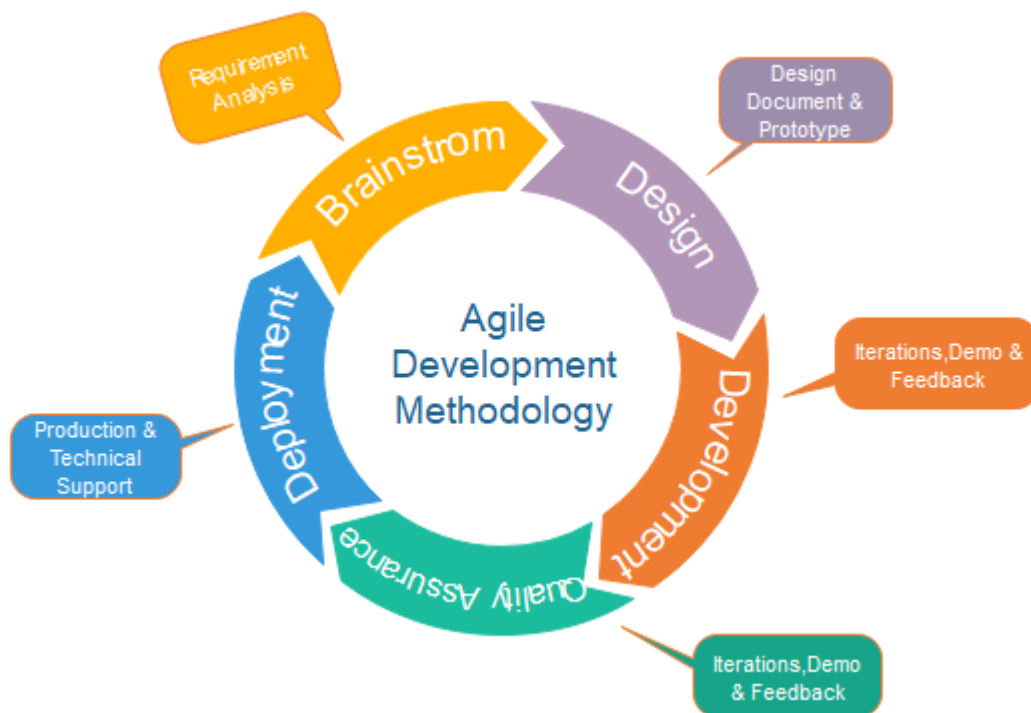


Fig 3.1: Agile Model

### **3.2 Occupancy Diagram:**



Fig 3.2: Use Case Diagram

**Description about Use Case**

### 3.2.1 Register

Anyone may log in to take full advantage of every advantage offered by this system. If they are not familiar with the system, they have to authenticate and log in before they may build a user profile.

<b>ID for instances of use</b>	UC-01	
<b>Employ Case The name</b>	Register	
<b>The Intention</b>	To add a new user profile to the MySQL database	
<b>Priorities are what</b>	A user's email address is a must.	
<b>Lead Actor Secondary Contributor</b>	Admin Accountant	
<b>Configure off the</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	To establish a fresh user profile
<b>During Condition</b>	Users are able to sign in.	
<b>A substitute flow</b>	Initial Sign in.	

### 3.2.2 Login

To use all of the system's functions, users are required to sign in with their username and password. The user must first register if they are yet to do so recently.

<b>ID for instances of use</b>	UC-02	
<b>Employ Case The name</b>	Create	
<b>The Intention</b>	Enter your system credentials to log in.	
<b>Priorities are what</b>	User must be signed up with a password and email location.	
<b>Lead Actor Secondary Contributor</b>	Admin Accountant, Customer	
<b>Configure off the</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To gain entry to the system
	2	To use each option that is exposed to a user
<b>During Condition</b>	Users can Logout anytime.	
<b>A substitute flow</b>	Signup	

### 3.2.3 Add/Delete/Edit Item

Users can manage an item form dashboard.

<b>ID for instances of use</b>	UC-03	
<b>Employ Case The name</b>	Add/Delete/Edit Item	
<b>The Intention</b>	Manage item/products	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Edit/Delete an item
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.4 Add/Delete Order

Users can manage an order form dashboard.

<b>ID for instances of use</b>	UC-04	
<b>Employ Case The name</b>	Add/Delete Order	
<b>The Intention</b>	Manage Order	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Order
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.5 Add/Delete Purchase

Users can manage a purchase form dashboard.

<b>ID for instances of use</b>	UC-05	
<b>Employ Case The name</b>	Add/Delete Purchase	
<b>The Intention</b>	Manage Purchase	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Purchase
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.6 Add/Delete Invoice

Users can manage an Invoice form dashboard.

<b>ID for instances of use</b>	UC-06	
<b>Employ Case The name</b>	Add/Delete Invoice	
<b>The Intention</b>	Manage Invoice	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin	

<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	To enter into system
	<b>2</b>	Create/Delete Invoice
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.7 Add/Delete Warranty

Users can manage a Warranty form dashboard.

<b>ID for instances of use</b>	UC-07	
<b>Employ Case The name</b>	Add/Delete Warranty	
<b>The Intention</b>	Manage Warranty	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	To enter into system
	<b>2</b>	Create/Delete Warranty
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.8 Add/Delete Return

Users can manage a Return form dashboard.

<b>ID for instances of use</b>	UC-08
--------------------------------	-------

<b>Employ Case The name</b>	Add/Delete Return	
<b>The Intention</b>	Manage Return	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Return
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.9 Add/Delete Account

Users can manage an Account form dashboard.

<b>ID for instances of use</b>	UC-09	
<b>Employ Case The name</b>	Add/Delete Account	
<b>The Intention</b>	Manage Account	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin Accountant	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Account
<b>During Condition</b>	Users must have to login.	

<b>A substitute flow</b>	Signup
--------------------------	--------

### 3.2.10 Add/Delete Journal

Users can manage a Journal form dashboard.

<b>ID for instances of use</b>	UC-10	
<b>Employ Case The name</b>	Add/Delete Journal	
<b>The Intention</b>	Manage Journal	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin Accountant	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Journal
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.11 Add/Delete Journal

Users can manage a Journal form dashboard.

<b>ID for instances of use</b>	UC-11	
<b>Employ Case The name</b>	Add/Delete Journal	
<b>The Intention</b>	Manage Journal	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	

<b>Lead Actor Secondary Contributor</b>	Admin Accountant	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Journal
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.12 Add/Delete Payment

Users can manage a Payment form dashboard.

<b>ID for instances of use</b>	UC-12	
<b>Employ Case The name</b>	Add/Delete Payment	
<b>The Intention</b>	Manage Payment	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin Accountant	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Payment
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.13 Add/Delete Receipt

Users can manage a Receipt form dashboard.

<b>ID for instances of use</b>	UC-13	
<b>Employ Case The name</b>	Add/Delete Receipt	
<b>The Intention</b>	Manage Receipt	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin Accountant	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Receipt
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.14 Generate Report

Users can manage a Report form dashboard.

<b>ID for instances of use</b>	UC-14	
<b>Employ Case The name</b>	Generate Report	
<b>The Intention</b>	Manage Report	
<b>Priorities are what</b>	User must be signed by entering a password and a valid email address.	
<b>Lead Actor Secondary Contributor</b>	Admin Accountant	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Create/Delete Report
<b>During Condition</b>	Users must have to login.	
<b>A substitute flow</b>	Signup	

### 3.2.15 Add to Cart

Users can add to cart products.

<b>ID for instances of use</b>	UC-15	
<b>Employ Case The name</b>	Add to Cart	
<b>The Intention</b>	Products add to a cart	
<b>Priorities are what</b>	User has to be an end-customer.	
<b>Lead Actor Secondary Contributor</b>	Customer	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	1	To enter into system
	2	Products add to cart
<b>During Condition</b>	Users don't have to login.	
<b>A substitute flow</b>	Home	

### 3.2.16 Checkout

Users can checkout products with address, name, phone and email.

<b>ID for instances of use</b>	UC-16	
<b>Employ Case The name</b>	Checkout	
<b>The Intention</b>	Products checkout	
<b>Priorities are what</b>	User has to be an end-customer.	
<b>Lead Actor Secondary Contributor</b>	Customer	
<b>Configure off The</b>	Button	

<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	To enter into system
	<b>2</b>	Products checkout
<b>During Condition</b>	Users don't have to login.	
<b>A substitute flow</b>	Home	

### 3.2.17 Claim Warranty

Users can claim warranty of a product.

<b>ID for instances of use</b>	UC-17	
<b>Employ Case The name</b>	Claim Warranty	
<b>The Intention</b>	Products claim warranty	
<b>Priorities are what</b>	User has to be an end-customer.	
<b>Lead Actor Secondary Contributor</b>	Customer	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	To enter into system
	<b>2</b>	Enter order number and claim warranty
<b>During Condition</b>	Users must have to purchase.	
<b>A substitute flow</b>	Home	

### 3.2.18 Claim Return

Users can claim return of a product.

<b>ID for instances of use</b>	UC-18	
<b>Employ Case The name</b>	Claim Return	

<b>The Intention</b>	Products claim Return	
<b>Priorities are what</b>	User has to be an end-customer.	
<b>Lead Actor Secondary Contributor</b>	Customer	
<b>Configure off The</b>	Button	
<b>Breakdown of Key Success Factors</b>	<b>Step</b>	<b>Action</b>
	<b>1</b>	To enter into system
	<b>2</b>	Enter order number and claim Return
<b>During Condition</b>	Users must have to purchase.	
<b>A substitute flow</b>	Home	

### 3.3 Activity Diagram

#### 3.3.1 Register

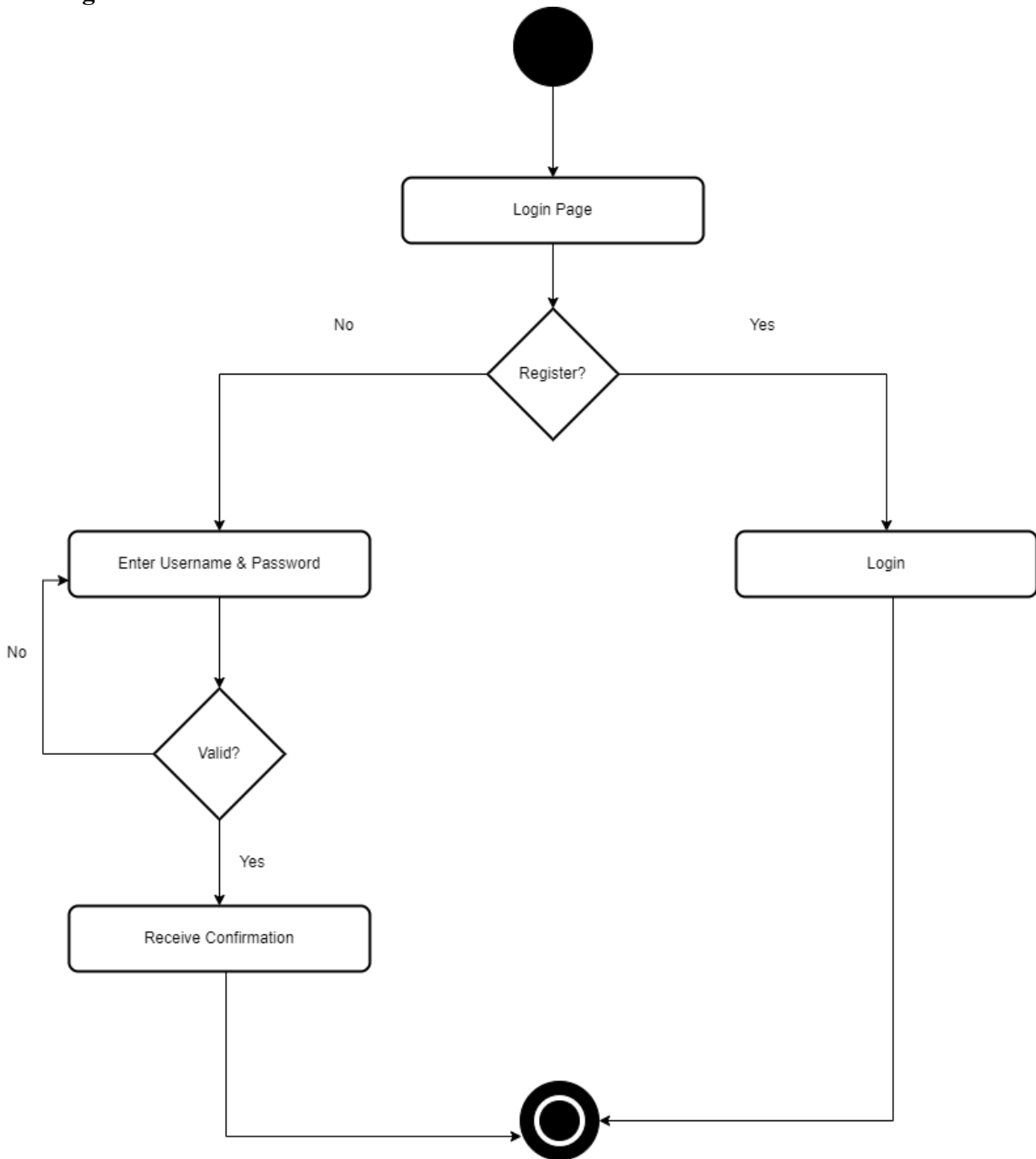
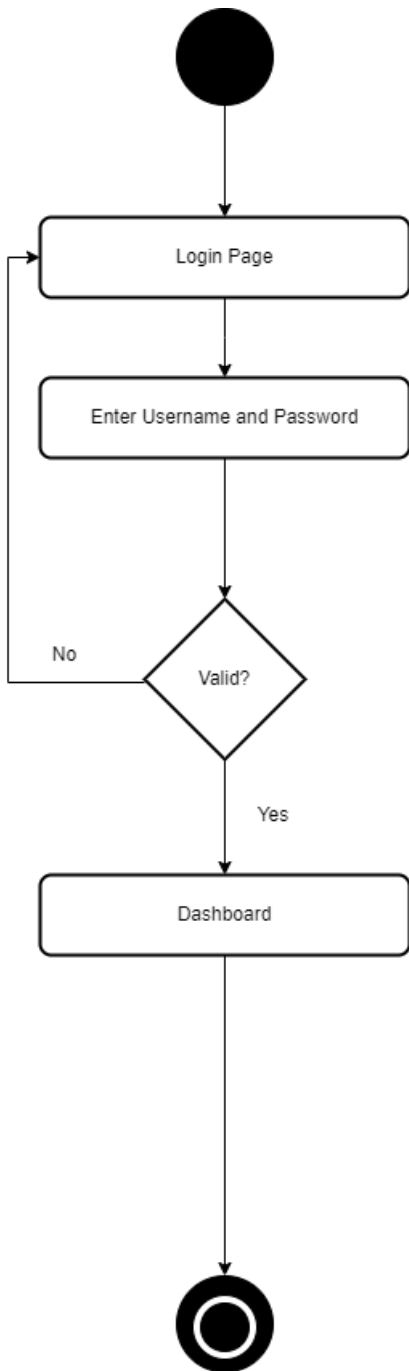


