GROCERY GALLERIA

Software Requirement Specification

Group CS 24

Date of Submission

21.09.2021

Table of Contents

Details of Project Supervisor, Co-supervisor, Advisors, and Clients

Proposed Project Supervisor (Academic Staff of UCSC):

Name of the supervisor: Dr. B H R Pushpananda

Signature of the supervisor:

Date: 21.09.2021

Proposed Project Co-Supervisor (Assigned by Course Coordinator):

Name of the co-supervisor: Miss. Amali Perera

Signature of the supervisor:

Date: 21.09.2021

List of Figures

List of Tables

Acronyms and Abbreviations

PC – Personal Computer

AWS – Amazon Web Services

GCP – Google Cloud Platform

PWA – Progressive Web App

ISP – Internet Service Provider

HTTPS – Secure Hypertext Transport Protocol

1 Introduction to Project.

1.1 Domain Description

After analyzing the essential services sector, we have recognized several existing problems and their impacts:

During the covid-19 pandemic situation, we all have seen that almost all the people had to face vast difficulties when finding day-to-day needs like food because of the lack of online food ordering platform availability in Sri Lanka. Although the supermarkets had the online ordering facility available, it is not capable of handling all the food delivery needs in Sri Lanka. And even the large-scale entrepreneurs like Supermarkets had assets to build or rebuild an entire online ordering platform to function continuously, most of the small-scale food entrepreneurs don't have that amount of wealth to spend for that kind of a system and so that these types of businesses became unprofitable and unsuccessful with this situation.

On the other hand, people had to wait so long without getting their essential needs efficiently in time because of the above-mentioned issues.

Based on the above-identified issues we discovered that the following outcomes are of necessity:

To save both the people and the small-scale entrepreneurs from the above-mentioned difficulties, a Centralized Online Grocery Ordering Platform shall be built for the usage of the general public and such enterprises. Which mostly sell essential items like vegetables, Fruits, Groceries, Fish & Meat. In simple terms, our teams' fundamental target is to create one single platform which makes available our nearest regional such small-scale shops where the customers can order their needs through this platform online and get deliver it to their doorstep.

Our platform "Grocery Galleria" has several new special features which are not existing in current related e-commerce platforms. And they are,

1. A customer can find all the essential item needs under one centralized online platform.
2. A customer can buy items online from their familiar regional shops.
3. A customer can order all the items that they need even from several shops within their region in one order. (Maximum up to 4)
4. Even a customer's shopping cart contains several shops; they can pay in one go for all the shops including the delivery fee.

5. A customer gets his/her item package in one delivery even he/she has ordered the items via several shops.

There are five major service user roles in this system except for the *Customer* user role. They are *Shop, Delivery Staff, Delivery Rider,* and the *System Staff*.

Shop

To sell items through this platform, a shop has to register first by providing the required information for the system., Once the provided information has been approved by the system staff, A shop can start to add items to their shop space provided by the system and start selling items to their shop region customers. So that shop will be provided with a shop dashboard to handle the orders along with the features like item stock updating, item price handling, etc. And also, they will receive an email to refer for the orders in case of power failures of the shop.

Delivery Staff

In this system, the delivery staff is the ones who are responsible for the order delivery handling, so for that, they have the delivery handling dashboard. By using that dashboard, they could easily manipulate the Upcoming, Ongoing, and Past orders. Whenever there is a new order, they can select that order if it should be delivered today according to the order timespan guidelines given to them.

Once they select an order for the delivery, the delivery dashboard will provide the optimum shop line-up that they should go to arrive at the customer location to minimize the cost. then they could manually assign a delivery rider for that order based on the reachability to the first shop based on the following criteria,

- Choose Bicycle or Three-wheel rider based on Order weight and items included in the order.
- Assign a rider based on availability status(online) and the rider location provided by the delivery dashboard.

Delivery Rider

A delivery rider can see the assigned order for them by using the simple mobile app which we are providing for them. And also, they have been provided with the optimum shop line-up that they should go to arrive at the customer location to minimize the cost. And also, with the several other features like Mark Order completion, etc.

System Staff

In this system, the system staff has the granted privileges to the system to manage the system efficiently. Staff will provide the permissions requested by the system and they have the granted permissions to take actions like remove the shop from the system, etc. And they are the ones who are responsible for the Shop registrations, Complaint handling, Report generation, etc

1.2 Current system & It's limitations

- Example flow of an existing similar system.

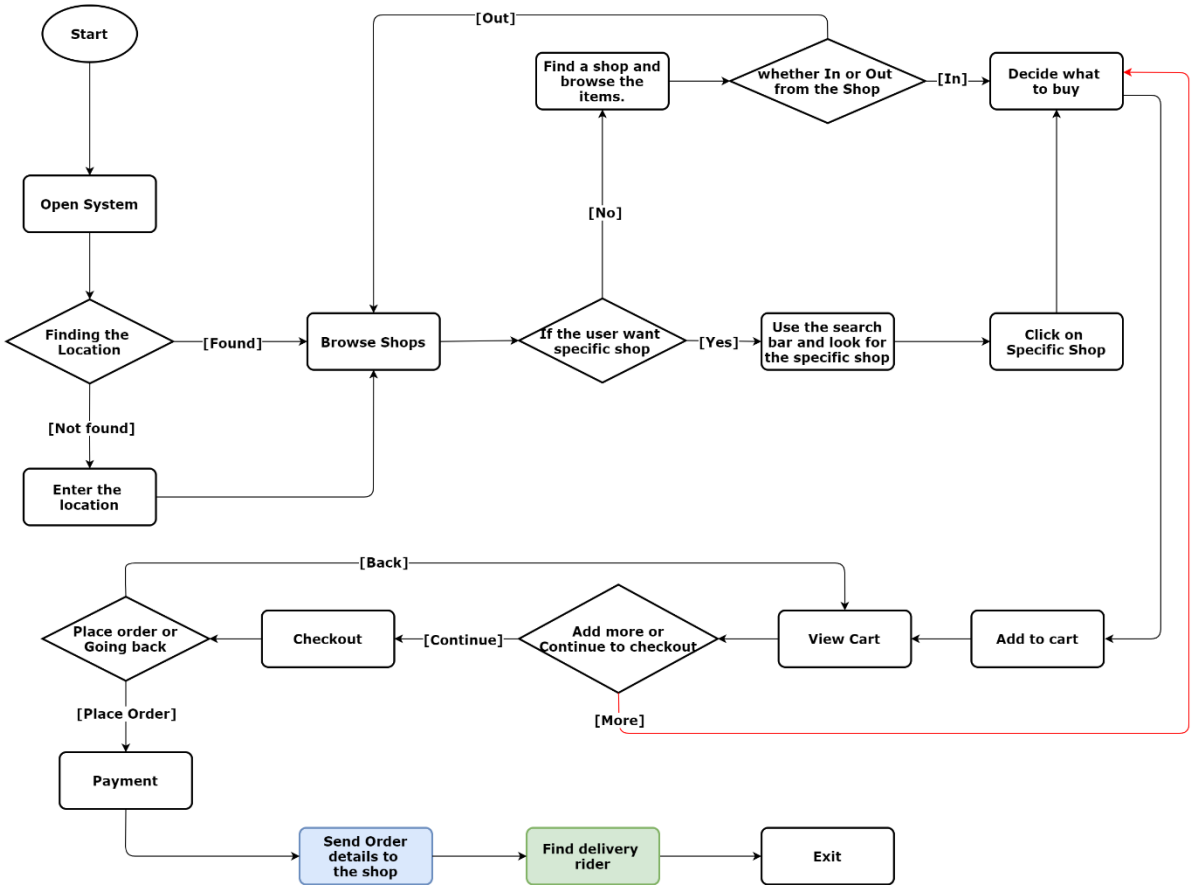


Figure 1: Example flow of existing similar systems.

- Above *Figure 1* provides the sample flow of currently existing systems.
- As from our system point of view, the main constraint of this process flow is that the customer can only purchase items from one particular shop in one order. Which is highlighted in red.
- So that, if the customer needs to buy items from different shops, they have to buy them using multiple orders. Thus, making the process inconvenient and expensive.

- The flow of the proposed system

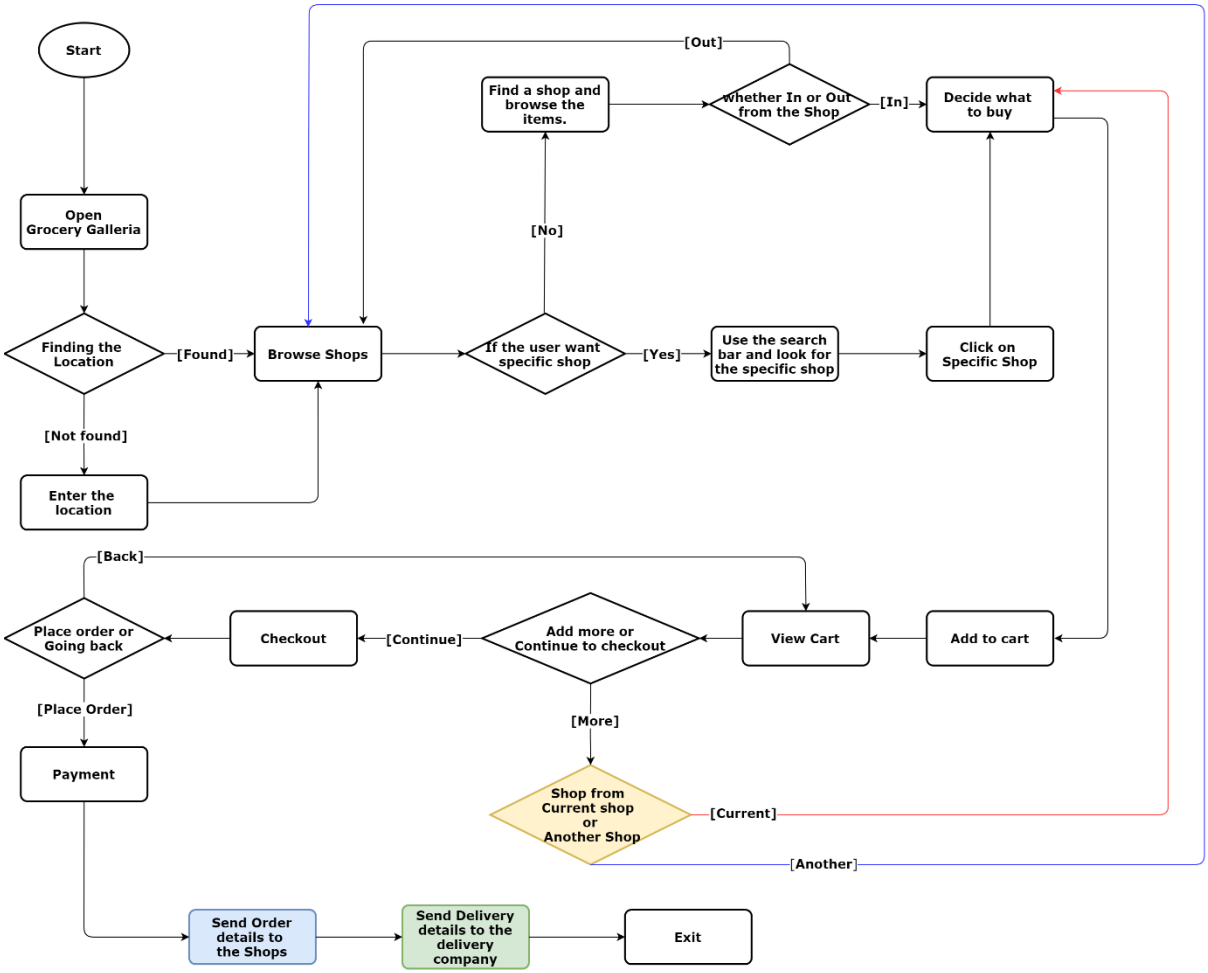


Figure 2: Flow of the proposed system

- Above *Figure 2* provides the new proposed flow from our system.
- The highlighted blue colored path shows the new proposed flow path to resolve the above *Figure 1* main constraint.
- By using the above flow, a customer can order items even from several shops in one order.
- And also we have changed the delivery method which is using in *Figure 1* flow, to cater to the new requirement. So that our system maintains delivery riders per suburb area, where the assigned delivery rider will collect the items from the shops and deliver them to the customer in one go.

1.3 Objectives & Goals

Our team goal is to build a user convenient online shopping platform for the entire small up to medium scale enterprises and for the general public to establish an uninterrupted trade connection between customers and such enterprises in any kind of situation in the country.

So that, we have defined the following objectives to achieve during the development of our system.

- To build the UI design simply and straightforwardly to make it easily understandable for their level of perception.
- To make a fully functional deployable robust product within the given time span.
- Test and debug the system in each development phase to improve the robustness of the system.
- To use software good practices when developing the system to improve the efficiency and throughput of the system.

1.4 Assumptions, Constraints & Limitations

1.4.1 Assumptions and Constraints

Customer

Constraints

- A new customer has to first register inside the system to get access to the "Grocery Galleria" website.
- Customers can only purchase items from the shops that are within the customers' region.
- Customers can only purchase items from four shops maximum in a single order.
- The customer cannot update the shopping cart, once the customer has proceeded to the checkout window.
- The customer cannot cancel the order once the order has been placed.
- Customers cannot search for an item from several shops simultaneously.
- There will be a maximum weight per order,50KG.

Assumptions

- Customer details provided by the customer are accurate. (E.g.: Contact details, Address, etc.)

Shop

Constraints

- The shop has to be eligible to be registered in the system. System staff will check registrations and verify the authenticity of the details that are submitted by the shop owner. Criteria will be,
    - Shop size: Enough to handle both online (the system) and offline (on-premise) sales.
    - Location: System will be deployed to cater to customers within a specified region, and these shops should also be within the region to be registered.
    - Shop type: Meat/Fish, Grocery Items, Vegetables. The shop owner will guarantee that shop has the items that it has been listed.

- Shop owners can sell the items via the system only in the suburban area.

- The shop owner will agree to remove an item from the system if the shop is running low on inventory.
- Shop Owner cannot sell higher than MRP if there the product has an MRP.

Assumptions

- The shop owner is aware of how much stock is currently available in the inventory and when it's low he/she will remove the item from the system as soon as possible.

Delivery Rider

Constraints

- A delivery rider has to deliver every order that he has assigned during the day.
- A delivery rider should have a smartphone with a power bank with him.
- The delivery vehicle is chosen depending on the weight of the order.

Assumptions

- Delivery riders have the required technical knowledge to handle an order through the material we provide. (System app, Google Maps, etc)

Delivery Staff

Constraints

- Delivery staff can only access the delivery-related information through the dashboard provided by the system.
- The delivery staff has to assign the riders to the orders based on the order delivery time-span guidelines provided by the system.

Assumptions

- None

System Staffs

Constraints

- None

Assumptions

- System staff will thoroughly examine the quality of the shops and the delivery riders to ensure systems quality of service.

1.4.2 Limitations

In Scope

- Order items through the web platform.
- User profile module.
- Manage to customize the order via different shop combinations.
- Payment through an online payment gateway.
- Shop module for item listing and order details handling.
- Delivery module for accepting and assign riders.
- Delivery Mobile App for riders.
- Weekly report generation
- Inventory handling system for Shops
- Real-time delivery tracking

Out of Scope

- Complete Shop Inventory System.
- Complete Delivery Management Module

2 Feasibility Study

The feasibility study for the system is done to gather the facts about the practicality of creating such a system. This section will help to better understand the product. The study is divided into five primary subtopics.

1) Technical Feasibility
2) Economic Feasibility
3) Legal and Ethical Feasibility
4) Operational Feasibility
5) Schedule Feasibility

2.1 Technical Feasibility

In this section, we measure the technical feasibility of our system based on users' perspectives as well as from the system perspective.

From the users' perspective, there will be several key features will be provided for the users to uplift the technical feasibility of the system.

“Grocery Galleria” is a web-based platform. This shall be implemented using HTML, CSS, and JavaScript as the frontend and the backend shall implement in PHP and SQL. The system shall prioritize mobile users over desktop users in most of the interfaces.

The system shall be deployed in a cloud provider with a service level agreement that shall handle the systems bandwidth load and processing power. There will be significant investment upfront to get the infrastructure running but once the system is deployed and scaled to multiple regions the costs of infrastructure shall be divided into the regions.

To demonstrate the payment, process a sandbox will be used; however, once the system is commercially deployed, the developers would provide the necessary funds to purchase a payment portal.

The development teams shall use freely available IDEs and Tools to develop the application throughout the whole time.

2.2 Economic Feasibility

In this section, we will be describing the Economic feasibility of the system from the view of the System perspective and as well as from the business model that we will be used.

2.2.1 Business model

Grocery galleria will be a Multisided Platform Business Model-based platform that connects three sides: end-customers, shop-owners, and delivery riders. And the business model between those user groups is as follows.

- Grocery-Galleria and Customer:- B2C
- Grocery-Galleria and Shop-Owner:- B2C-online intermediaries
- Grocery-Galleria and Delivery-riders:- B2C- online intermediaries

Grocery Galleria has 2 different revenue streams:

Commission on orders: As the B2C-online intermediaries model, we will be taking a 3% cut from every shop order placed through the site as the revenue from the shop to the grocery galleria.

Delivery fees: Delivery cost will be a minimum of 120 LKR per order and a maximum of 160 LKR will be charged from the customer. And 20 LKR will be taken from each delivery as the delivery revenue to the grocery galleria.

2.2.1.1 Operating Costs

2.2.1.1.1 Personal cost

There will be no additional personal cost regarding the system as the system doesn't need specialized training to use the system. The development team shall ensure the system have an adequate level of usability among all stakeholders.

2.2.1.1.2 Cost for payment methods

The payment gateway costs LKR 39.00 plus 3.9% of the transferred amount of each transaction. This is something that needs to be considered because some orders can be cheap enough that Rs.39.00 + 3.9% of that order is a significant deduction from the profit. So, LKR 39.00 from each transaction with another additional charge will be an effect directly the profit.

2.2.1.2 Operating Benefits

2.2.1.2.1 Reduced personal cost

By using this system, people can reduce the manual cost of shopping like traveling cost, physical waste of the body, etc. since the people can order items online

And also, the shop owners can sell items online without worrying about the costs like calculating the total, customer handling, and physical waste of the body. And they don't need to invest in a delivery team or system. That means by deploying this system shall reduce the personal cost of the main stakeholders of the system.

2.2.1.2.2 Increase the revenue

This system shall increase the revenue of the shop owners. since they can sell items through this online platform it will increase the customer audience. And also, since people tend to start buying from small-scale shops, it will increase the no of deliveries within a town and so that it will increase the revenue of the typical delivery rider as well.

2.2.1.2.3 Customer Satisfaction

By using this system, the customers can purchase items from several shops available within reach of a click as well in a single order. This is an introduced new feature for these types of platforms so that It increases customer satisfaction certainly. And also, the customer can buy items directly from the nearest well-known shop using this system.

In this system, the customers can make complaints about the system services which is highly valuable to increase the customer's satisfaction.

2.2.2 System Costs

2.2.2.1 One Time Costs

2.2.2.1.1 Hardware and Software cost

Customers and Shops can use existing personal smartphones to access the system. Shops can use a PC to access a desktop version of the system for convenience. If the shop hasn't got a PC there will be an initial onetime cost for the shop owners.

2.2.2.1.2 Development team cost

As the development team cost, we will not have any software tools cost since we use free and open-source tools for the development and testing purposes. Although we will have the costs like hardware wastage of our personal computers, monthly internet cost, etc as personal costs.

2.2.2.2 Operational Costs

2.2.2.2.1 Utility costs

Once it has been deployed after the implementation there will be costs for electricity and Telecom ISP charges due to the system needs to be up and running 24/7.

2.2.2.2.2 Hosting Costs

To make a reliable service, the system shall be deployed in a popular cloud platform that has a solid service level agreement. The team has researched the idea and found out that, when the system is deployed in the AWS Singapore region (for lower latency and bandwidth costs) there is approx. 38000LKR(171USD)[1] per month cost for hosting and for bandwidth usage. This can be further reduced by using caching mechanisms for reducing the bandwidth costs.

2.2.3 System Benefits

2.2.3.1 Quantitative benefits

2.2.3.1.1 Uses of free software

In this system, we will be using freeware software bundles as our development tools mostly. So that we can reduce the personal cost and improve the robustness of the system since they are errorless in nature because of the collaborative development community available.

2.2.3.1.2 Generate Reports

By using this system, the shop owners and delivery companies can keep track of the business income that they have gained through this system by getting Weekly, Monthly & Annual income reports. The system staff shall generate those reports and send that to the relevant stakeholders.

2.2.3.1.3 System managements

The system management shall be mainly carried out by the system staff and they can manage all the other stakeholders' information through the system. However, the other stakeholders can manage their responsible part of the system management using the system without system staff.

[1] https://calculator.aws/#/estimate?id=322747e2bb04c9415462a5a36c017ee1c91ecdb6

2.3 Ethical and Legal Feasibility

- The system guarantees that the system user information will be secure inside the system and will not be granted to any third parties for any other purposes.
- Any occurrence of substantial changes inside the system should be notified to the relevant user of the system.
- The system users should be responsible for the information that he/she is preserving inside the system. (Ex: Shop owner's item pricing)
- The delivery riders should adhere to the delivery agreement provided by the system.
- The system privileges will be granted to the system staff to manage the system efficiently.
- Verified and Approved payment gateways will be used for transactions.
- The system will be designed with best practices to ensure the security of the system.
- The basic revenue analysis reports shall be generated by the system staff and relevant details shall be sent to the relevant other users of the system.
- Proper documentation and coding practices shall be maintained during the development.
- Shop owners and Delivery riders' eligibility shall be analyzed by the system staff before finalizing the registration.
- The whole system will be developed using free, and open-source resources, and any external code plugging that we might have to use will be acknowledged and give credits to its rightful owner.
- According to the Electronic Transactions, Act, No. 19 of 2006(as amended) (the 'ETA'), Currently the system does not break any rules and regulations given by the act under the law of Sri Lanka. And the system shall obligate the guidelines given by the ETA act in Sri Lanka in future development.[2]

[2] https://www.dailymirror.lk/features/E-commerce-in-Sri-Lanka-A-legal-overview/185-189694

2.4 Operational Feasibility

In our system, the final end-users will be Customers, Shop Owner, Delivery Staff, Delivery riders, and System staff.

Development-wise, the team members are knowledgeable on how the different technologies should be used to make this system fully functional and the team members are logistically planning each development phase according to the given timespan. System maintenance will be handled by the team members once it's deployed.

To make this system operationally feasible in the real world, we have to sustain the user convenience of the system with a strong business model that makes fairtrade for all the users of the system including system staff. So that, we will provide the following features and guidelines for the users to make this system operationally feasible.

- Customer queue will be maintained based on the first-in-first-out mechanism to function the system without slowdowns or crashes under high-web traffic during peak times. So that we handle the server load by accepting only a limited no of customers inside the system for some time.
- So that, Customer login session will be maintained based on a timespan scheduled by the system to improve the system throughput as well as the productivity of the system.
- Customers will be able to order all the essential needs from their familiar shops located in their house region and pay for that order online.
- Customers will get the order package to their doorstep within 24hours maximum via the delivery service of our system.
- Real-time stock updating will be maintained so that the customers will only see the available items from the shops during the moment of he/she accessing the system. And the site will be refreshed during the period of the customer session to update the available stocks for each customer.
- In case of sudden item unavailability, when after the order has been placed, the customer will receive the refund amount to their bank account which was used to place the order within a maximum of two weeks.
- Shop owner only has to process the upcoming orders based on the order process time guidelines and other information given to them by our system and handover the package to the deliver rider coming from our system.
- Shop owners will receive an email to refer for the orders in case of power failures of the shop.

- Shop owners can manage the item stock by using the shop dashboard provided by the system.
- Delivery rider only has to collect the item packages from the shops in an order that he has assigned into and deliver it to the customer. Once they've done an order they will deliver another and so on. So that they have been provided with a delivery app that mainly gives the optimum shop line-up that they should go to arrive at the customer location for each order to minimize the cost. And also with the several other features like Mark Order completion, etc.
- Only Bicycle and Three-wheel owners will be registered as delivery riders and they will have an ice-box with their vehicle to deliver the frozen items fresh to the customer. Ice-box will be provided to them for a reasonable price on registration.
- There will be system hotlines that the users can contact in case of inquiry about system services and make complaints.
- System staff has the granted privileges to take the actions about the shops and delivery riders based on their activity inside the system to maintain the quality of the service provided by the system.
- Shop owners and delivery riders will receive the revenue to their account on weekly basis.
- System staff and the delivery staff will be permanent staff of our system and they will receive their salary on monthly basis.

In a conclusion, we can say our system would be operationally feasible in the real world according to the above operational features given.

Once it has been deployed to the public, soon after the implementation. There will be some time for users to adapt to the system and there will be minor changes while. So that to minimize that issue, the development team will continuously be getting feedback from the users during the development as well as after the deployment to uplift the system efficiency as much as we could.

2.5 Schedule Feasibility

"Grocery Galleria" intends to be completed within a one-year time frame. Because of the stable requirements, the system is more suitable to develop using the waterfall development methodology. The grant chart of the proposed timeline is attached on page **Error! Bookmark not defined.**.

We are using project management tools like GitHub to schedule and manage our works more organized. And considering the current context, we are already halfway through the process at the moment. Therefore, it is possible to finish the development of the "Grocery Galleria" meeting end of the time frame.

3 Requirements.

3.1 Stakeholder Identification (Actors)

System stakeholder identification is as follows,

- Customers:- The primary user of the system.
- Shop owners:- The core service provider to the system.
- Delivery Staff:- Handling the delivery process of the system.
- Delivery Rider:- Handle the orders and delivers them to the user customer.
- System Staff:- Handling the overall process of the system.

3.2 Use Case Diagram

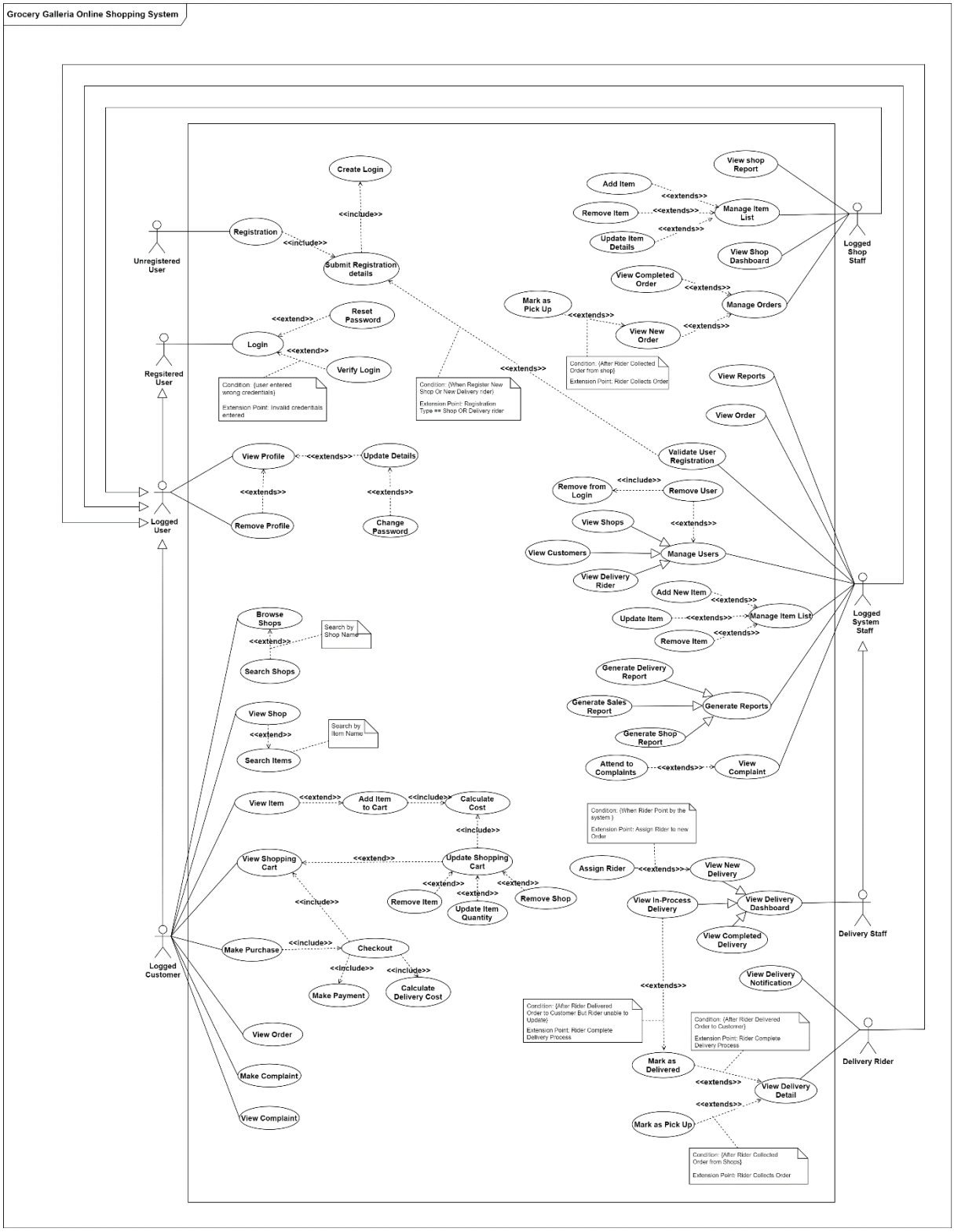


Figure 3: Use Case Diagram

3.3 Functional requirements of the project

Refer to the Appendix for the use-case narratives.

3.4 Quality Attributes

Following Quality Attributes are found out to ensure the system meets the required excellence. In each subtopic, we will go through them one by one concisely.

3.4.1 Availability

The system shall be available 24/7. By using a reputed cloud service provider like AWS[3], or GCP[4] they provide an uptime of >95% monthly. For a cost of roughly Rs.40000[5]. Going with a cloud provider shall give the system more stable uptime and, in the future, we can also scale rapidly. In summary to ensure system availability,

- Using a cloud provider for infrastructure.
- The system shall log the activities, in case of omission or crash of the system, the administrator of the system will be informed via email.
- The system shall be reliable. It shall log the errors and recover from faults.
- Daily database backups shall ensure the system is recoverable in an event of failure.
- The system displays a UI if the system is down for maintenance.
- The system will be operational only during business hours (9 AM – 9 PM). Any other time system will be accessible but a UI will be shown to the Customer that the system will be available during business hours.
- Items are served first-come, first-serve basis. If one customer pays first then the other, the Item will be given to the first customer and the second customer will be notified and reimbursed if necessary.

3.4.2 Modifiability

The system shall be developed component-wise, such as the logging module and the orders module. Ensuring each module depends on the minimal dependency of another module, thus archiving a modifiable system. Each component will give out interfaces that can be interconnected with each other making the inner changes of such components will not affect the whole system.

[3] https://aws.amazon.com/compute/sla/
[4] https://cloud.google.com/compute/sla
[5] Approx. 170USD

The system shall be using configuration files to get the credentials and details that are needed to run the system in one centralized place (ex: Config/ directory), making the system easy to configure.

Developers will be using the database connector PDO. In a change of RDBMS, minor changes in queries and code will suffice[6]. As for interfaces developers shall use architecture like MVC. Thus, making the interfaces act independently of the system making it more modifiable[7].

In summary,

- The system shall be developed similarly to MVC architecture. Each component shall have minimal dependency on the other.
- Configuration files are kept in a specified directory. Making changes easier.
- Unified code conventions shall be used throughout.
- As an RDBMS connector, PDO shall be used as changes are minimal when migrating to a different RDBMS.
- PHP/MySQL is a widely used language/RDBMS in both Windows and Linux Operating Systems. Changing the runtime environment shall not be a problem.
- Code is commented on and given self-explanatory variables and function names to easy understanding of the code structure.

3.4.3 Performance

To ensure the system is performed under acceptable terms under performance following,

- Implementing a Customer Queue to keep an optimal number of customers within the system to make sure a better service is given to them.
- The system shall be stress tested for high traffic to the system. By using PWA technology lot of static content can be cached. Making fewer HTTP requests to the server[8].

[6] https://code.tutsplus.com/tutorials/pdo-vs-mysqli-which-should-you-use--net-24059
[7] https://www.freecodecamp.org/news/the-model-view-controller-pattern-mvc-architecture-and-frameworks-explained/
[8] https://developers.google.com/web/ilt/pwa/caching-files-with-service-worker

3.4.4 Security

In the system, passwords will be not stored plaintext. Minimum required number of details shall be collected from each user to ensure in case of a data breach, not all personal information of a user is given out. (Ex: Birthday of a user is not required to system functionality; this will be not stored by the system). Also, the system will provide suggestions for strong password practices to the user.

All the critical activities such as creating a user, deleting a user will be logged out by the system with timestamps. And such activities shall be done by the authorized personnel only. The system will use Role-Based Authentication to control which components it has access to.

The system will utilize a payment gateway for handling payments. This will ensure that payments are handled securely and such sensitive details leave no trace in the system itself.

In summary,

- Passwords are not stored in plaintext. Developers shall take acceptable measures to store such details.
- Critical system events will be logged.
- Databases will be backed up between reasonable periods to ensure the system can be recovered after an incident.
- A payment gateway is used to securely handle customer payments. This external component shall ensure that sensitive details such as credit card numbers are not stored with the system.
- Role-based authentication to control users' privileges to access the system components.

3.4.5 Usability

The system is web-based and responsive. The system shall be developed in mobile-first driven design. System UI shall be designed similar to existing popular online shopping websites to ensure smooth customer onboarding.

- The system will provide hints for users if the user is stuck during a process. Such as tooltips.
- UIs will be clean and less cluttered and shall follow acceptable UX design principles.
- Consistency of the UIs shall be maintained throughout the system

4 Proposed System's Architecture

4.1 Derived System Architecture

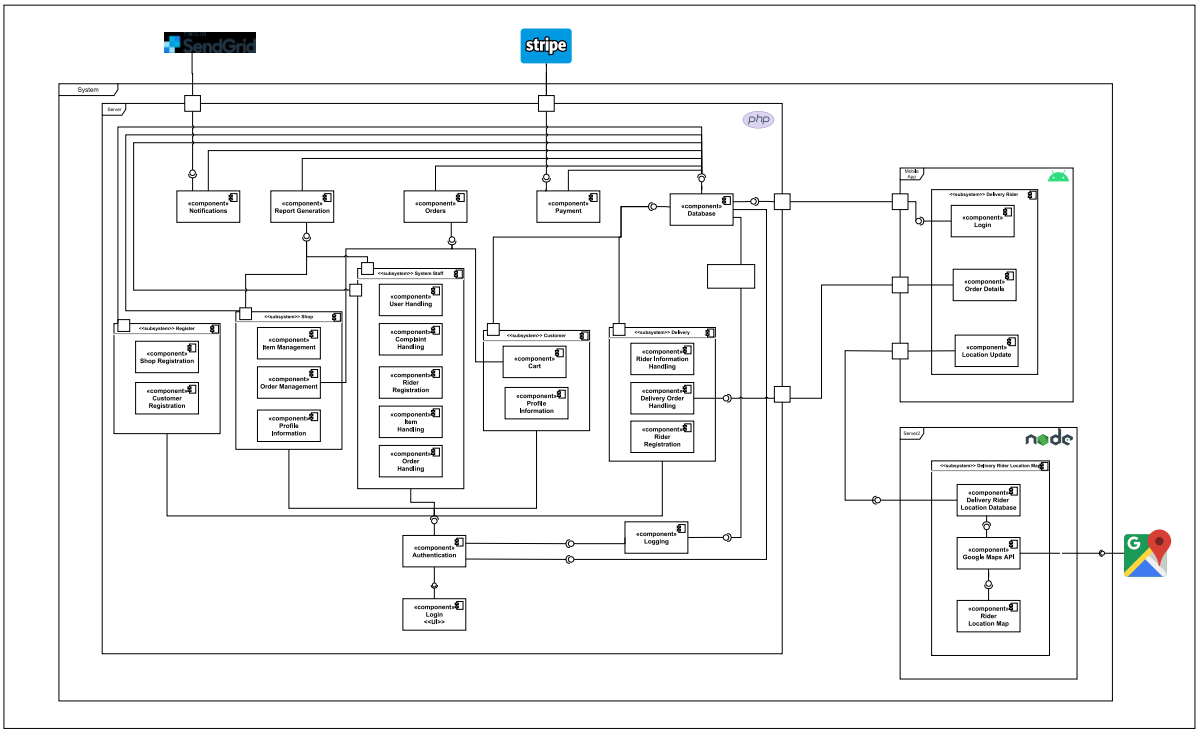


Figure 4: Component Diagram

A summary about the above components are as follows,

- Customer, Shop, Delivery, System Staff Components: UI interfaces are given by the system to respective stakeholders.
- Register Component: Used to register customers and shops. Customers are given access to the system after verifying their email. Shops are verified by the staff before giving access to the system.
    - Shop Registration Module: Gets the shop details from the shop staff
    - Customer Registration Module: Gets the customer details from the customer.
- Login Component: UI to get the login details from the user. This module also validates the given input.
- Authentication Component: Verifies the given details that are entered into the system to the user then authorizes them based on the role.
- System Staff subsystem: Modules for the system staff.
    - Report Generation Component: Used in generating reports weekly and monthly.
    - Complaints Component: Used in handling stakeholder issues within the system. Used solely by system staff.

    - Rider Registration: Acquire details from the riders when interviewing them for recruitment. And add suitable candidates.
    - Item Handling: Adding items to systemwide usage.
    - Order Handling: Ability to view order details and update them.
- Customer Subsystem
    - Cart: Temporary stores items in the while customer places the order.
    - Profile information: View and Update profile information. Also, customers can delete their profiles there.
- Delivery Subsystem
    - Rider information handling: Update Rider information
    - Delivery Order Handling: Assigning Riders to Orders. View completed and pending order details.
    - Rider Registration: Add riders to the system.
- Delivery Rider Subsystem (Mobile Application)
    - Login Component: Login using the credentials that given to the Staff of the system.
    - Order Details: Get and display order details querying the server.
    - Location Update: Send periodically updates to the location update server.
- Rider Location Map:
    - Delivery Rider Location Database: Real-time database for keeping track of updates of the rider location.
    - Google Maps API: Sends and gathers information to display rider locations and renders a live map.
    - Rider Location Map: Communicate with the Google Map API to render the live map of the customer.
- Notification Component: Module to send the notifications to the stakeholder via email or SMS.
- Payment Component: To communicate with the payment gateway through the given API.
- Orders Component: Record current and past orders done by the system.
- Logging Component: Store critical system events for non-repudiation purposes. Uses MySQL Binary logging and for PHP application PSR-3 Complaint logging module.

Refer to appendix for Use Case narratives (*page 7474*) and Activity Diagrams (*page **Error! Bookmark not defined.***).