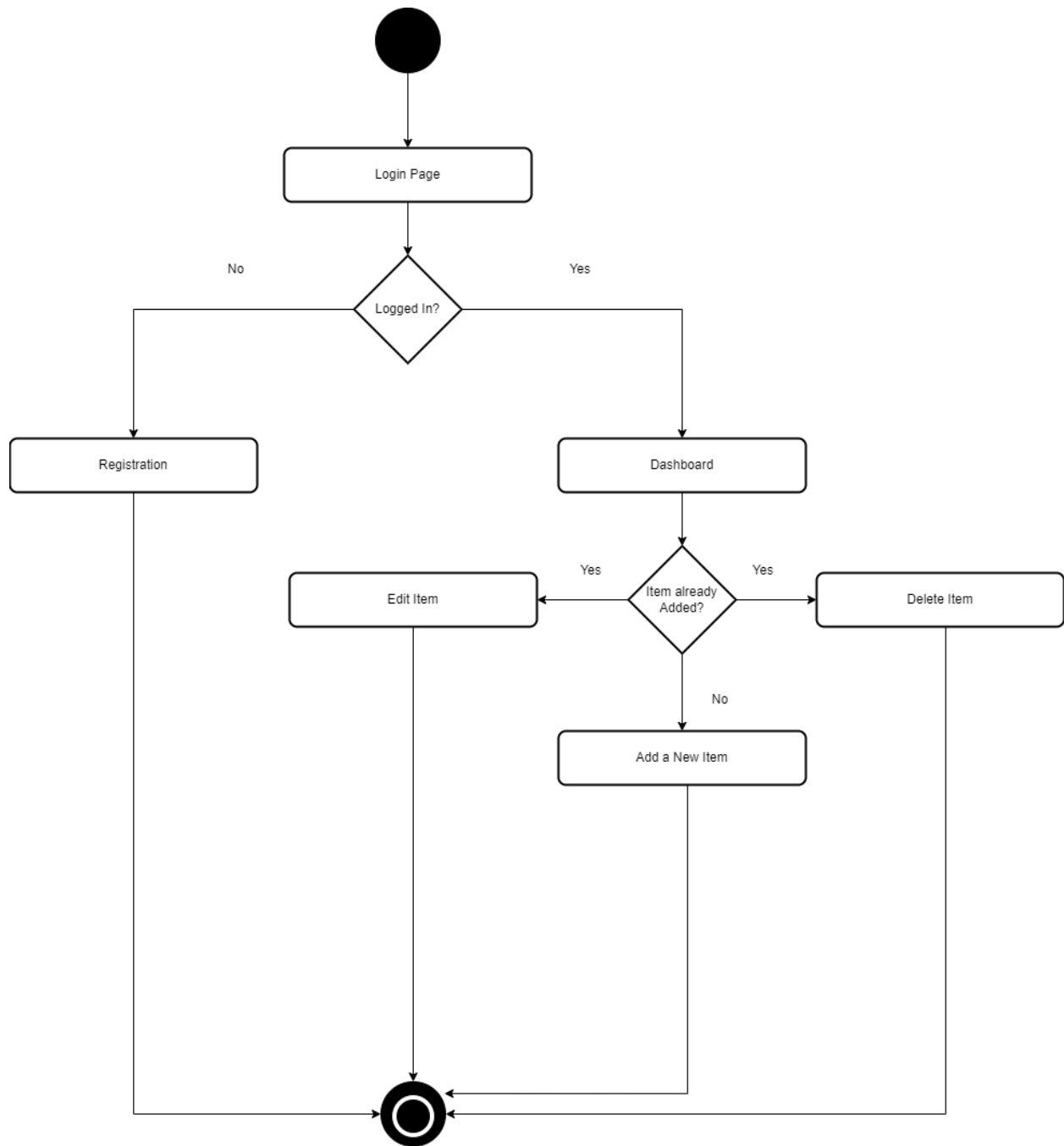
Figure 3.3.1: Register

#### 3.3.2 Login



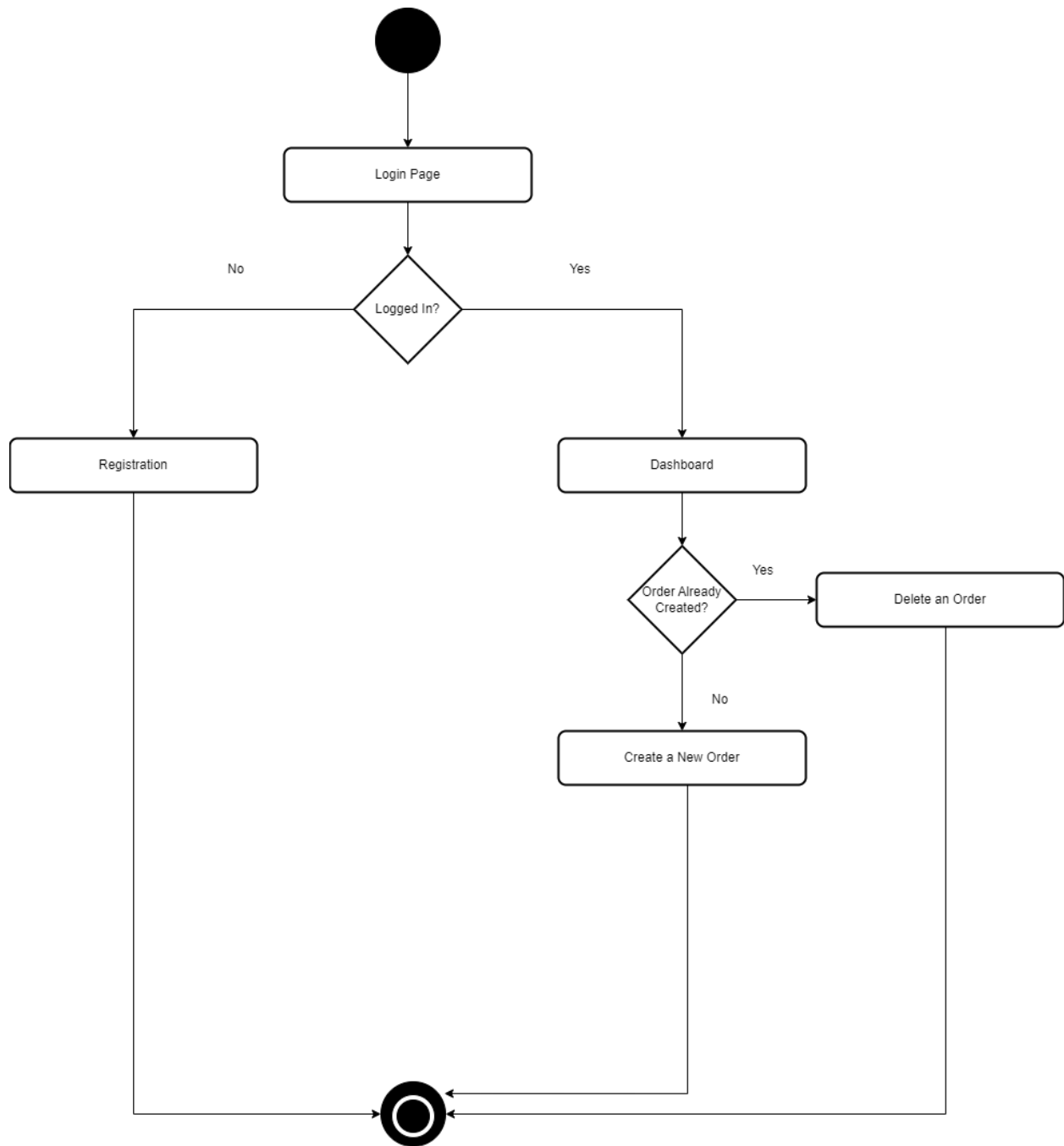
**Figure 3.3.2: Login**

### **3.3.3 Add/Delete/Edit Item**



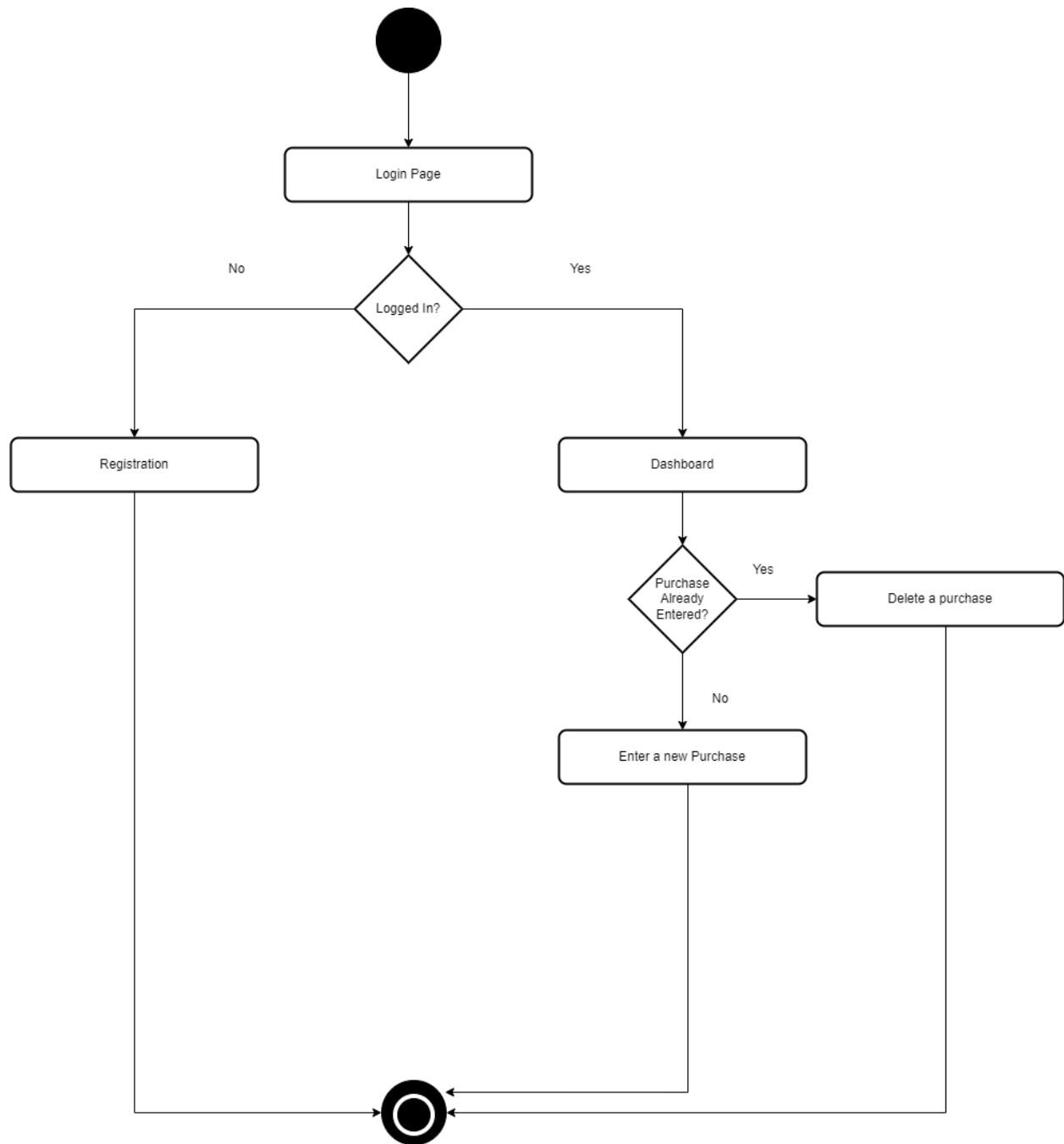
**Figure 3.3.3: Add/Delete/Edit Item**

### 3.3.4 Add/Delete Order



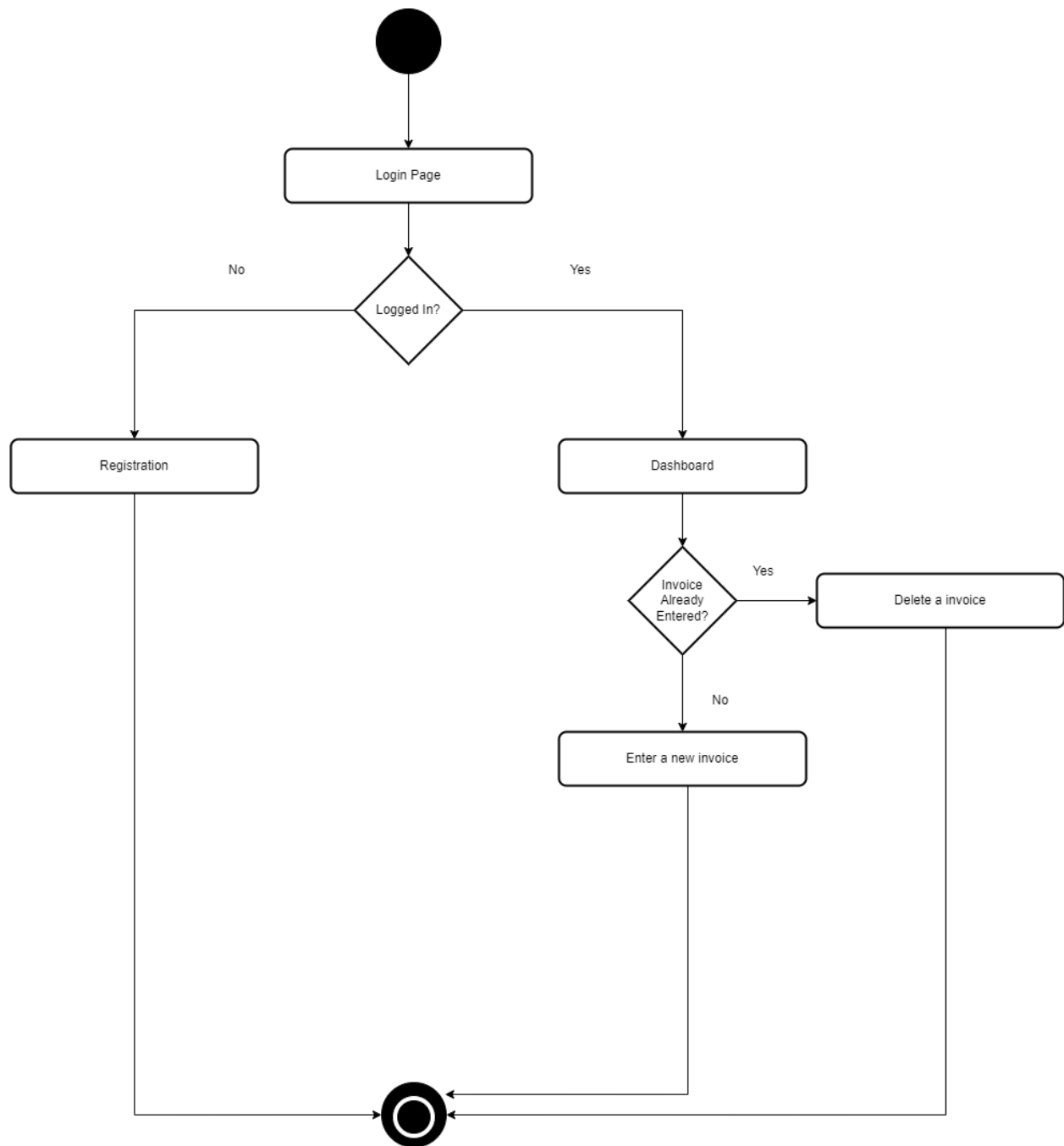
**Figure 3.3.4: Add/Delete Order**

### **3.3.5 Add/Delete Purchase**



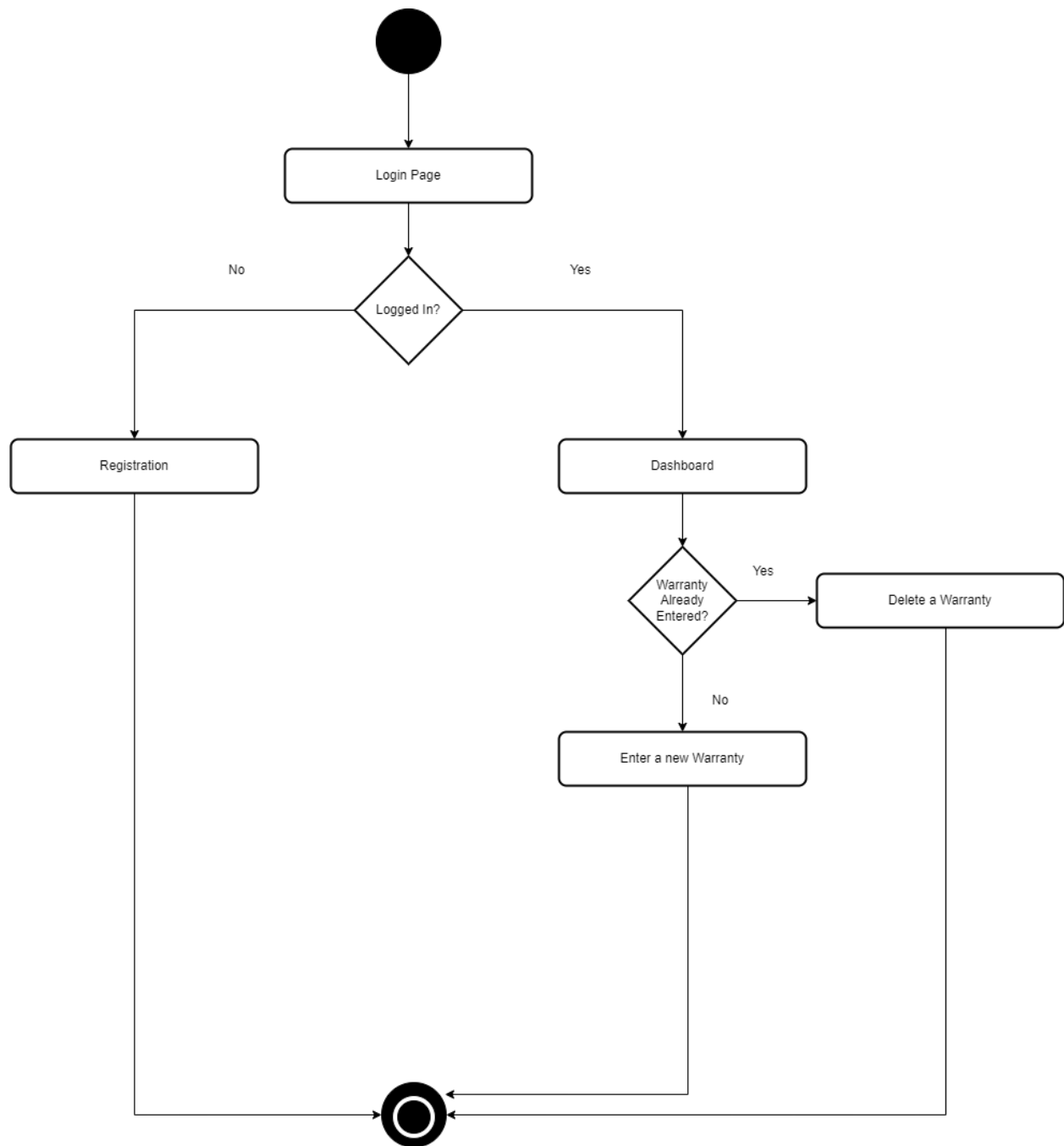
**Figure 3.3.5: Add/Delete Purchase**

### 3.3.6 Add/Delete Invoice



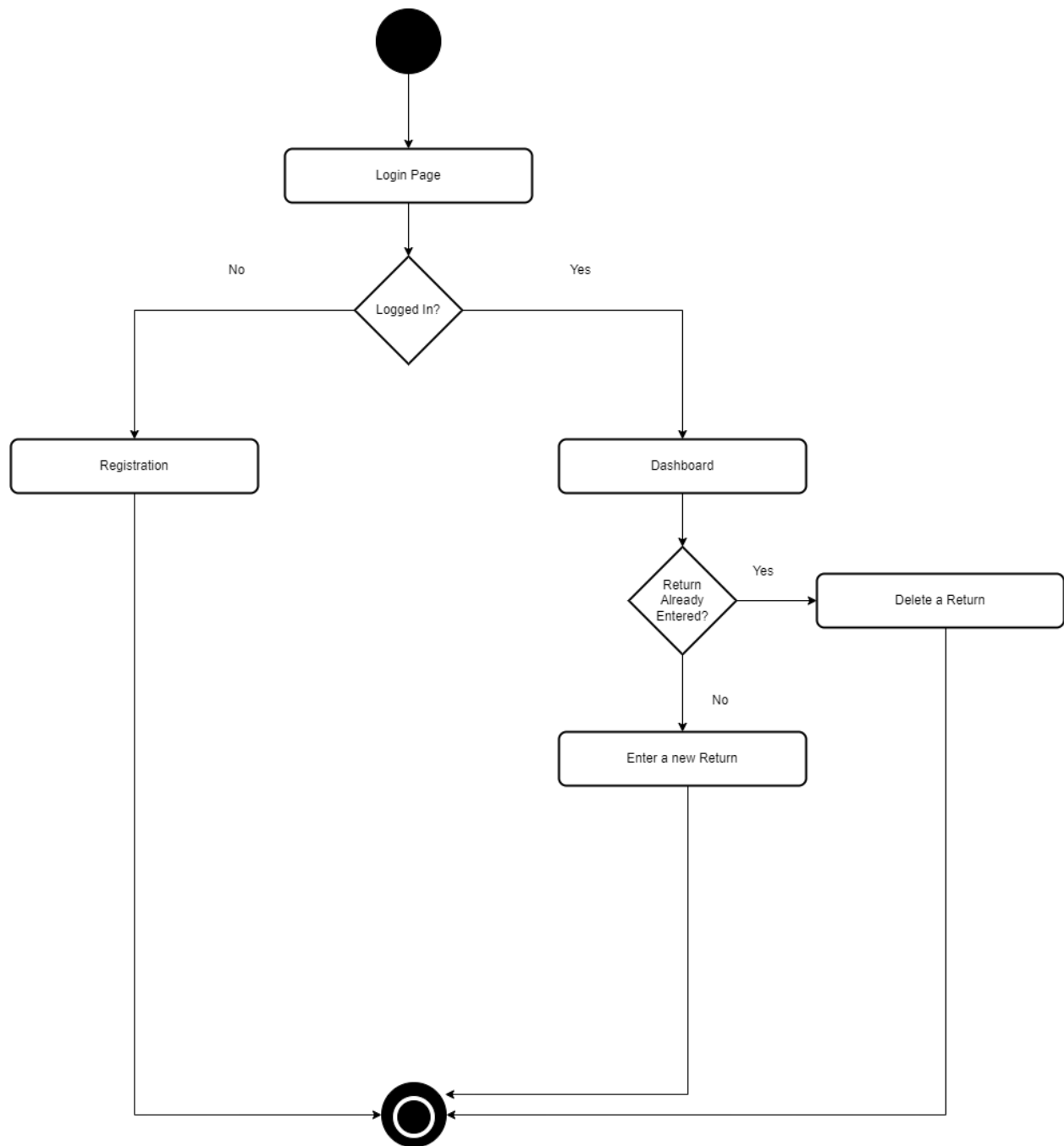