5 System's Design

5.1 Class diagrams

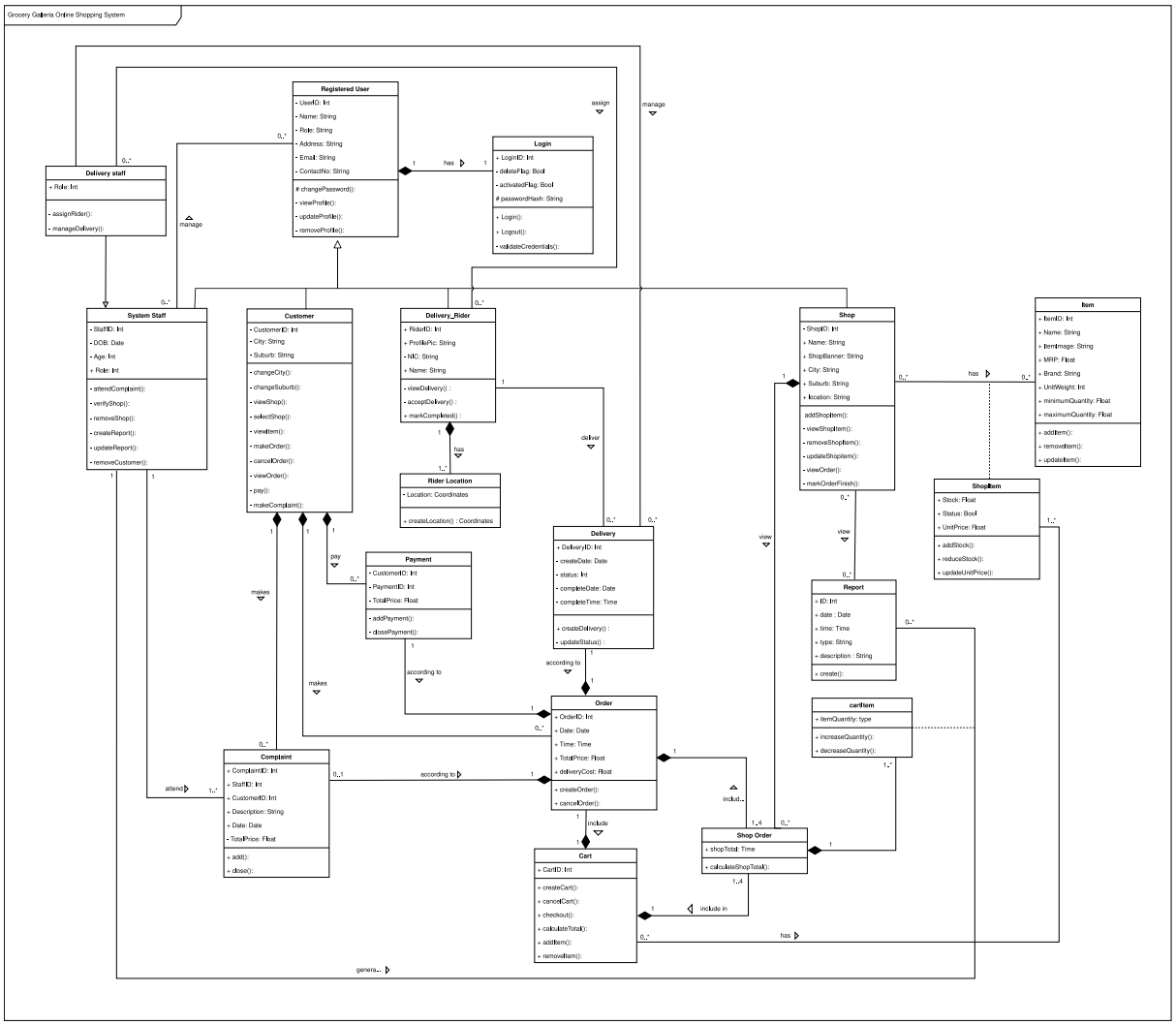


Figure 5: Class Diagram

5.2 ER Diagram

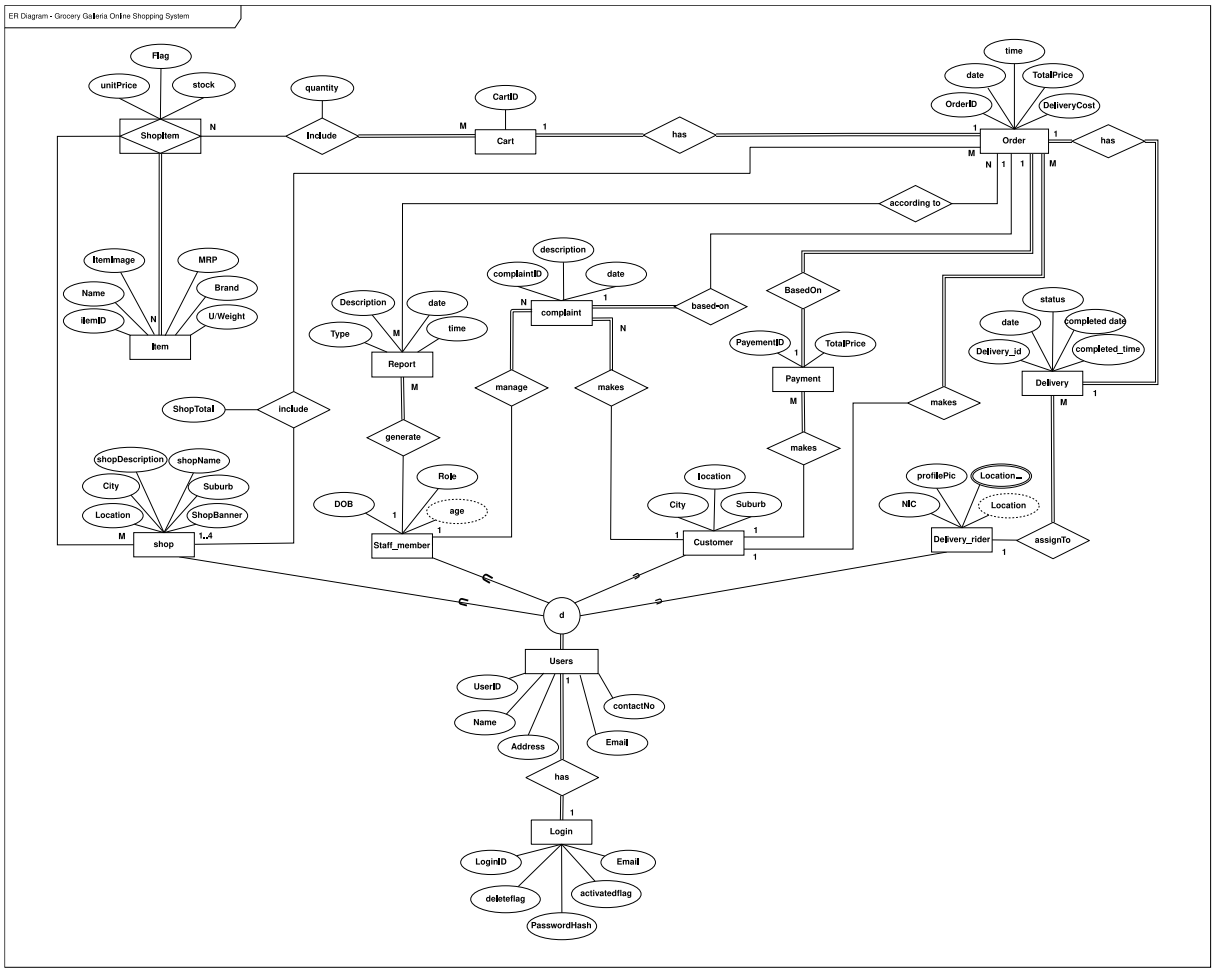


Figure 6: ER diagram

5.3 Sequence diagrams

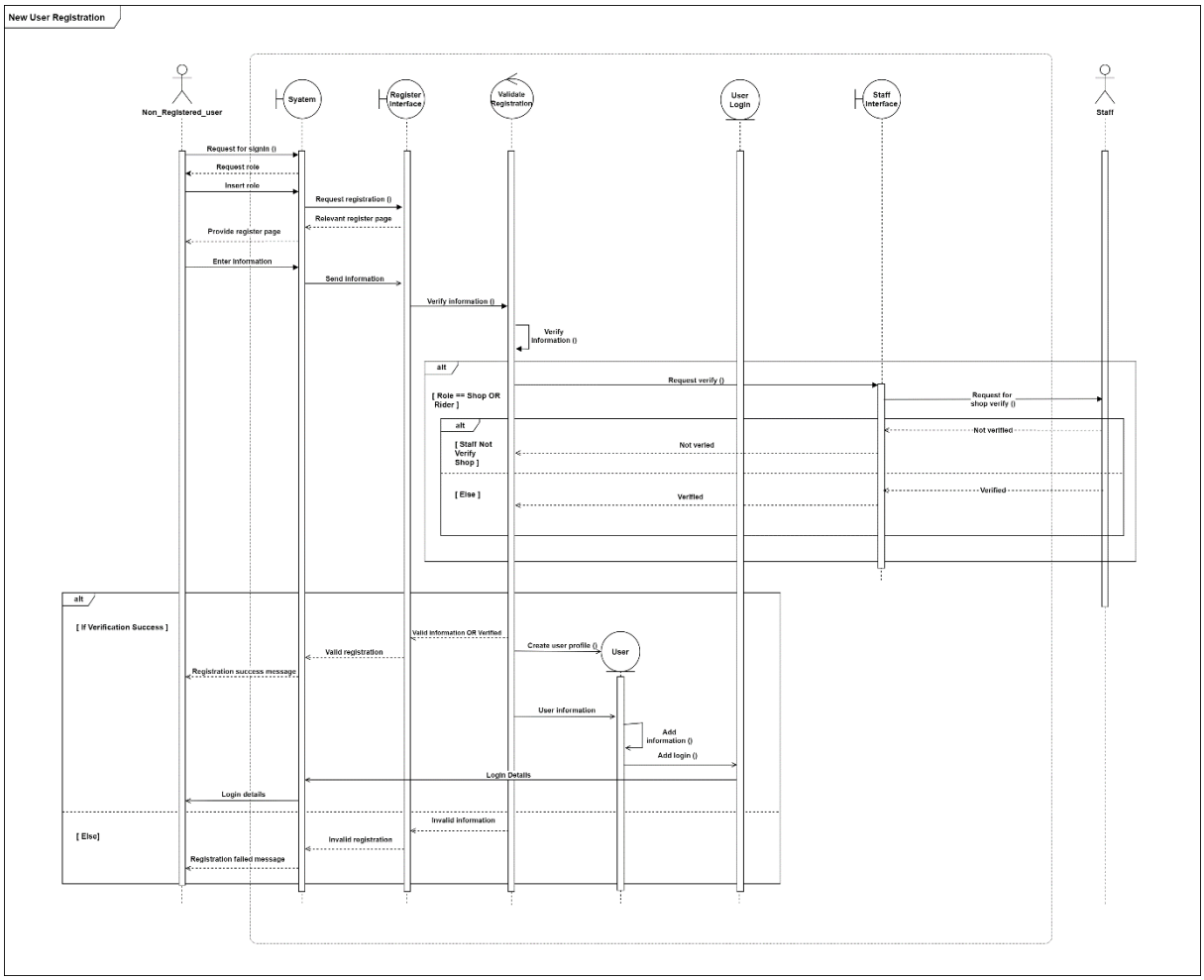


Figure 7: New User Registration

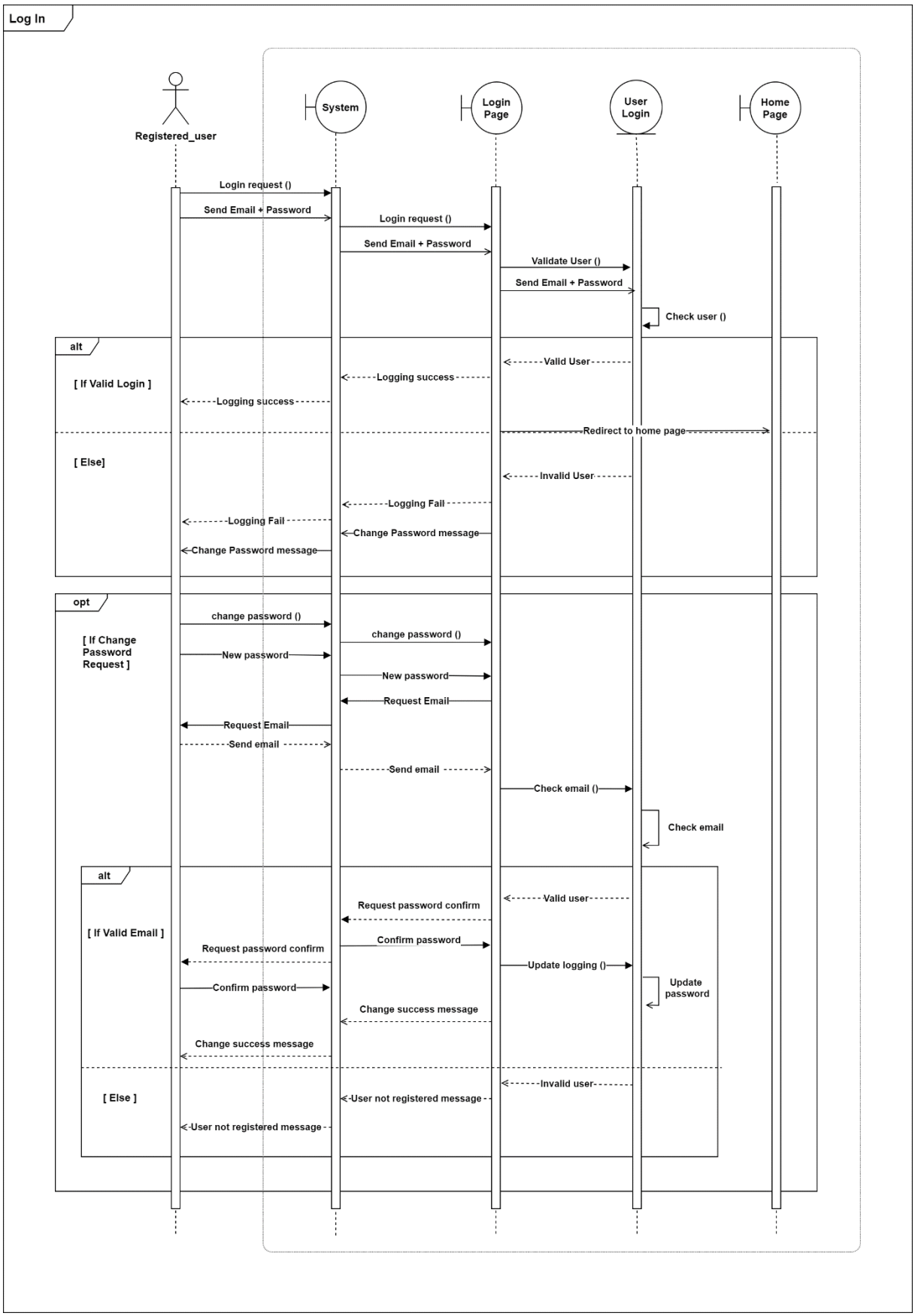


Figure 8: Login

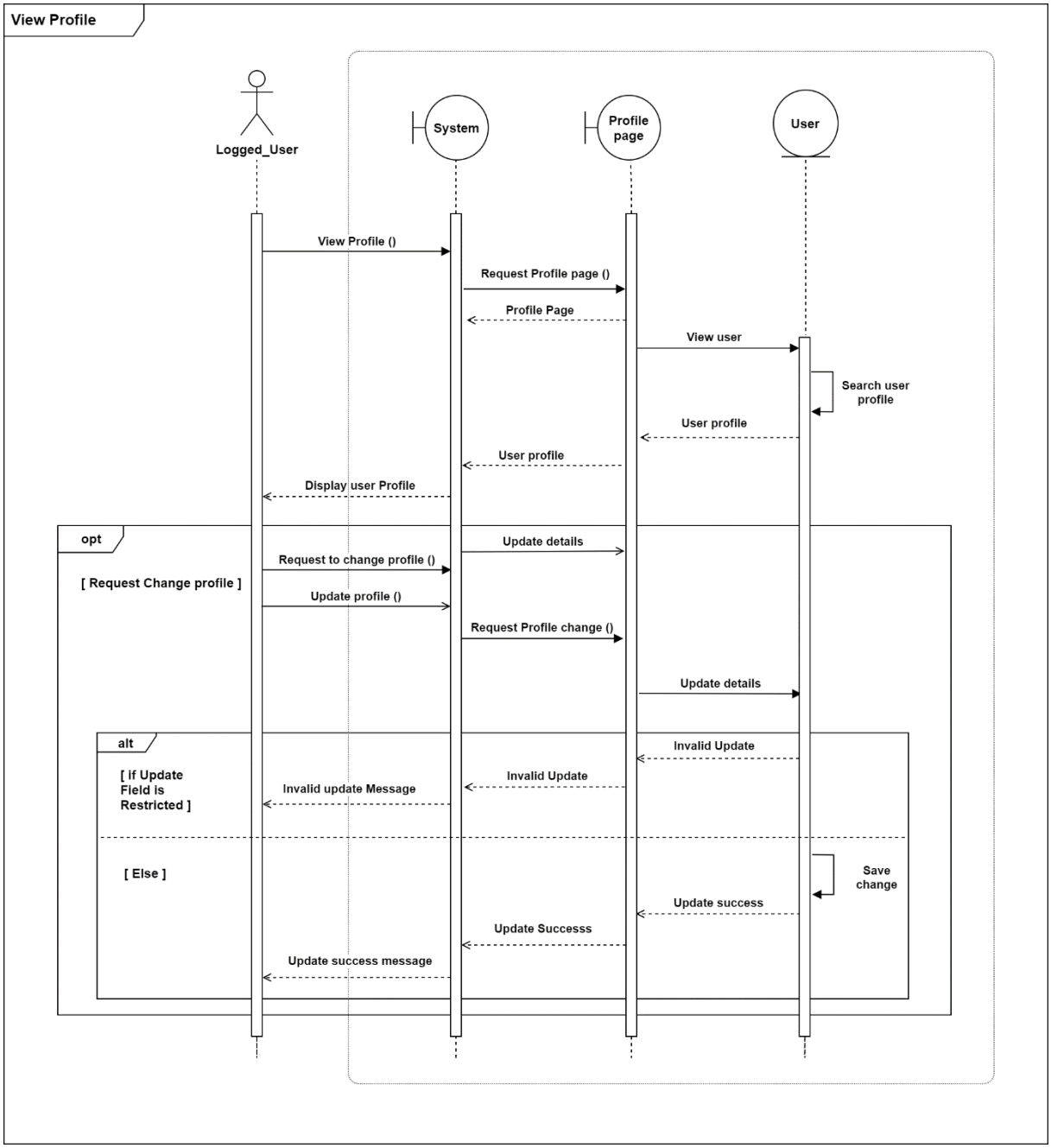


Figure 9: View Profile

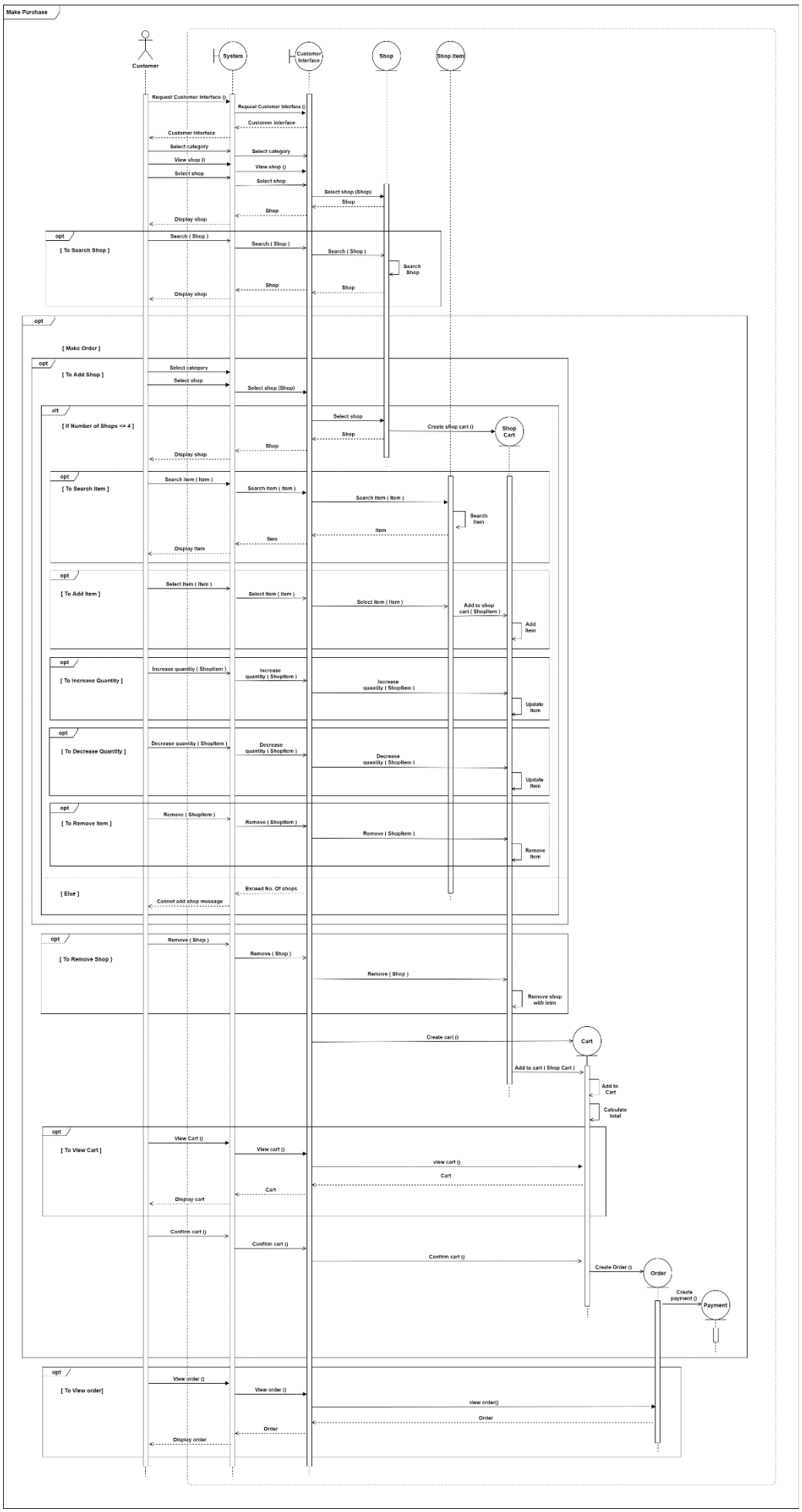


Figure 10: Make Purchase

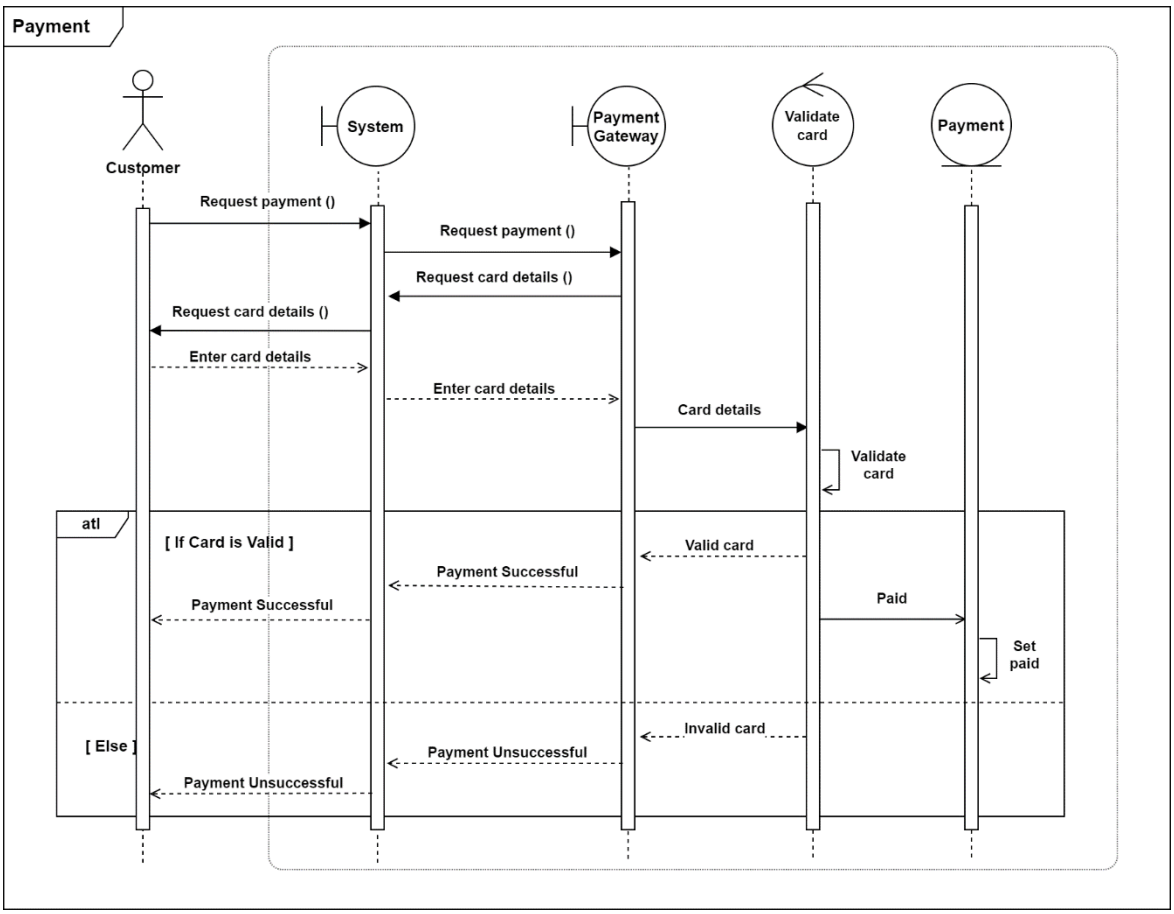


Figure 11: Payment

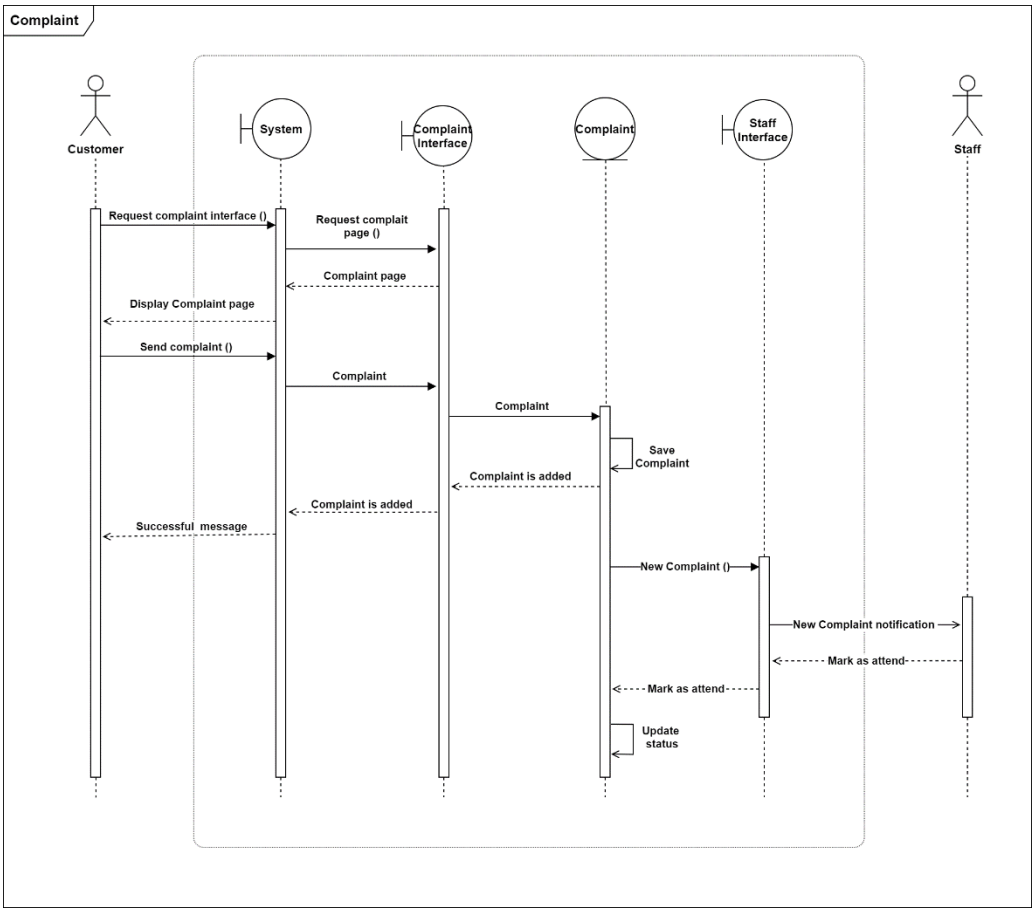


Figure 12: Complaint

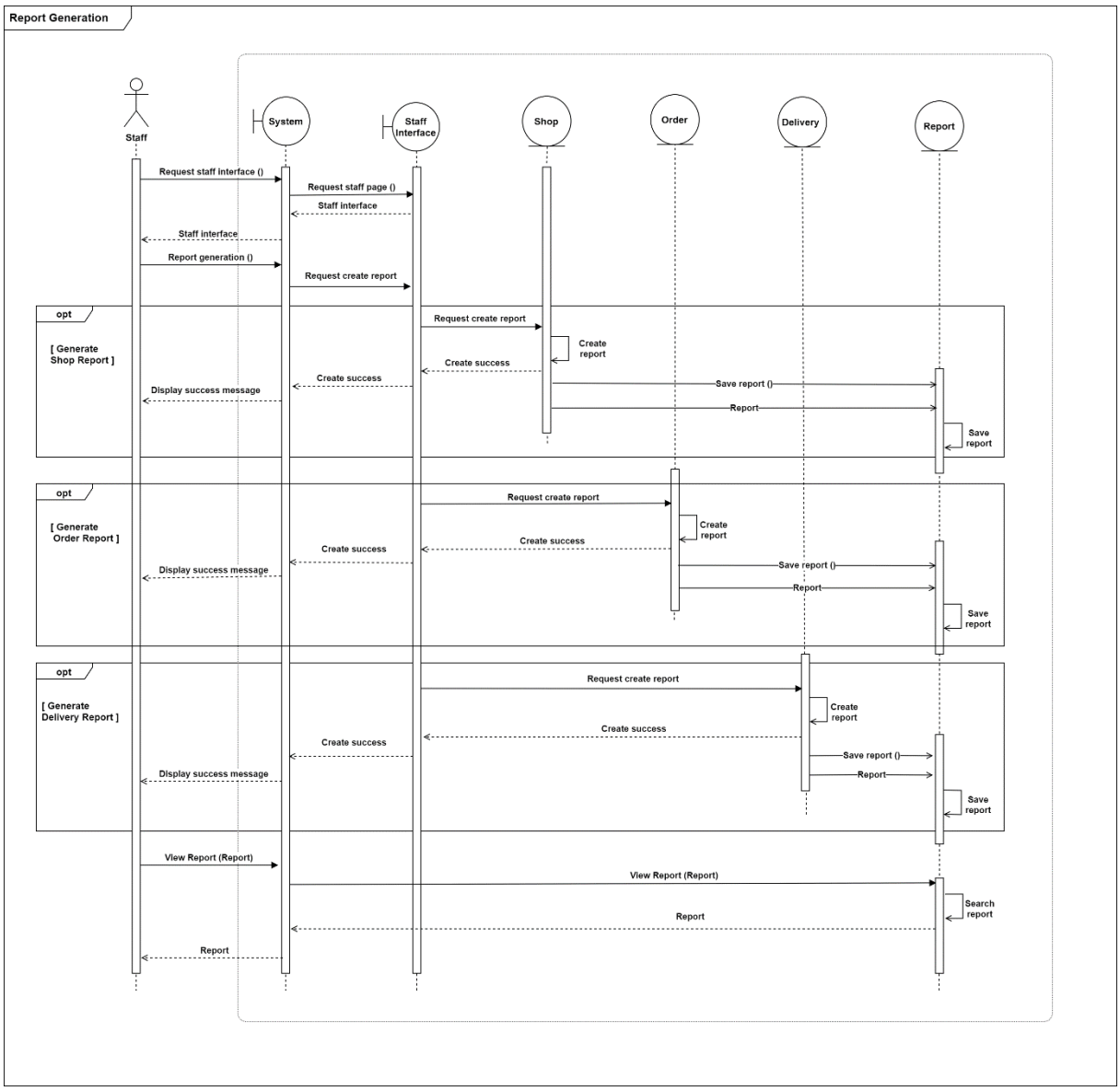


Figure 13: Report Generation

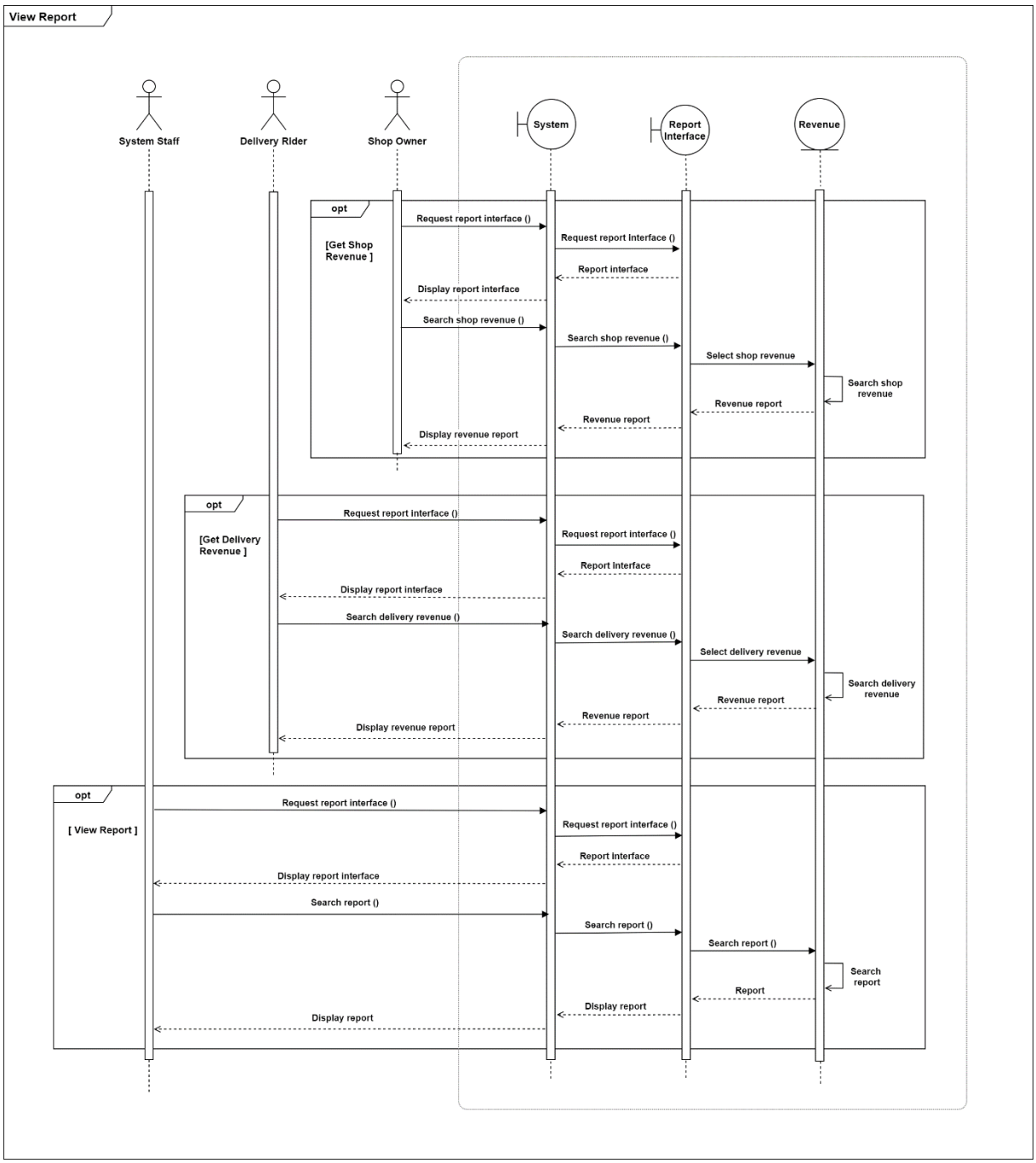


Figure 14: View Report

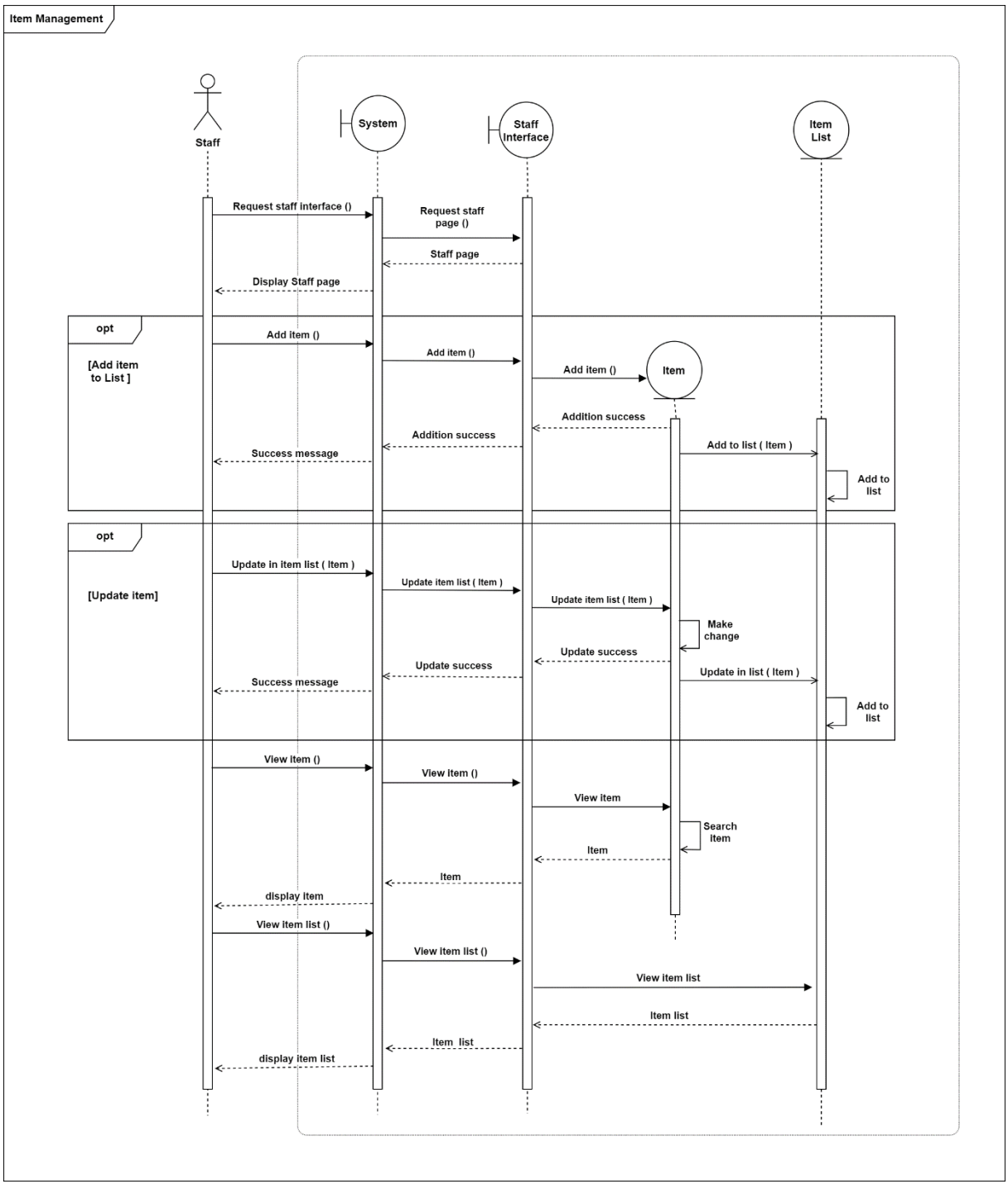


Figure 15: Item Management

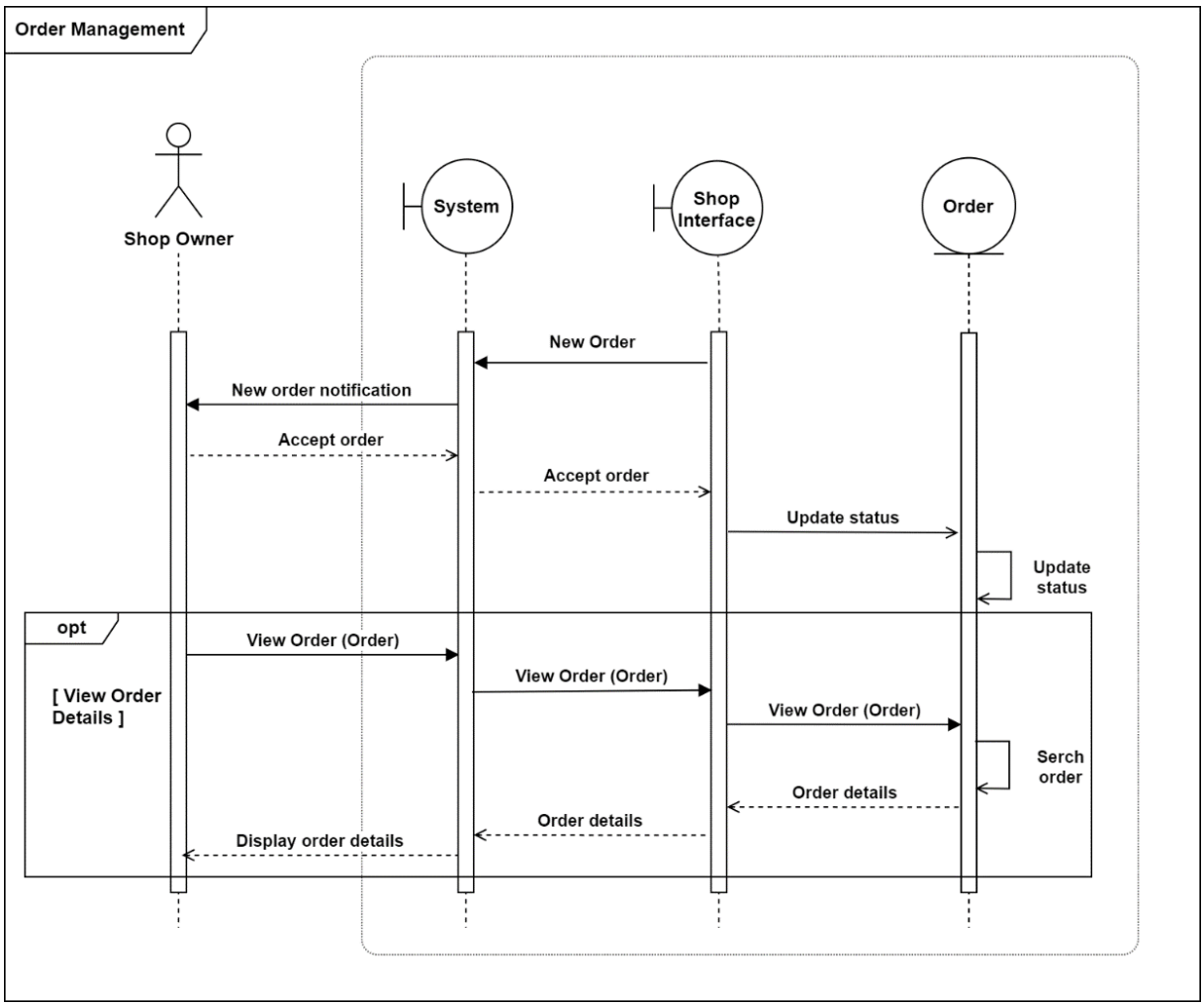


Figure 16: Order Management

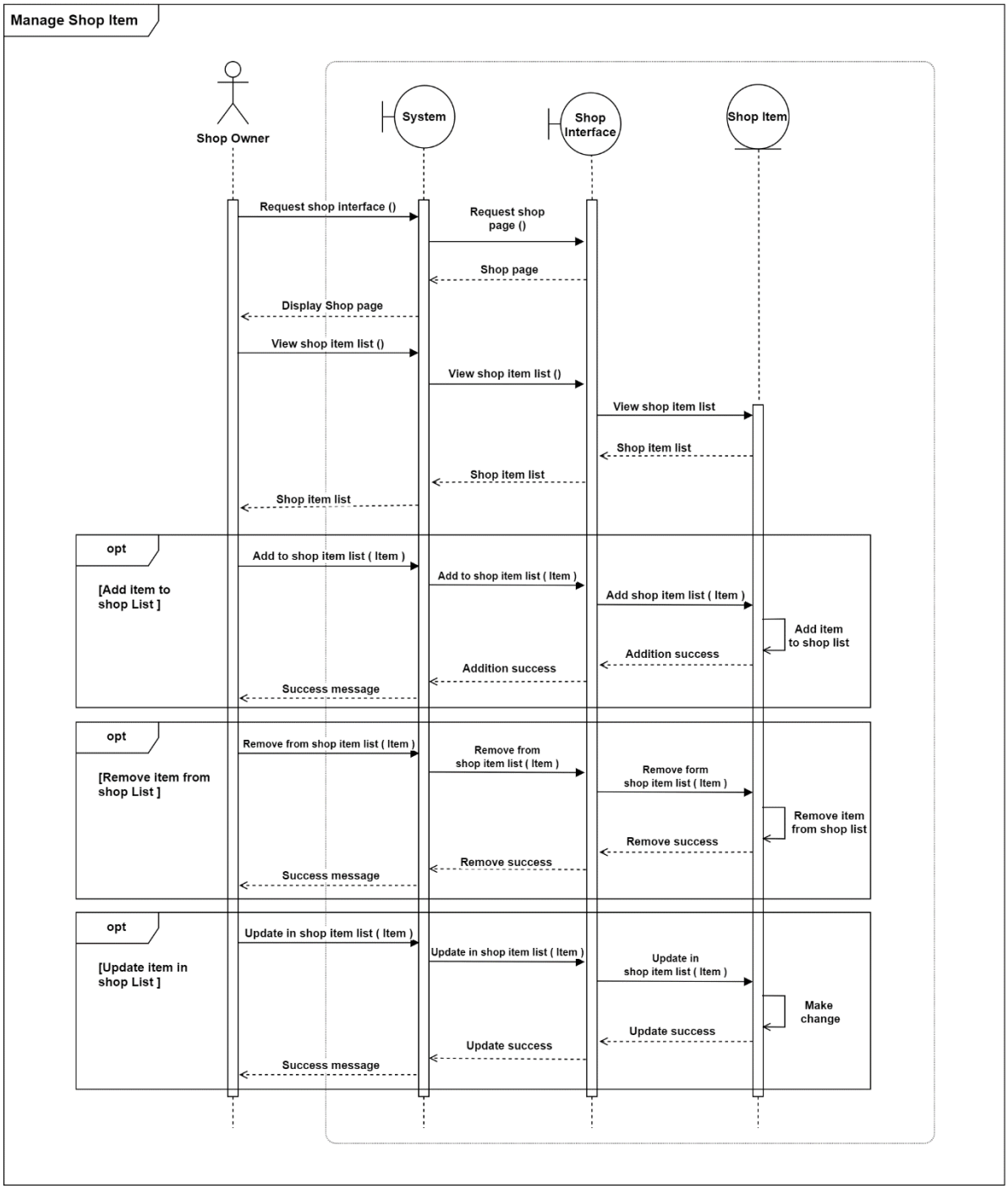


Figure 17: Manage Shop Item

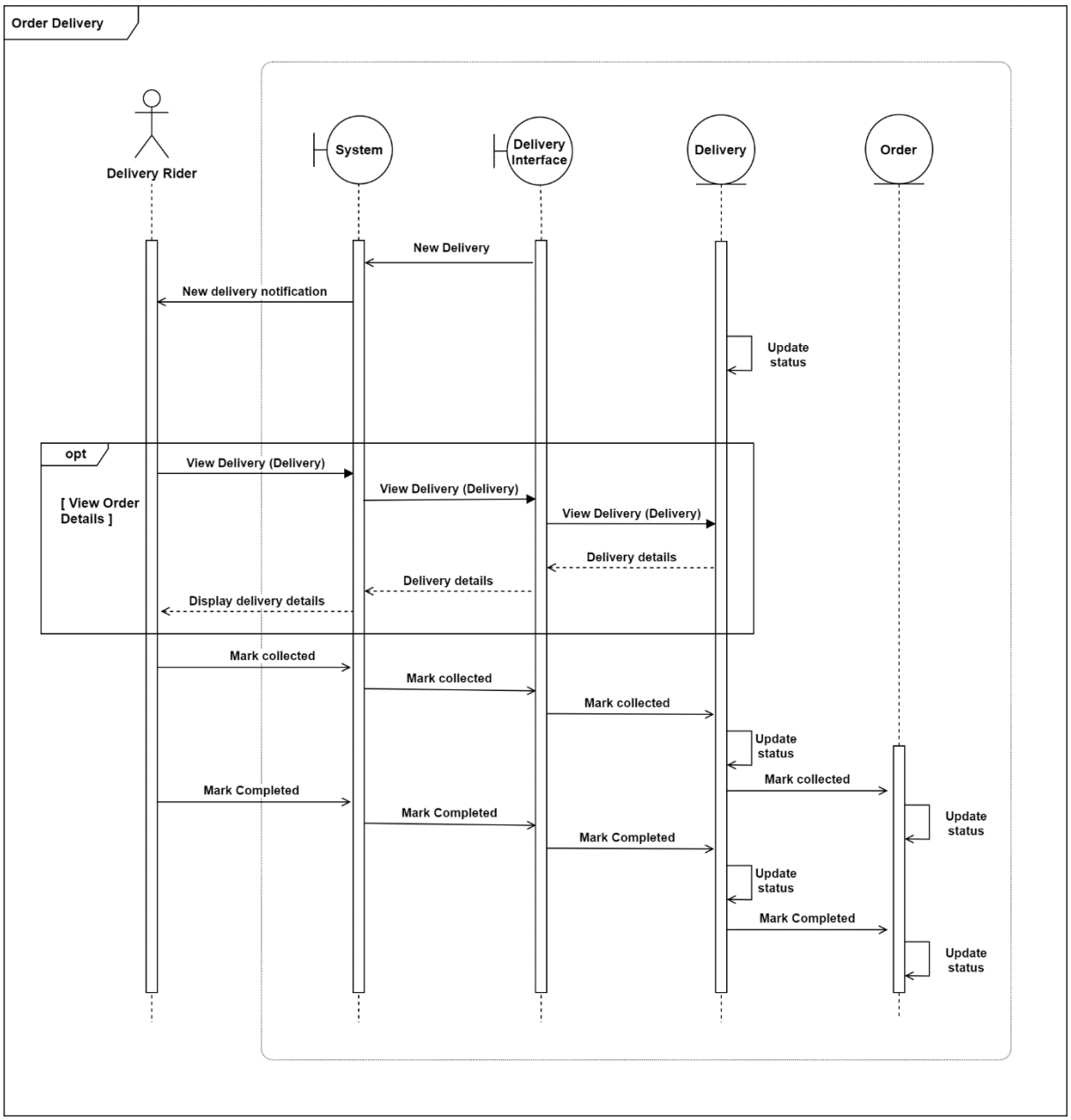
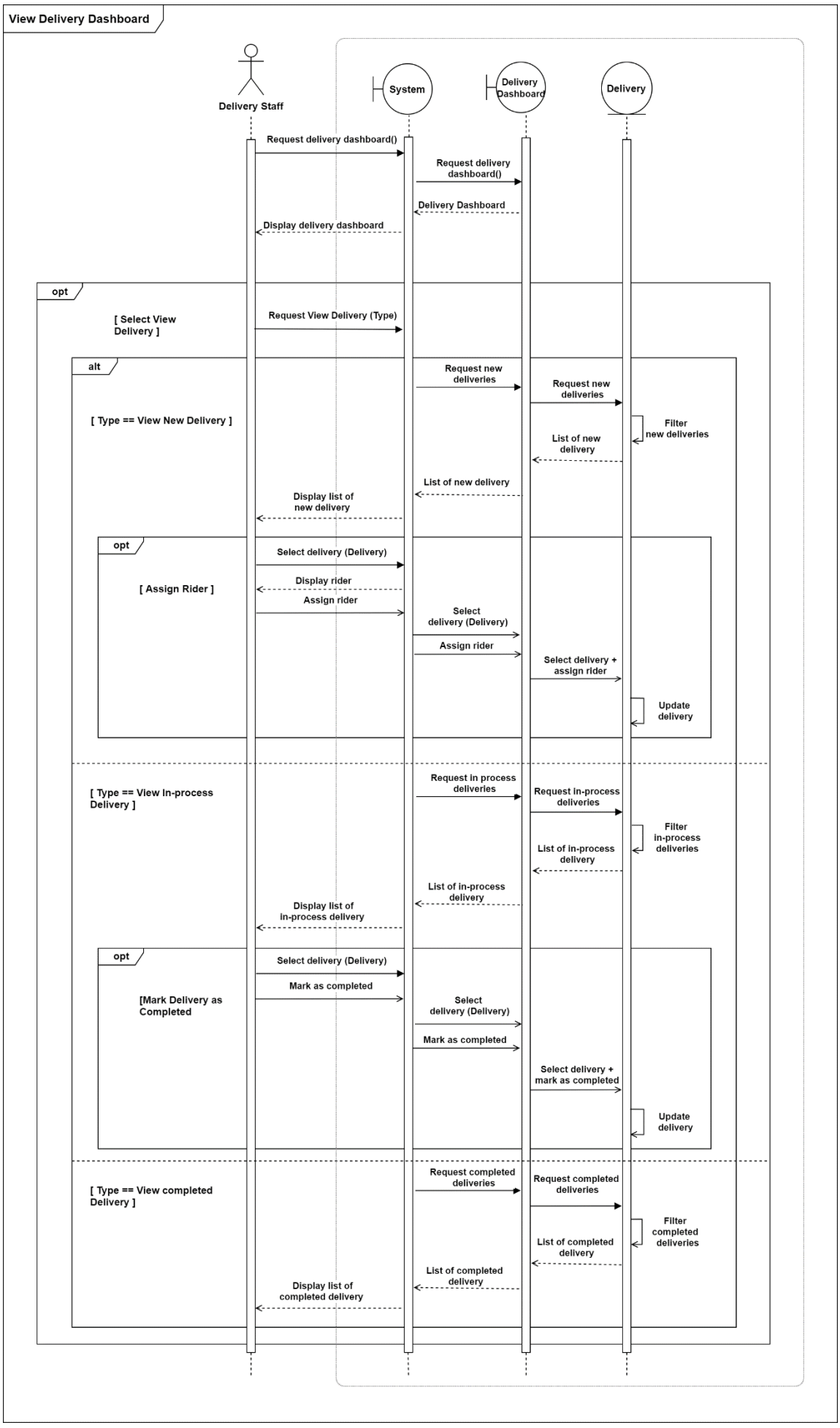


Figure 18: Order Delivery


View Delivery Dashboard
Delivery Staff
System
Delivery Dashboard
Delivery
Request delivery dashboard()
Request delivery dashboard()
Delivery Dashboard
Display delivery dashboard
opt
[Select View Delivery]
Request View Delivery (Type)
alt
[Type == View New Delivery]
Request new deliveries
Request new deliveries
Filter new deliveries
List of new delivery
List of new delivery
Display list of new delivery
opt
[Assign Rider]
Select delivery (Delivery)
Display rider
Assign rider
Select delivery (Delivery)
Assign rider
Select delivery + assign rider
Update delivery
[Type == View In-process Delivery]
Request in process deliveries
Request in-process deliveries
Filter in-process deliveries
List of in-process delivery
List of in-process delivery
Display list of in-process delivery
opt
[Mark Delivery as Completed
Select delivery (Delivery)
Mark as completed
Select delivery (Delivery)
Mark as completed
Select delivery + mark as completed
Update delivery
[Type == View completed Delivery]
Request completed deliveries
Request completed deliveries
Filter completed deliveries
List of completed delivery
List of completed delivery
Display list of completed delivery