**Figure 3.3.6: Add/Delete Purchase**

### 3.3.7 Add/Delete Warranty



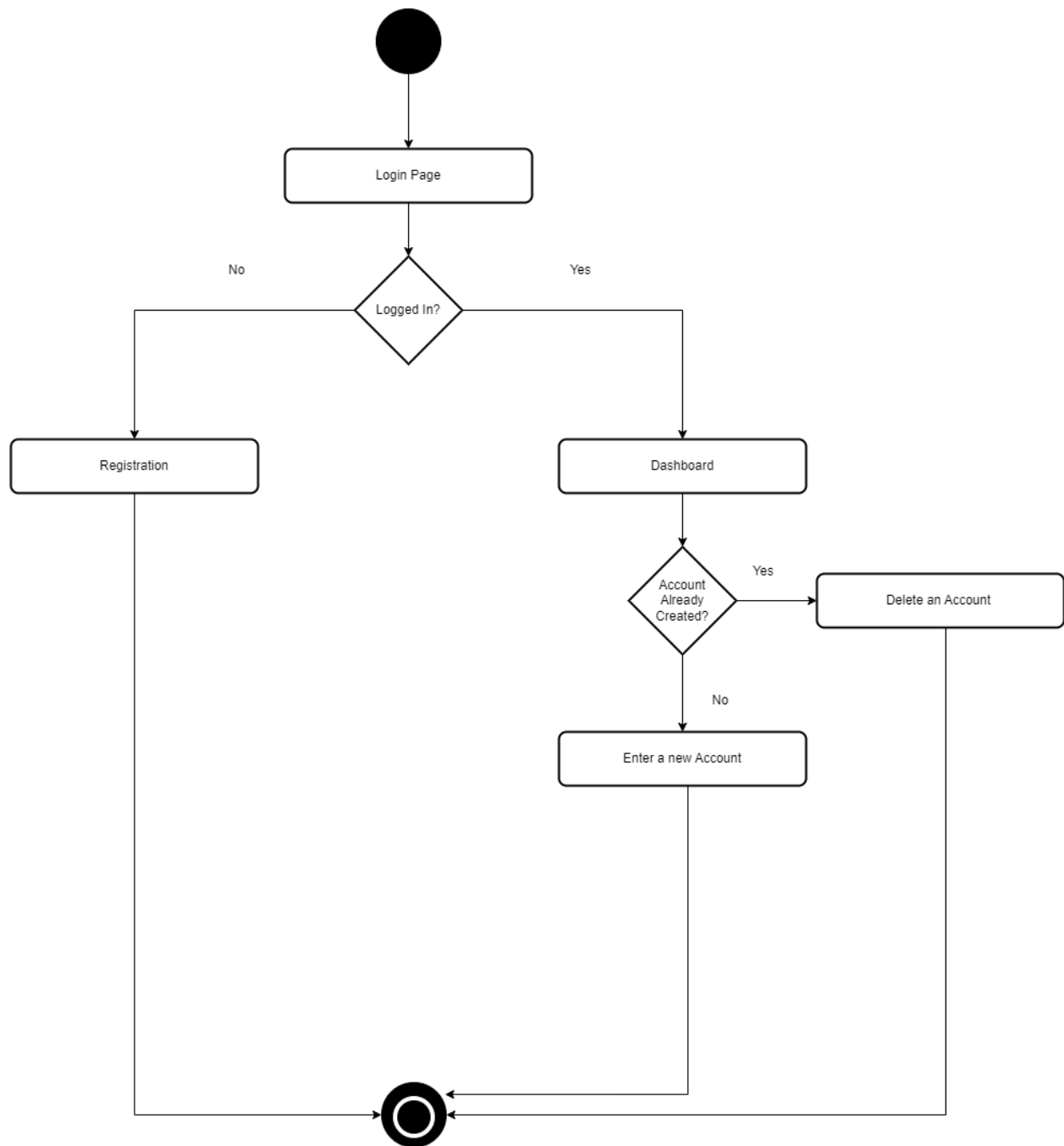
**Figure 3.3.7: Add/Delete Warranty**

### 3.3.8 Add/Delete Return



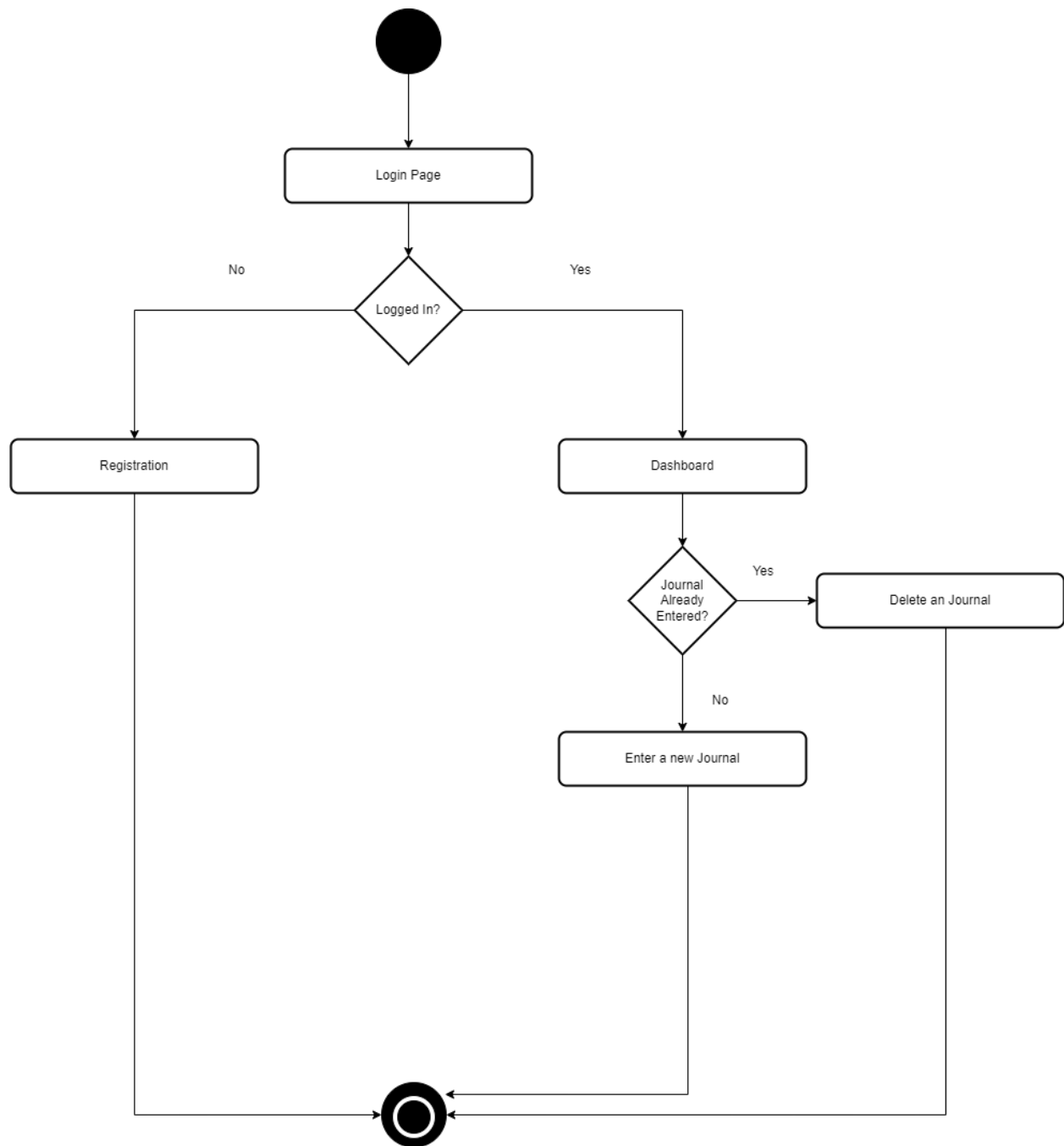
**Figure 3.3.8: Add/Delete Return**

### 3.3.9 Add/Delete Account



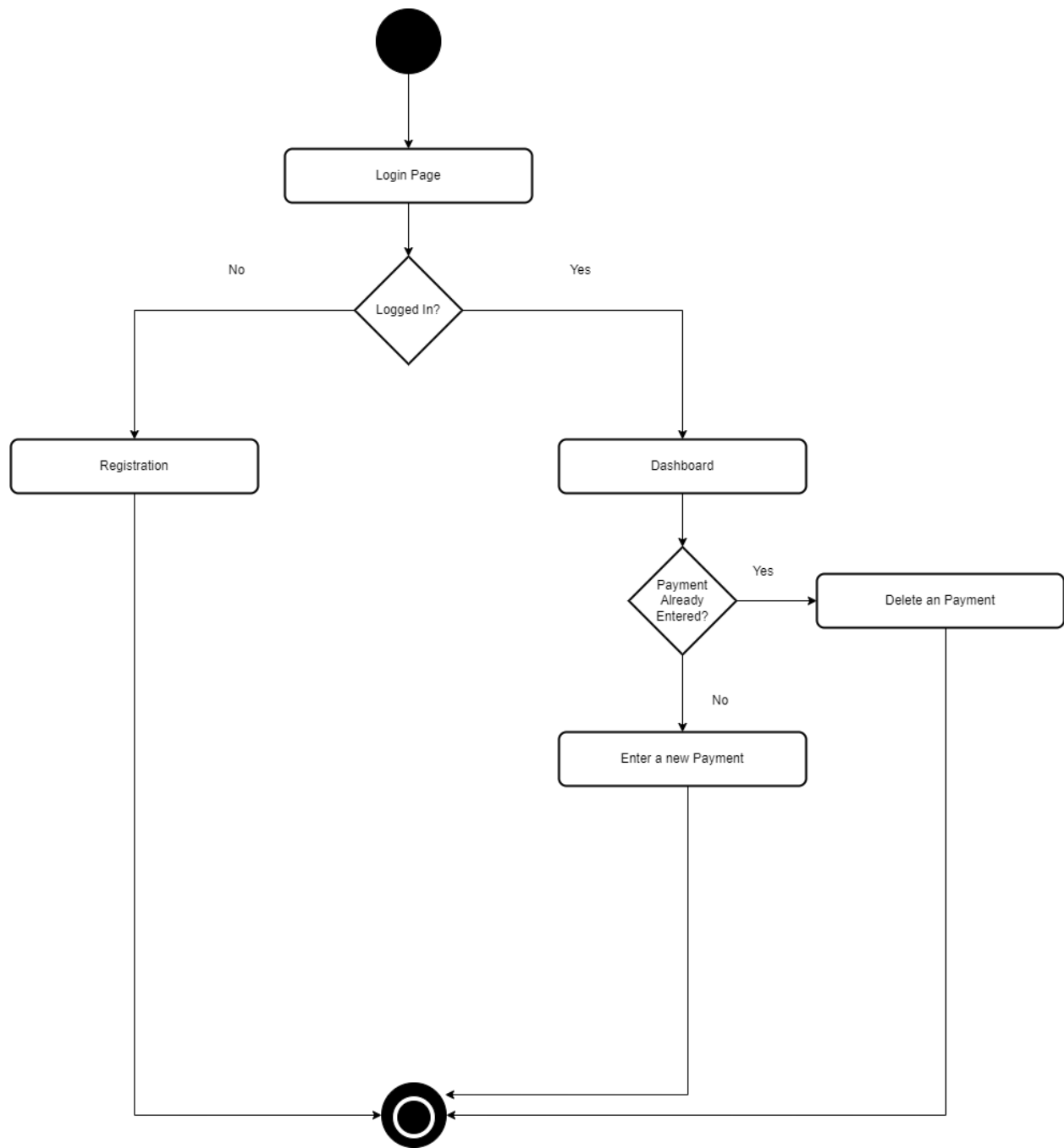
**Figure 3.3.9: Add/Delete Account**

### 3.3.10 Add/Delete Journal



**Figure 3.3.10: Add/Delete Journal**

### 3.3.11 Add/Delete Payment



**Figure 3.3.11: Add/Delete Payment**

### 3.3.12 Add/Delete Receipt

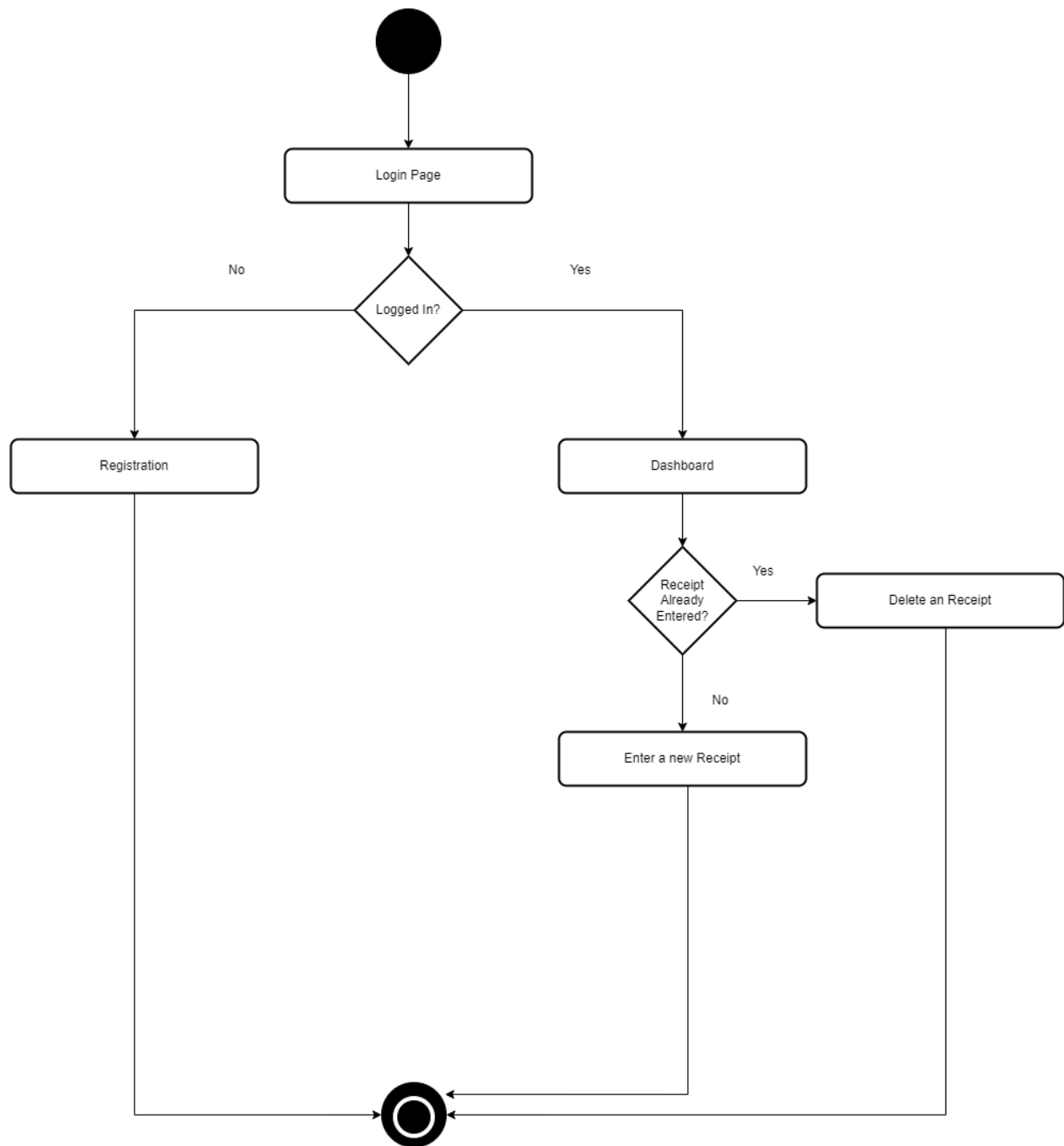
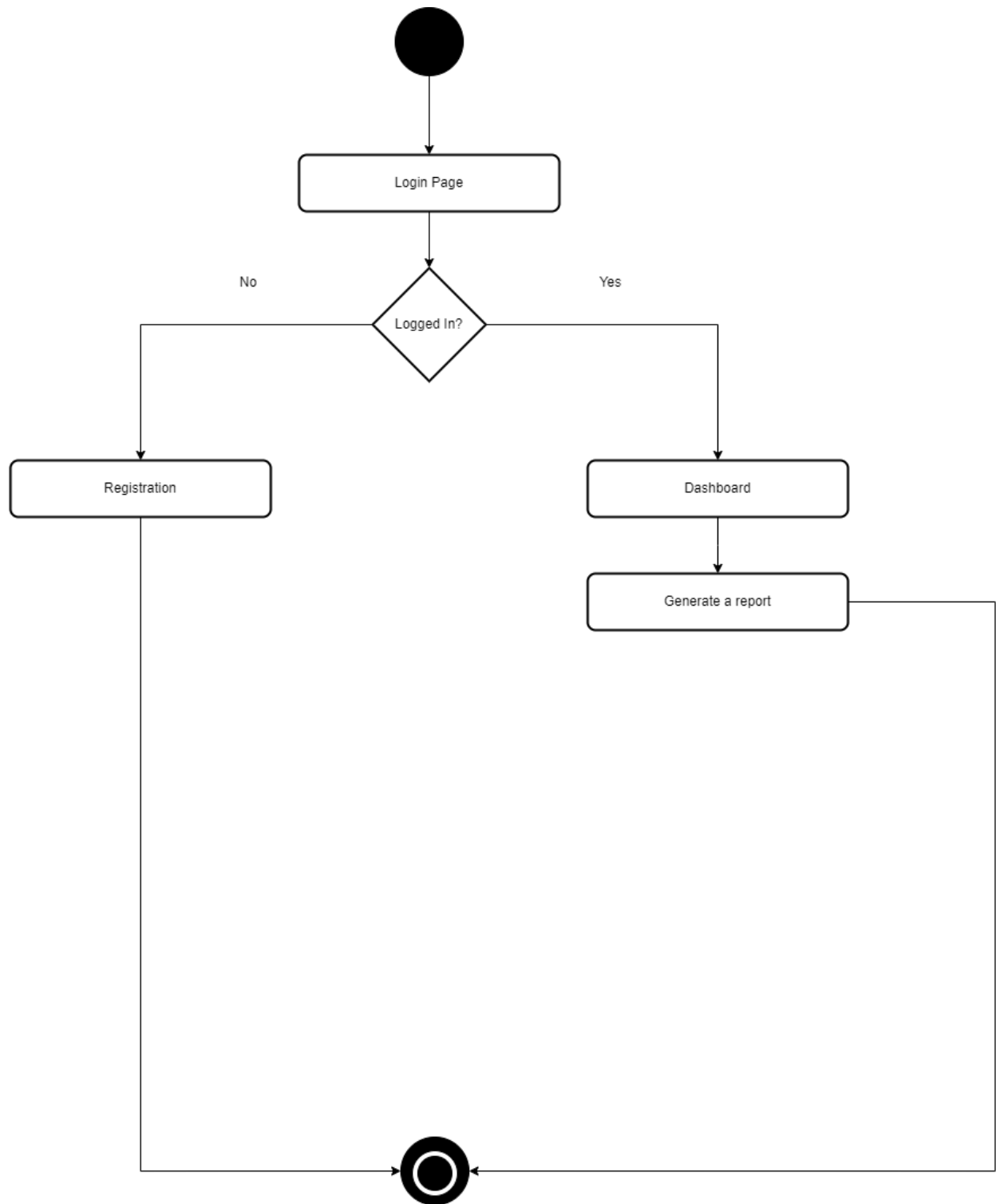