5.4 Activity diagrams

5.4.1 Customer

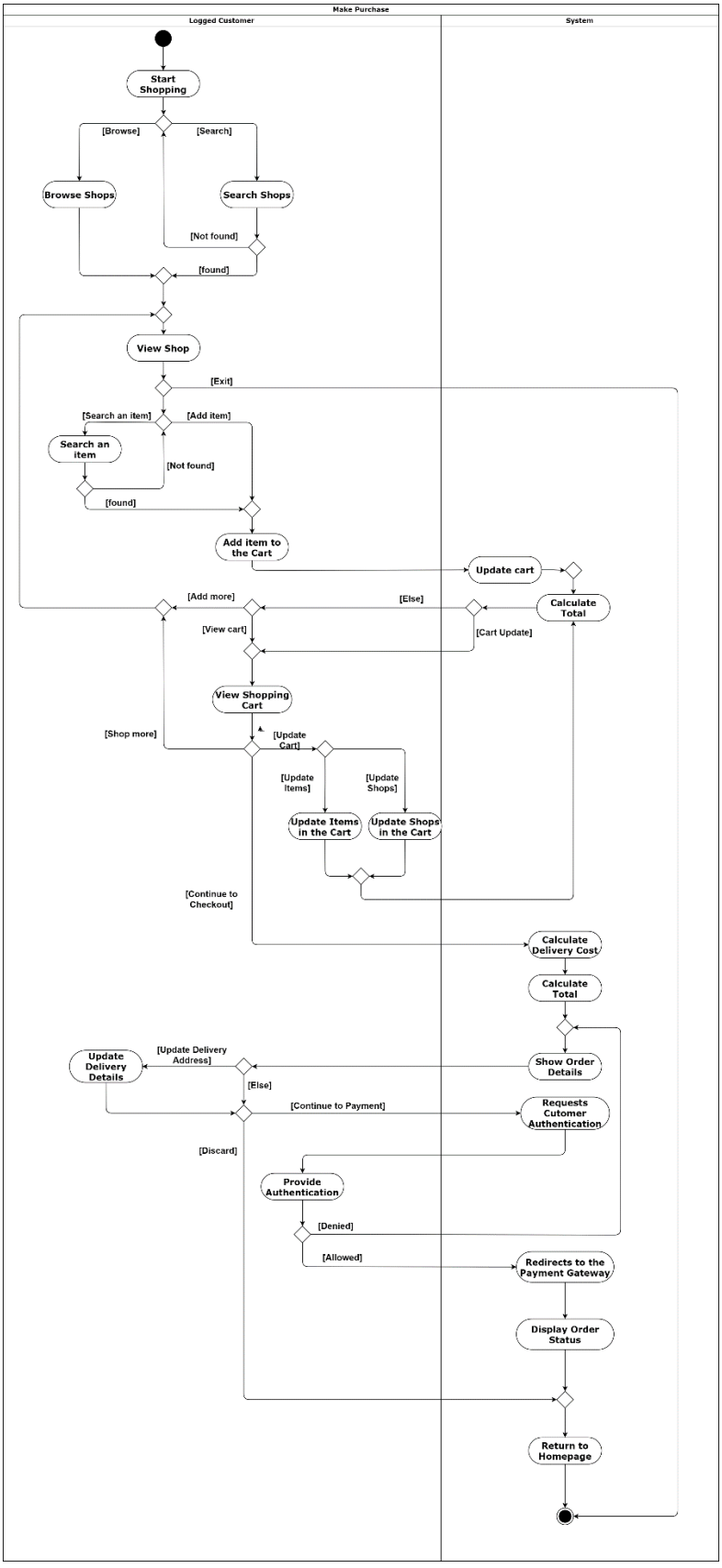


Figure 20: Customer - Make Purchase

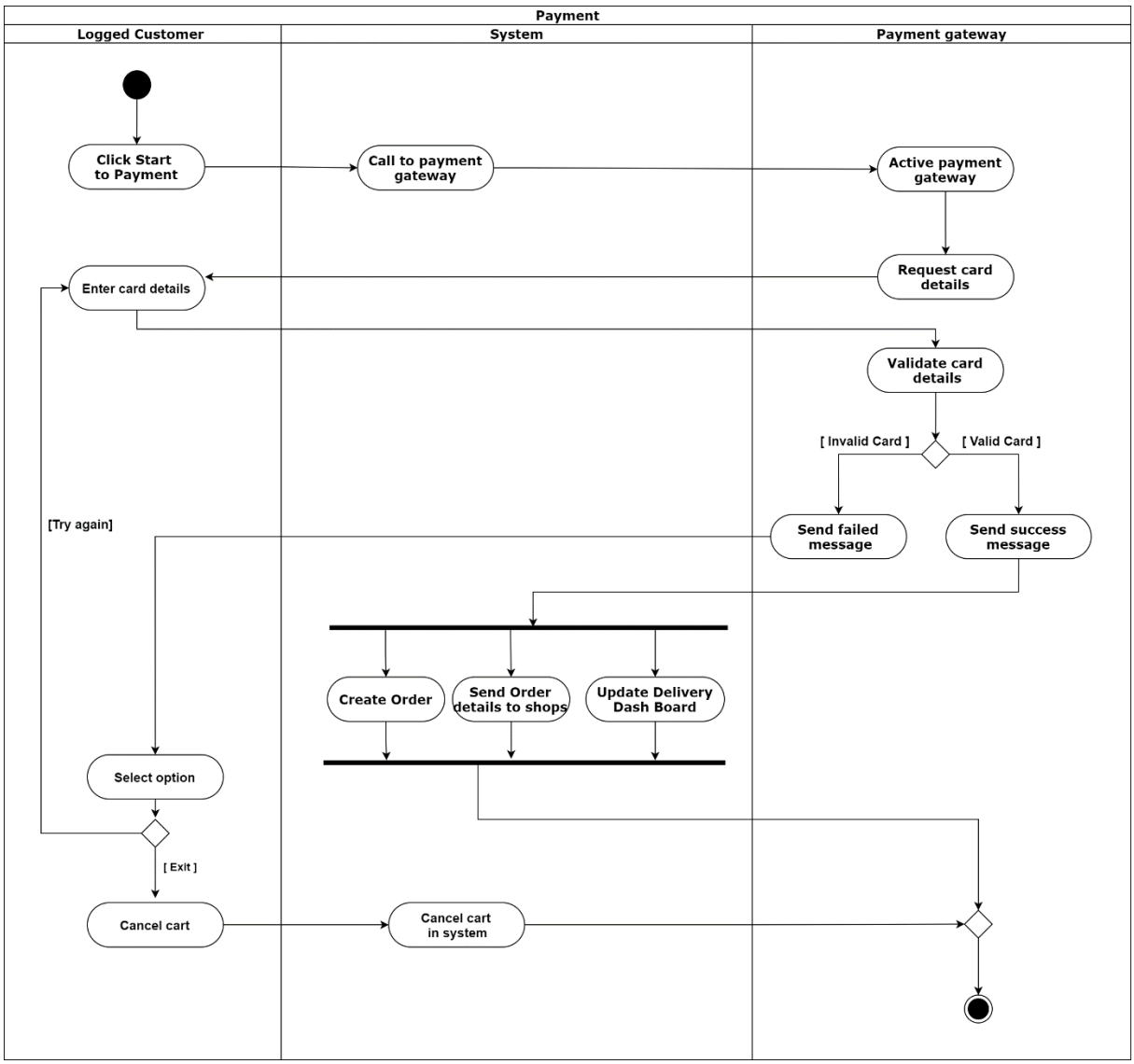


Figure 21: Customer – Payment

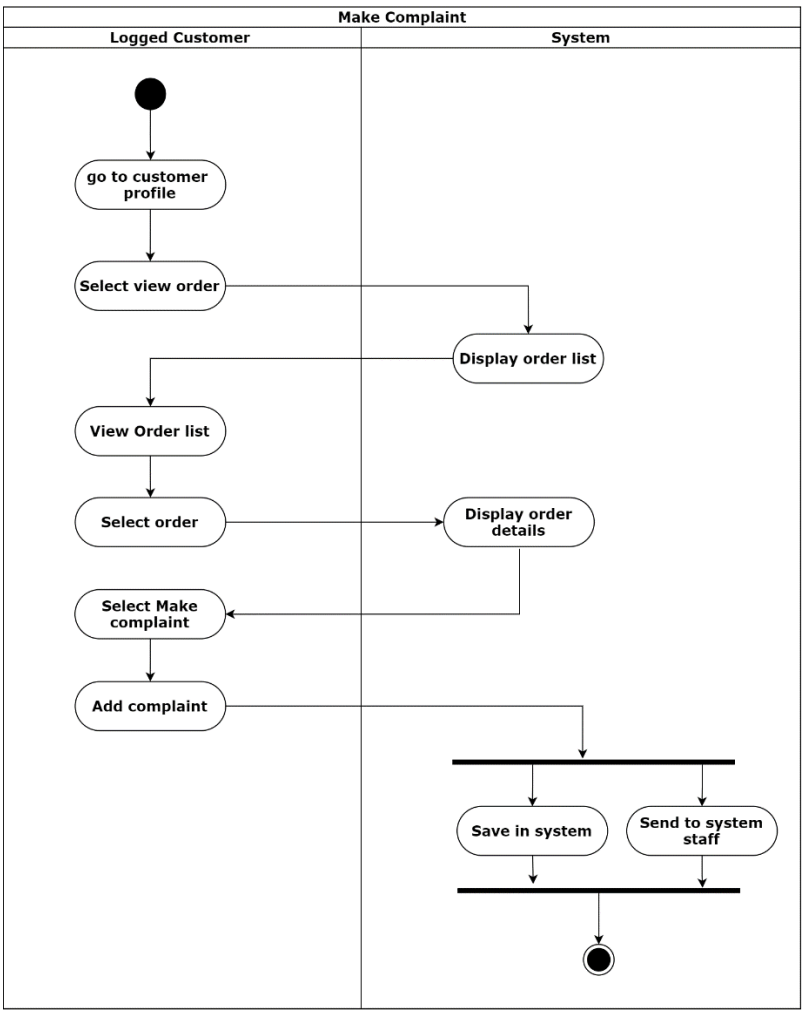


Figure 22: Customer - Make Complaint

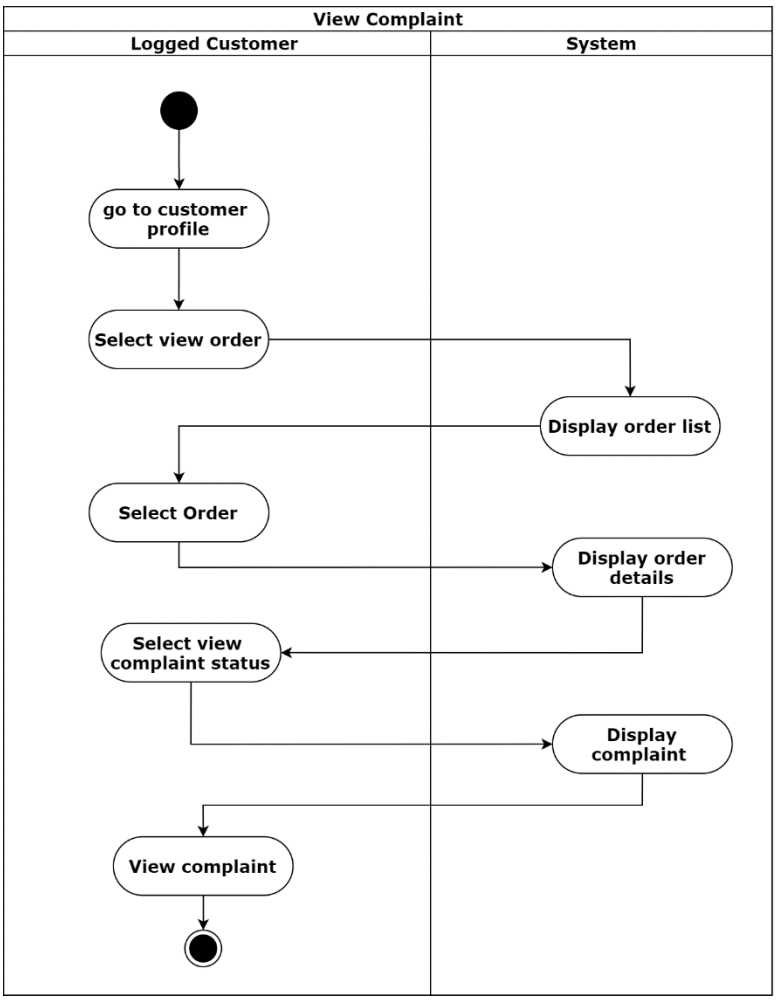


Figure 23: Customer - View Complaint

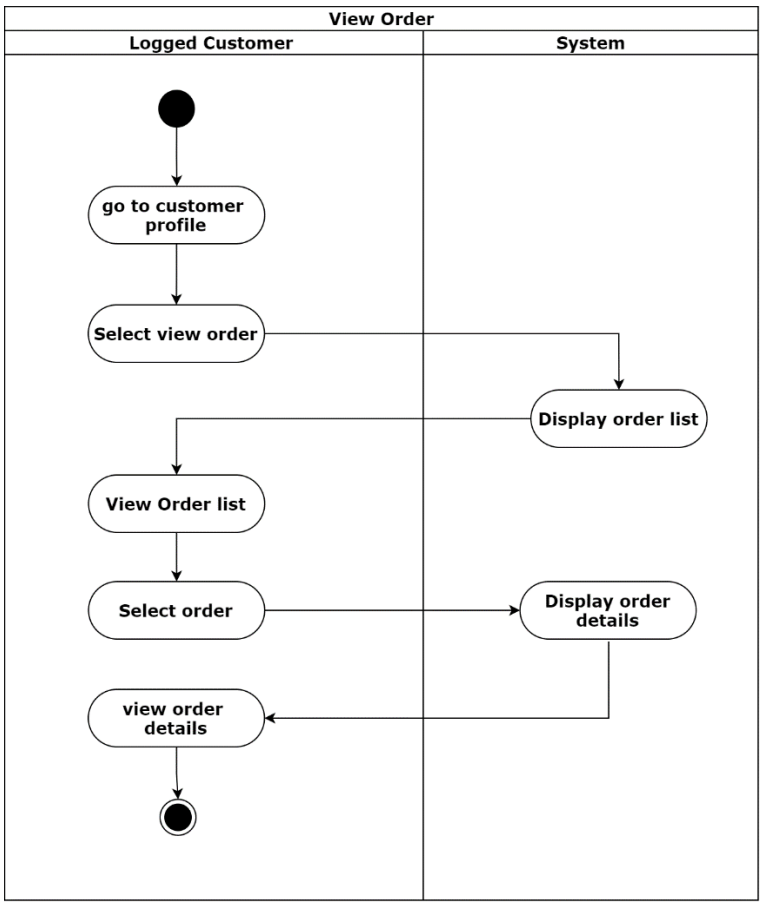


Figure 24: Customer - Logged Customer

5.4.2 Delivery

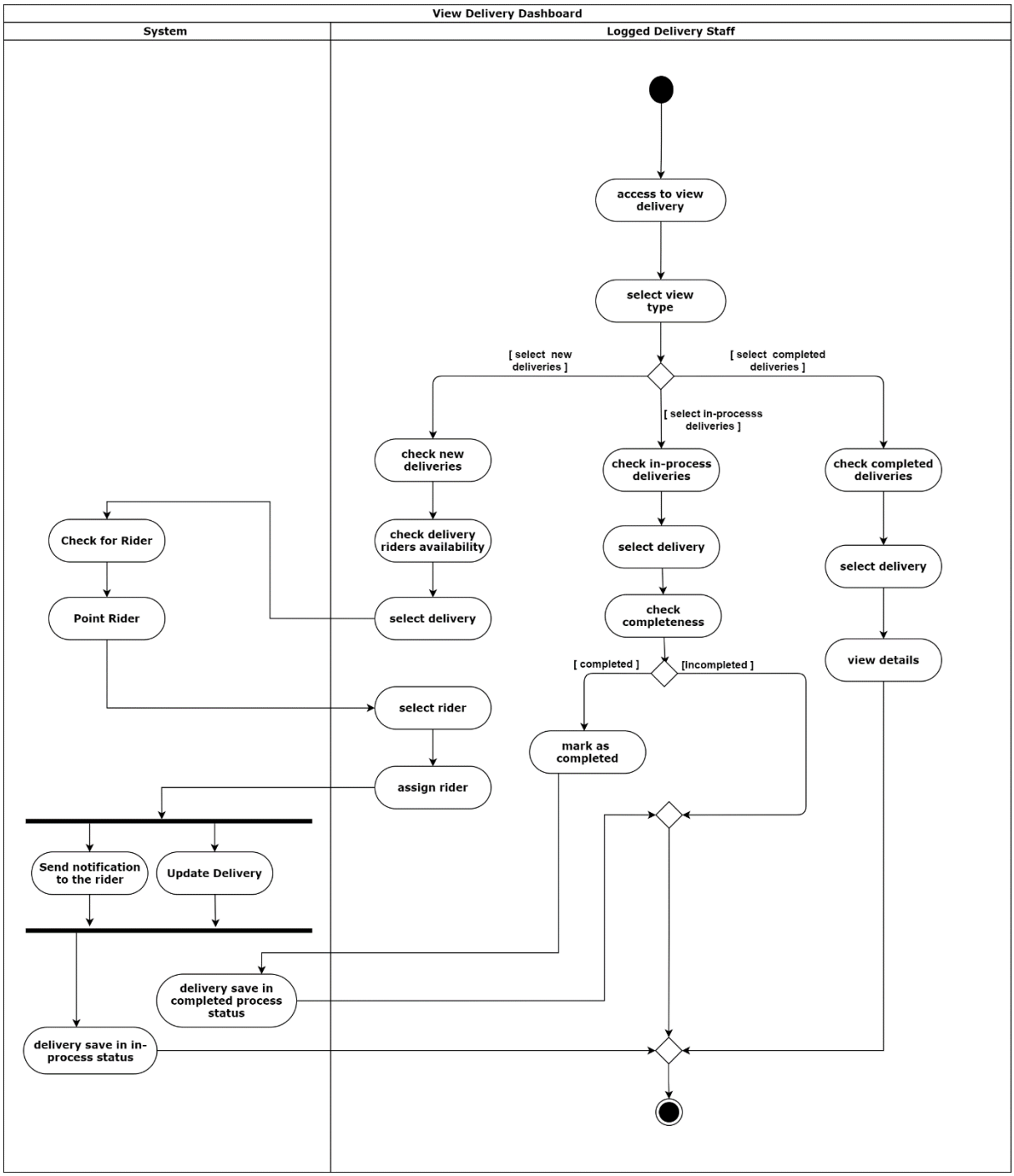


Figure 25: View Delivery Dashboard

5.4.3 System Staff

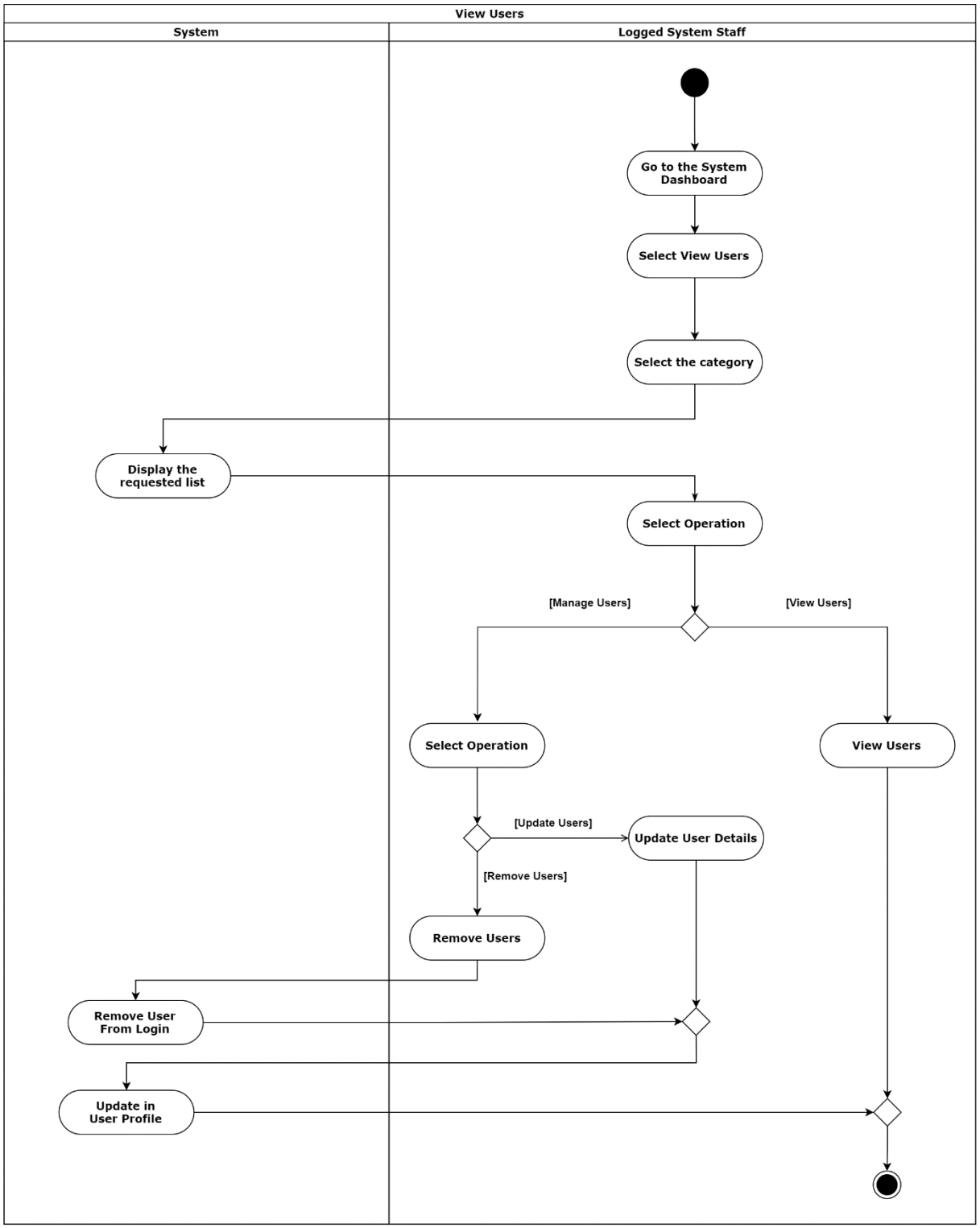


Figure 26: System Staff - View Users

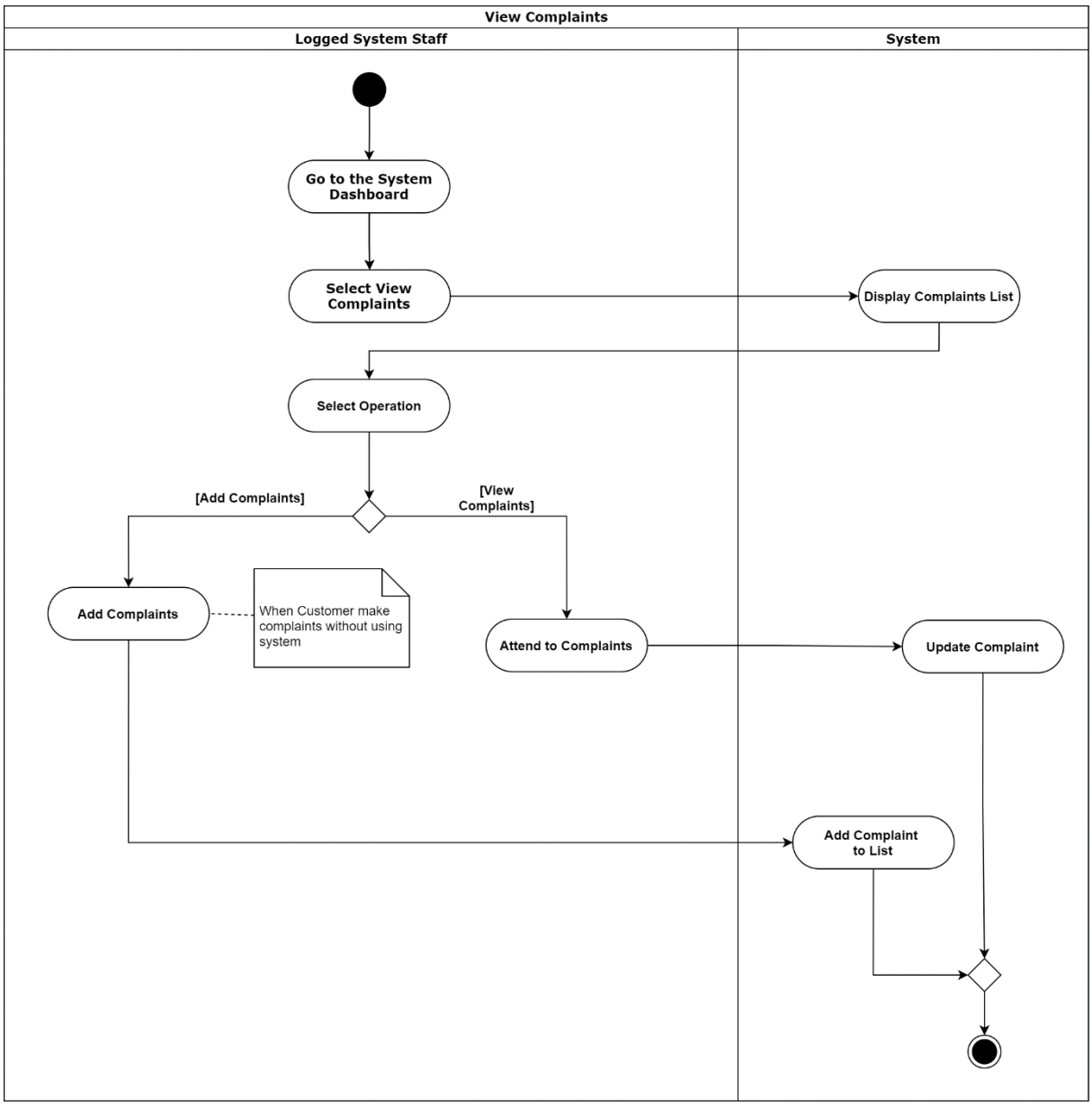


Figure 27: System Staff - View Complaints

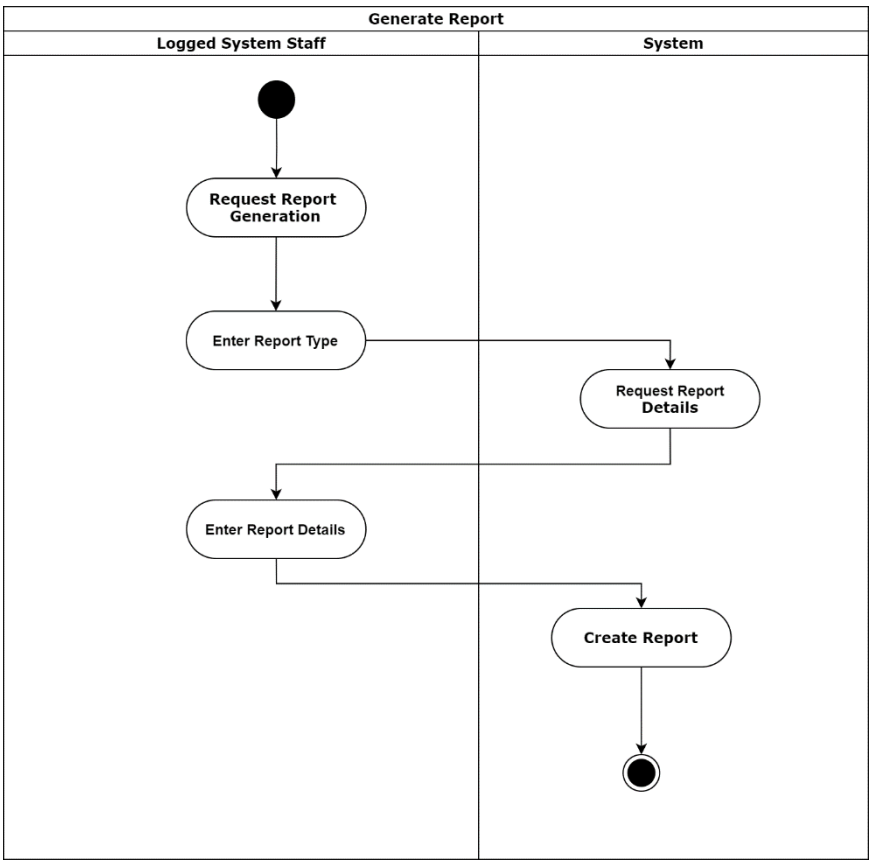
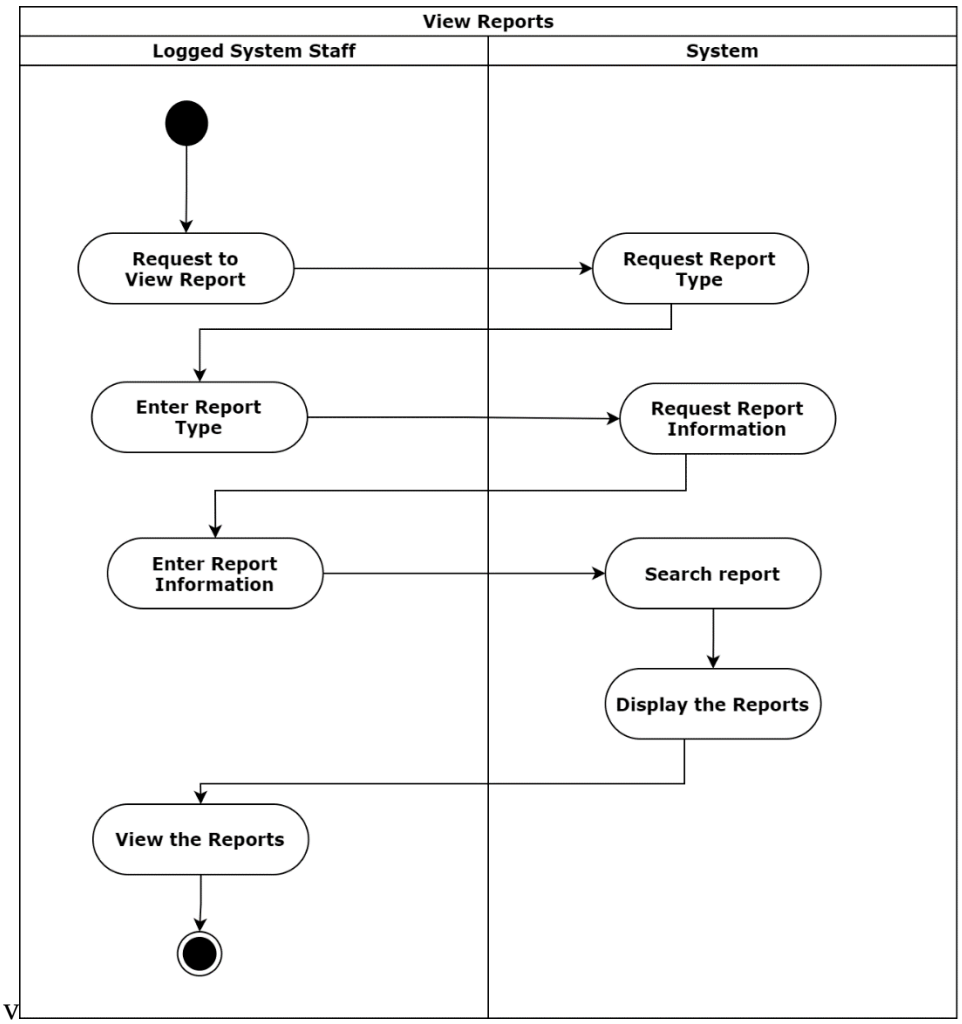


Figure 28: System Staff - Generate Report



V

Figure 29: System Staff - View Reports

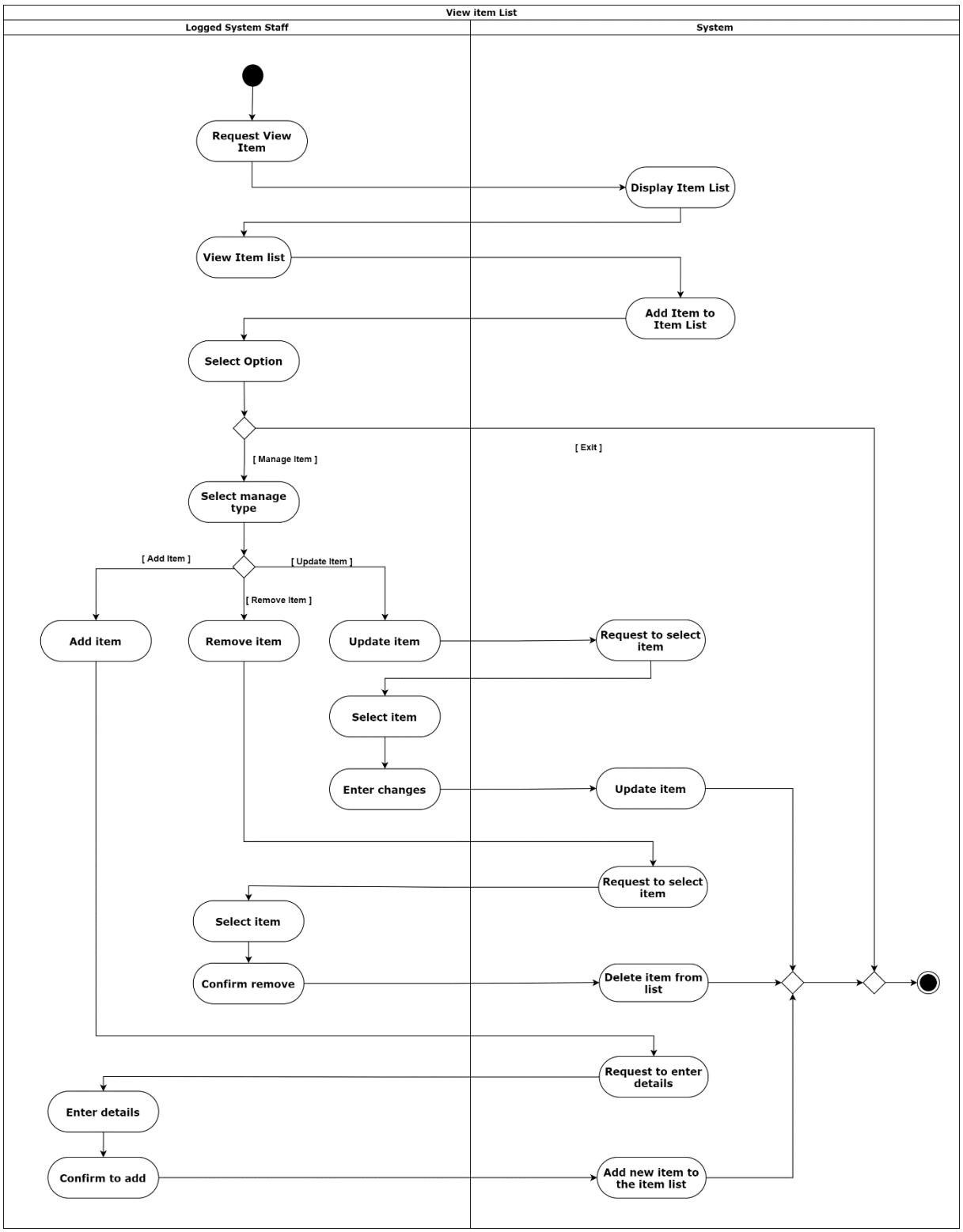


Figure 30: System Staff - View Item List

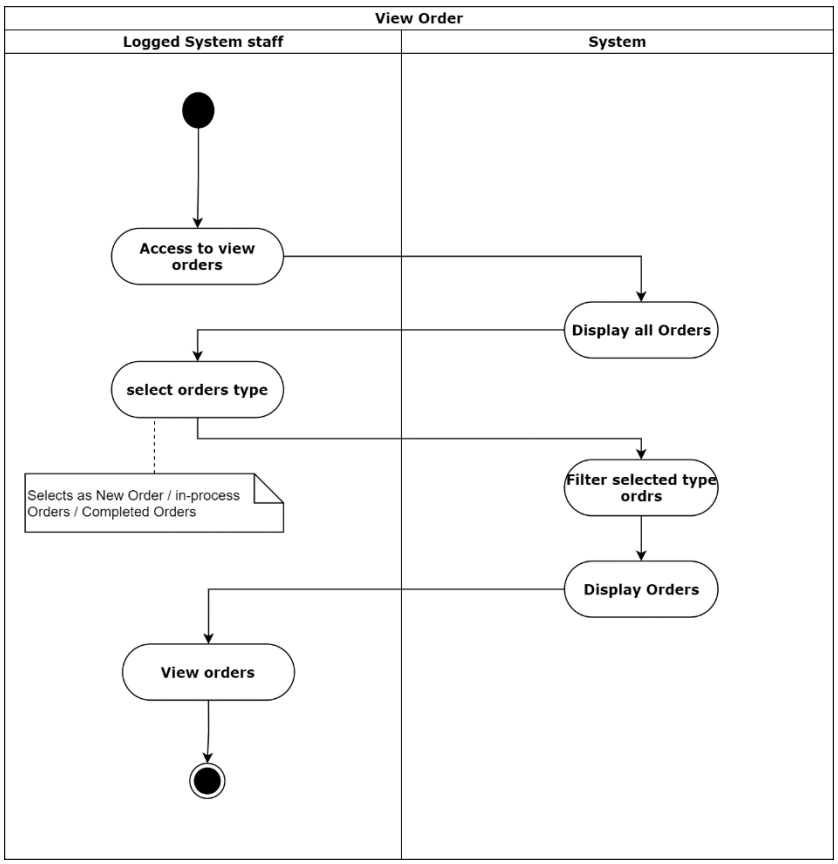


Figure 31: System Staff - View Order

5.4.4 Delivery Rider

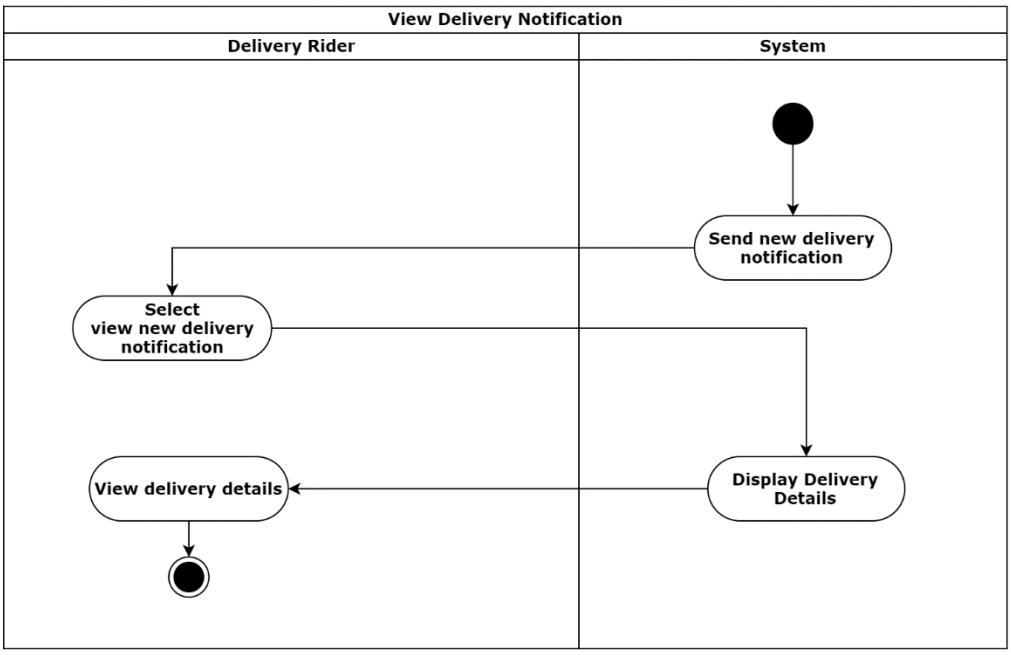


Figure 32: Delivery Rider - View Delivery Notification

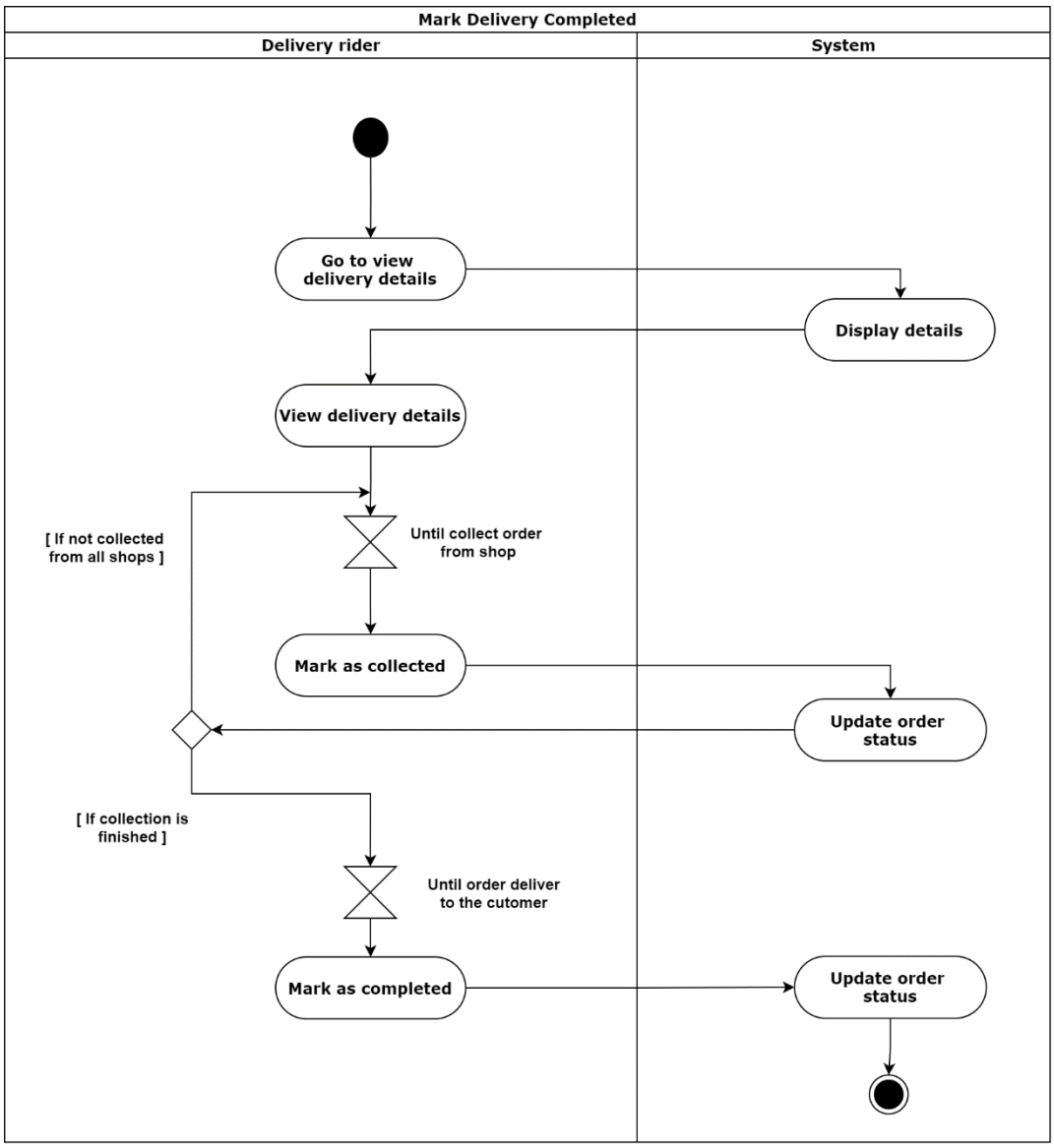


Figure 33: Delivery Rider - Mark Delivery Completed

5.4.5 Shop Staff

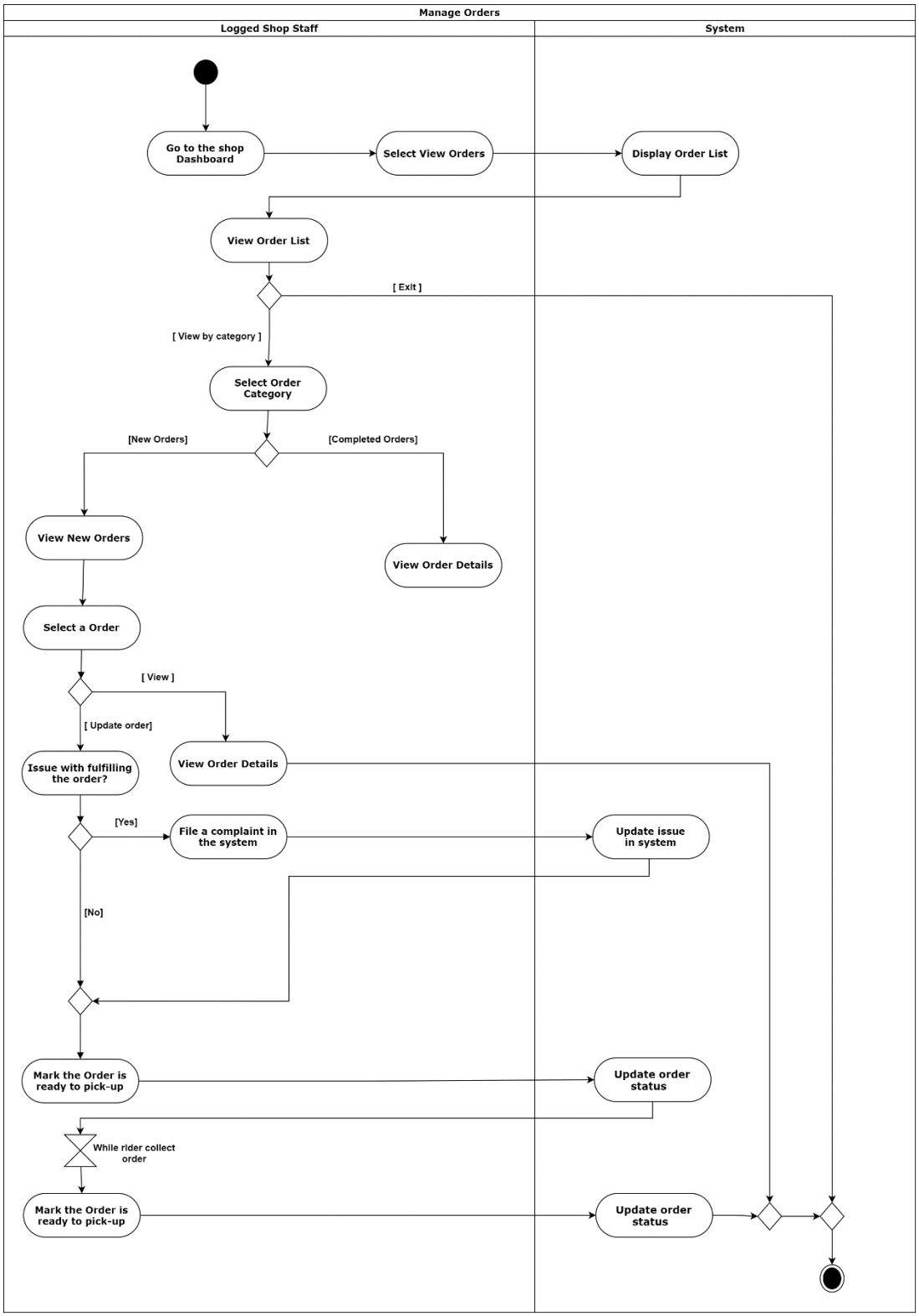


Figure 34: Shop Staff - Manage Orders

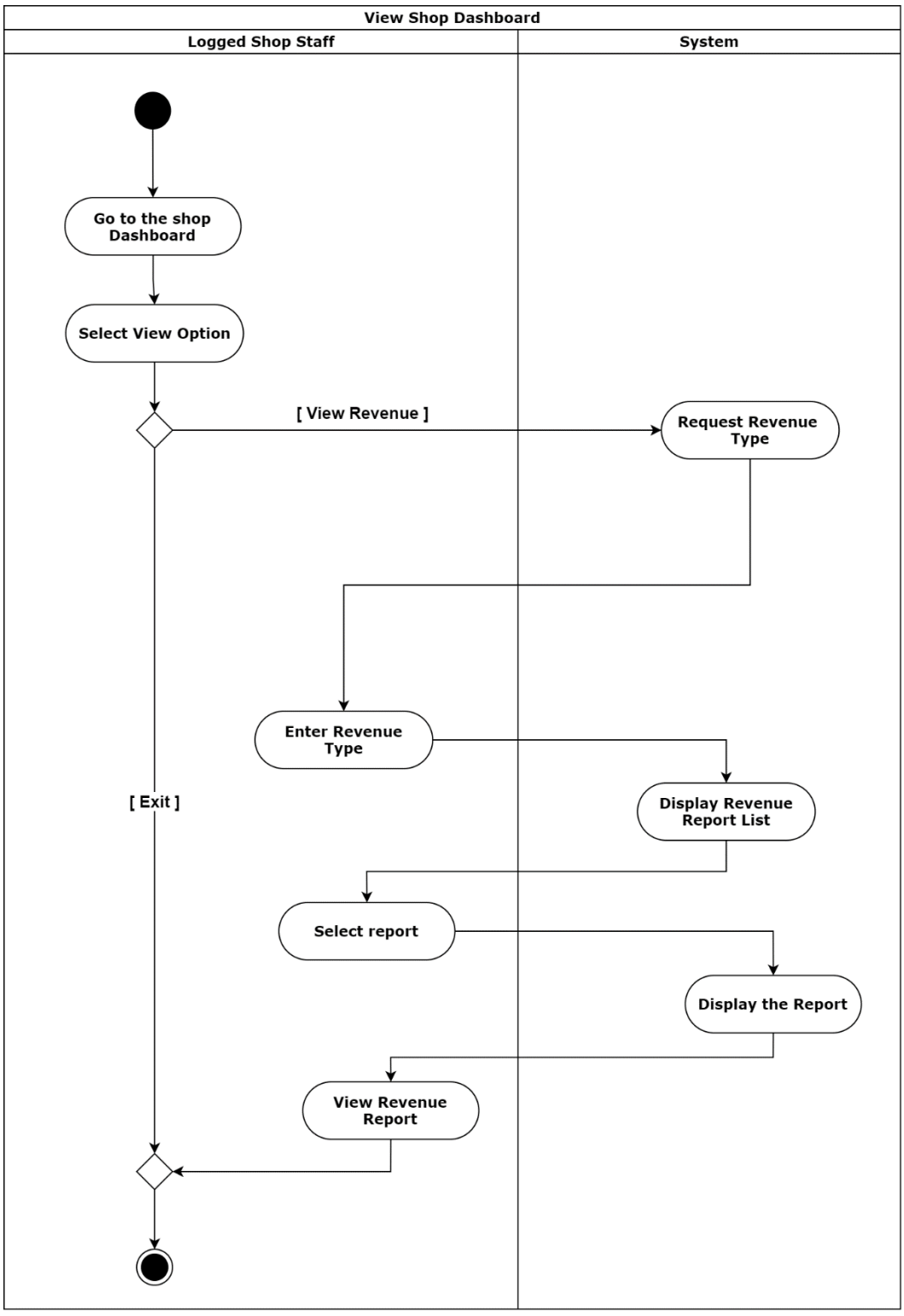


Figure 35: View Shop Dashboard

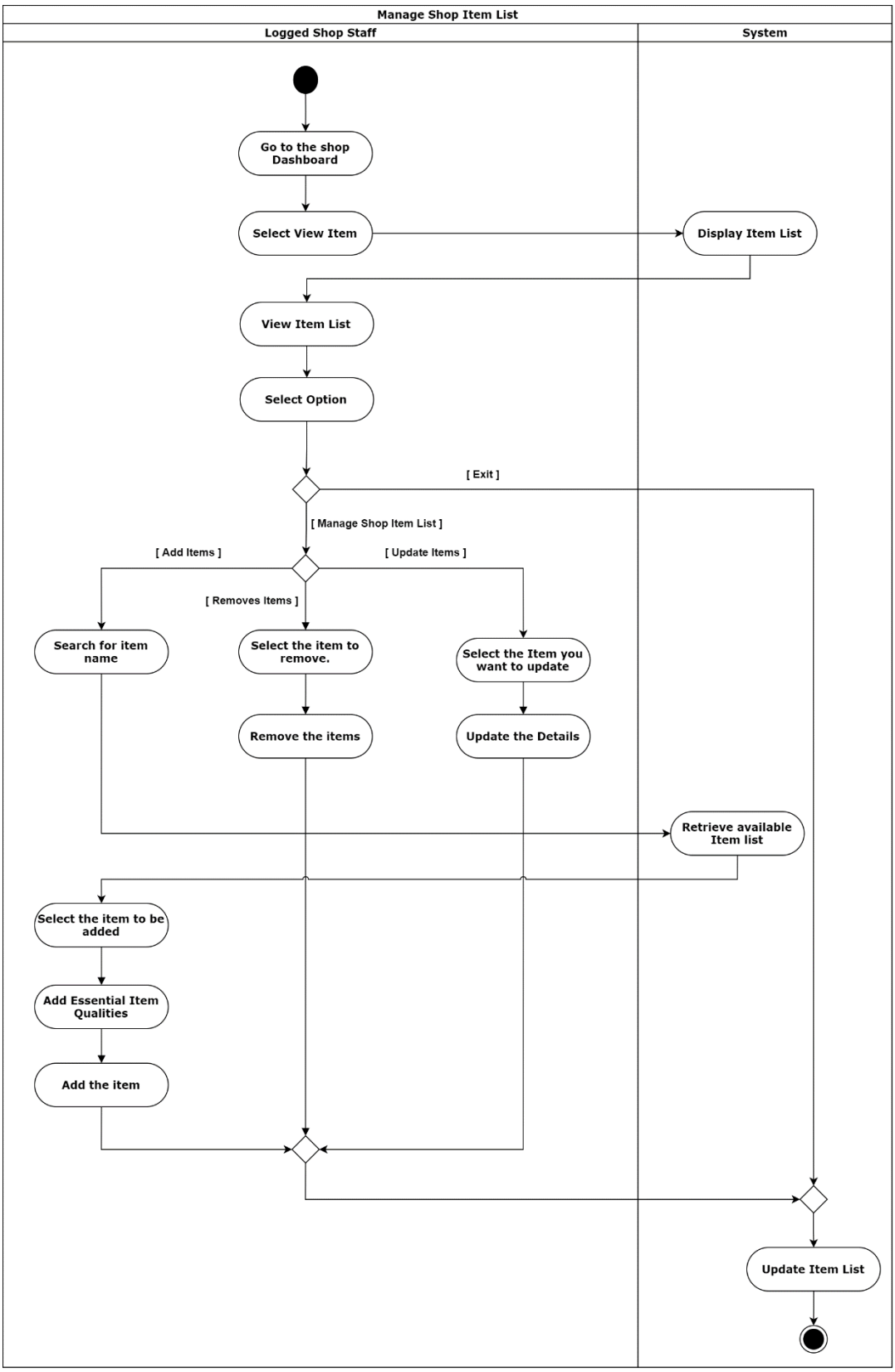


Figure 36: Shop Staff - Manage Shop Item List

5.4.6 Common Use Cases

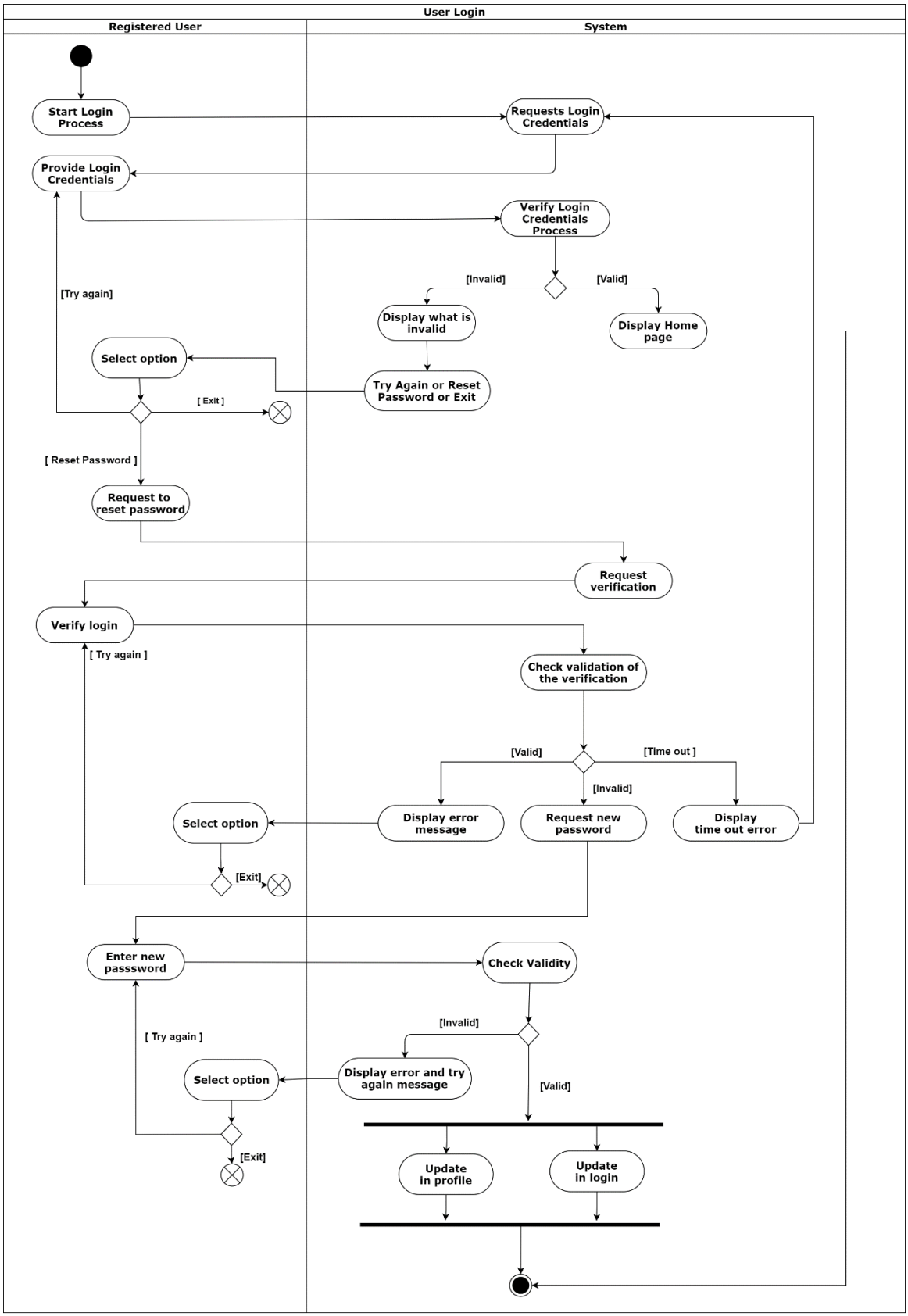


Figure 37: Common Use Case - User Login

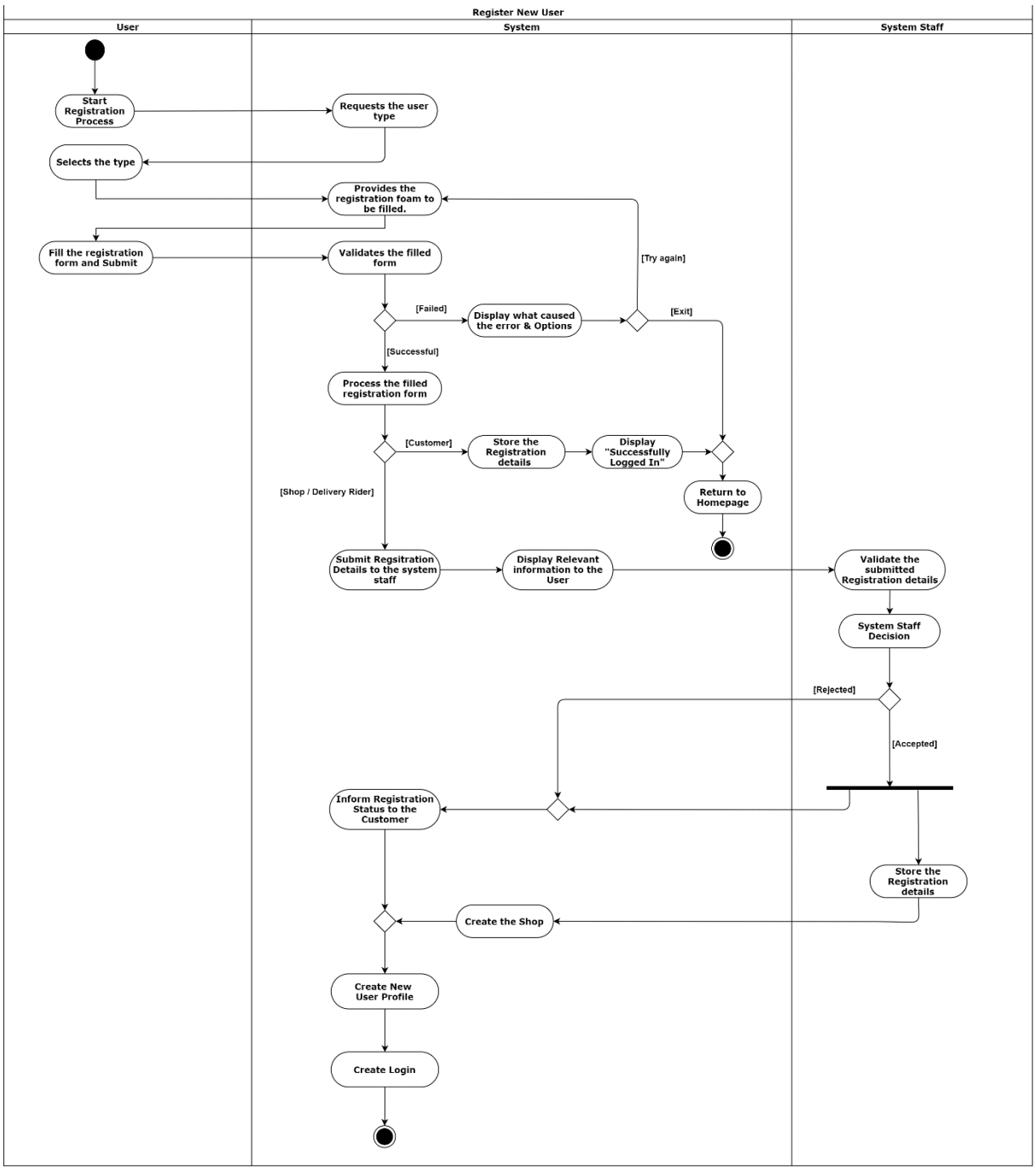


Figure 38: Common Use Case - Register New User

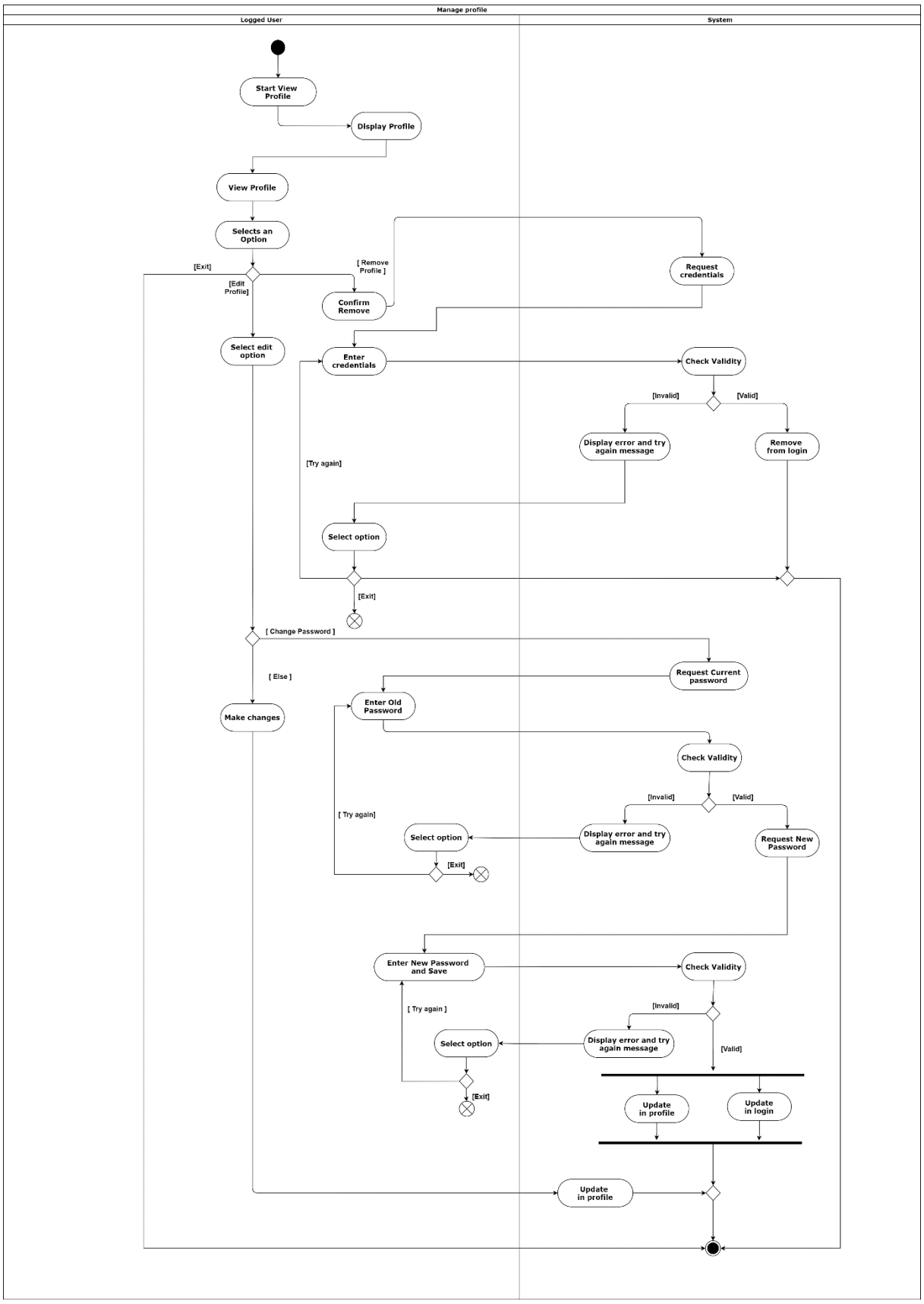


Figure 39: Common Use Case - Manage Profile

5.5 State transition diagrams

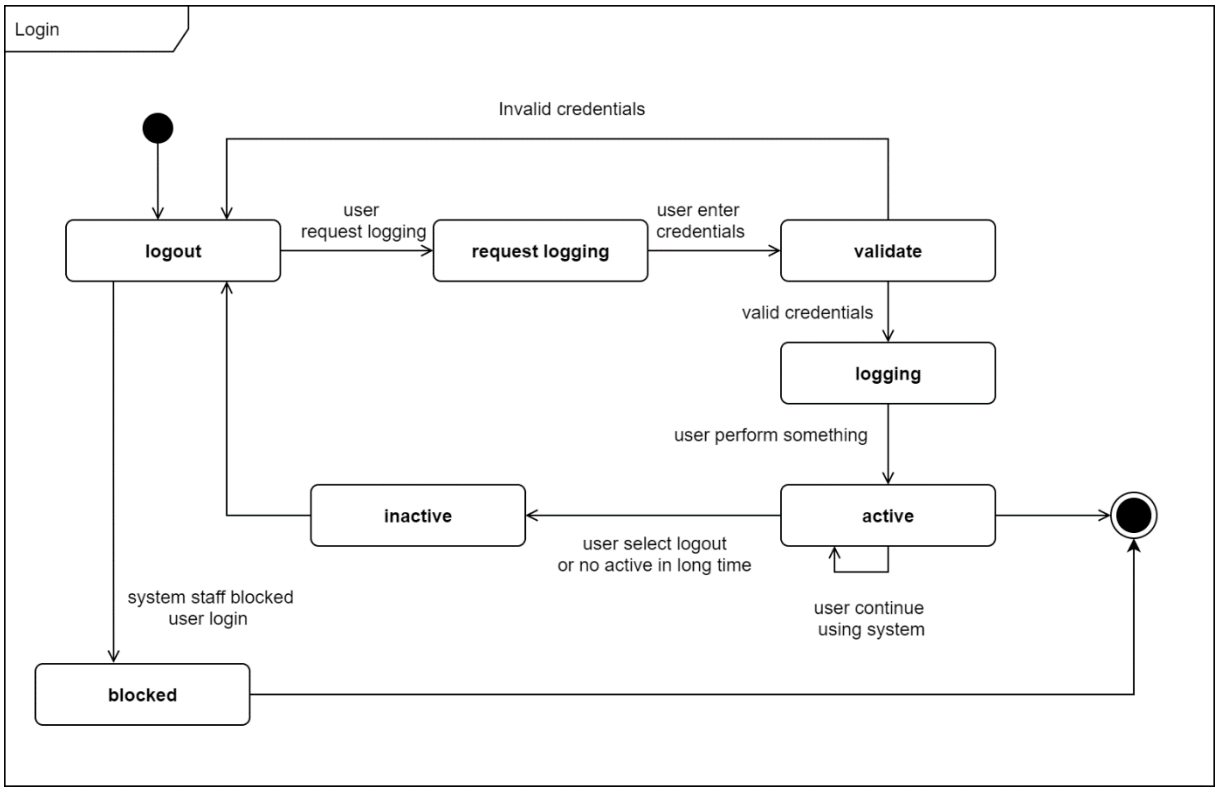


Figure 41: Login State Transition

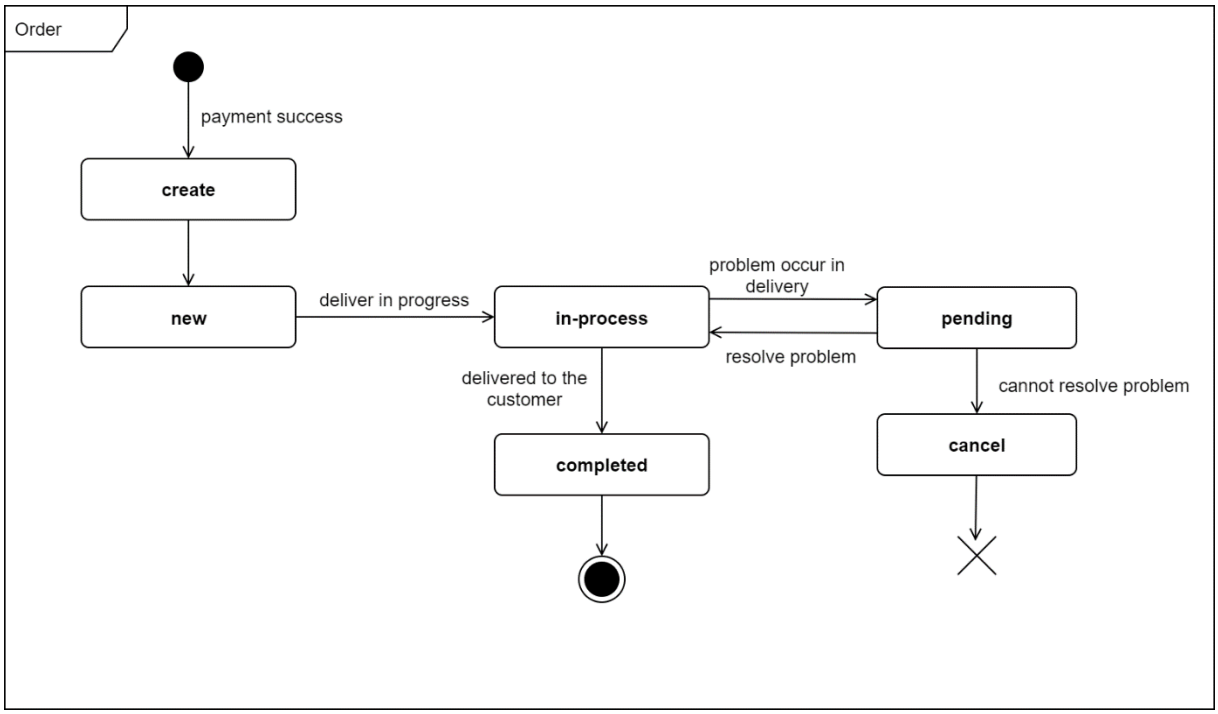


Figure 40: Order State Transition

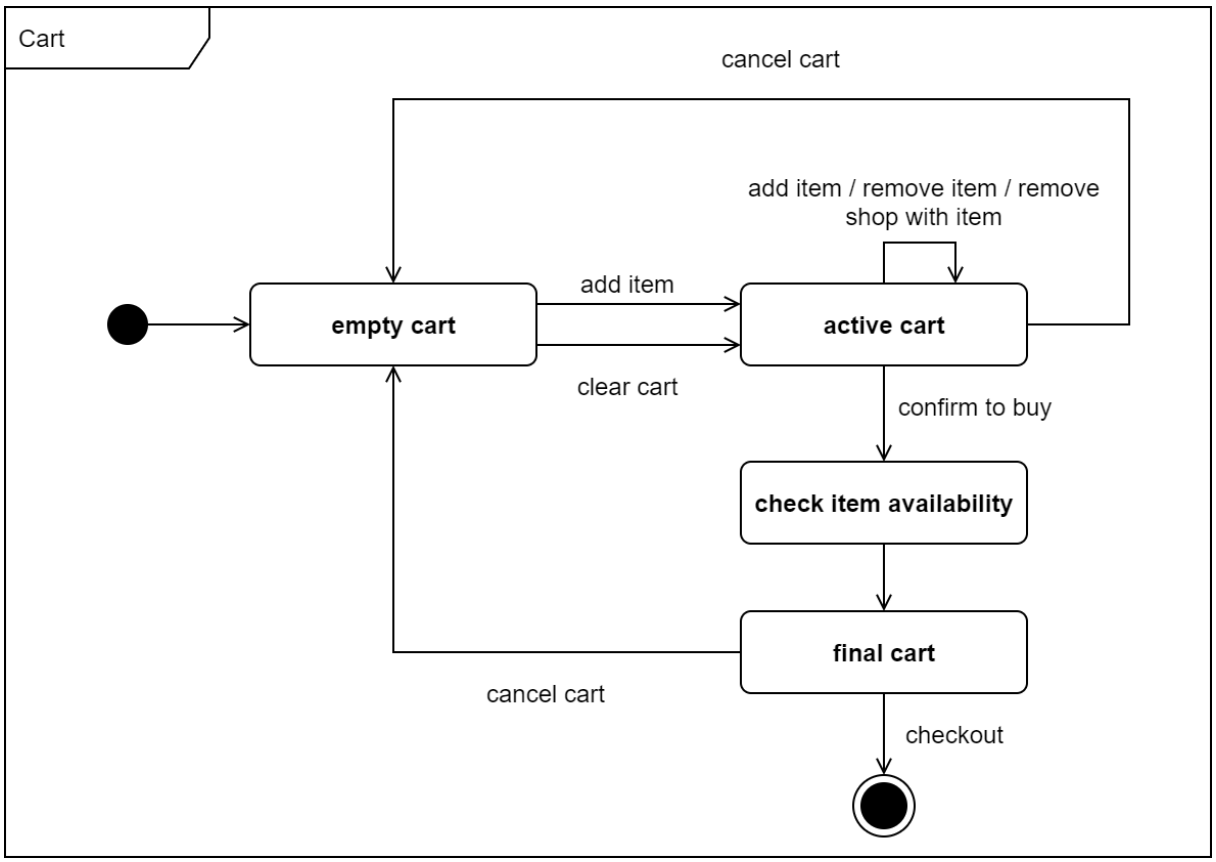


Figure 44: Cart State Transition

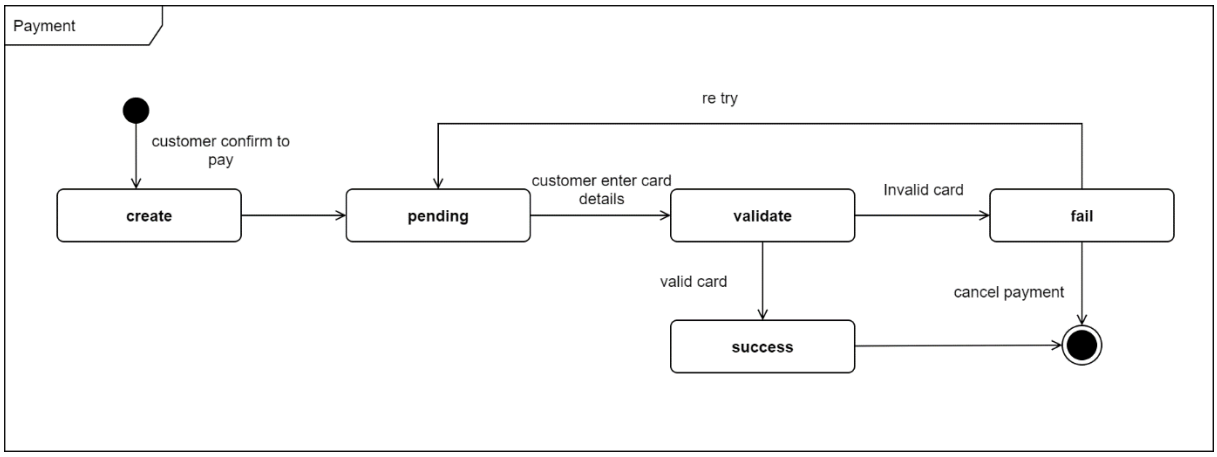


Figure 43: Payment State Transition

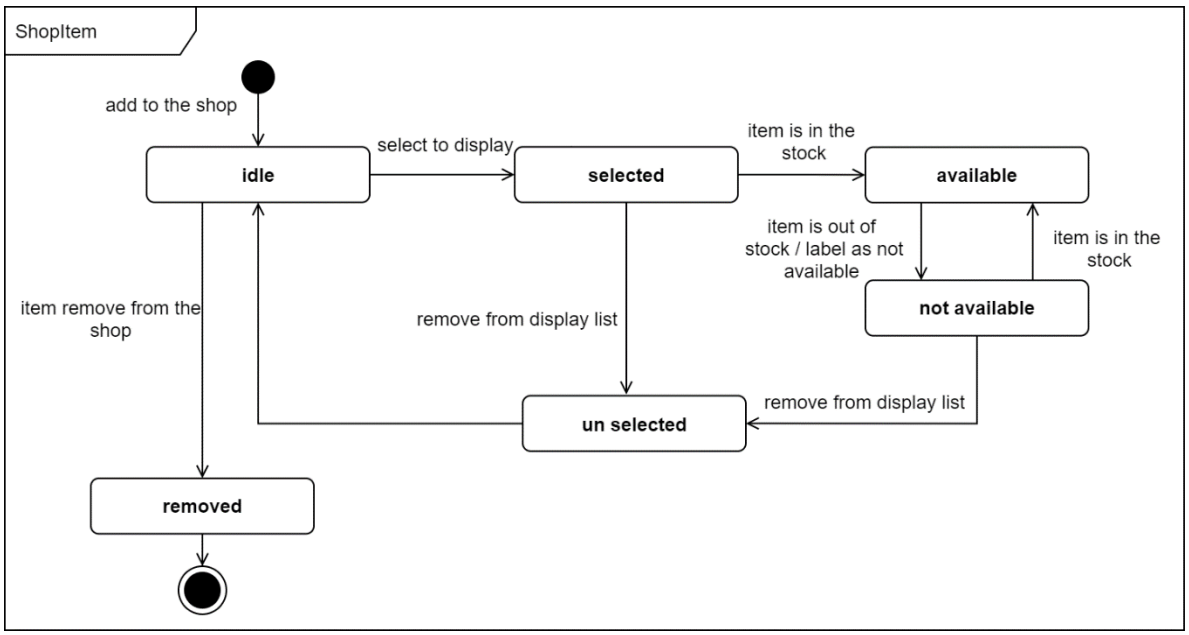


Figure 45: Shop Item State Transition

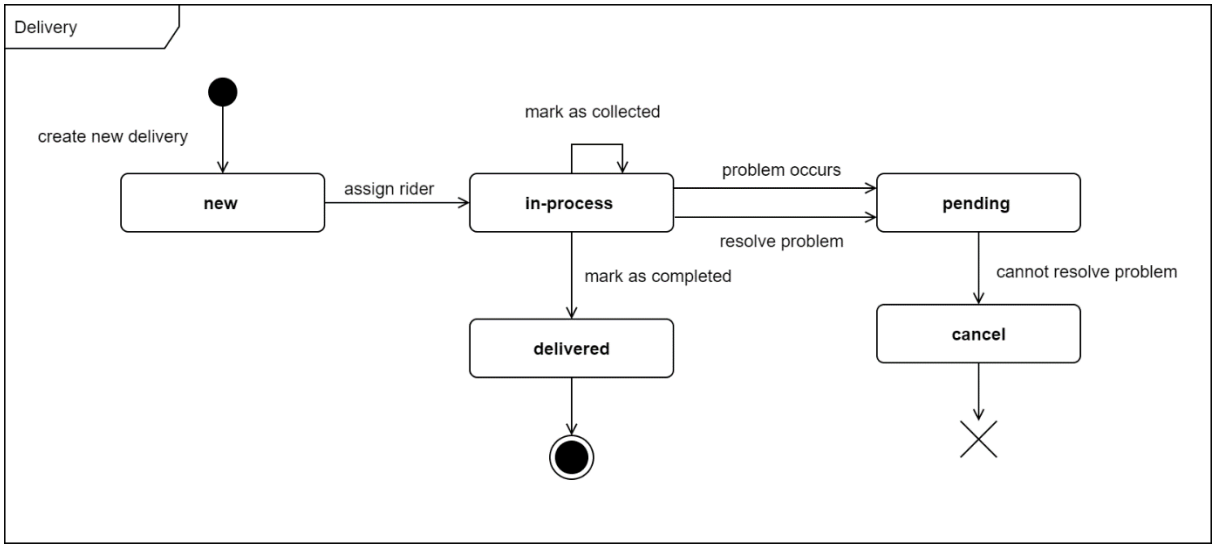