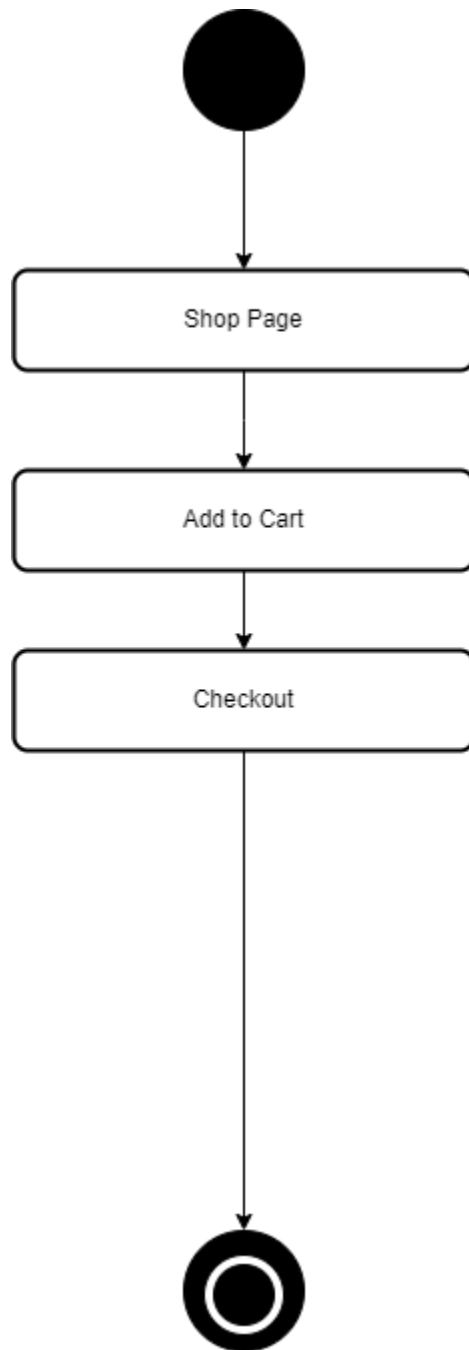
Figure 3.3.12: Add/Delete Receipt

### 3.3.13 Generate Reports



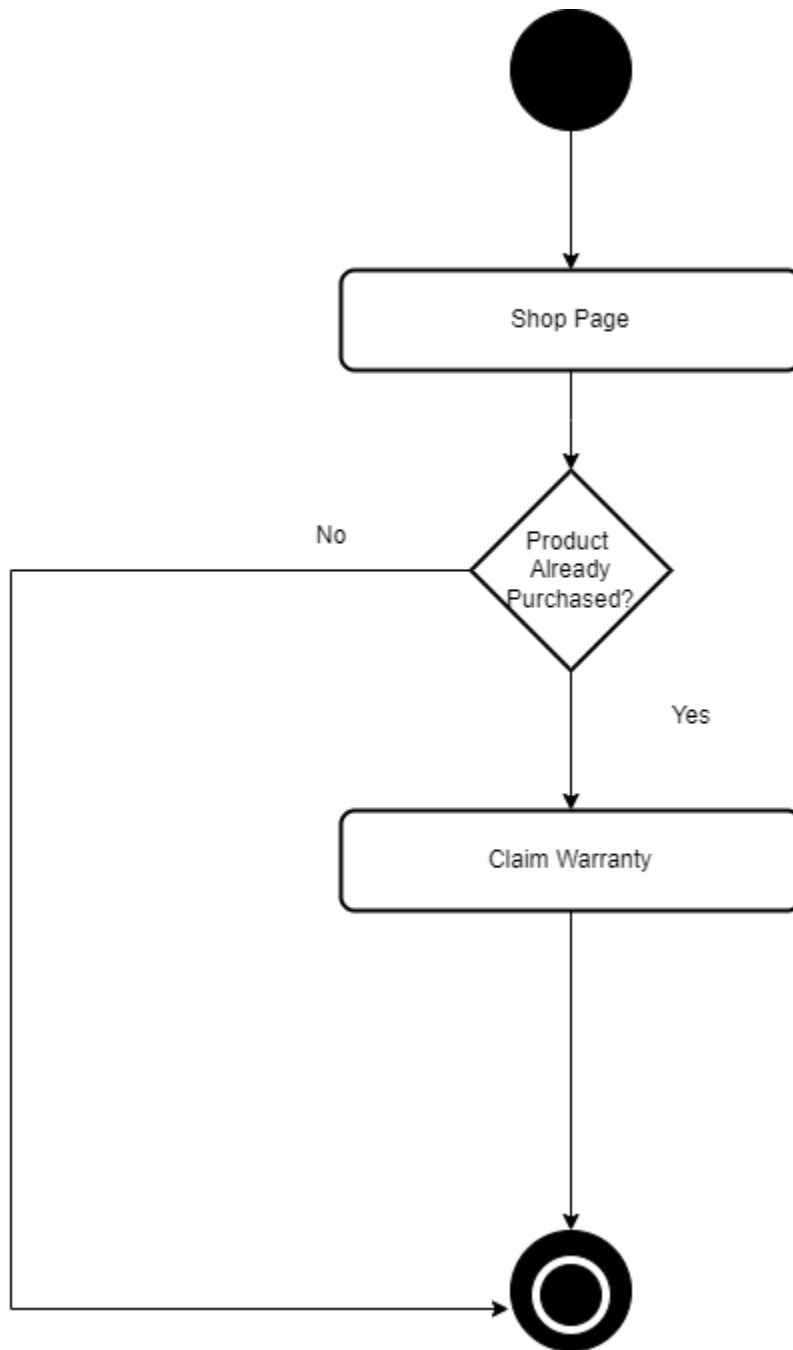
**Figure 3.3.13: Generate Reports**

### **3.3.14 Buy a Product**



**Figure 3.3.14: Buy A Product**

### **3.3.15 Claim Warranty**



**Figure 3.3.15: Claim Warranty**

### **3.3.16 Claim Return**

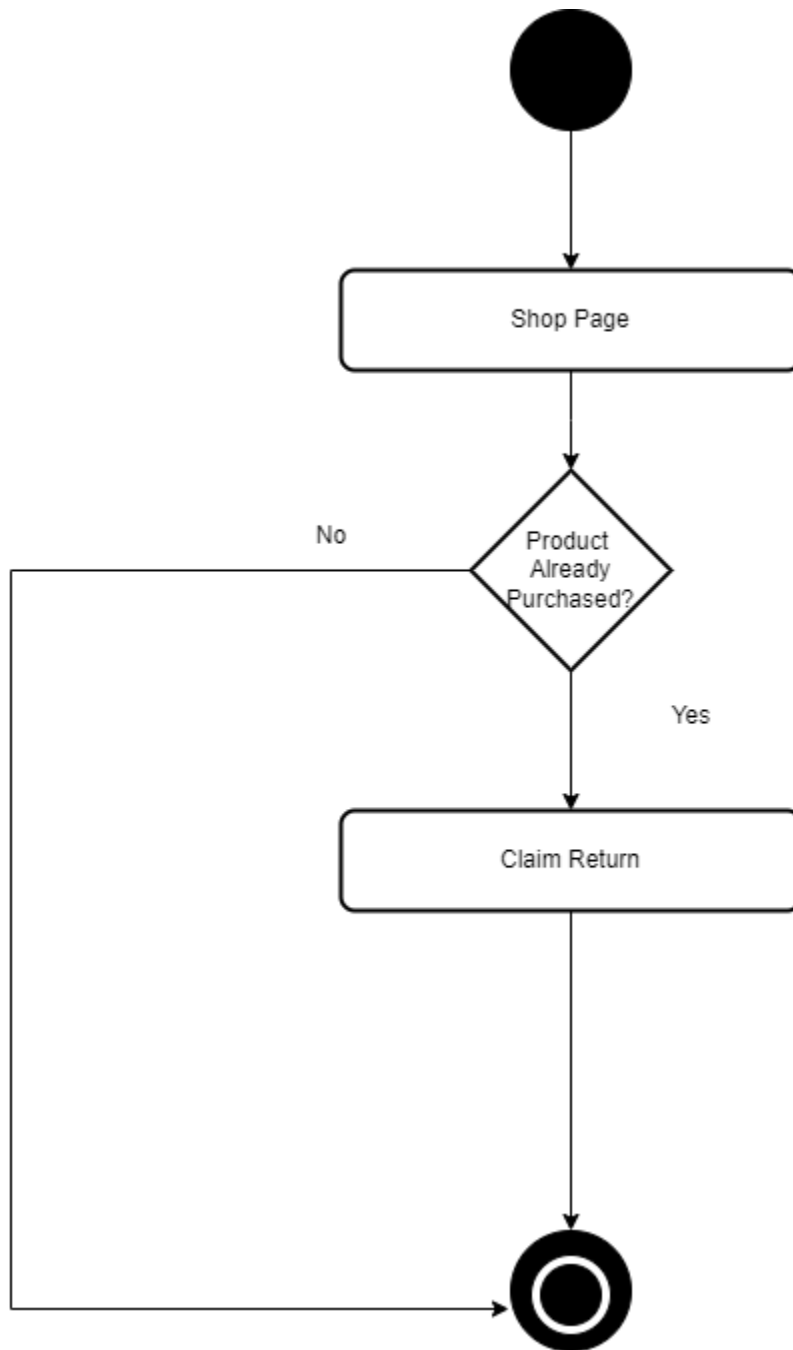


Figure 3.3.16: Claim Return

### 3.4 Sequence Diagram

#### 3.4.1 Login, Sign Up

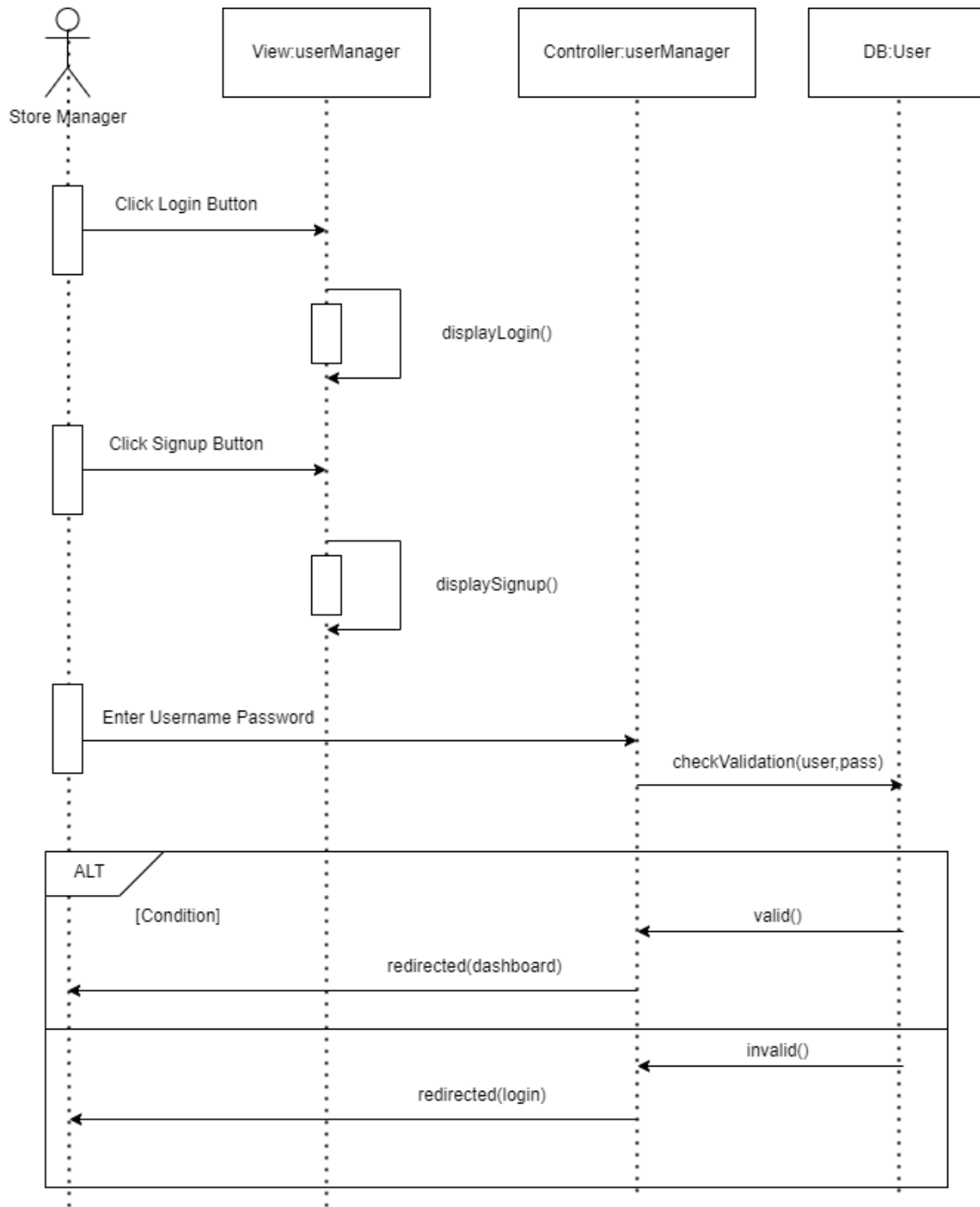
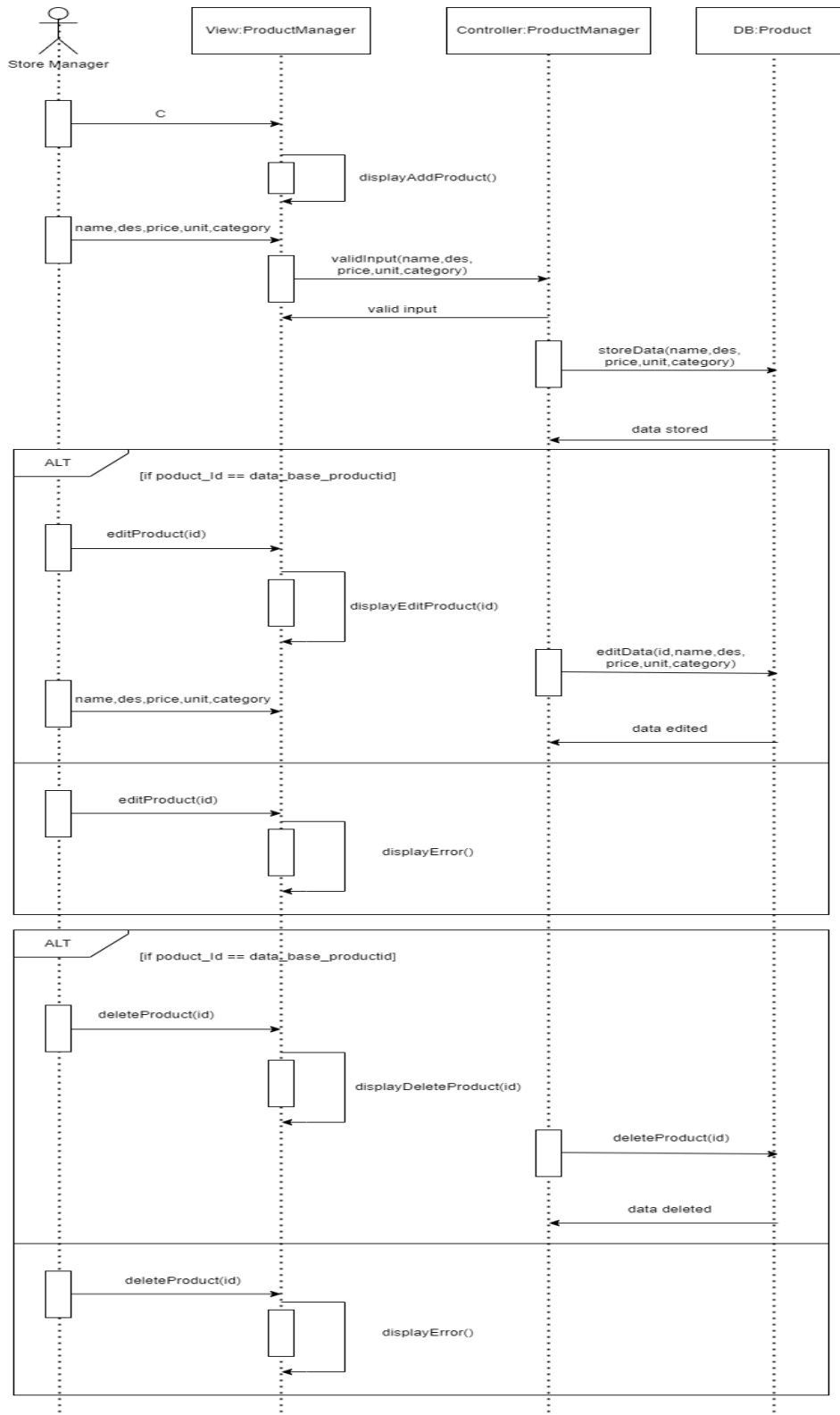


Figure 3.4.1: Login, Sign Up

### 3.4.2 Add/Delete/Edit Item



**Figure 3.4.2: Add/Delete/Edit Item**  
**3.4.3 Add/Delete Order**

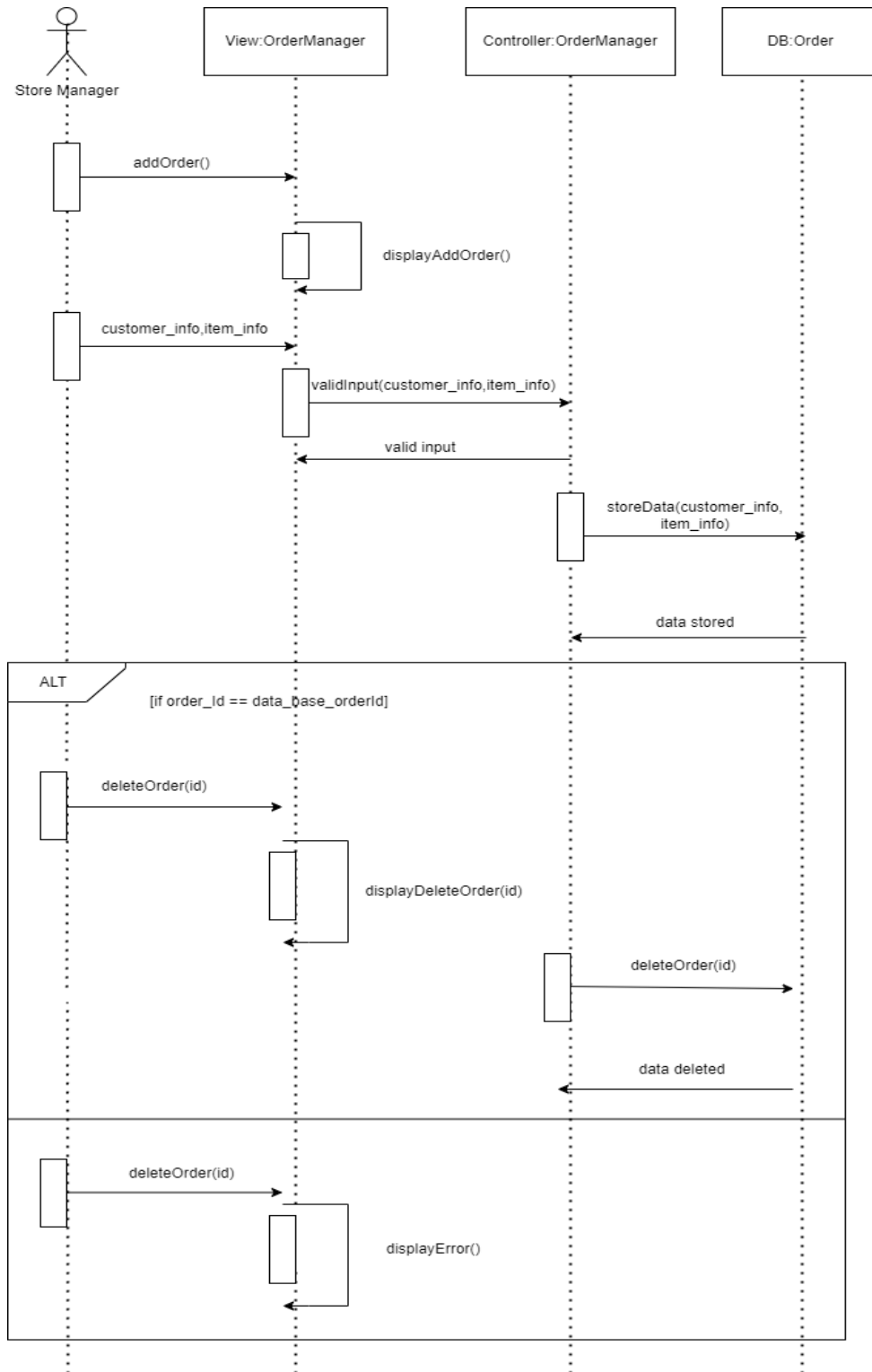


Figure 3.4.3: Add/Delete Order

### 3.4.4 Add/Delete Purchase

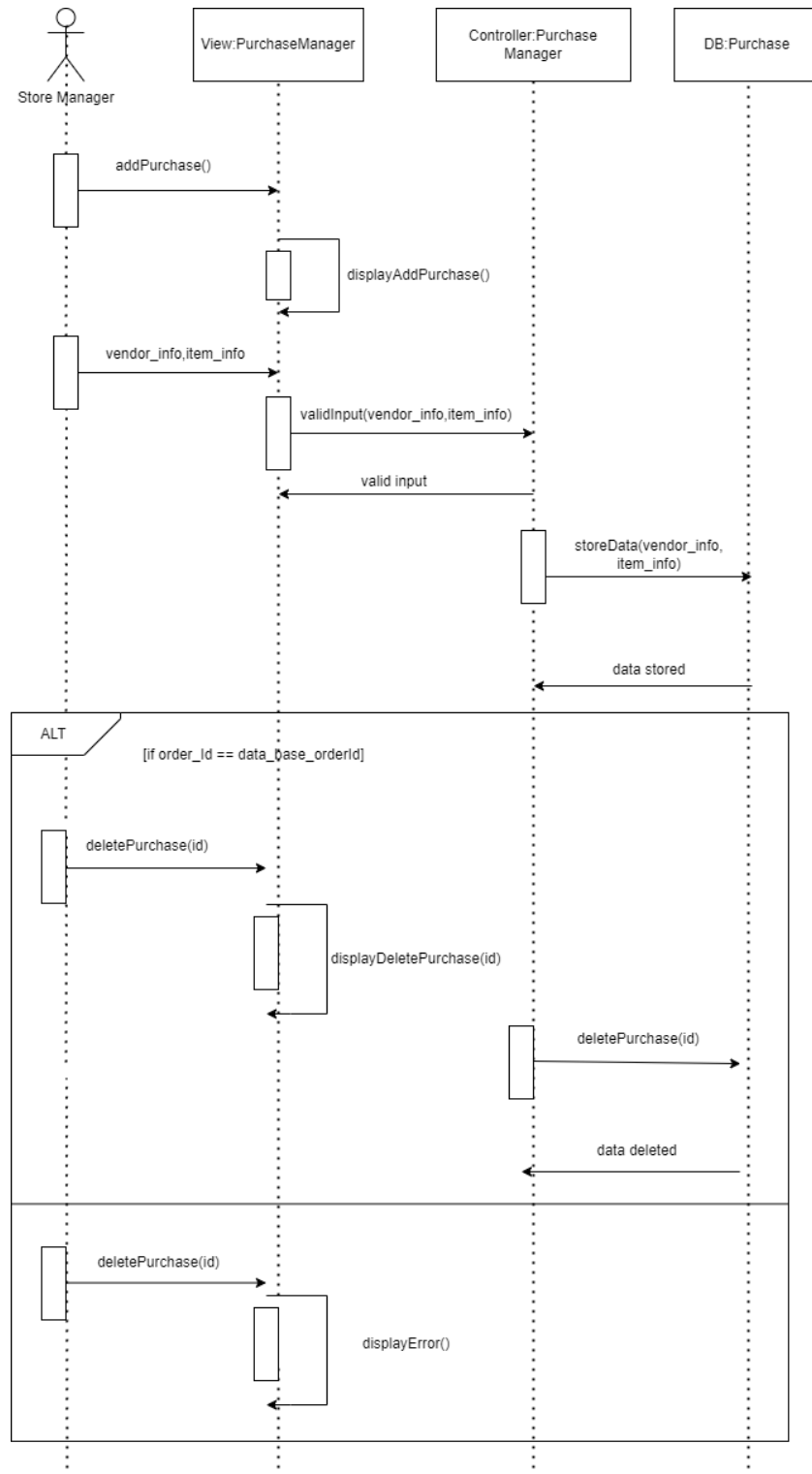


Figure 3.4.4: Add/Delete Purchase

### 3.4.5 Add/Delete Invoice

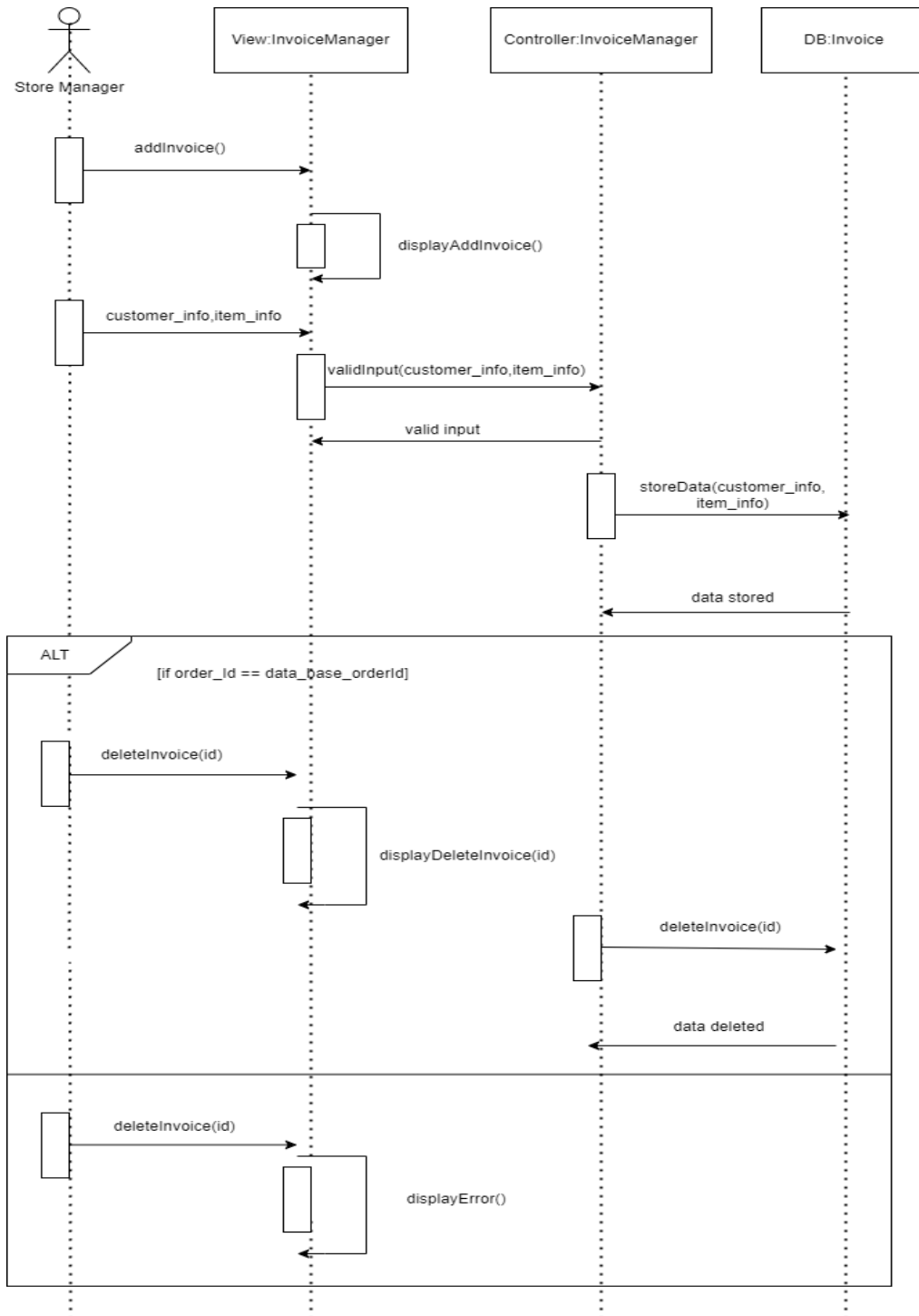


Figure 3.4.5: Add/Delete Invoice

### 3.4.6 Add/Delete Warranty

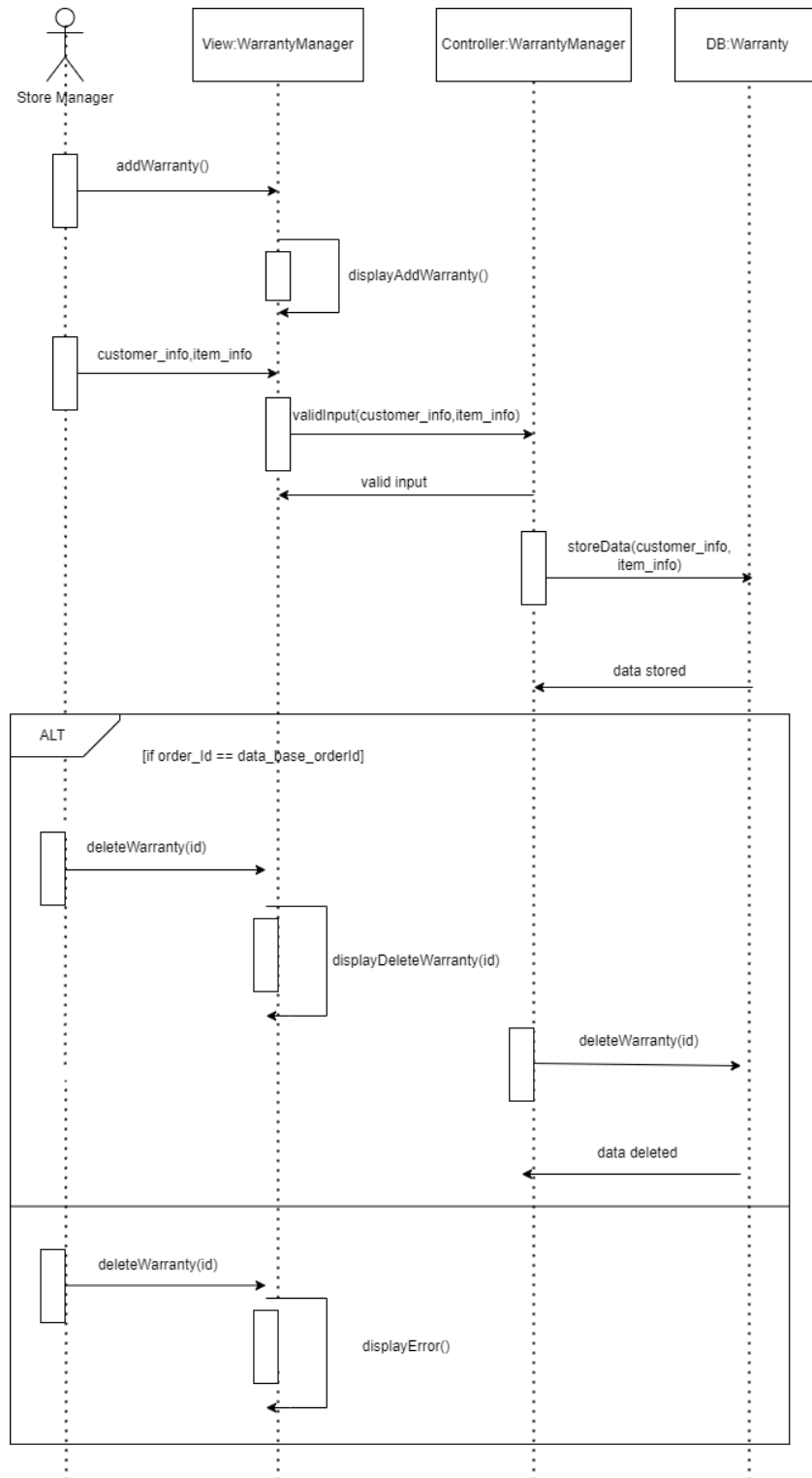
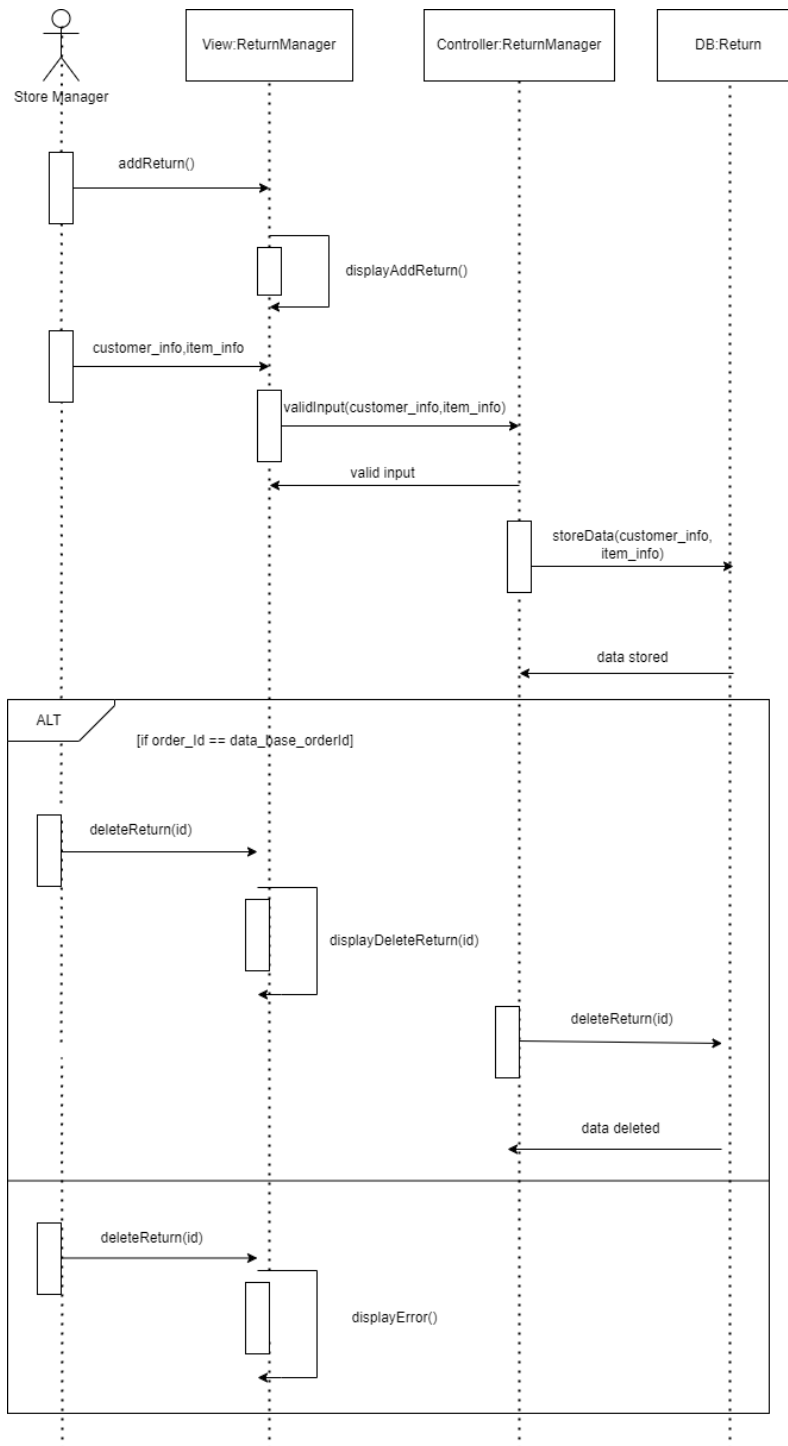


Figure 3.4.6: Add/Delete Warranty

### 3.4.7 Add/Delete Return



**Figure 3.4.7: Add/Delete Return**

### 3.4.8 Add/Delete Accounts

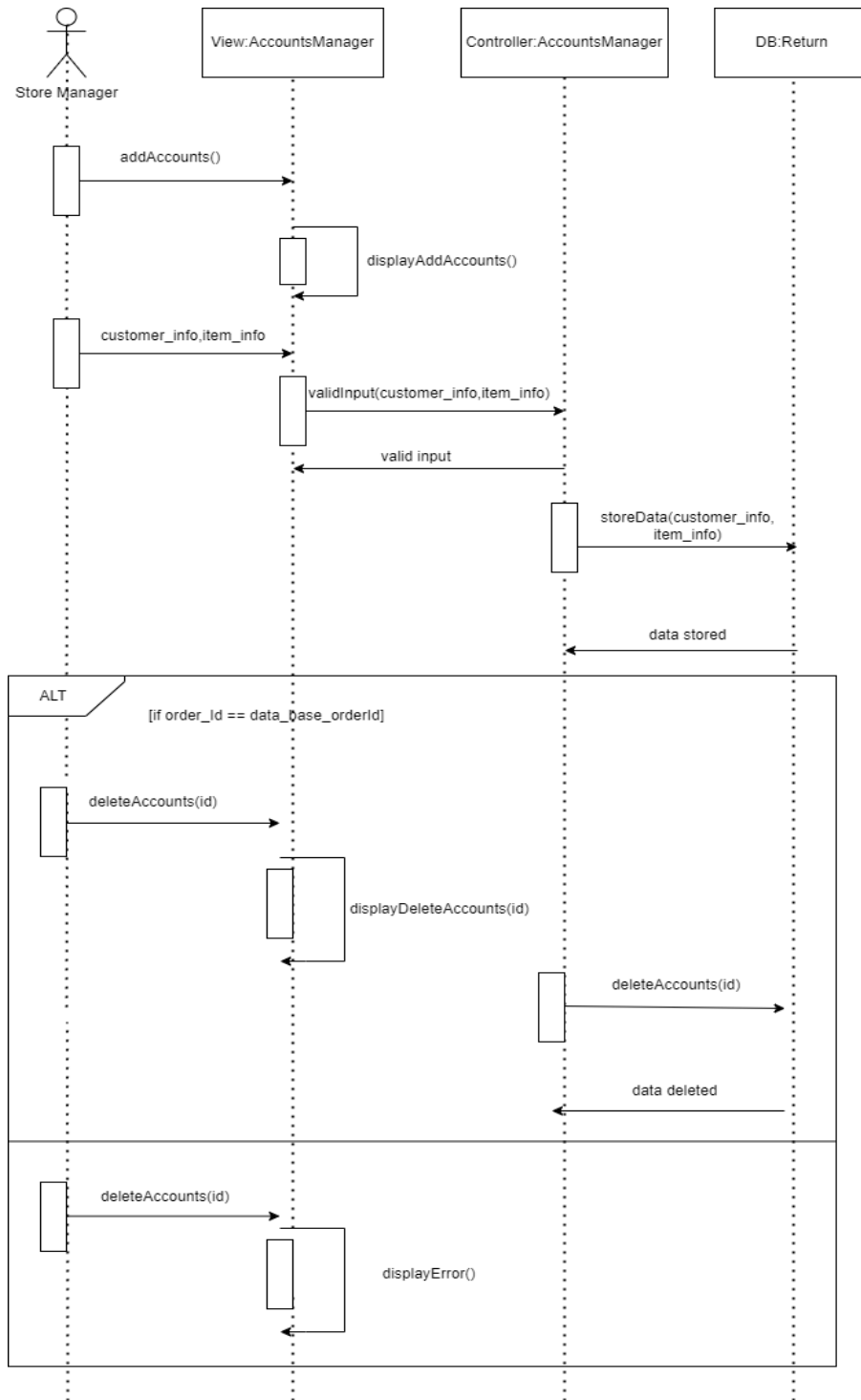
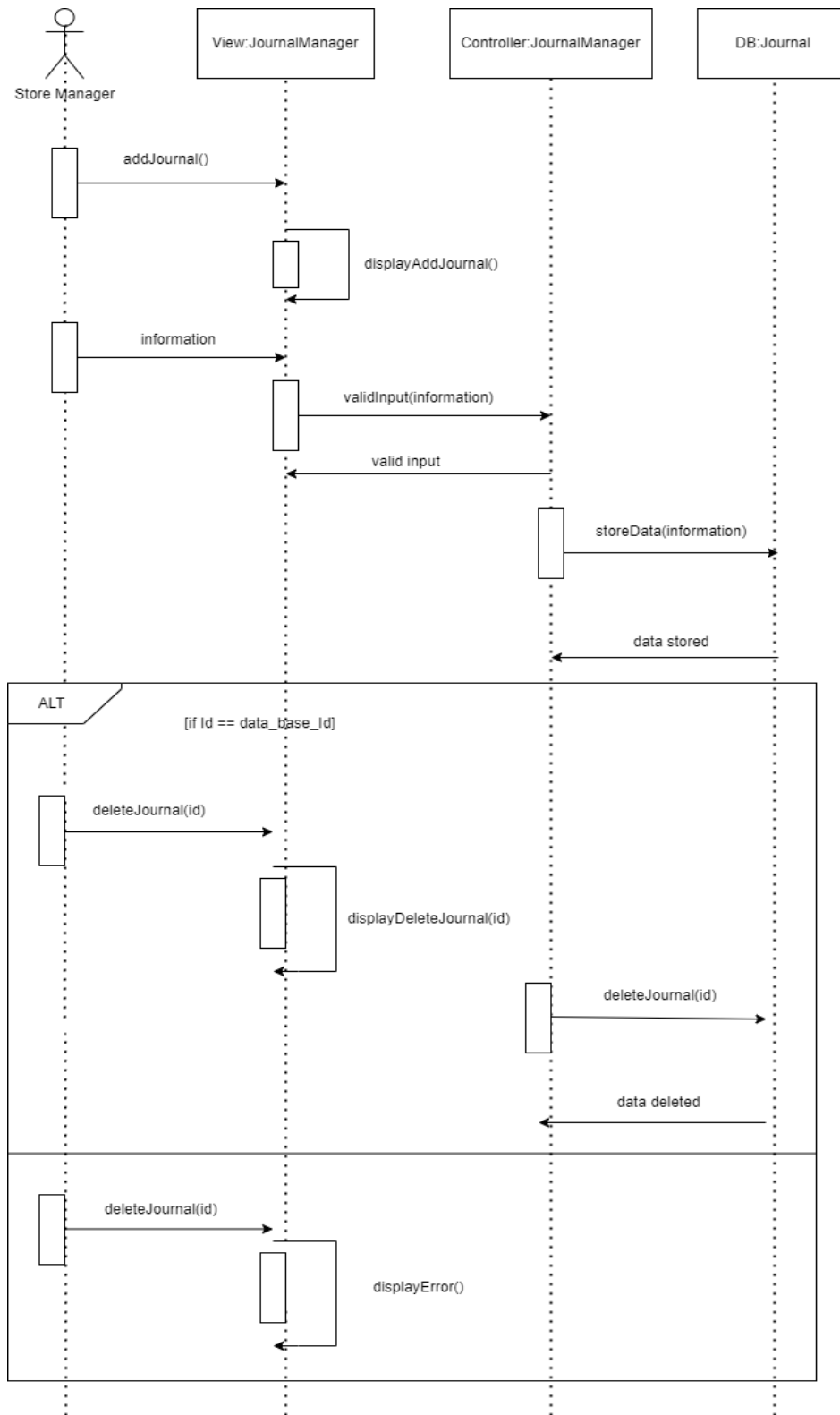


Figure 3.4.8: Add/Delete Accounts

### 3.4.9 Add/Delete Journal



**Figure 3.4.9: Add/Delete Journal**  
**3.4.10 Add/Delete Payment**

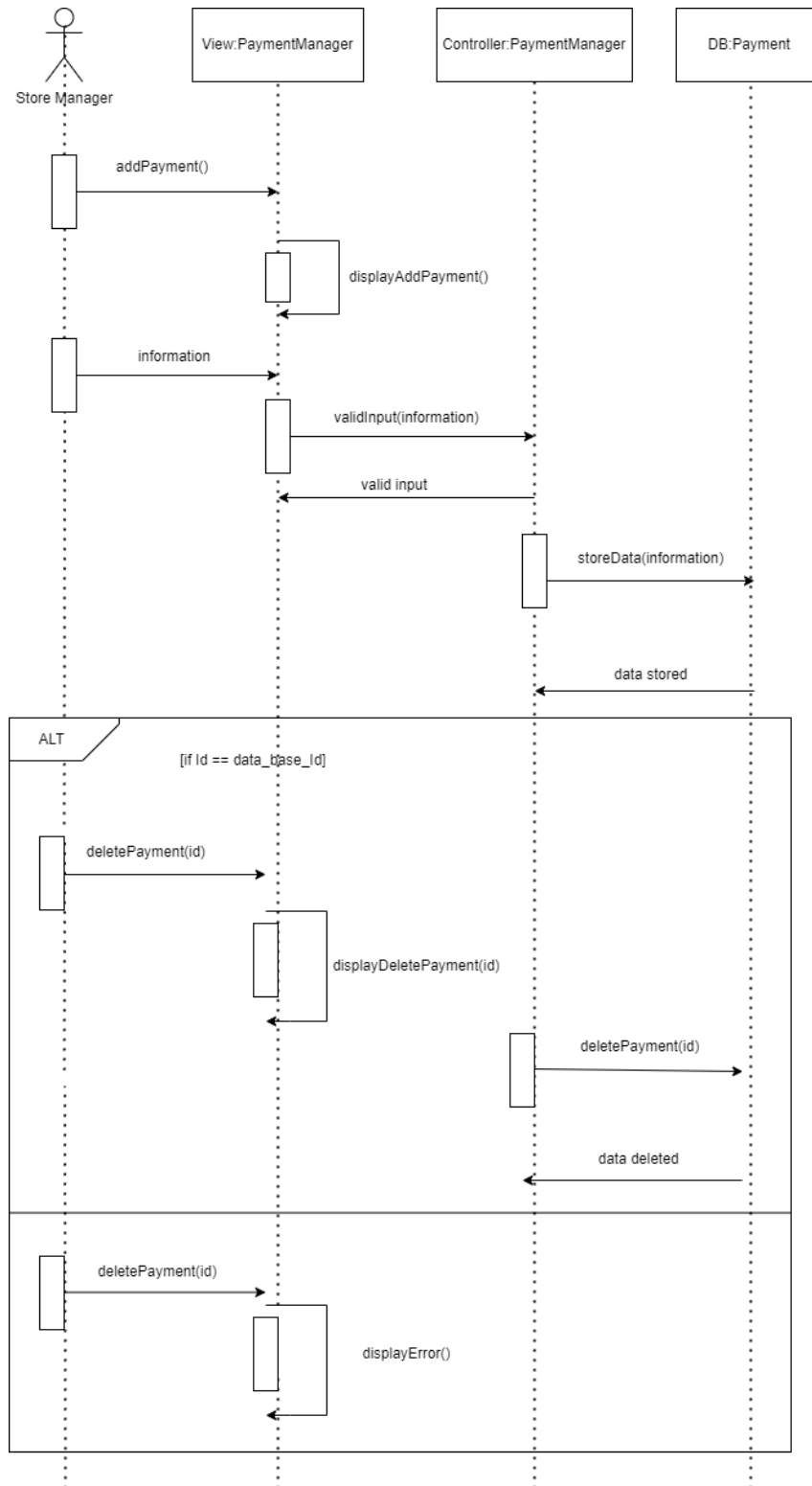
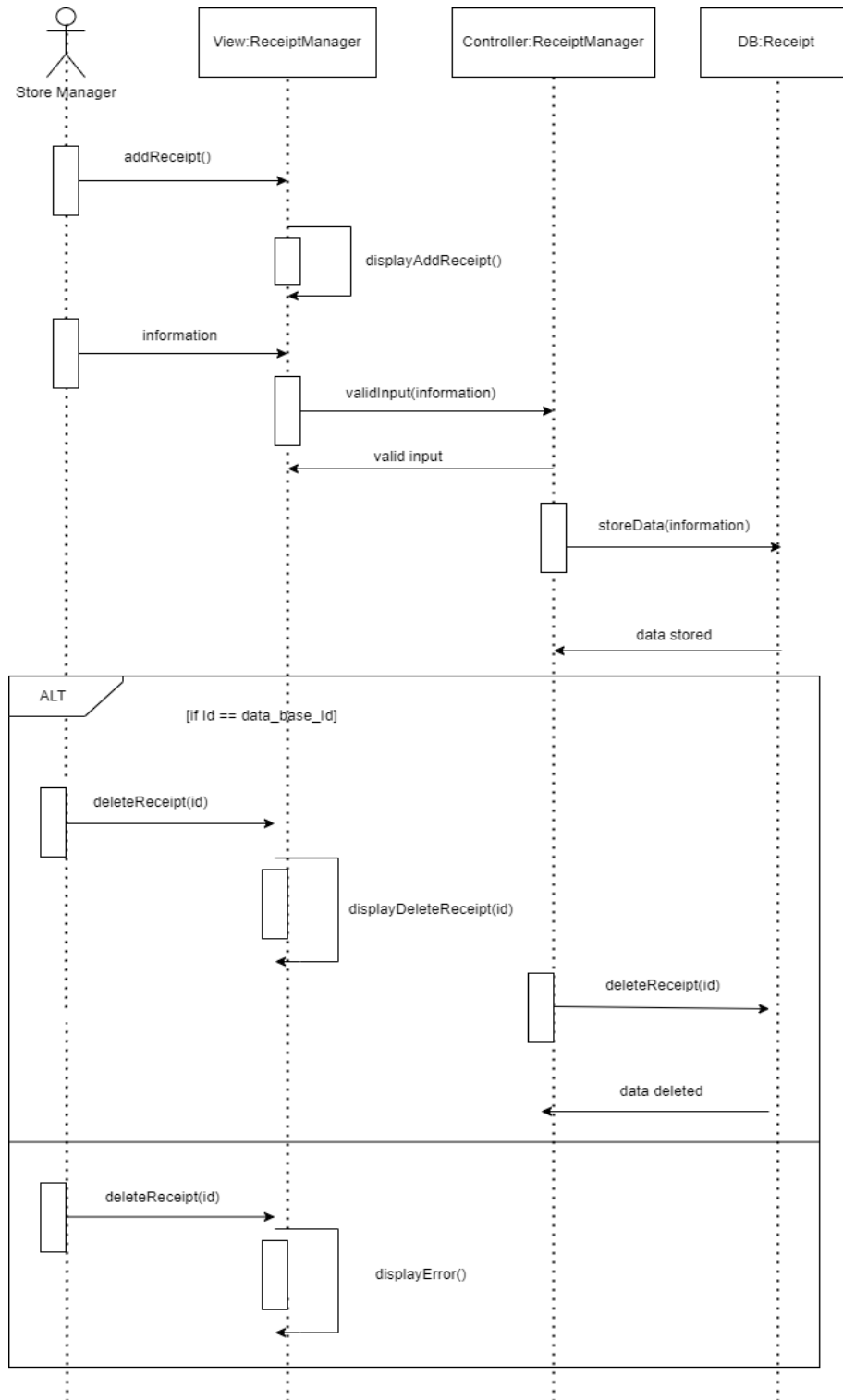