Figure 46: Delivery State Transition

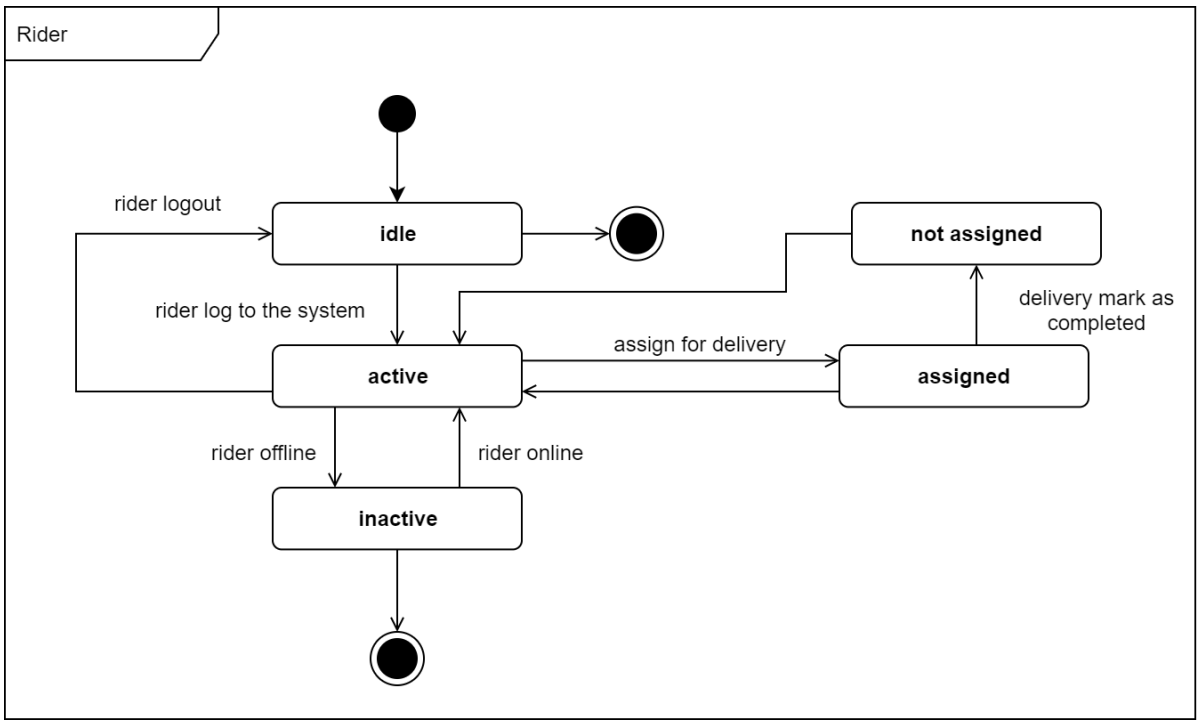


Figure 48: Role State Transition

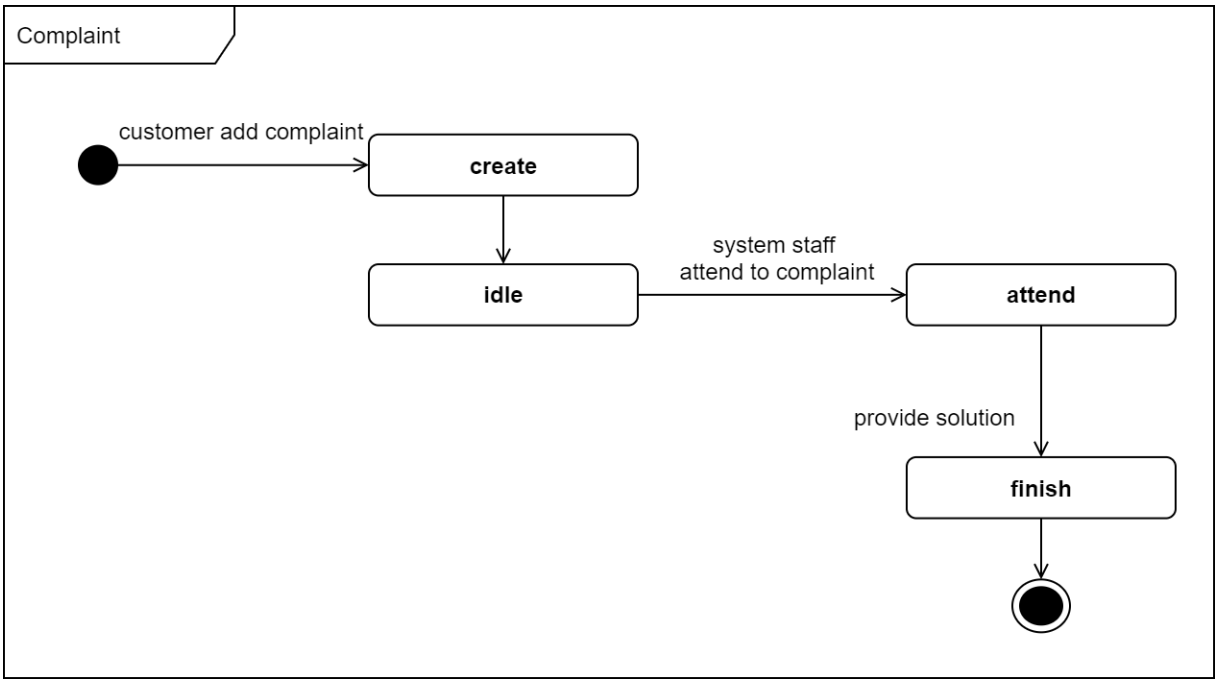


Figure 47: Complaint State Transition

6 User Interface Flow Diagram using wireframes

6.1 UI Flow

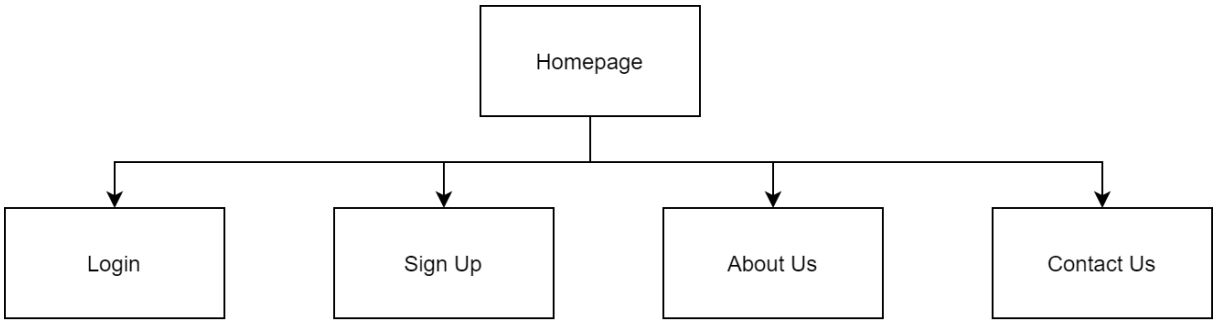


Figure 49:Homepage Flow

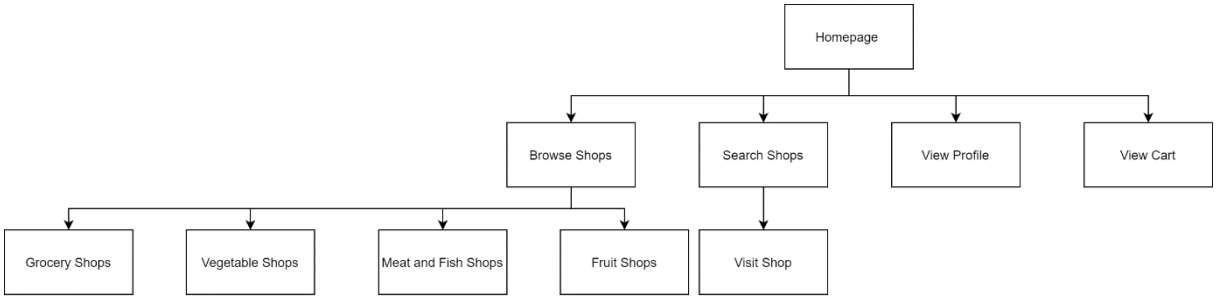


Figure 50: Customer Flow

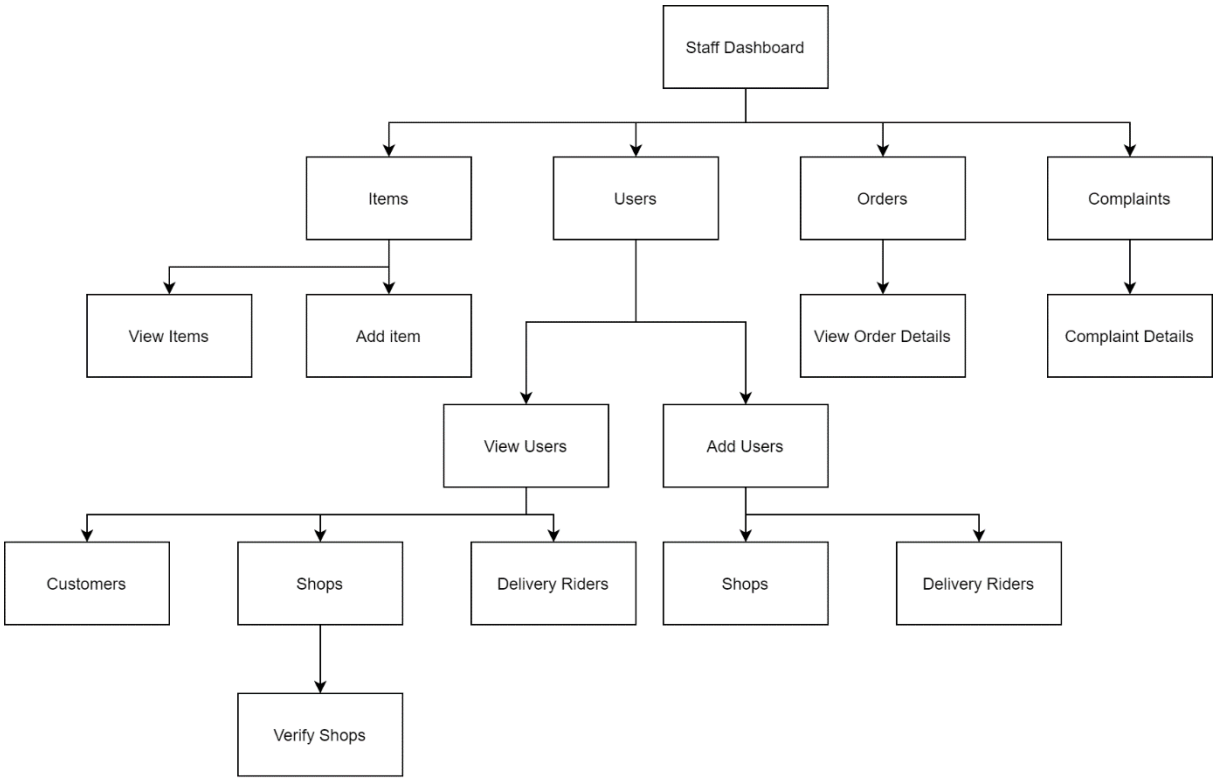


Figure 51: System Staff Flow

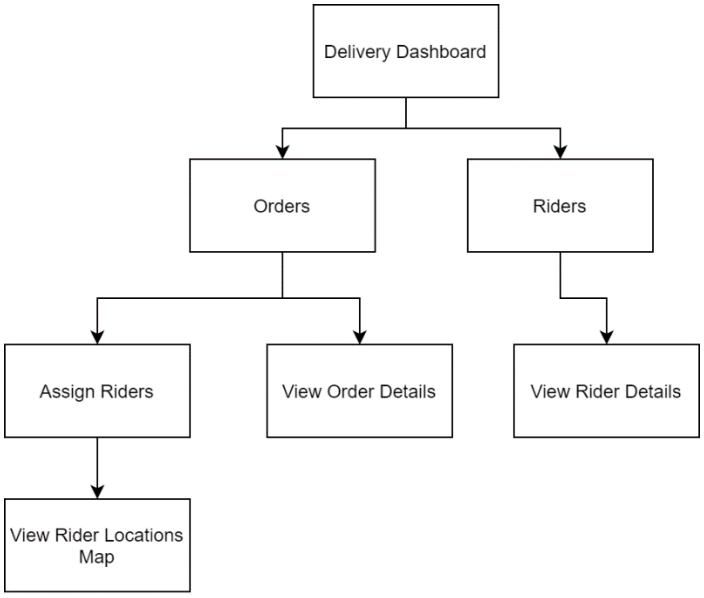


Figure 52: Delivery Staff Flow

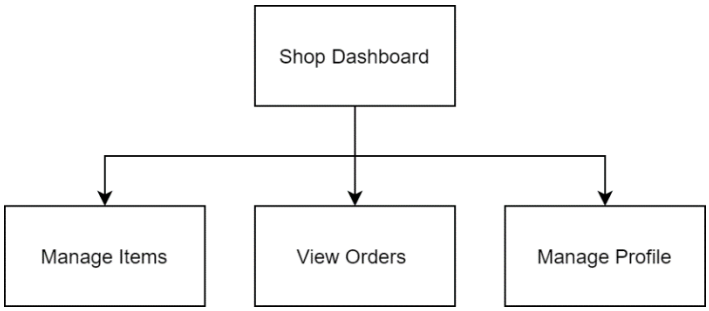


Figure 53: Shop Staff Flow

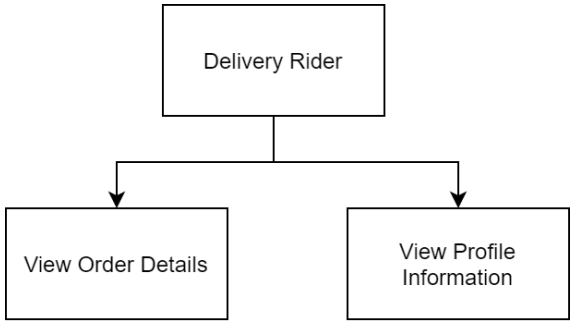


Figure 54: Delivery Rider Flow

6.2 User Interfaces

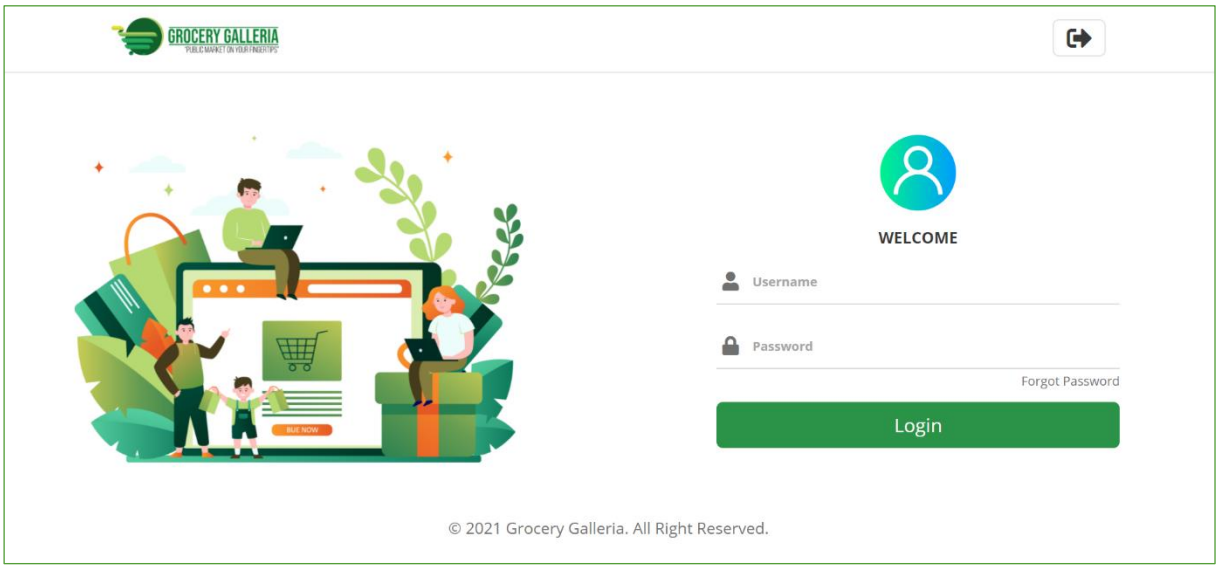


Figure 55: Login UI

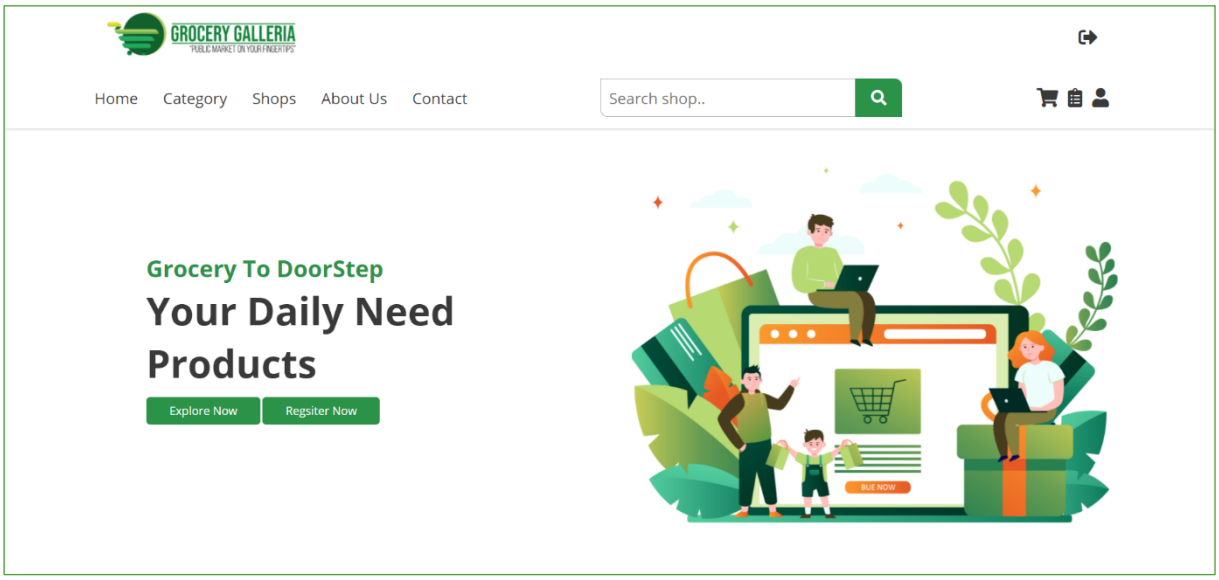


Figure 56: Front Page UI

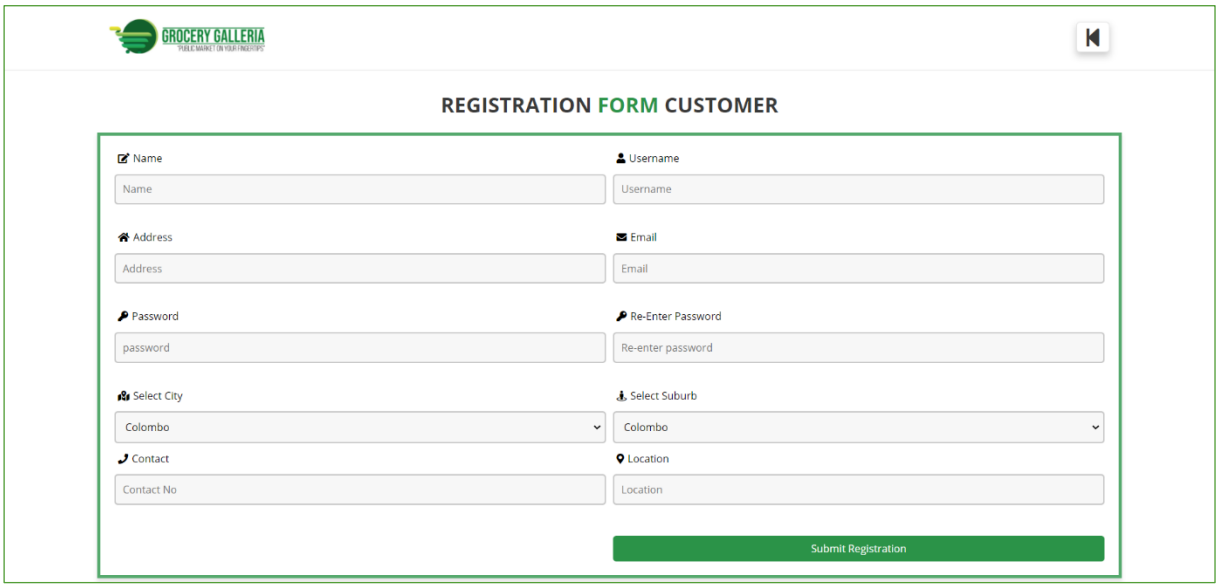


Figure 57: Registration Form UI

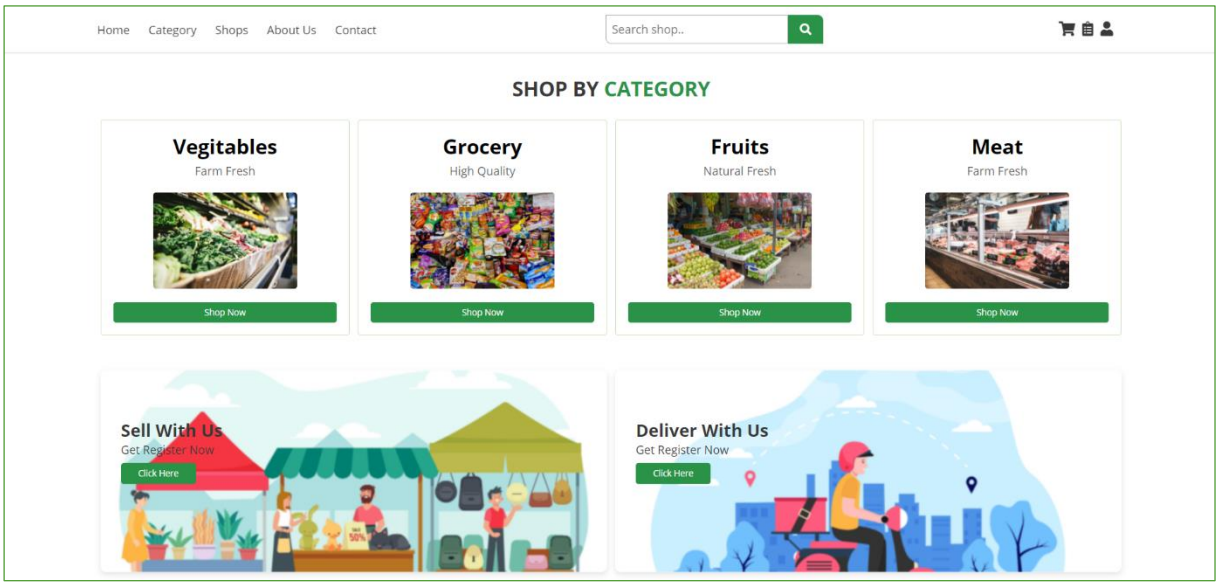


Figure 58: Shop Category & User Card UI

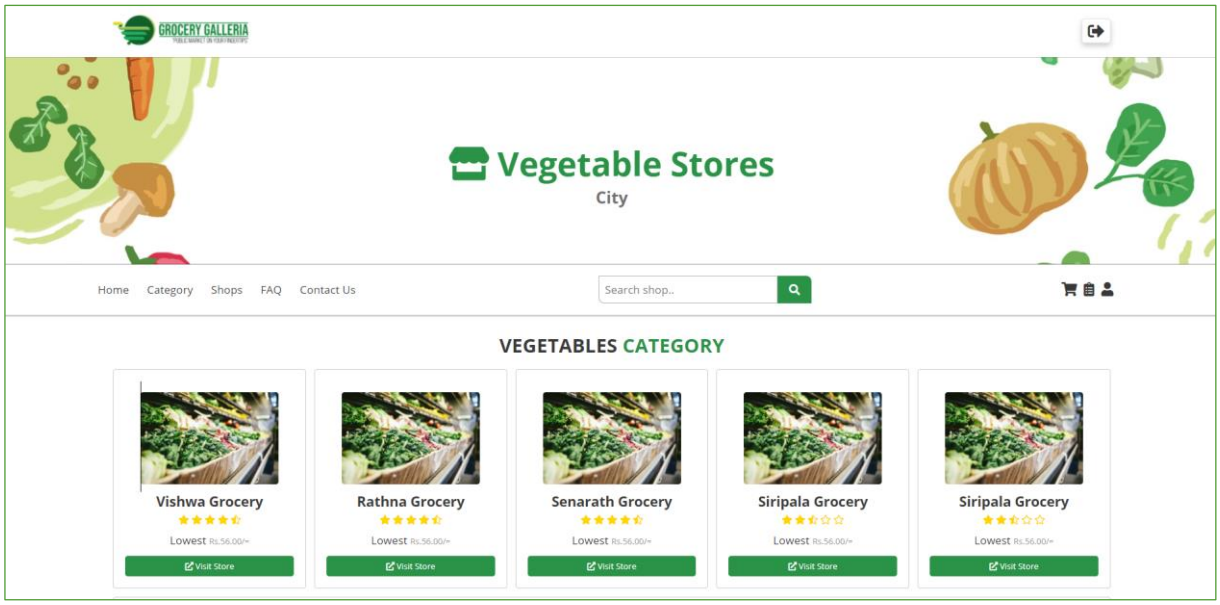


Figure 59:Vegetable Shop Category UI

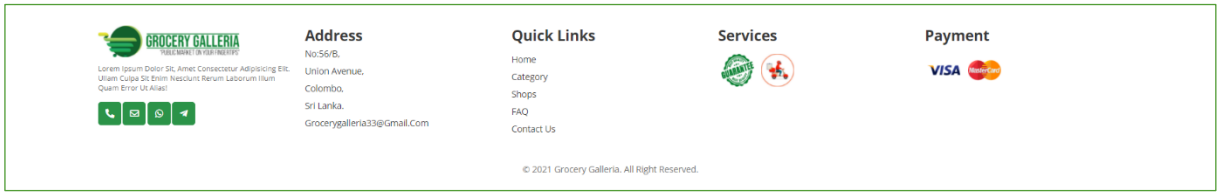


Figure 61: Footer design UI

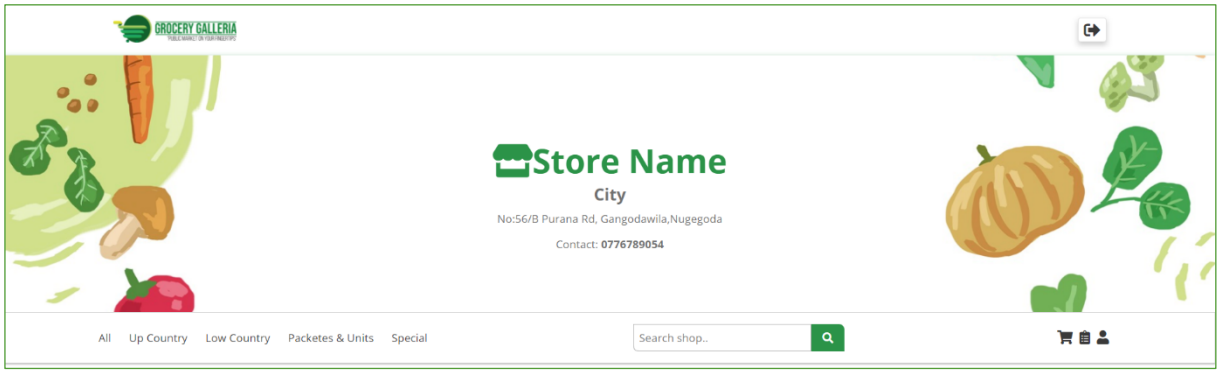


Figure 60: Shop Page UI

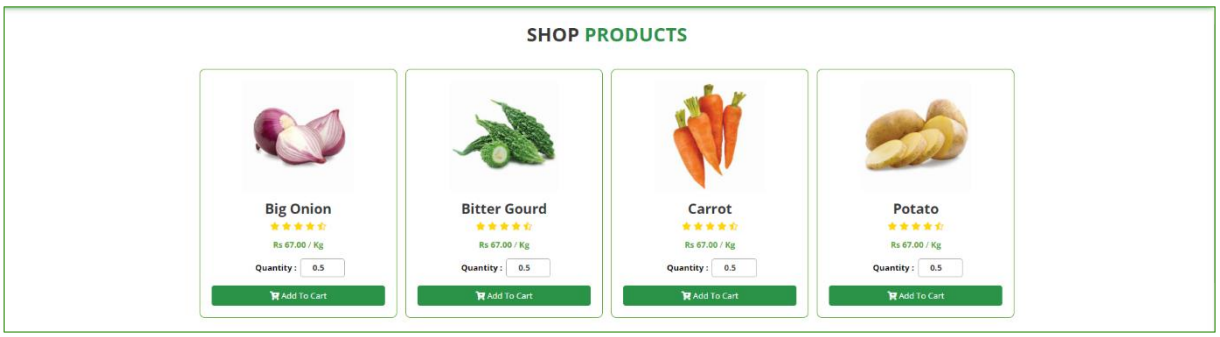


Figure 62:Shop Item Card Design

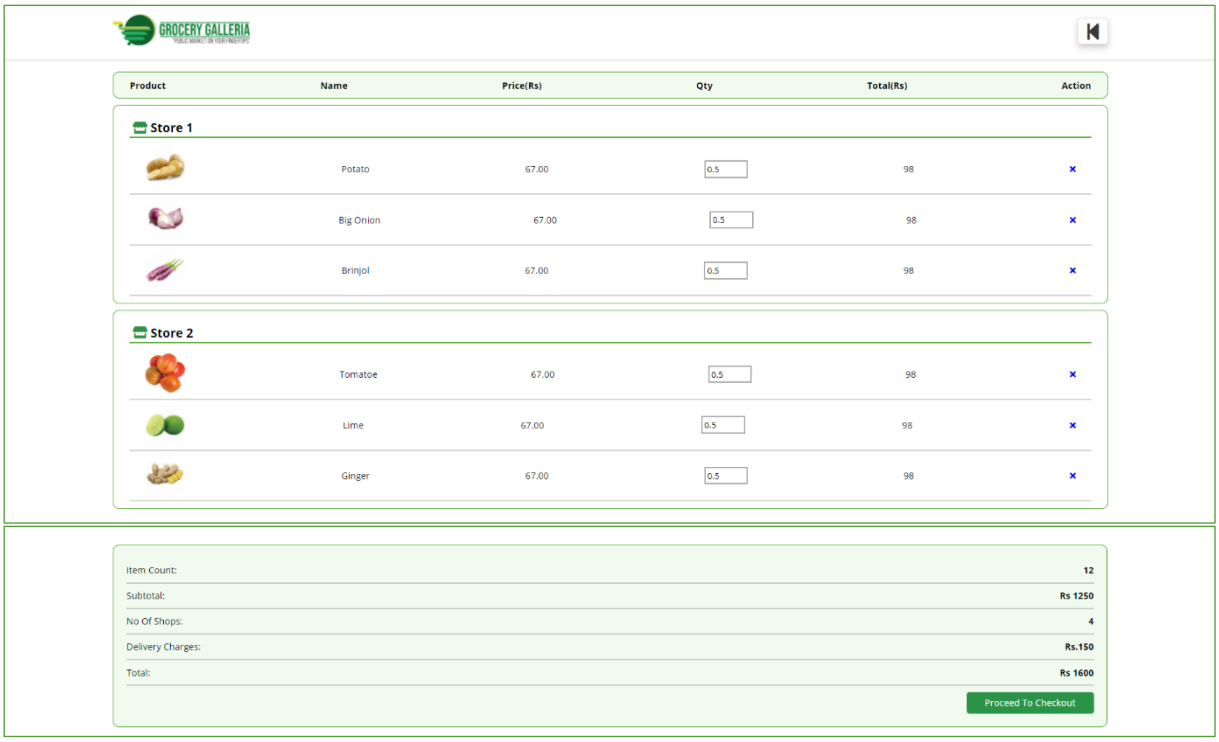


Figure 65: Customer Cart UI

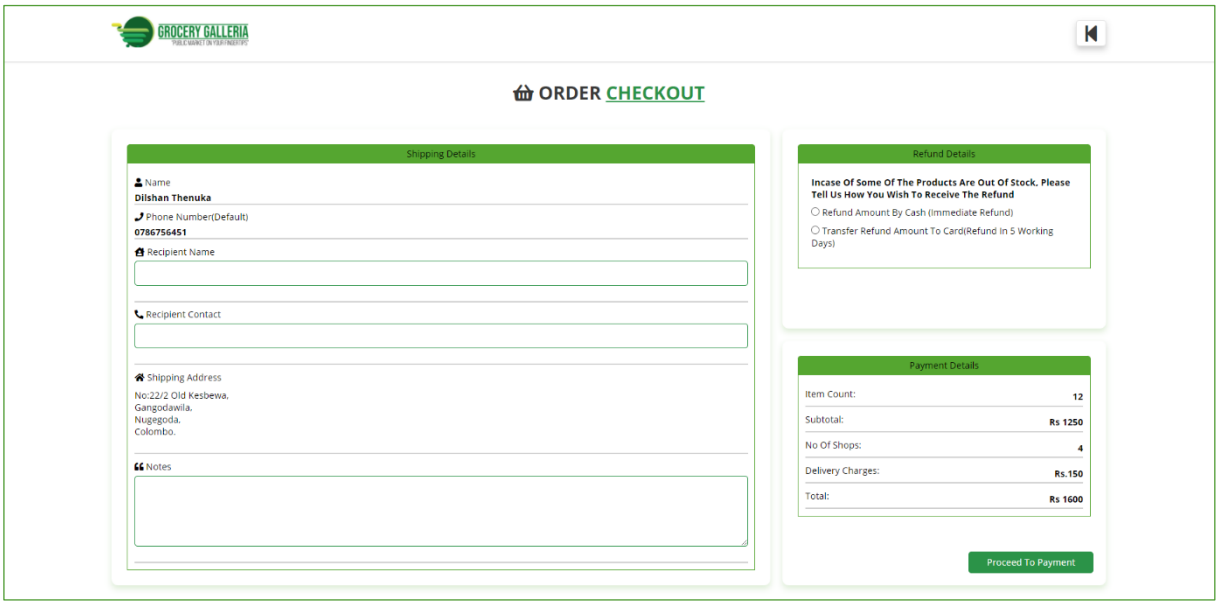


Figure 63: Order Checkout Page UI

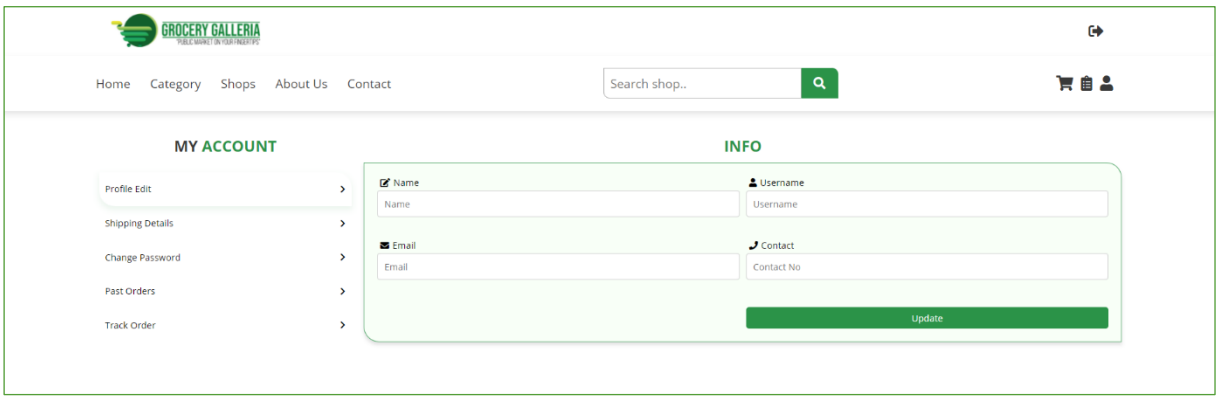


Figure 64: Customer Profile Page UI

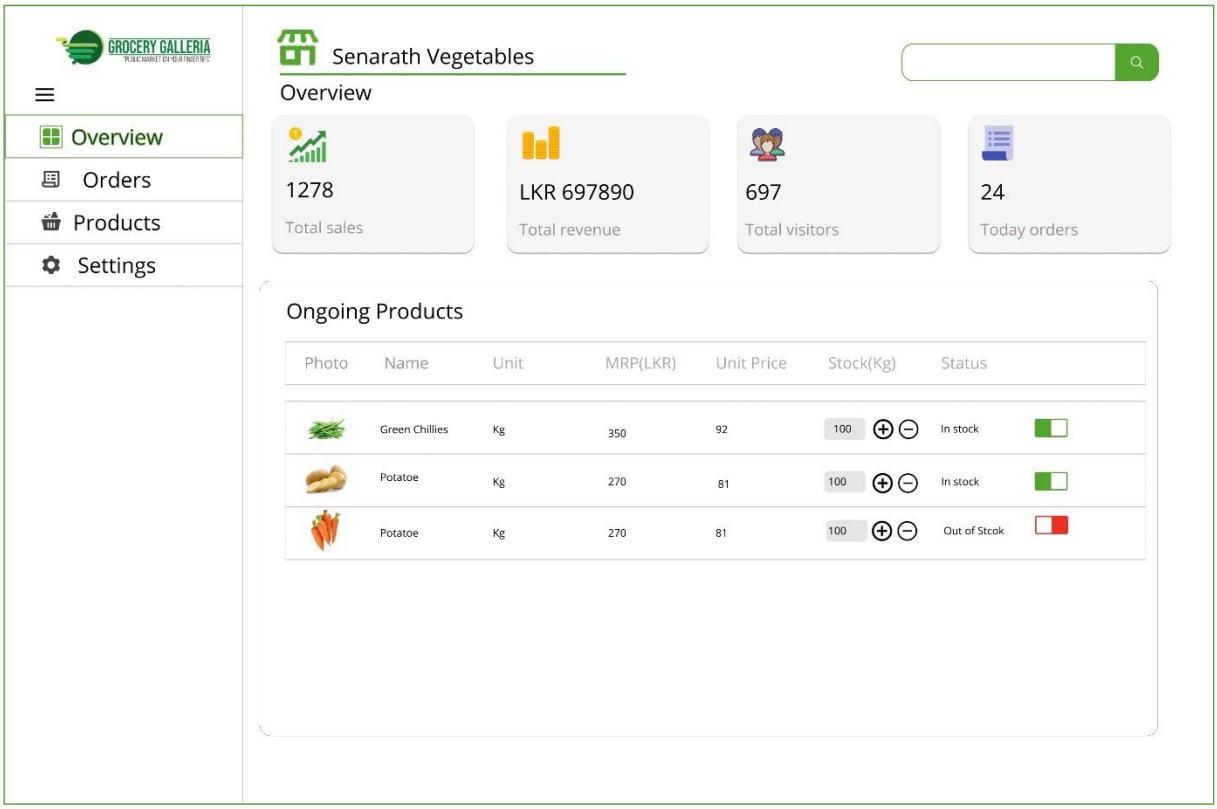


Figure 67: Shop Dashboard UI

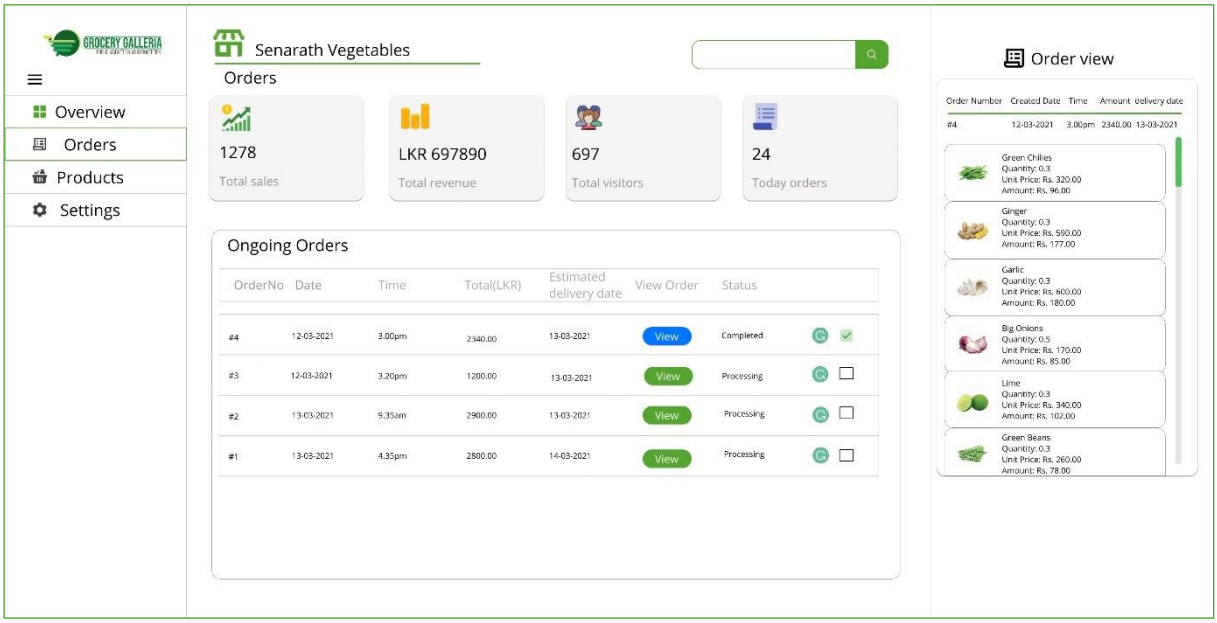


Figure 66: Shop Order View UI

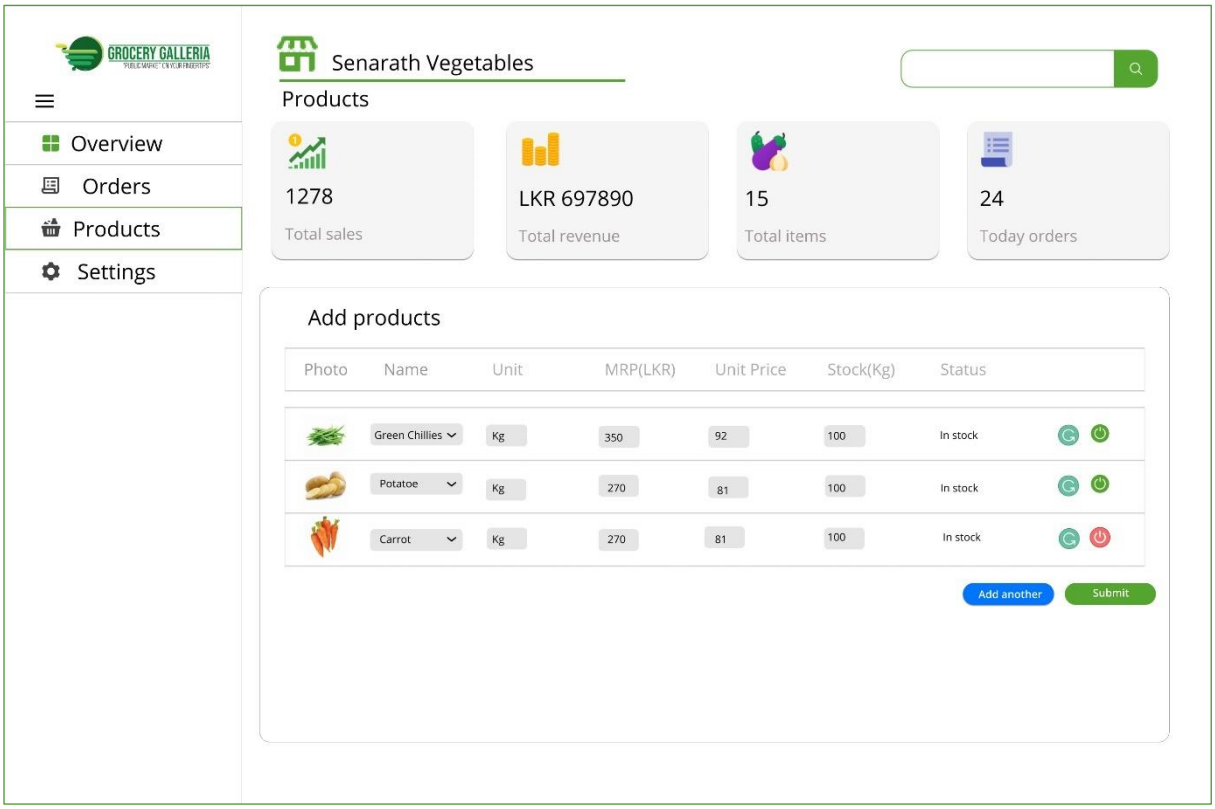


Figure 68:Shop Items Handling UI

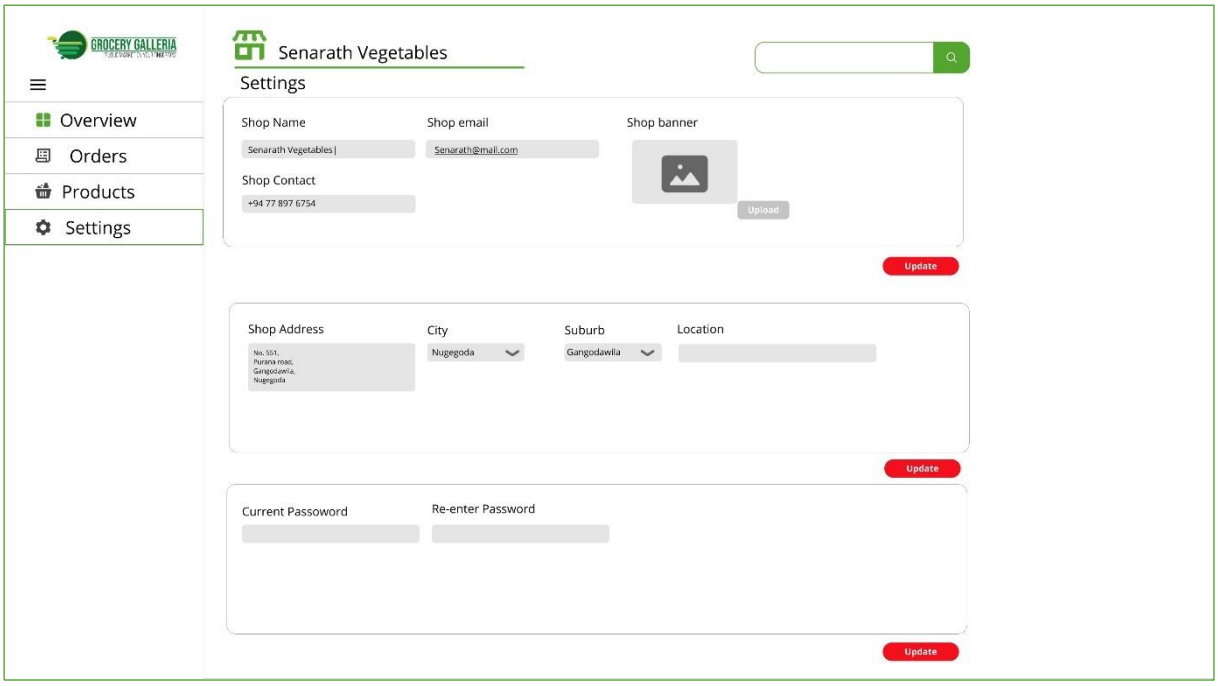


Figure 69: Shop Profile Page UI

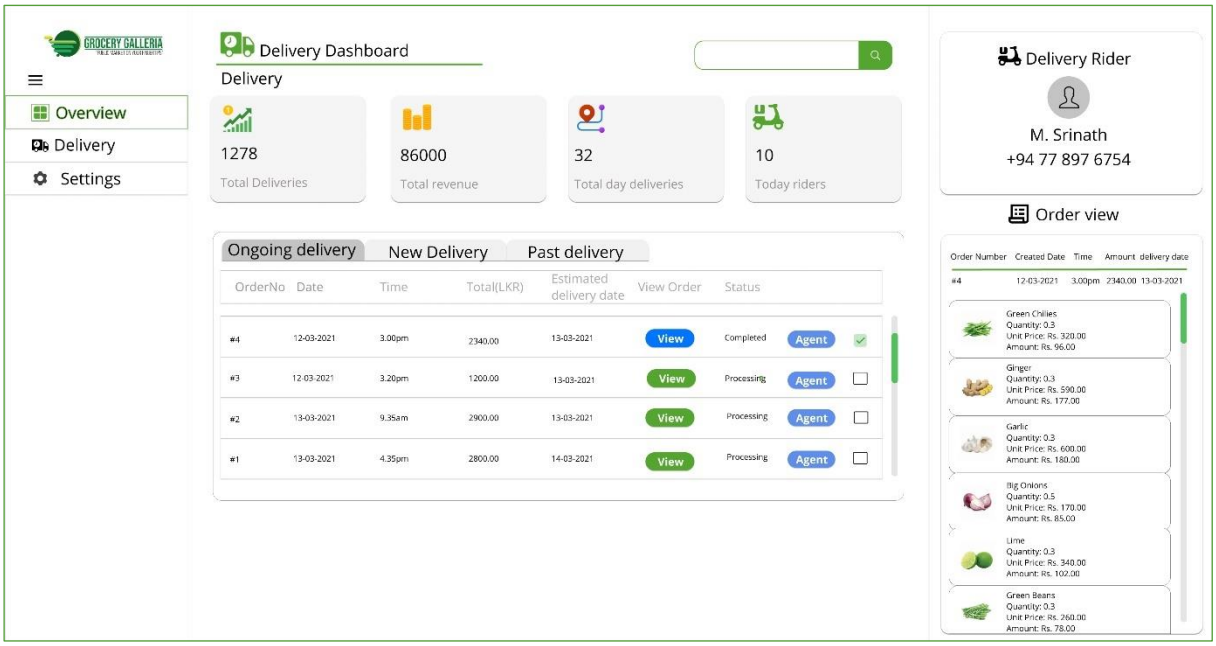


Figure 71: Delivery Dashboard UI

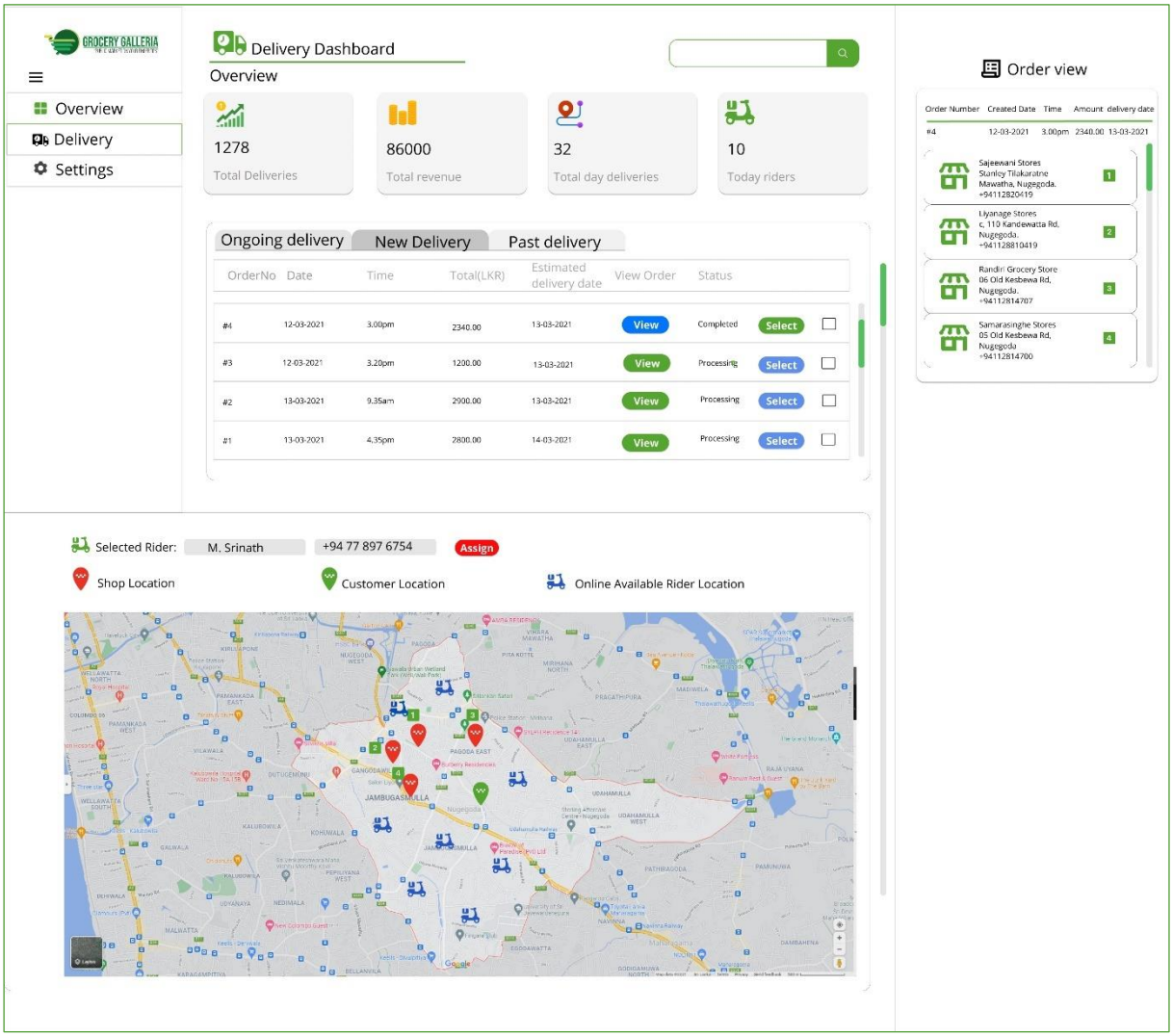


Figure 70: Order Details UI

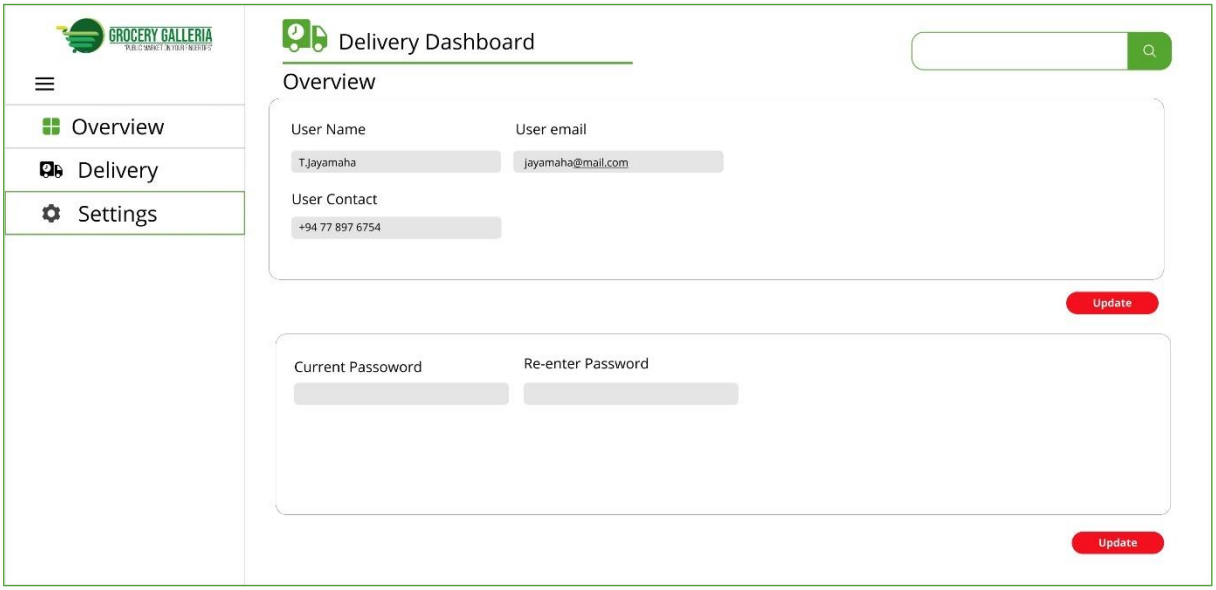


Figure 72: Delivery Profile Page UI

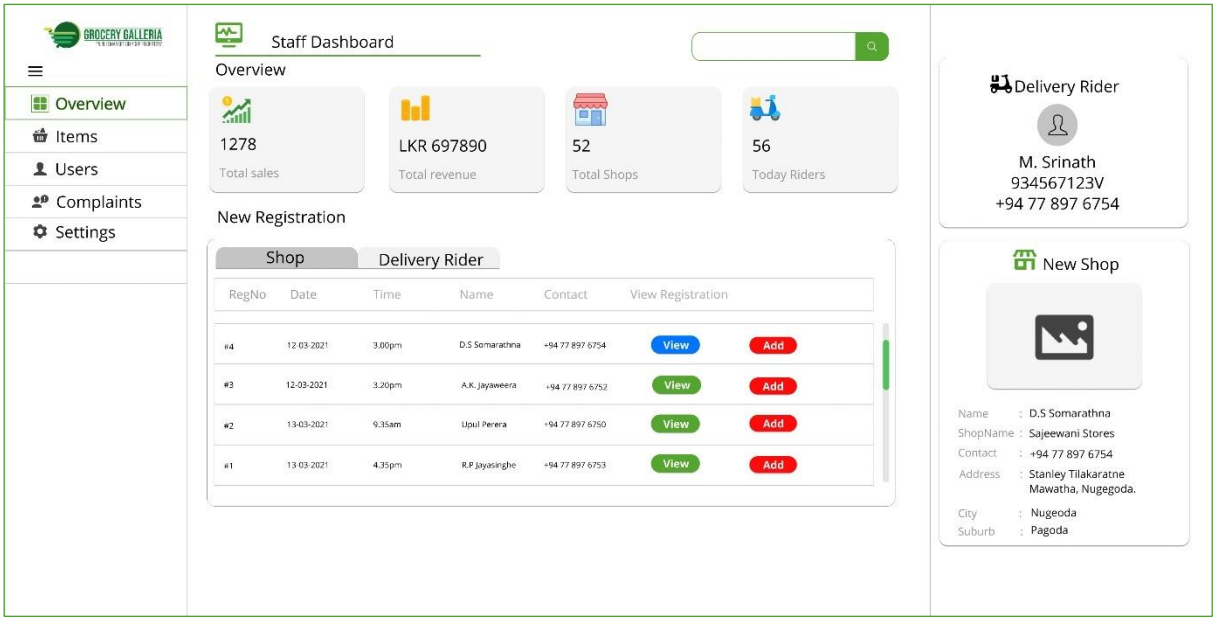


Figure 73: Staff Dashboard UI

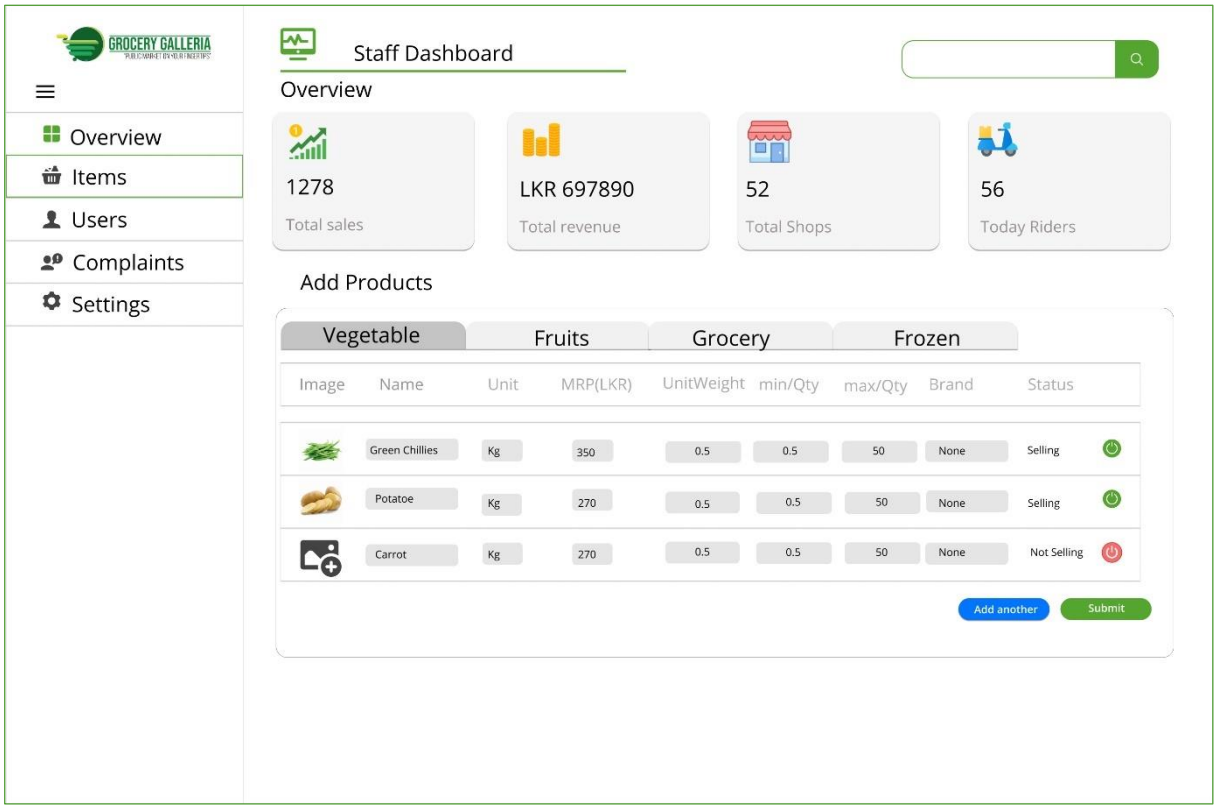


Figure 75: Staff Add Items UI

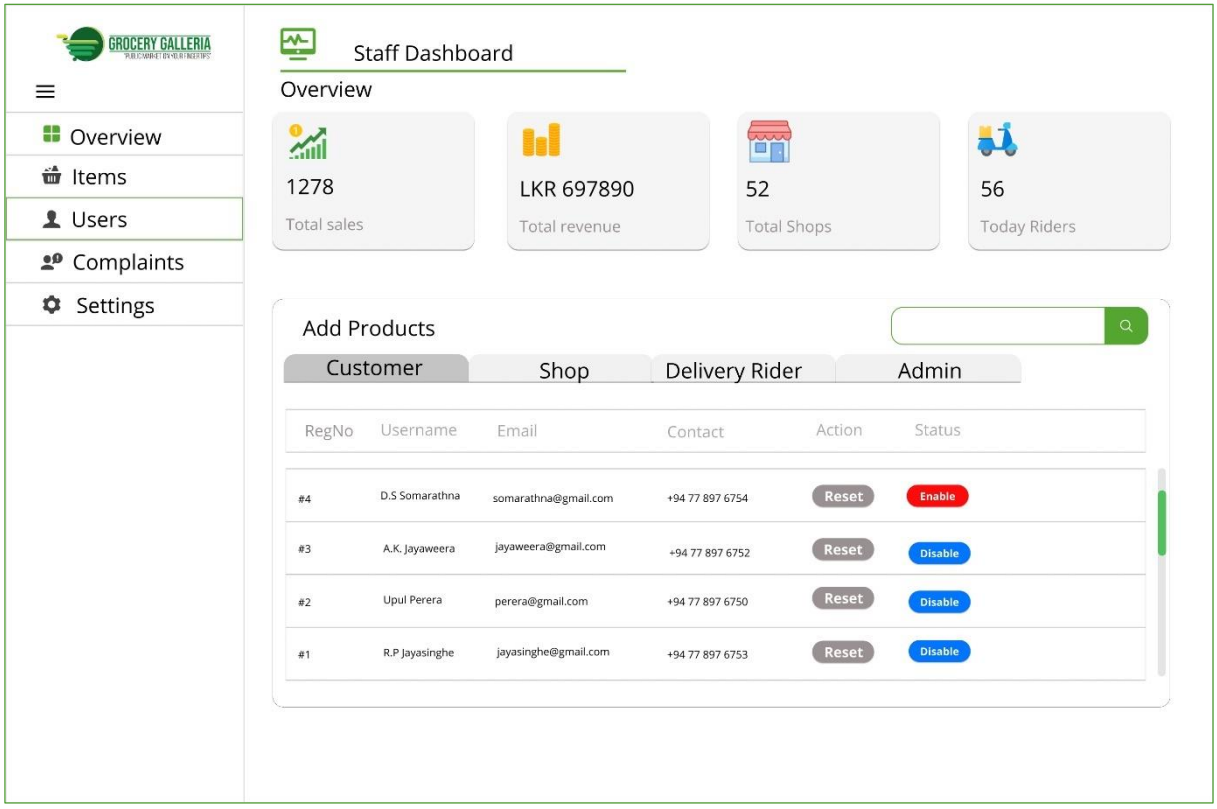


Figure 74: Staff User Handling UI

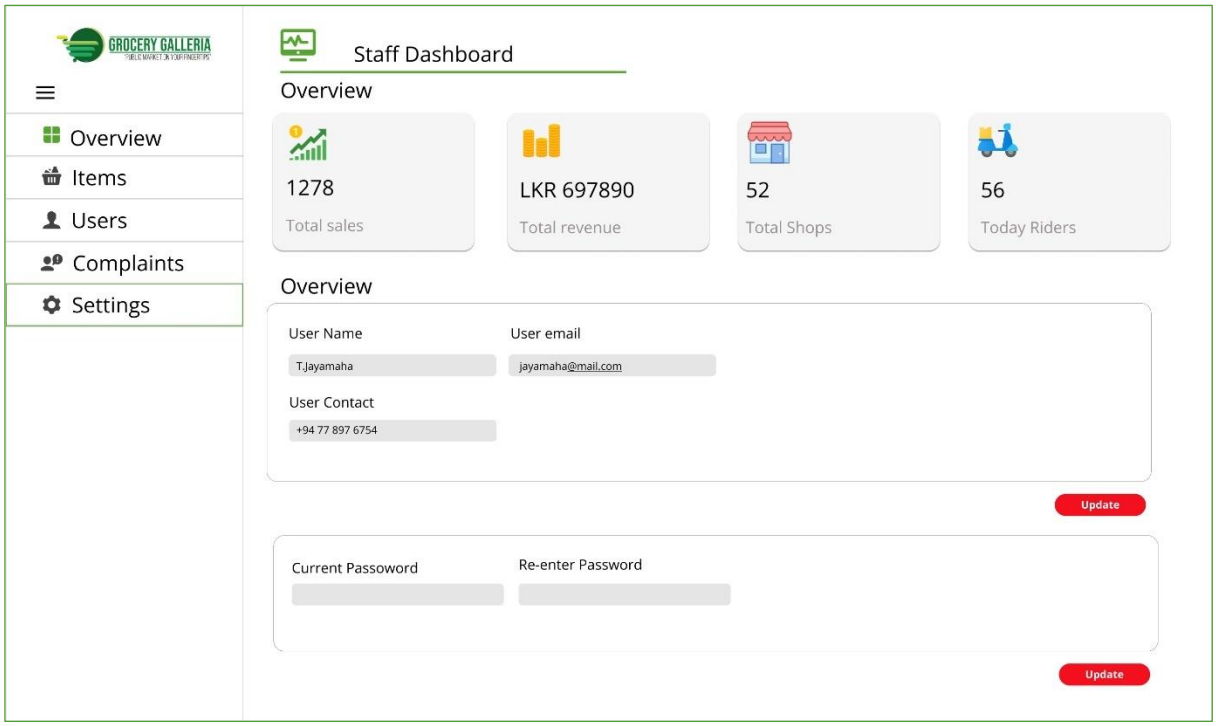


Figure 77: Staff Dashboard Profile UI

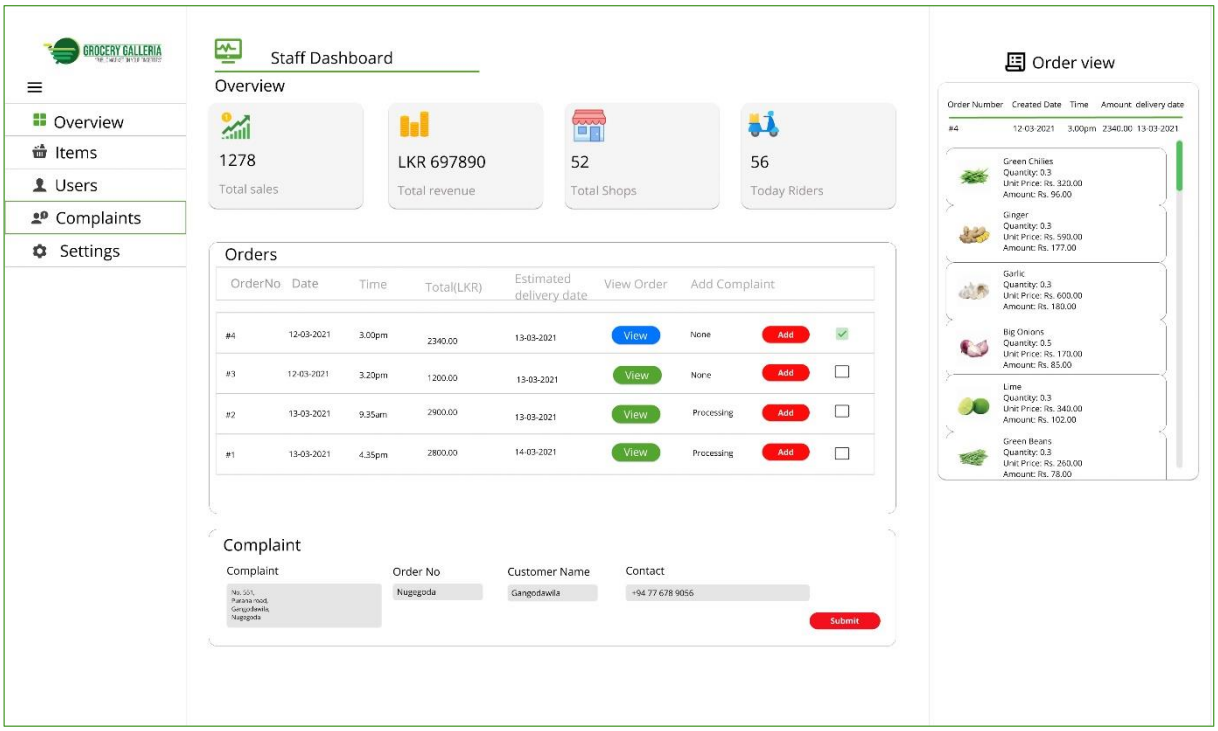


Figure 76: Staff Complaints Handling UI

7 Declaration

*We as members of the project titled "**Grocery Galleria**", Certify that we will carry out this project according to the guidelines provided by the coordinators and supervisors of the course as well as we will not incorporate, without acknowledgment, any material previously submitted for a degree or diploma in any university. To the best of our knowledge and brief, the project work will not contain any material previously published or written by another person or ourselves except where due reference is made in the text of appropriate places.*