Figure 3.4.10: Add/Delete Payment

### 3.4.11 Add/Delete Receipt



**Figure 3.4.11: Add/Delete Receipt**  
**3.4.12 Generate Reports**

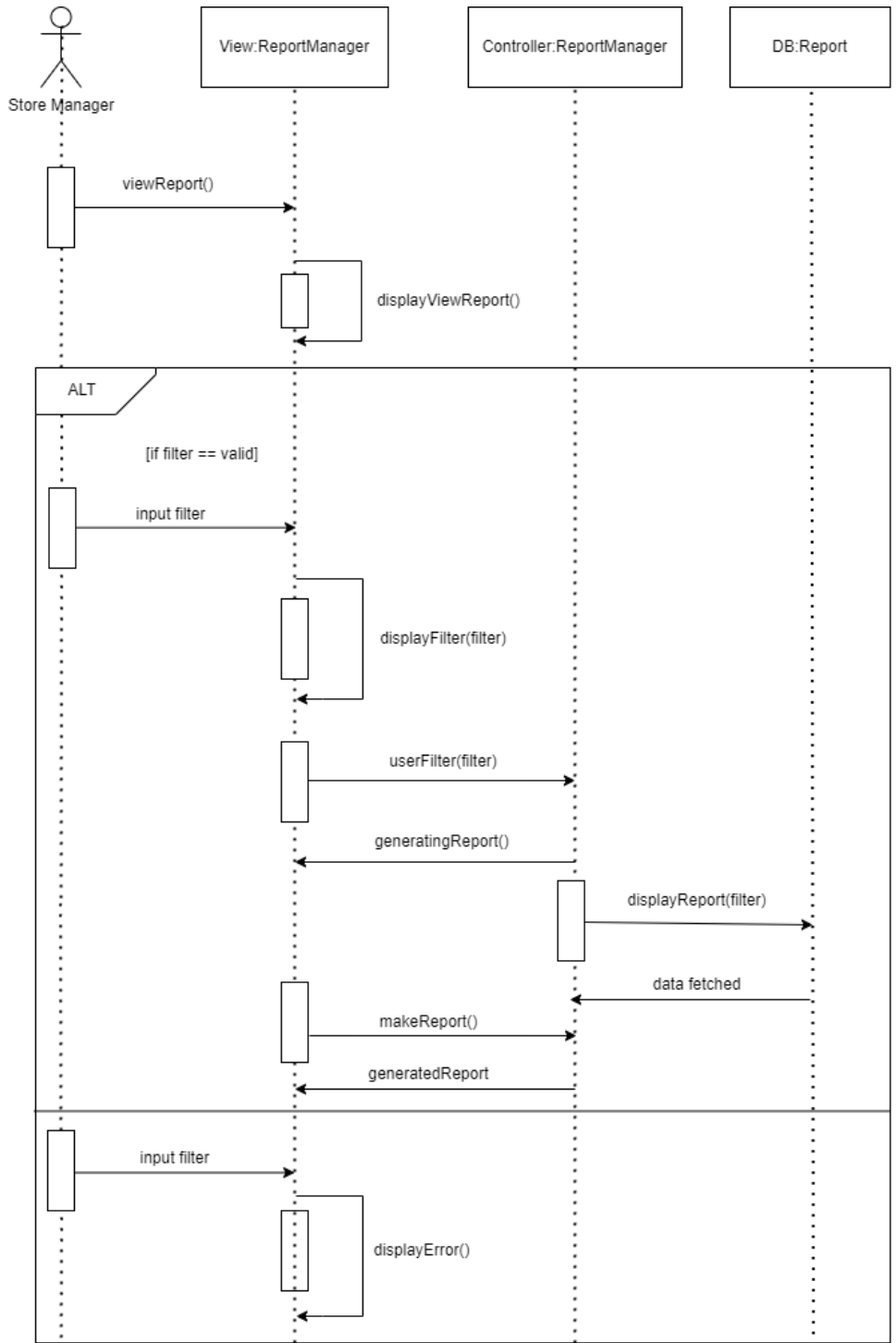


Figure 3.4.12: Generate Reports

### 3.4.13 Add To Cart

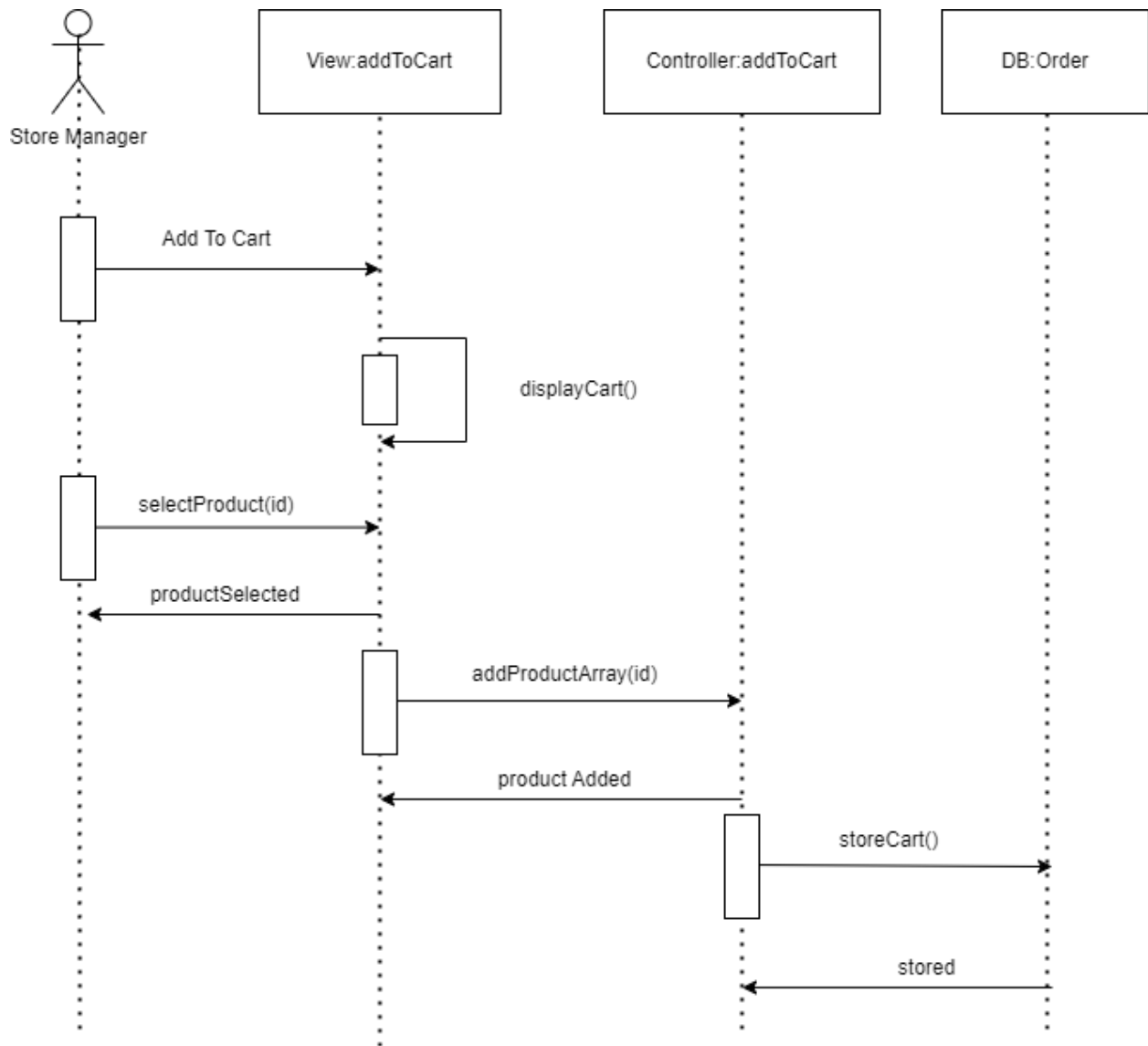


Figure 3.4.13: Add To Cart

### 3.4.14 Checkout

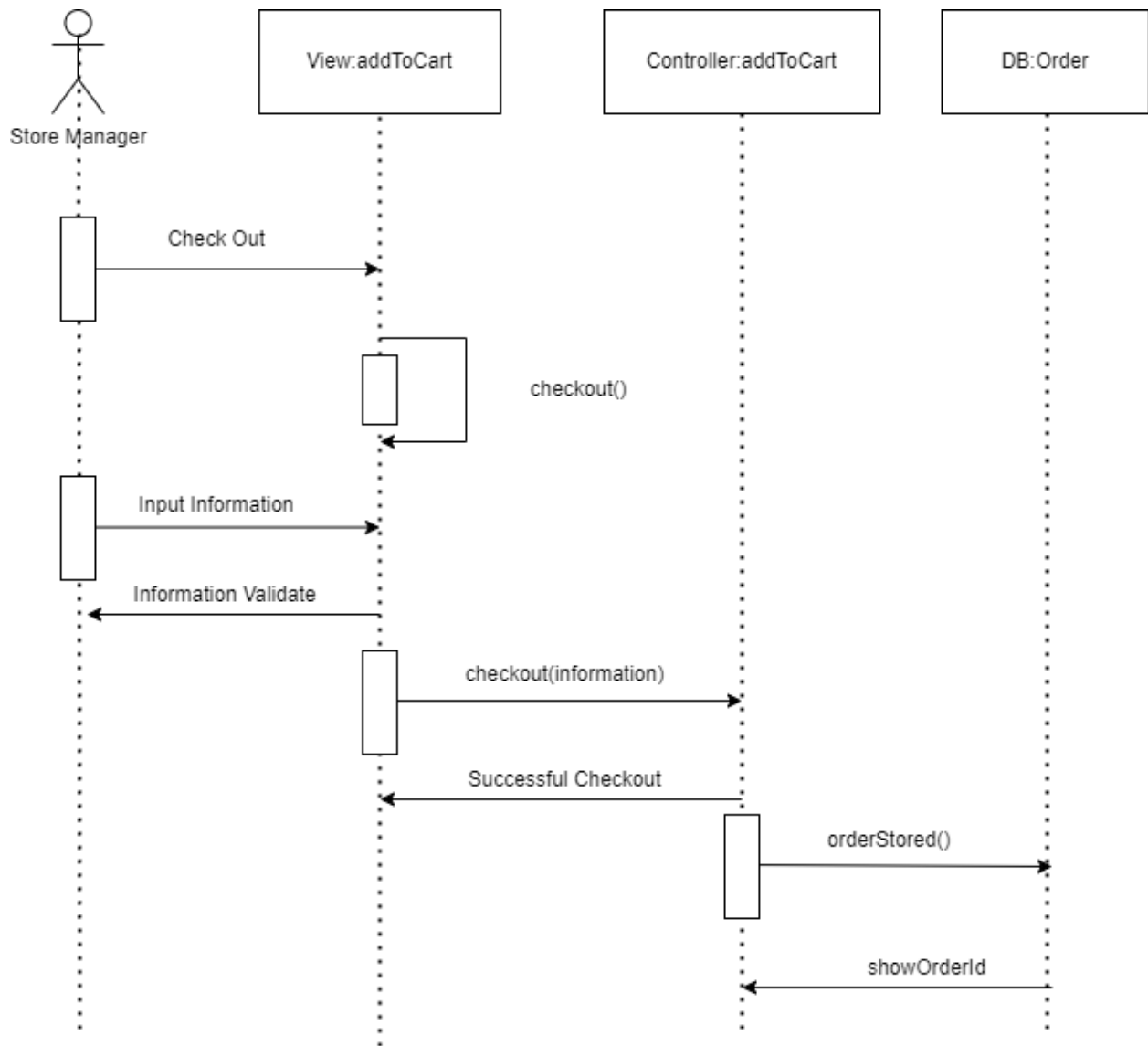


Figure 3.4.14: Checkout

### 3.4.15 Claim Warranty

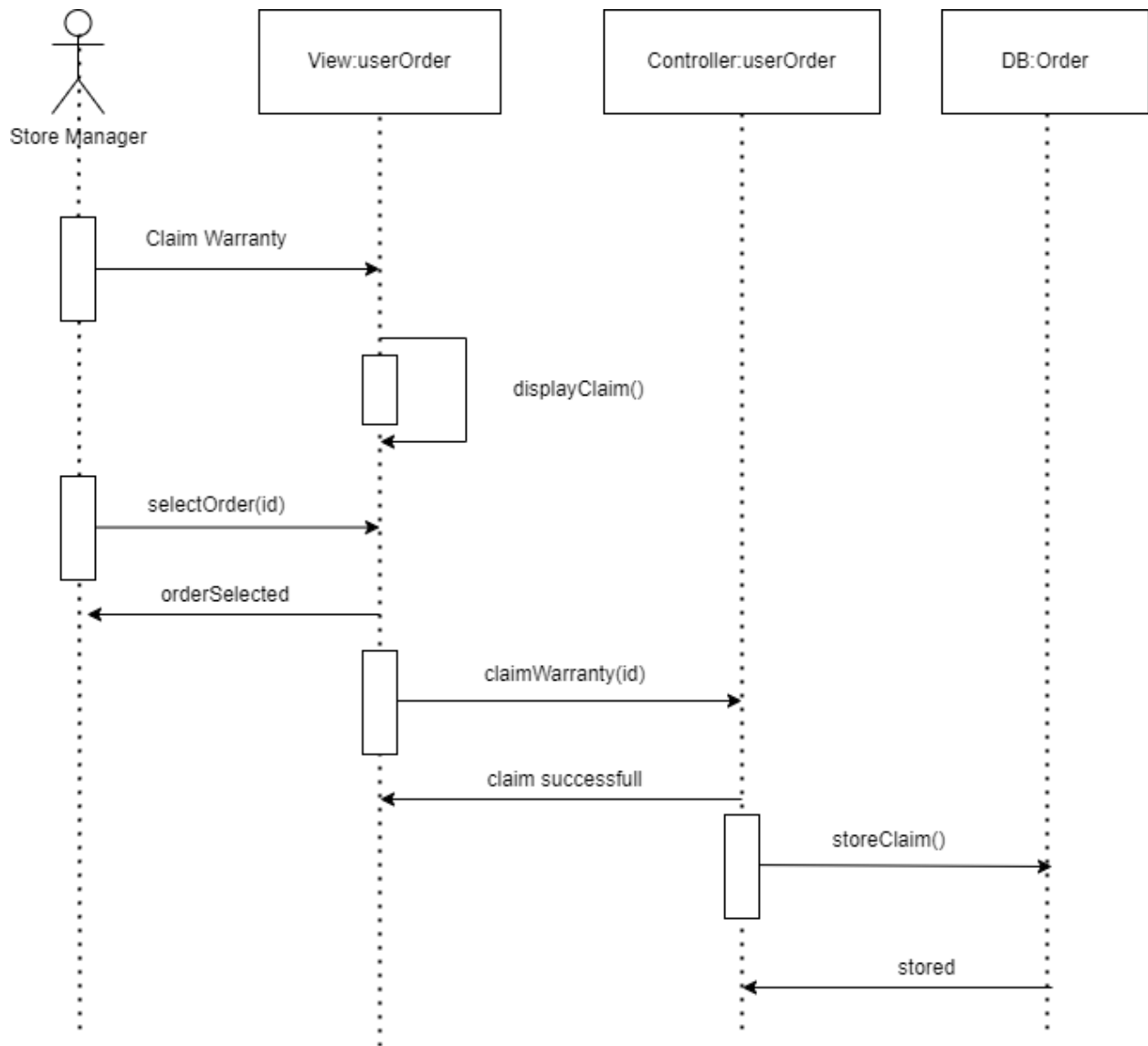


Figure 3.4.15: Claim Warranty

### 3.4.16 Claim Return

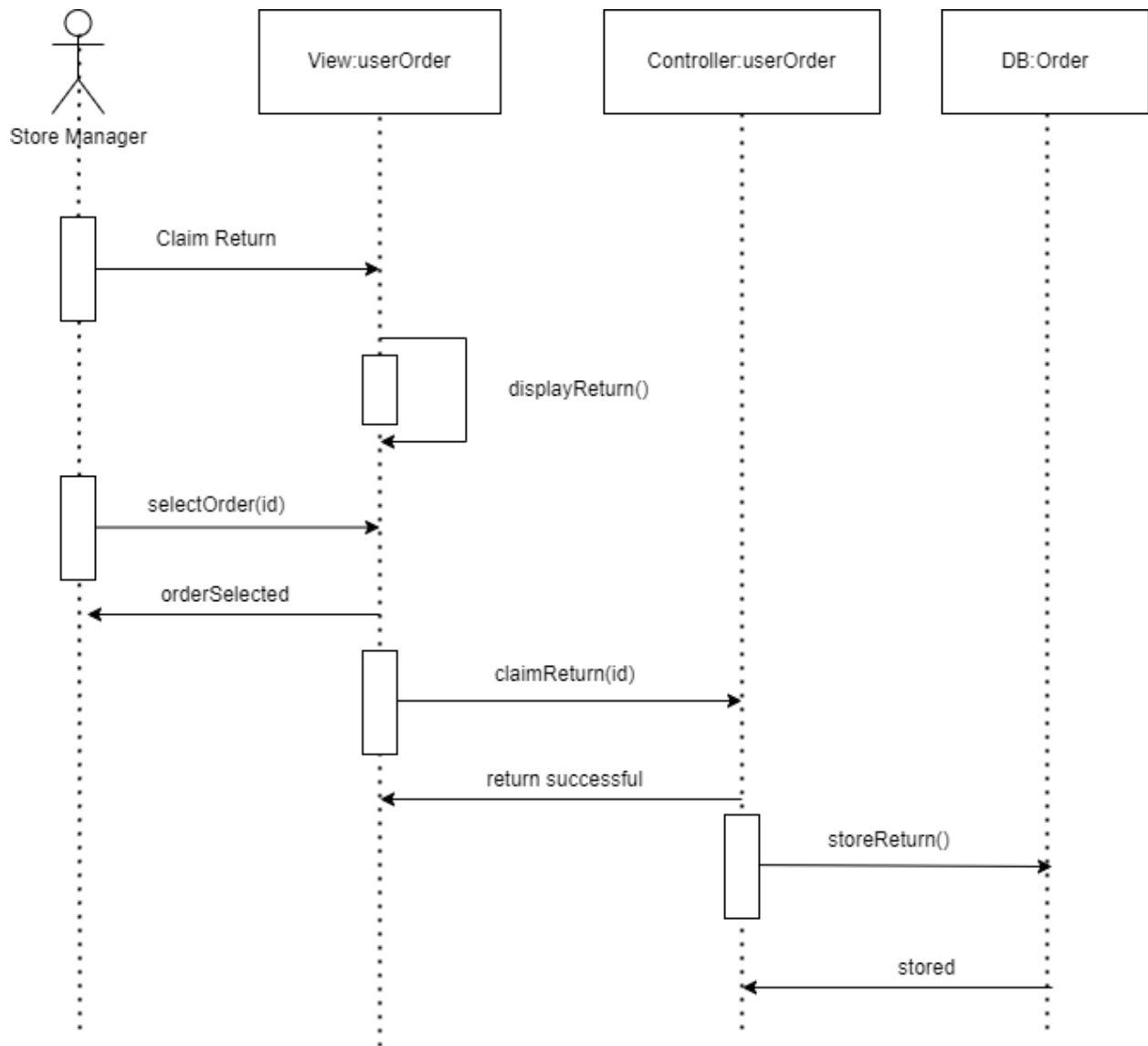


Figure 3.4.16: Claim Return

### 3.5 Block Diagram

#### 3.5.1 What is Block Diagram?

A block diagram is a system diagram where the main components or functions are represented by blocks joined by lines indicating the associations between the blocks. In engineering, they are

widely utilized in process flow diagrams, software, hardware, and electrical design.

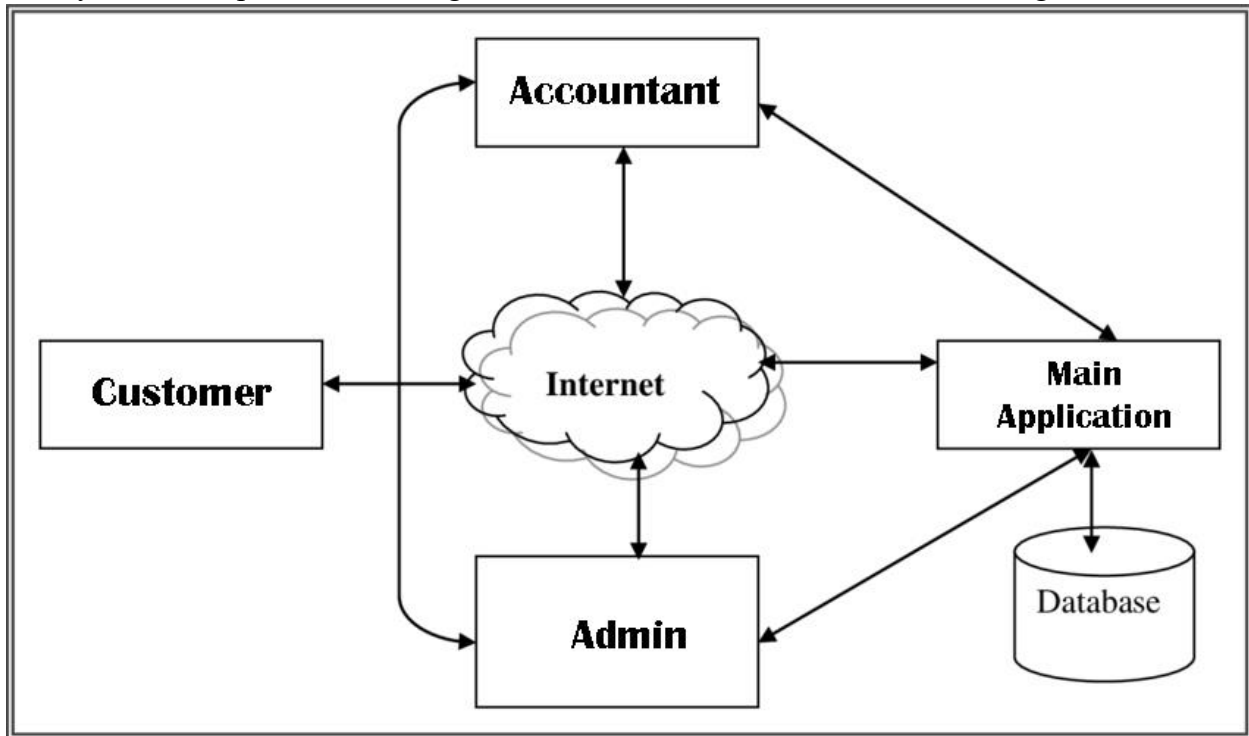
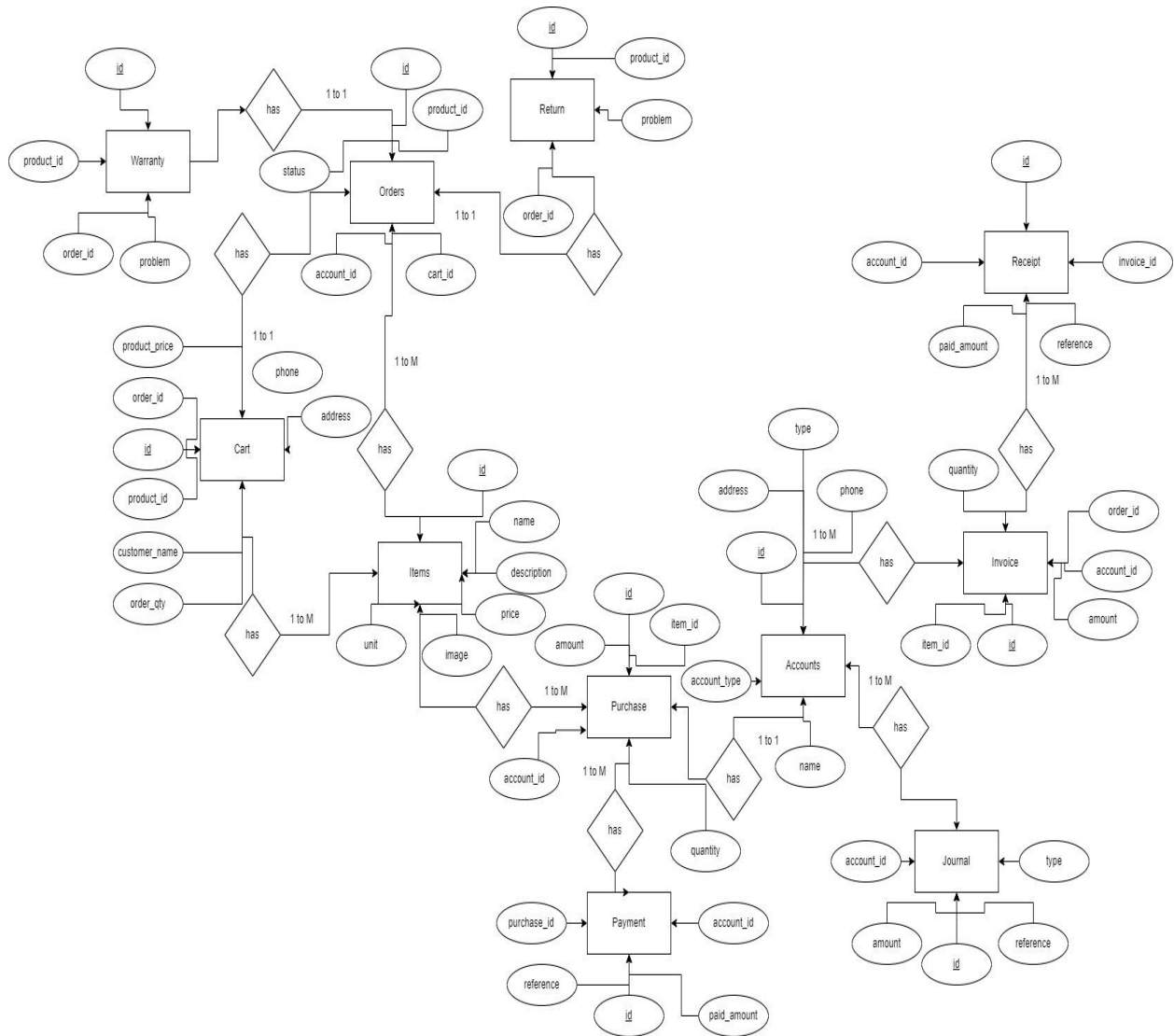


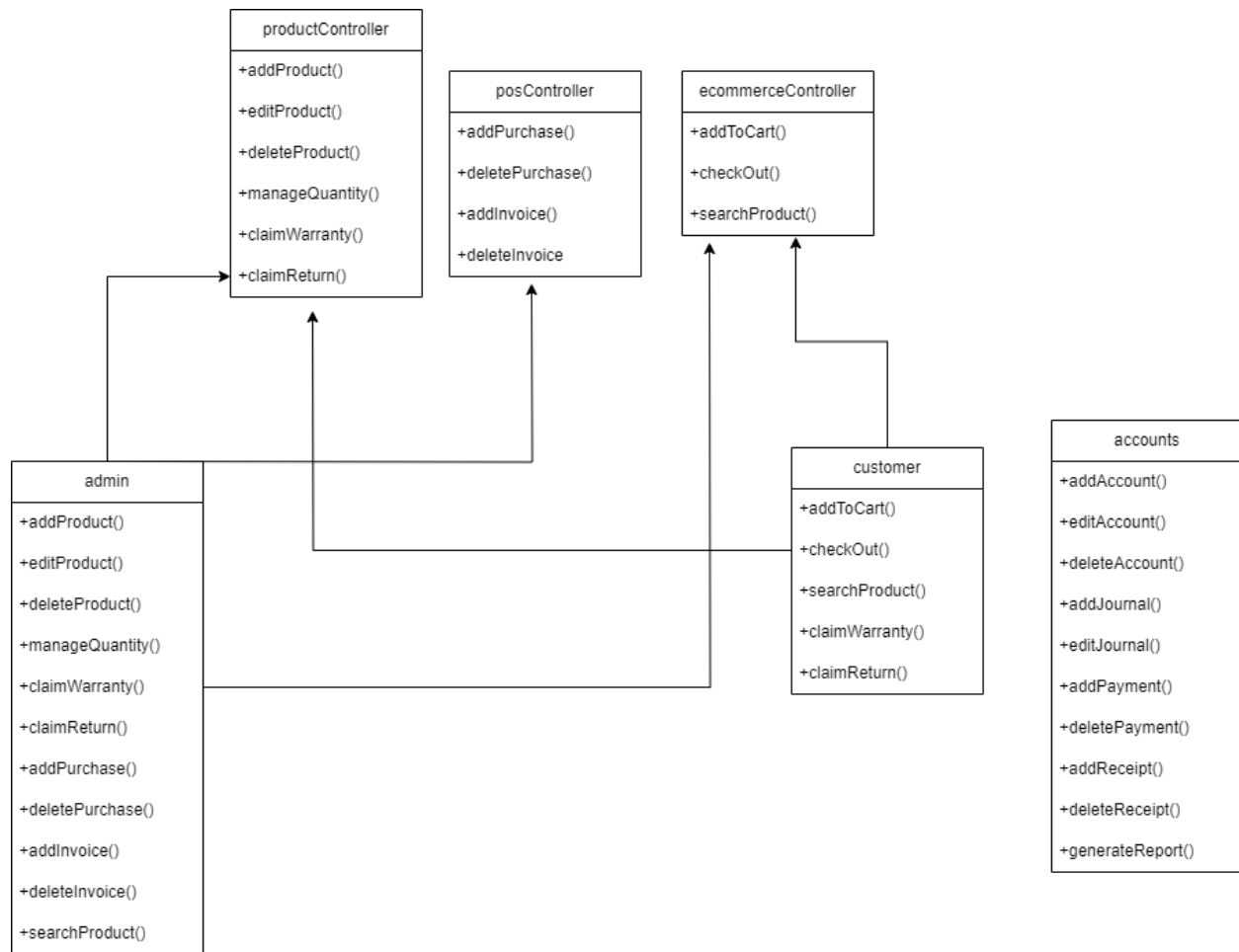
Figure 3.5.1: Block Diagram

### 3.6 ER Diagram



**Figure 3.6: ER Diagram**

### 3.7 Class Diagram



**Figure 3.7: Class Diagram**

# CHAPTER 4

## System Assessment

### 4.1 Feature Testing

Additions are added to the system or shifted, and this is considered to be done through feature testing. Every single function and capability has a unique characteristic about it. These features are designed for boosting the web application's potency, trustworthiness, safety, and utility.

#### 4.1.1 Feature Testing

Characteristics	Priority	Description
Login	1	I need to authenticate as the admin using the login.
Logout	2	The login session must stop after logging out.
By agreeing to sign up	1	The admin must be registered for first time use of this application
Manage Product	1	The admin can add/edit/delete a product of this application
Manage Customer	1	The admin can view customer information of this application
Manage Order	1	The admin add/delete/view orders of this application
Manage Sale	1	The admin add/delete/view sales of this application
Manage Purchase	1	The admin add/delete/view purchases of this application
Manage After Sales	1	The admin add/delete return and warranty of this application
Reports	1	The admin view reports of this application
Journal	1	The admin add/delete/view journal of this application
Order	1	The customer can place an order form this application
Track	1	The customer can track an order/warranty form this application

Table 4.1: Table of Rankings

### 4.2 Methods of Assessment

#### 4.2.1 Method of Screening

I have employed two distinct assessment methods techniques for guaranteeing that my software is of the greatest quality. The majority of my work was on functional and structural testing.

- White box testing, often referred to as glass box testing or caked structural testing, is an assessment of the interior workings of a system or component.
- Black box testing, often referred to as functional testing, is testing that ignores a system's or component's interior operations in favor of relying solely on the outputs exerted in response to preset inputs and execution requirements.

## 4.2.2 Successful or unsuccessful Levels

I construct successful and losing prerequisites for unit testing. Based on which information inputs operate properly and which are not, I figure out the successful or unsuccessful criterion. It has been proven that well-functioning data meet the pass criteria. Additionally, the rest of the input data is seen as a failure criteria. Some pass/fail standards are listed below:

- One considers a system crash to be a failure scenario.
- Only when a criterion has a perfect score during testing is it considered to have passed.
- Improper data display by the system should also be considered a failure criteria.

## 4.2.3 Timetable for Assessing

Assessing Phase	Time
Construct a test technique	A week in a row
Units of testing	In the process of evolution
Test component	In the process of evolution
Testing for coupling	A week in a row
User interface validation	A week in a row
Testing dosages	A week in a row
Testing for effectiveness	A week in a row
Testing for mobility	A week in a row

Table 4.2: Timetable for Assessing

## 4.2.4 Testing Schedule

In order to set up a testing environment, the equipment and software required for the tester to execute the test case as requested must be given. In addition to using gear 4.2.3 the program, network configuration is required to carry out the plans for testing. I made advantage of a few crucial spaces for the testing setup. In the section below:

- Review facts
- ApacheWeb server
- MSQlDatabase server
- Front-end Linux operating system
- Setting up the environment for the back end
- Browser on an internet connection
- Applications and System

#### 4.2.5 Test Cases

A test case is a collection of requirements or guidelines intended to determine if a system can perform more effectively under test conditions. We understand that every system may possess flaws and drawbacks. These kinds of things happen all the time while designing software.

Additionally, the solutions for those issues typically use software testing methodologies. But if we overlook the aforementioned issues, the system development process as a whole can collapse.

As such, sufficient testing needs to be done for the development process. I've created a few test cases to aid with the application's testing. I will now discuss my test scenarios:

Test Cases:

A test case is a collection of regulations and specifications intended to ascertain whether a system can perform more effectively under test environments. We understand that mistakes can happen in any system. These kinds of situations happen all the time while designing software. Moreover, strategies for software testing are typically used to solve those problems. If we overlook those issues, though, the system development process as a whole would not accomplish. Thus, sufficient evaluation is required for the development process. I've derived a number of test cases with the objective to test my system application. I'll now review my test cases:

## 1. Sign upward

An instance of Test #01		Name of instance of test: Register			
A structure:		Subsystem's functionality: N/A			
Based By: Erfan		intended Date: 01-January-2024			
Accomplished by:		Underway Date:			
A succinct explanation: The system will save records of new customers.					
Pre-Scopes:					
<ul style="list-style-type: none"> <li>User's name, email address, and password must be active.</li> </ul>					
Describe	Electronic Mail	Secret Key	Anticipated Outcome	Move on/Error	Observation
Erfan	erfan	4456	In addition a valid email.	Fail	
		2312	The email box and name are compulsory.	Fail	
1224			The name is not genuine.	Fail	
Erfan	erfan35-2610@diu.edu.bd	123456789	Register Successfully	Error	
Post-Scopes: The application can now be launched by the user.					

## 2. Login

An instance of Test #02		Name of instance of test: Login			
A structure:		Subsystem's functionality: N/A			
Based By: Erfan		intended Date: 01-January-2024			
Accomplished By:		Underway Date:			
A succinct explanation: The system will verify your reliability and authorization.					
Pre-Scopes:					
<ul style="list-style-type: none"> <li>Users will be redirected to the login page if they have not authenticated by my system while entering into the admin dashboard.</li> </ul>					
Describe	Electronic Mail	Secret Key	Anticipated Outcome	Move on/Error	
erfan@test.test		4456	The credentials don't match our records	Fail	
		2312	The email field is required	Fail	
			The email and password field is required	Fail	
erfan35-2610@diu.edu.bd		123456789	Login Successfully	Error	
Post-Scopes: User can access the admin dashboard					

### 3. Logout

An instance of Test #03		Name of instance of test: Logout	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: The user session is finished on the system.			
Pre-Scopes:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Describe	Electronic Mail	Secret Key	Anticipated Outcome
The strike Log out button	The session came to its end.	Ok	
Close Browser tab	The session came to its end.	Ok	
After-Scopes: The user can re-login with a new password and address			

### 4. Add Product

An instance of Test #04		Name of instance of test: Add Product	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will add a product			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The strike Add button	Add product	Ok	
Save	Product added	Ok	
Post-conditions: User can now view products in list			

### 5. Delete Product

An instance of Test #05		Name of instance of test: Delete Product	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will delete a product			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The strike Delete button	Delete product	Ok	
Confirm	Product deleted	Ok	
Post-conditions: User can't view deleted products in list			

## 6. Add Purchase

An instance of Test #06		Name of instance of test: Add Purchase	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will register a purchase bill			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Add button	Add Purchase	OK	
Save	Purchase added	OK	
Post-conditions: User can now view Purchases in list			

## 7. Delete Purchase

An instance of Test #07		Name of instance of test: Delete Purchase	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will delete a purchase bill			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Delete button	Delete Purchase	OK	
Confirm	Product deleted	OK	
Post-conditions: User can't view deleted Purchase in list			

## 8. Make Payment

An instance of Test #08		Name of instance of test: Make Payment	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will register a payment voucher			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Payment button	Make Payment	OK	
Save	Paid	OK	
Post-conditions: User can now view Payments in list			

## 9. Add Sale

An instance of Test #09		Name of instance of test: Add Sale	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will register a sell invoice			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Add button	Add Sale	OK	
Save	Sale added	OK	
Post-conditions: User can now view Sales in list			

## 10. Delete Sale

An instance of Test #10		Name of instance of test: Delete Sale	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will delete a sell invoice			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Delete button	Delete Sale	OK	
Confirm	Sale deleted	OK	
Post-conditions: User can't view deleted Sale in list			

## 11. Receive Payment

An instance of Test #11		Name of instance of test: Receive Payment	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will register a receipt			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Receive button	Receive Payment	OK	
Save	Paid	OK	
Post-conditions: User can now view Payments in list			

## 12. Add Refund

An instance of Test #12		Name of instance of test: Add Refund	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will register refund voucher			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Refund button	Refund Sale	OK	
Save	Refund added	OK	
Post-conditions: User can now view Refunds in list			

## 13. Delete Refund

An instance of Test #12		Name of instance of test: Delete Refund	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will delete refund voucher			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Delete button	Delete Refund	OK	
Confirm	Refund deleted	OK	
Post-conditions: User can't view deleted Refund in list			

## 14. Add Claim

An instance of Test #14		Name of instance of test: Add Claim	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will register a claim voucher			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Claim button	Claim Sale	OK	
Save	Claim added	OK	
Post-conditions: User can now view Claims in list			

## 15. Delete Claim

An instance of Test #15		Name of instance of test: Delete Claim	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will delete claim voucher			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Delete button	Delete Claim	OK	
Confirm	Claim deleted	OK	
Post-conditions: User can't view deleted Claim in list			

## 16. Add Journal

An instance of Test #16		Name of instance of test: Add Journal	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will delete register journal			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Journal button	Journal Sale	OK	
Save	Journal added	OK	
Post-conditions: User can now view Journals in list			

## 17. View Reports

An instance of Test #17		Name of instance of test: View Repots	
A structure:		Subsystem's functionality: N/A	
Based By: Erfan		intended Date: 01-January-2024	
Accomplished By:		Underway Date:	
A succinct explanation: System will show report			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			

Button	HTTP Response	Ok/Error	Comment
The Strike Report button	Show Report	OK	
Set Months	Show detailed report	OK	

Post-conditions: User can now view reports month-wise

## 18. Manage Order

An instance of Test #18	Name of instance of test: Manage Order		
A structure:	Subsystem's functionality: N/A		
Based By: Erfan	intended Date: 01-January-2024		
Accomplished By:	Underway Date:		
A succinct explanation: User can manage orders			
Pre-conditions:			
<ul style="list-style-type: none"> <li>Users must have login.</li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Order button	Show Order	OK	
Set Status	Status Updated	OK	

Post-conditions: User can now track orders

## 19. Place Order

An instance of Test #18	Name of instance of test: Place Order		
A structure:	Subsystem's functionality: N/A		
Based By: Erfan	intended Date: 01-January-2024		
Accomplished By:	Underway Date:		
A succinct explanation: User can place an order			
Pre-conditions:			
<ul style="list-style-type: none"> <li></li> </ul>			
Button	HTTP Response	Ok/Error	Comment
The Strike Add to cart button	Item added to cart	OK	
The Strike Checkout button	Checkout page	OK	
The Strike Place order button	Order placed	OK	

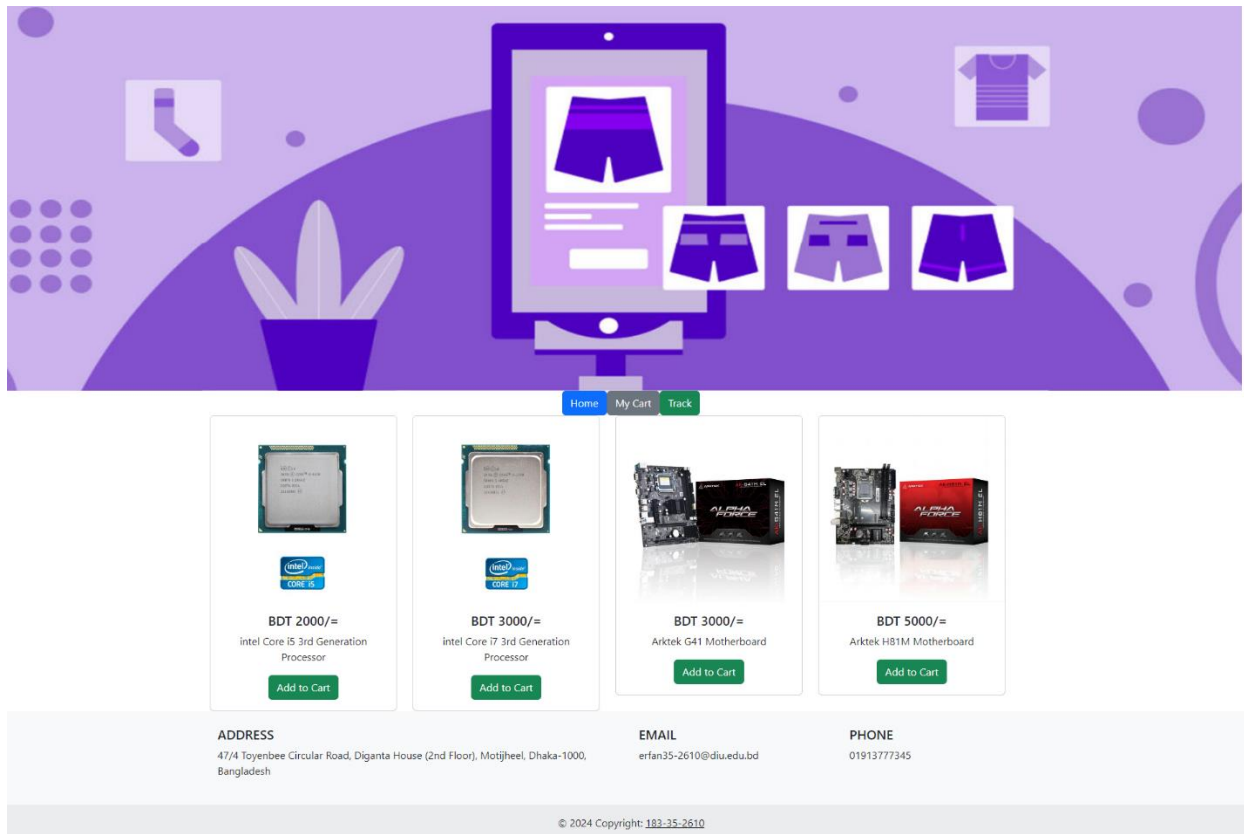
Post-conditions:

# CHAPTER 5

## Client Guide

### 5.1 Order Place

A user can place order by follow few steps. First user has to hit add to cart button of his desired item then go to my cart then fill up name, mobile number, address and payment method then hit place order button. In that page user can delete item from his cart and update item quantity of his cart. After successfully place order user will get and order number. With this number user can track order and claim warranty in future.



[Home](#) [My Cart](#) [Track](#)

Customer Name

Erfan

Mobile Number

01913777345


Address

Dhaka

Payment Type

Cash On Delivery

[Complete Order](#)

Image	Product Name	Price	Quantity	Update	Delete
	Intel Core i5 3rd Generation Processor	2000	<input type="text" value="1"/>	<a href="#">Update</a>	<a href="#">Delete</a>

[Cancel](#)

ADDRESS	EMAIL	PHONE
47/4 Toyenbee Circular Road, Diganta House (2nd Floor), Motijheel, Dhaka-1000, Bangladesh	erfan35-2610@diu.edu.bd	01913777345

© 2024 Copyright: 183-35-2610

[Home](#) [My Cart](#) [Track](#)

Tracking#

Tracking Type

Order

[Track](#)

Your Order Status

Date:                      Order No.

---

BDT /=

Tracking Order

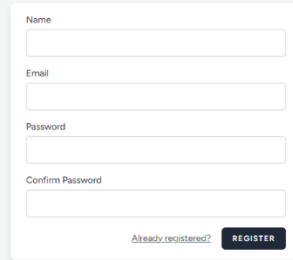
—————

Claim #	Product Name	Remarks
ADDRESS	EMAIL	PHONE
47/4 Toyenbee Circular Road, Diganta House (2nd Floor), Motijheel, Dhaka-1000, Bangladesh	erfan35-2610@diu.edu.bd	01913777345

© 2024 Copyright: 183-35-2610

## 5.2 Register & Login

To access admin panel user, have to register in this system with name, email & password. Then user can login the admin panel.



A registration form with four input fields: Name, Email, Password, and Confirm Password. Below the fields are a link for 'Already registered?' and a 'REGISTER' button.

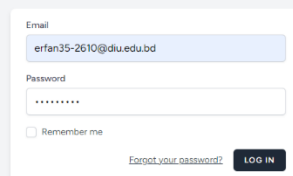
Name

Email

Password

Confirm Password

[Already registered?](#)



A login form with two input fields: Email and Password. Below the fields are a 'Remember me' checkbox, a link for 'Forgot your password?', and a 'LOG IN' button.

Email

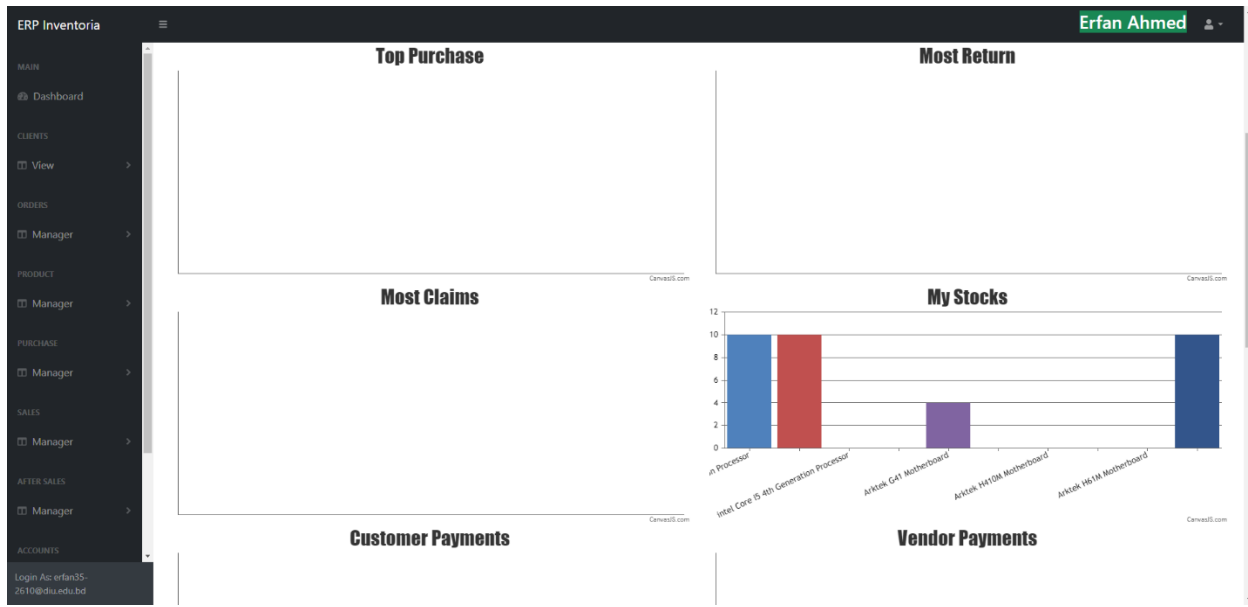
Password

Remember me

[Forgot your password?](#)

### 5.3 Dashboard

Here user can see the monthly business analytics such as profit loss, top selling, stock, due, claims etc.



## 5.4 Clients Area

Here user can manage vendor and customer information.

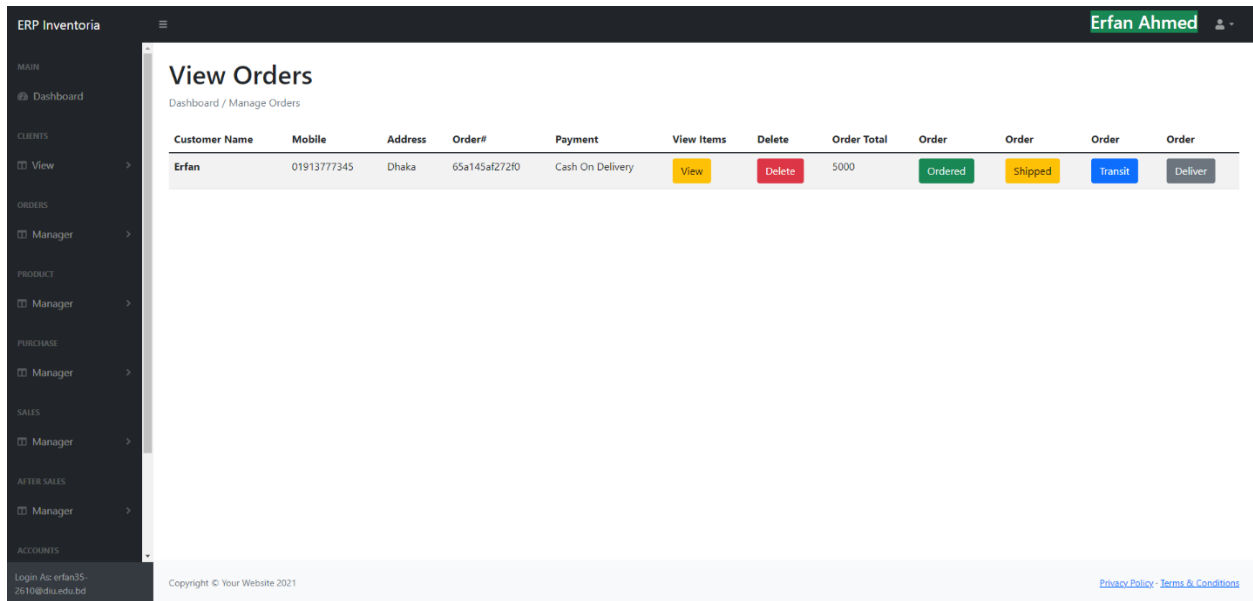
**View Vendor**  
Dashboard / Manage Vendor

Bill#	Mobile	Name	Address	Due
65a145d44177b	01913777345	Erfan	Dhaka	0

Copyright © Your Website 2021 [Privacy Policy](#) [Terms & Conditions](#)

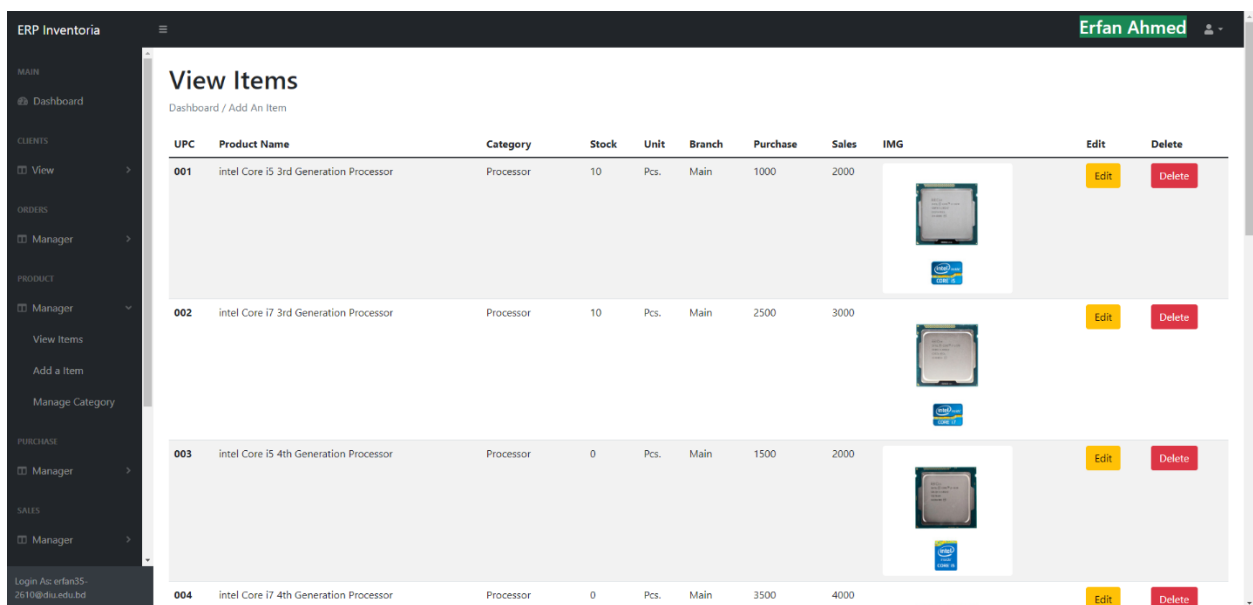
## 5.5 Orders Area

Here user can manage order and information and delete order and change status.



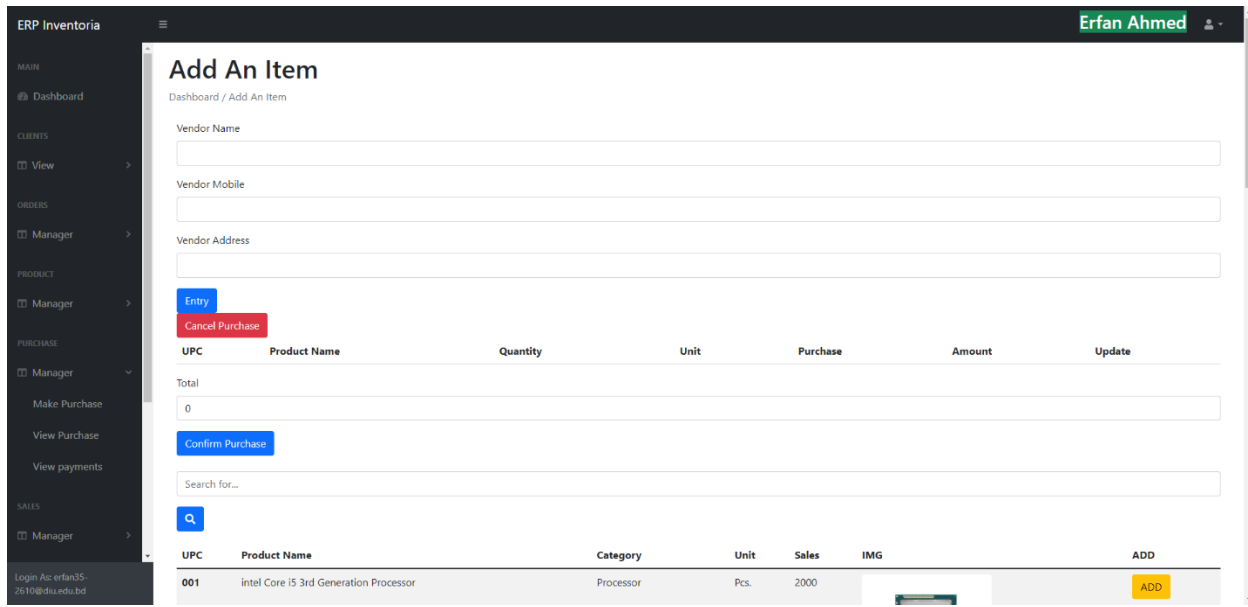
## 5.6 Product Area

Here user can manage product and information and upload, edit and delete order.



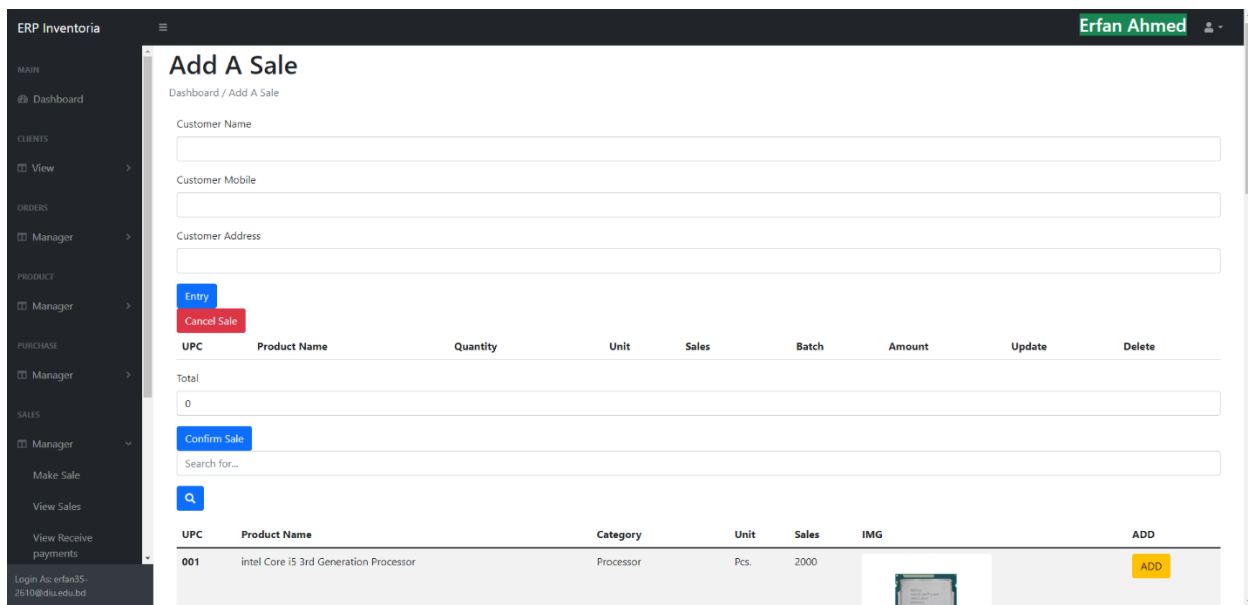
## 5.7 Purchase Area

Here user can manage purchase and information and pay, edit and delete purchase.



## 5.8 Sale Area

Here user can manage sale and information and pay, edit and delete sale.



## 5.9 After sales Area

Here user can manage refund and warranty claim information.

**View Items**  
Dashboard / Add An Item

Product Name	Mobile No	Invoice No	Batch No	Quantity	Price	Total	Delete
intel Core i5 3rd Generation Processor	01913777345	65a145d44177b	65a14395085f	1	2000	2000	<a href="#">Delete</a>

## 5.10 Journal Area

Here user can entry and manage journal.

**Add An Category**  
Dashboard / Add Journal

Date:

Account Name:

Account Type:

Amount:

Remarks:

[Submit](#)

## 5.11 Profit & Loss Report

Here user can see full detailed report by selecting dates.

ERP Inventoria Erfan Ahmed

**Sale wise Profit & Loss Report**  
Dashboard / Sale wise Profit & Loss Report

From:

To:

Bill #	Invoice #	Product Name	Purchase Price	Sales Price	Sale Quantity	Profit
65a143950f85f	65a145d44177b	intel Core i5 3rd Generation Processor	1000	2000	1	1000
65a143950f85f	65a145d44177b	Arktek G41 Motherboard	2800	3000	1	200

Invoice #	Product Name	Sales Price	Return Quantity	Return Amount
65a145d44177b	intel Core i5 3rd Generation Processor	2000	1	2000

Claim #	Product Name	Batch	Problem	Demurrage
2000CLAIM	Arktek G41 Motherboard	65a143950f85f	No Display	200

**Total Profit = BDT 0/=**

Copyright © Your Website 2021 [Privacy Policy](#) - [Terms & Conditions](#)

## 5.12 Journal wise Report

Here user can see full detailed report with journals by selecting dates.

ERP Inventoria Erfan Ahmed

**Sale wise Profit & Loss Report with Journal**  
Dashboard / Sale wise Profit & Loss Report with Journal

From:

To:

Bill #	Invoice #	Product Name	Purchase Price	Sales Price	Sale Quantity	Profit
65a143950f85f	65a145d44177b	intel Core i5 3rd Generation Processor	1000	2000	1	1000
65a143950f85f	65a145d44177b	Arktek G41 Motherboard	2800	3000	1	200

Invoice #	Product Name	Sales Price	Return Quantity	Return Amount
65a145d44177b	intel Core i5 3rd Generation Processor	2000	1	2000

Date	Type	Account Name	Amount	Remarks
12-01-2024	Expense	Store Rent	5000	January

Claim #	Product Name	Batch	Problem	Demurrage
2000CLAIM	Arktek G41 Motherboard	65a143950f85f	No Display	200

**Total Profit = BDT -5000/=**

# CHAPTER 6

## Conclusion

### 6.1 Project Summary

Ecommerce with Ecommerce with Inventory Management & Accounting is a business development application their user can maintain his item stocks, accounting and ecommerce website in a single system. User don't have to maintain different application for these three tasks. Moreover, user can maintain warranty and claims through this system. Customer can place order and track his orders and future he can claim warranty and track his claim. It is an web base application so that it can run over any devices like desktop pc, laptop and smart phones.

In just a brief period of time, I finished this project from planning to development, and the system is now operational. Regular updates to this system have to happen as the application moves forward.

### 6.2 The constraints

I'm still developing and in the early phases. I am reluctant to complete my system's performance for my project. There are a number of constraints. That's:

- A lot of features may still be developed for this. User role wise authorization is not implemented to this system
- Branch wise authorization is not implemented to this system
- Product serial wise sale is not implemented to this system

### 6.3 Obstacle and Gains

The process of creating this system was fraught numerous difficulties that I had to overcome in order to finish. Maybe obstacles help us establish our worth. A route to success is paved with challenges, obstacles, and the triumph of conquering obstacles and problems. Along with several challenges and barriers, I also encountered when I started working on this project. I'm using MySQL and Laravel, two completely unfamiliar languages, in my project. When I first started, I had some difficulty picking up the language. I'm having some trouble with CRUD operations. I'm having some issues connecting to the local server and database. I overcame all of these issues after four months of continuous labor, and my system has now gained maturity smoothly. Every component of my system functions flawlessly. My most significant achievement will be if the user finds the system easy to use.

## Final Year Project Erfan V3

---

### ORIGINALITY REPORT

---

**4%**  
SIMILARITY INDEX

**2%**  
INTERNET SOURCES

**0%**  
PUBLICATIONS

**4%**  
STUDENT PAPERS

---

### PRIMARY SOURCES

---

**1** Submitted to Staffordshire University  
Student Paper 1%

---

**2** Submitted to Segi University College  
Student Paper 1%

---

**3** [dspace.daffodilvarsity.edu.bd:8080](https://dspace.daffodilvarsity.edu.bd:8080)  
Internet Source <1%

---

**4** Submitted to Midlands State University  
Student Paper <1%

---

**5** Submitted to Kuala Lumpur Infrastructure  
University College  
Student Paper <1%

---

**6** Submitted to Laureate Education Inc.  
Student Paper <1%

---

**7** Submitted to University of Lagos  
Student Paper <1%

---

**8** Submitted to Asia Pacific University College of  
Technology and Innovation (UCTI)  
Student Paper <1%

---

**9** Submitted to National Economics University

Student Paper

<1%

---

10

Submitted to Sheffield Hallam University

Student Paper

<1%

---

11

www.coursehero.com

Internet Source

<1%

---

12

Submitted to Birkbeck College

Student Paper

<1%

---

13

Submitted to i-CATS University College

Student Paper

<1%

---

14

www.technologyuk.net

Internet Source

<1%

---

Exclude quotes Off

Exclude matches Off

Exclude bibliography Off