Index Number	Name of the Student	Signature
19001789	K.K.W. VISHWAJITH	
19001673	S.A. DILSHAN THENUKA	
19001541	S.S.D. SANDUNI SANDEEPA	

8 Appendix

8.1 Use case narratives

8.1.1 Common Use Cases

Use Case ID:	1		
Use Case Name:	**Registration**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	*6/13/2021*	*Date Last Updated:*	

Actors:	Unregistered User (Unregistered Customers, Shops, Delivery Rider)
Purpose:	Allows an unregistered user to register to the system by submitting a sign-up form.
Preconditions:	None
Description:	To use the system functionalities, the users should have to register inside the system first. So that, unregistered users should have to register using the registration process by submitting a registration form to the system.
Exceptions:	None
Special Requirements:	None
Postconditions:	The **Submit Registration details** use case must be triggered.
Assumptions:	None

Table 1: Unregistered User – Registrations

Use Case ID:	2		
Use Case Name:	**Submit Registration details**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Unregistered User (Unregistered Customers, Shops, Delivery Rider)
Purpose:	Directs the signup form to the system staff for further verifications.
Preconditions:	The unregistered user should fill the sign-up form before submitting it.
Description:	Unregistered user fills the sign-up form provided by the system and process the sign-up foam and directs the foam to the system staff for the further verification. If an immediate validation error is caused while processing the submitted form by the system, it will be displayed to the user and him/she has to resubmit the foam before proceed further.
Exceptions:	• Unregistered customer registration will be handled directly by the system without final verification from system staff. • If the registration is not successful, the user has to resubmit the sign-up form.
Special Requirements:	None
Postconditions:	The sign-up form must be directed to the system staff and after validation, user registration by system staff creates new login in the system.
Assumptions:	None

Table 2:Unregistered User - Submit Registration Details

Use Case ID:	3		
Use Case Name:	**Create Login**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Unregistered User (Unregistered Customers, Shops, delivery Rider)
Purpose:	Create a login in the system and save credentials in the system database
Preconditions:	Submit Registration Details
Description:	After the unregistered user submits registration details then if submitted details are valid create a login for the user.
Exceptions:	• Submitted details are not valid • Disapproval by system staff
Special Requirements:	Need to have the valid data entered and system staff approvement for a new shop or new delivery rider registration.
Postconditions:	None
Assumptions:	None

Table 3: Unregistered User - Creare Login

Use Case ID:	4		
Use Case Name:	**User Login**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Registered User
Purpose:	Allows a registered user to login to the system:
Preconditions:	The user must be a registered user.
Description:	When a registered user enters Username & Password. The **Verify Login** use case verifies the login credentials and if the login verification failed, the appropriate message will be shown to the registered user.
Exceptions:	None
Special Requirements:	The Reset Password option must be displayed and the **Reset Password** use case must be triggered if the user requests it.
Postconditions:	The **Verify Login** use case must be triggered.
Assumptions:	None

Table 4:Registered User - User Login

Use Case ID:	5		
Use Case Name:	**Reset Password**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Registered User
Purpose:	Allows the user to reset the password for a given username
Preconditions:	Registered users must be unable to log in.
Description:	i. User Enters Username. ii. User Enters Email or Mobile telephone number for the respective account. iii. If the account details are correct Then send a reset link to your Email or Mobile Number. Reset Password Update database iv. Else Re-enter details
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 5:Registered User - Reset Password

Use Case ID:	6		
Use Case Name:	**Verify Login**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Registered User
Purpose:	Verifies the login of a registered user.
Preconditions:	The **Login** use case must be triggered.
Description:	Once the registered user enters the login credentials using the Login process, it will be redirected to the Verify Login process to validate the login credentials. There could be four results from this Verify Login process, they are following, i. User name & Password is correct: The message "Successfully Logged in.' ii. The username is correct but Password is incorrect OR The username is incorrect. Message "Wrong Username or Password."
Exceptions:	If the login failed, the registered user has to resubmit the login credentials.
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 6:Registered User - Verify Login

Use Case ID:	7		
Use Case Name:	**View Profile**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged User
Purpose:	Allows logged every user to view the profile.
Preconditions:	None
Description:	Once the user logged in, the user can view the profile page and check the profile details. And the user can update their profile, change the password, as options from the view profile page.
Exceptions:	None
Special Requirements:	Update Profile, Change Password, options should be displayed to trigger the relevant use case when the user wants it.
Postconditions:	None
Assumptions:	None

Table 7:Logged User - View Profile

Use Case ID:	8		
Use Case Name:	**Update Profile**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged User
Purpose:	Allows logged users to update the profile details.
Preconditions:	**View Profile** use case must be triggered.
Description:	When the logged user wants to update the profile information like Email, Name, Contact, Billing Address, etc. Then the user can use this process. The user has to refill the relevant fields of the provided form and hit the save.
Exceptions:	If the user missed pressing the save button before leaving, the update will not be saved.
Special Requirements:	Updated data must be valid to the system
Postconditions:	Under the update profile, the user can change the password. Then Change password use case must be triggered.
Assumptions:	None

Table 8:Logged User - Update Profile

Use Case ID:	9		
Use Case Name:	**Change Password**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged users to change the current password.
Preconditions:	**Update Profile** use case must be triggered.
Description:	If the logged user needs to change the current password in case of any security issue, then the user can change it using this process.
Exceptions:	If the user enters a wrong password, then the user cannot update the password.
Special Requirements:	The user has to include the current password first.
Postconditions:	None.
Assumptions:	Once the user changes the password, the previous password will not be valid anymore.

Table 9:Logged User - Change Password

Use Case ID:	10		
Use Case Name:	**Remove Profile**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged User
Purpose:	Allows logged every user to remove the profile from the system.
Preconditions:	None
Description:	If the user no longer wants to work with the system, they can remove their profile from the system.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 10::Logged User - Remove Profile

8.1.2 Customer

Use Case ID:	11		
Use Case Name:	**Browse Shops**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	The logged customer can browse the shops available on the site.
Preconditions:	The customer has to log into the system to browse the shops.
Description:	Once the customer logged into the system, he/she can see the available shops on the site on the homepage. And then the customer can browse the shops on the homepage. It has several shops which are categorized into All, Vegetables, Fruit, Fish, Meat and Grocery.
Exceptions:	None
Special Requirements:	The search Shops option should be displayed to trigger the **Search Shops** use case if the customer wants it.
Postconditions:	None
Assumptions:	None

Table 11:Logged Customer - Browse Shop

Use Case ID:	12		
Use Case Name:	**Search Shops**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to search a shop available on the site.
Preconditions:	**Browse shops'** use case must be triggered.
Description:	Once the customer logged in to the site, if the customer needs to directly buy the items from a particular shop, then the customer can search the shop by providing the shop name on the search bar.
Exceptions:	If the searched shop is not available, the not available message will be displayed.
Special Requirements:	Shop search suggestions should be provided when typing the shop name on the search bar by the customer.
Postconditions:	None
Assumptions:	None

Table 12:Logged Customer - Search Shops

Use Case ID:	13		
Use Case Name:	**View Shop**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to view the chosen shop.
Preconditions:	The customer should select a shop first to view the shop.
Description:	Once the customer selects a shop by clicking the shop and then he/she can view the inside of the shop. When the customer is inside, the customer can see: i. The shop items with pricing. ii. The shop details
Exceptions:	None
Special Requirements:	Search item field and the Add to cart options should be displayed to trigger the **Search Items** and **Add to cart** use cases if the customer requests it by clicking.
Postconditions:	Customers can search for items in the shop if they need to.
Assumptions:	Customers can only view the items which are relevant to the shop.

Table 13:Logged Customer -View Shop

Use Case ID:	14		
Use Case Name:	**Search Items**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to search items inside the shop.
Preconditions:	The **View Shop** use case must be triggered.
Description:	Once the customer is inside a shop, he/she can search for an item that is relevant to the shop. Only the items available inside the shop can be searched.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 14:Logged Customer - Search Items

Use Case ID:	15		
Use Case Name:	**View Item**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Logged customers customer can view items and item details.
Preconditions:	Must be entered in to shop before viewing items.
Description:	Once customers enter the selected shop, they can view a list of items. And also, customers can select items and can view details of them.
Exceptions:	None
Special Requirements:	None
Postconditions:	If the customer wants, they can be added items to the cart. Then **Add to cart** use case must be triggered.
Assumptions:	None

Table 15:Logged Customer - View Item

Use Case ID:	16		
Use Case Name:	**Add to cart**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to add items to the shopping cart from the selected shop.
Preconditions:	The **View Item** use case must be triggered.
Description:	Once the customer insides a shop, he/she can add items to the shopping cart from that shop. When adding the items to the shopping cart, the **Calculate Total** use case must be triggered to calculate the total of the items.
Exceptions:	None
Special Requirements:	None
Postconditions:	The **Calculate Shop Total, Calculate Cart Total** use cases must be triggered.
Assumptions:	None

Table 16:Logged Customer - Add to Cart

Use Case ID:	17		
Use Case Name:	**Calculate Cost**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to get the total value of the items in the shopping cart.
Preconditions:	**Add to cart** or **Update Shopping Cart** use case must be triggered.
Description:	Whenever the customer adds items to the cart or updates the cart, the Calculate total process must be triggered to have the current total value of the shopping cart.
Exceptions:	None
Special Requirements:	None
Postconditions:	The calculated value must be redirected to the Shopping cart.
Assumptions:	None

Table 17:Logged Customer - Calculate Cost

Use Case ID:	18		
Use Case Name:	**View Shopping Cart**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/14/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows the logged customer to view the shopping cart and get the total item cost.
Preconditions:	None
Description:	The logged customer can simply view the shopping cart by clicking the cart icon on the home page and it will trigger the View Shopping cart use case and It will show the following details, i. Shop list with the total price. ii. Separated item lists with details for each shop in the cart and the total price for each. iii. Update Shopping cart options.
Exceptions:	None
Special Requirements:	The update shopping card option should be displayed to trigger the **Update Shopping cart** use case if the customer requests it.
Postconditions:	None
Assumptions:	None

Table 18:Logged Customer - View Shopping Cart

Use Case ID:	19		
Use Case Name:	**Update Shopping Cart**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/14/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customer to update their shopping cart.
Preconditions:	The **View shopping cart** use case must be triggered.
Description:	Customer can update the shopping cart as an option when he/she view the shopping cart. Customers can update the shopping cart in two ways: i. Update the shops in the cart**.** If a customer needs to remove a shop from the cart with the items. ii. Update the items in the cart**.** If the customer needs to change the quantity or remove an item from a shop.
Exceptions:	None
Special Requirements:	None
Postconditions:	The **Calculate Total** use case must be triggered.
Assumptions:	The only use case is where the customer can update the shopping cart before proceeding to checkout.

Table 19:Logged Customer - Update Shopping Cart

Use Case ID:	20		
Use Case Name:	**Remove Item**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to remove the items in the shopping cart.
Preconditions:	**Shopping cart Update** must be triggered.
Description:	When the shopping cart is updating, the items included in the cart can be removed.
Exceptions:	None
Special Requirements:	None.
Postconditions:	**Calculate Cost** must be triggered.
Assumptions:	None

Table 20:Logged Customer - Remove Item

Use Case ID:	21		
Use Case Name:	**Remove Shop**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to remove the shops from the shopping cart.
Preconditions:	**Shopping cart Update** must be triggered.
Description:	The selected shop can be removed with all items in the shopping cart.
Exceptions:	None
Special Requirements:	None.
Postconditions:	**Calculate Cost** must be triggered to calculate the new total value of the cart and calculate shop totals.
Assumptions:	None

Table 21:Logged Customer - Remove Shop

Use Case ID:	22		
Use Case Name:	**Update Item Quantity**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	To change selected item quantity in the shopping cart
Preconditions:	**Shopping cart Update** must be triggered.
Description:	Customers can change item quantity in the shopping cart. They can increase or decrease. They can change the quantity in a way that does not exceed the maximum and minimum quantity values given to each item.
Exceptions:	The customer has restricted to change item quantity if exceeding minimum and maximum quantity values when they increasing or decreasing item quantities.
Special Requirements:	None.
Postconditions:	None
Assumptions:	None

Table 22:Logged Customer - Update Item Quantity

Use Case ID:	23		
Use Case Name:	**Make Purchase**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to make purchases through the system.
Preconditions:	None
Description:	A logged customer can make purchasing through this process. For that customer has to browse items from the shops and add items to the cart from the relevant shops before make proceed to the purchasing. And then after the customer can proceed to the checkout and purchase the system.
Exceptions:	None
Special Requirements:	Proceed to checkout button should be displayed in the shopping cart area to trigger the Make purchase use case if the customer requests it.
Postconditions:	**Checkout** use case must be triggered.
Assumptions:	None

Table 23: Logged Customer - Make Purchase

Use Case ID:	24		
Use Case Name:	**Checkout**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows the logged customer to proceed to finalize the order.
Preconditions:	**Make purchase** use case must be triggered.
Description:	Once the logged customer is done with the shopping, he/she can proceed to finalize the order and make the payments for that. The flow is: i. View the final shopping cart and get the item total. ii. Get the delivery cost from the Calculate Delivery cost process. iii. Calculate the total payment (Item total + delivery cost). iv. Get customer approval. v. Make payment
Exceptions:	If the payment failed, the customer can retry again without losing the shopping cart.
Special Requirements:	The update delivery details option should be displayed to trigger the **Update delivery details** use a case in case of billing address is different from the delivery address which makes it available to the customer to change it if need.
Postconditions:	**View Shopping cart, Calculate Delivery Cost, Make Payment** use cases must be triggered.
Assumptions:	• Once the payment is completed, the customer cannot cancel the order.

	• If the customer approval or the payment failed the order will be fully canceled and only the shopping cart will remain in the profile. • Once the order is successful, the shopping cart will be cleared for the next order.

Table 24:Logged Customer - Checkout

Use Case ID:	25		
Use Case Name:	**Make Payment**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to make payments through the platform.
Preconditions:	**Checkout** use cases must be triggered.
Description:	Once the customer allows proceeding further with the payment, the customer will redirect to the payment gateway to make the payment via card and finalize the order placement.
Exceptions:	If the payment has failed, the order will not be placed and the customer will return to the checkout window and the customer can try again to make the payment.
Special Requirements:	The customer has to approve the pop-up message to proceed further with the payment process to make the placement.
Postconditions:	The order placement details should be sent to the relevant stakeholders to process the order.
Assumptions:	The customer can make the payment only as a card payment (No cash on delivery). Customers can take the refund as an immediate cash refund in case of sudden item unavailability.

Table 25:Logged Customer - Make Purchase

Use Case ID:	26		
Use Case Name:	**Calculate Delivery Cost**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	
Actors:	Logged Customer		
Purpose:	Allows checkout process to get the total delivery cost.		
Preconditions:	**Checkout** use case must be triggered.		
Description:	When the customer is in the checkout window, he/she can see the total delivery cost based on the number of shops in the shopping cart.		
Exceptions:	None		
Special Requirements:	The calculated delivery cost must be shown in the checkout process window.		
Postconditions:	None		
Assumptions:	Standard delivery costs will be maintained for every shop.		

Table 26: Customer - Calculate Delivery Cost

Use Case ID:	27		
Use Case Name:	**View Orders**		
Created By:	S.A Dilshan Thenuka	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Allows logged customers to view the orders.
Preconditions:	None
Description:	This area will provide all the orders the customer has made from this system including past and current order details as well, as a list of orders. However, the orders will be sorted based on the date which makes the new orders on the top. The customer provides several valuable information regarding the order.
Exceptions:	If the customer has no orders made through the system, the view order area will be empty.
Special Requirements:	The customer should be able to view the order detail by selecting a particular order.
Postconditions:	None
Assumptions:	None

Table 27:Customer - View Orders

Use Case ID:	28		
Use Case Name:	**Make Complaint**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Log customers can make complaints against the order that was made early.
Preconditions:	To make a complaint, the customer must have placed an order before.
Description:	If there is an issue in order, the customer can make a complaint against it via the system.
Exceptions:	The customer cannot complain if they have not placed a single order.
Special Requirements:	At least one order must have been made to make a complaint.
Postconditions:	The complaint must be redirected to the system staff.
Assumptions:	None

Table 28:Logged Customer - Make Complaint

Use Case ID:	29		
Use Case Name:	**View Complaint**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Customer
Purpose:	Log customers can view complaints that they made before.
Preconditions:	One or more complaints must have been made before viewing the complaints.
Description:	If the customer has complained about an order, they will be able to view it and see the solutions received from the system staff.
Exceptions:	The customer cannot complain if they have not placed a single order.
Special Requirements:	At least one order must have been made to make a complaint.
Postconditions:	The complaint must be redirected to the system staff.
Assumptions:	None

Table 29:Logged Customer - View Complaint

8.1.3 Delivery

Use Case ID:	30		
Use Case Name:	**View Delivery Notification**		
Created By:	S.S.D.S. Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Delivery Rider
Purpose:	View delivery notification when mobile app updated with a new delivery.
Preconditions:	None
Description:	Delivery Rider can view delivery notifications through the system when system staff assigns them to the new delivery process.
Exceptions:	None
Special Requirements:	Delivery riders must be pre-assigned by the delivery staff.
Postconditions:	None
Assumptions:	None

Table 30:Delivery Rider - View Delivery Notification

Use Case ID:	31		
Use Case Name:	**View Delivery Details**		
Created By:	S.S.D.S. Sandeepa	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Delivery Rider
Purpose:	View delivery details for a delivery process.
Preconditions:	None
Description:	The assigned delivery rider can view delivery details (customer information, shop addresses, order details, etc.) of the relevant order to accomplish the delivery process.
Exceptions:	None
Special Requirements:	Delivery riders must be pre-assigned by the delivery staff.
Postconditions:	**Mark as Collected** use case trigger when rider mark as the collected when he collect an order from every shop and **Mark as DElivered** use case trigger when rider marks the relevant deliver as complete.
Assumptions:	Only the rider assigned by the delivery company can view the relevant delivery details.

Table 31:Delivery Rider - View Delivery Details

Use Case ID:	32		
Use Case Name:	**Mark as Delivered**		
Created By:	S.S.D.S. Sandeepa	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged Delivery Staff/Delivery Rider
Purpose:	To confirm the delivery process of relevant order is finished
Preconditions:	The delivery staff has to trigger **View In-process Delivery** if they mark the particular delivery as delivered and the Delivery rider has to pass the **View Delivery Detail** use case before the mark as delivered.
Description:	After the rider hand over the ordered items to the relevant customer delivery rider or delivery staff can mark that delivery is completed. Then particular delivery disappears from in-process status and saves in completed status.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	Each delivery is marked as completed only after it has actually been completed and only if the relevant rider does not mark it as completed then delivery staff will do it.

Table 32:Logged Delivery Staff / Delivery Rider - Delivery Mark as Delivered

Use Case ID:	33		
Use Case Name:	**Mark as Pick up**		
Created By:	S.S.D.S. Sandeepa	Last Update By:	
Date Created:	6/15/2021	Date Last Updated:	

Actors:	Logged Delivery Rider
Purpose:	Delivery rider to mark as collected (Pickup) when order is collected from a shop.
Preconditions:	View Delivery Details use case must be triggered before the mark as pick up.
Description:	Since items are ordered from several shops for one order, when the rider picked up the shop orders from each shop, they can mark the relevant shop order as pick up(collected).
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	Shop Order is marked as pick up only after it has been collected from the shop.

Table 33:Delivery Rider - Mark as Pick up

Use Case ID:	34		
Use Case Name:	**View Delivery Dashboard**		
Created By:	SSDS Sandeepa	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Delivery Staff
Purpose:	View all types of deliveries and view all things related to the delivery
Preconditions:	None
Description:	Under view, delivery company can be selected three types of deliveries. i. View new deliveries ii. View in-process deliveries (not completed deliveries) iii. View completed deliveries
Exceptions:	None
Special Requirements:	None
Postconditions:	Depending on the choice of the delivery staff, he can view new delivery, completed delivery, and in-process delivery.
Assumptions:	None

Table 34: Logged Delivery Staff - View Delivery Dashboard

Use Case ID:	35		
Use Case Name:	**View New Delivery**		
Created By:	SSDS Sandeepa	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Delivery staff
Purpose:	View all new deliveries that have to deliver to a customer.
Preconditions:	New delivery can be view through under **view delivery** process.
Description:	When a customer places an order through the system, the delivery dashboard updates for the delivery process. All new deliveries can be viewed by the delivery staff and can assign delivery rider work in the company for each new delivery to transport relevant orders to the customer.
Exceptions:	None
Special Requirements:	None
Postconditions:	**Assign rider** is trigger when assigning riders to each new delivery.
Assumptions:	None

Table 35:Logged Delivery Staff - View New Delivery

Use Case ID:	36		
Use Case Name:	**View In-process Delivery**		
Created By:	SSDS Sandeepa	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Delivery staff
Purpose:	To view all in-process deliveries
Preconditions:	Processing delivery can be view through under the **View Delivery Dashboard** process.
Description:	Once a delivery rider assigns to the order delivery by delivery staff, it is saved in the in-process status until that delivery process is finished by the rider. Staff can view all the unfinished processing deliveries with relevant details.
Exceptions:	None
Special Requirements:	None
Postconditions:	Depending on the choice of the delivery staff, he can view new delivery, completed delivery, and in-process delivery.
Assumptions:	None

Table 36:Logged Delivery Staff - View In-process deliveries

Use Case ID:	37		
Use Case Name:	**View Completed Deliveries**		
Created By:	SSDS Sandeepa	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Delivery Staff
Purpose:	To view all completed deliveries with relevant details
Preconditions:	Completed delivery can be view through under **view delivery** process
Description:	After in-process delivery is marked as completed then that delivery saves in completed status. Delivery company staff can view all these completed deliveries with every relevant detail (customer details, order details, shop details, delivery cost, times, and others) that is concerned with the delivery process.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	Completed delivery details auto take from system and staff members cannot update or change those details.

Table 37:Logged Delivery Staff - View Completed Deliveries

Use Case ID:	38		
Use Case Name:	**Assign Delivery Rider**		
Created By:	SSDS Sandeepa	Last Update By:	
Date Created:	6/13/2021	Date Last Updated:	

Actors:	Logged Delivery Staff
Purpose:	Assign riders to each new delivery
Preconditions:	Must be triggered to **View New Delivery** before the assigned rider to the new delivery
Description:	The delivery staff can assign a rider to each new delivery by selecting a rider from the riders list to collect orders from shops and deliver them to the customer.
Exceptions:	None
Special Requirements:	Suitable delivery riders are shown by the system.
Postconditions:	After assigning rider, particular delivery disappears from the new status and saves in the in-process status.
Assumptions:	None

Table 38: Logged Delivery Staff - Assign Delivery Rider

8.1.4 Shop

Use Case ID:	39		
Use Case Name:	**View Shop Dashboard**		
Created By:	K.K.W. Vishwajith	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	Get an overview of the shop.
Preconditions:	The user must be logged in to the system.
Description:	Shop staff can see the overview details of the shop in the system like the number of orders, total income for the current day, and other related information to the shop.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 39: Logged Shop Staff - View Shop Dashboard

Use Case ID:	40		
Use Case Name:	**View Report**		
Created By:	K.K.W. Vishwajith	Last Update By:	
Date Created:	06/24/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	View all reports related to the hop
Preconditions:	Must activate **View Overview**
Description:	Shop Staff can view all reports like order, revenue, sales, etc. report that related to the shop. Here staff can see reports like annually, monthly, or weekly generated.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	The shop will only see relevant reports that relate to them

Table 40: Logged Shop Staff - View Report

Use Case ID:	41		
Use Case Name:	**Manage Orders**		
Created By:	K.K.W. Vishwajith	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	View all orders in the shop that are ordered by the system.
Preconditions:	None
Description:	See orders to the shop. With item list and the quantity of them. Also, will have the delivery rider details who will pick it up. Here shop staff can view new and completed orders separately.
Exceptions:	None
Special Requirements:	None
Postconditions:	**View Completed Order or View New Order** must be triggered according to the shop staff selection for the view orders
Assumptions:	None

Table 41:Logged Shop Staff - Manage Orders

Use Case ID:	42		
Use Case Name:	**View New Orders**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	View all new orders that the shop has received.
Preconditions:	**Manage Order** must be triggered before the view new delivery
Description:	Shop staff can view newly arrived orders that not any action taken before.
Exceptions:	None
Special Requirements:	None
Postcondition	**Mark as Pick up when** rider collected particular order from the shop.
Assumptions:	None

Table 42:Logged Shop Staff - View New Orders

Use Case ID:	43		
Use Case Name:	**Mark as Pick Up**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	Mark as picks up when rider collects relevant order from the shop.
Preconditions:	**View New Order** must be triggered.
Description:	Shop staff can mark selected orders as pick up(collected) when riders pick up shop orders from the shop.
Exceptions:	None
Special Requirements:	None
Postcondition	After mark as pick up relevant order, then it is saved in completed order status.
Assumptions:	None

Table 43:Logged Shop Staff - Mark as Pick up

Use Case ID:	44		
Use Case Name:	**View Completed Orders**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	View all completed orders of the shop.
Preconditions:	**Manage Order** must be triggered before the view new delivery
Description:	Shop staff can view newly arrived orders that not any action taken before.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 44:Logged Shop Staff - View Completed Order

Use Case ID:	45		
Use Case Name:	**Manage Item List**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	View and Manage all item list with relevant details.
Preconditions:	None
Description:	Staff can manage which items are available for selling on the platform. In this scenario, they can, 1. Add new items: Select items from a predefined list of items provided by the system. 2. Pause items from the listing: Remove items from the list of items that a customer can view. Used as if the shop doesn't have the item in the inventory. 3. Remove an item: Completely remove a product from the listing. Users can still add them using step 1. 4. Update item: Change item qualities.
Exceptions:	None
Special Requirements:	The shop owner must pause an item in the list if such an item has low inventory. As a standard rule of thumb if an item has less inventory than daily offline inventory sales. Remove it from the system.
Postconditions:	**Add, Remove or Update item** is trigger with the shop staff selection.
Assumptions:	None

Table 45: Logged Shop Staff – Manage Item List

Use Case ID:	46		
Use Case Name:	**Add Item**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	Add a new item to the shop item list
Preconditions:	None
Description:	By selecting a system item list shop staff can add a new item to the shop item list.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 46:Logged Shop Staff - Add Item

Use Case ID:	47		
Use Case Name:	**Remove Item**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	Remove selected items from the shop item list.
Preconditions:	None
Description:	Shop staff can remove unnecessary items from the shop item by selecting item by item.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 47:Logged Shop Staff - Remove Item

Use Case ID:	48		
Use Case Name:	**Update Item**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged Shop Staff
Purpose:	Update selected item qualities.
Preconditions:	None
Description:	Shop staff can update details of the item like price, stock, etc. By the system. Only allowed fields can be changed.
Exceptions:	Restricted fields cannot update by the system staff.
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 48:Logged Shop Staff - Update Item

8.1.5 System Staff

Use Case ID:	49		
Use Case Name:	**Validate User Registration**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	Approve register process of the users like shop or delivery rider.
Preconditions:	Before validating registration details unregistered customer has to S**ubmit Registration Details** according to the registration form
Description:	When a shop or delivery rider registers, the system staff is responsible for validating the data and approving the registration. So, after unregistered users submit registration details, then system staff check the validity of the detail and provide approval or de approval to the particular user.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 49:Logged System Staff - Validate User Registration

Use Case ID:	50		
Use Case Name:	**View Order**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	System staff can view all orders according to category wise.
Preconditions:	None
Description:	Allows system staff to view all orders with all details like order details, customer details, and delivery details. System staff will be able to these orders according to category. Like new orders, ongoing orders completed orders or, pending orders.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 50:Logged System Staff - View Order

Use Case ID:	51		
Use Case Name:	**Manage Users**		
Created By:	K.K.W. Vishwajith	Last Update By:	K.K.W. Vishwajith
Date Created:	06/23/2021	Date Last Updated:	06/23/2021

Actors:	Logged System Staff
Purpose:	View and manage other users and profiles.
Preconditions:	None
Description:	Manage the system accounts such as, • Customers • Users • Shops Doing actions such that, • Register Users: Delivery Companies and For Shops • Deleting Users • Updating User details: such as resetting the password
Exceptions:	None
Special Requirements:	None
Postconditions:	**View user** and **Remove user** use cases will be triggered according to the relevant function.
Assumptions:	None

Table 51: Logged System Staff - Manage Users

Use Case ID:	52		
Use Case Name:	**View Users**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	To view other users in the system.
Preconditions:	None
Description:	Sometimes the system staff wants to know the information of other users in the system. In such cases, system staff is allowed to monitor their data with a restriction level. So, system staff can view details of the shop, delivery rider, and customer as well as other system staff members.
Exceptions:	Cannot view data that is restricted to view.
Special Requirements:	Only privileged staff members can view other user's details.
Postconditions:	None
Assumptions:	None

Table 52:Logged System Staff - View User

Use Case ID:	53		
Use Case Name:	**View Customer**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	View Customer details
Preconditions:	View customer can do according to the **View User.**
Description:	System staff can view customer details to perform some system functionalities. Eg:- to check the validation of the customer when the customer makes a complaint
Exceptions:	Cannot view data that is restricted to view.
Special Requirements:	Only privileged staff members can view other user's details.
Postconditions:	None
Assumptions:	None

Table 53:Logged System Staff - View Customer

Use Case ID:	54		
Use Case Name:	**View Shop**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	View Shop Details
Preconditions:	View shop can do according to the **View User.**
Description:	System staff can view shop details to perform some system functionalities. Eg:- to check the validation of the shop or audit to check shop reliability.
Exceptions:	Cannot view data that is restricted to view.
Special Requirements:	Only privileged staff members can view other user's details.
Postconditions:	None
Assumptions:	None

Table 54:Logged System Staff - View Shop

Use Case ID:	55		
Use Case Name:	**View delivery Rider**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	View rider details
Preconditions:	View Rider can do according to the **View User.**
Description:	System staff can view the delivery rider’s details to perform some system functionalities.
Exceptions:	Cannot view data that is restricted to view.
Special Requirements:	Only privileged staff members can view other user's details.
Postconditions:	None
Assumptions:	None

Table 55:Logged System Staff - View Delivery Rider

Use Case ID:	56		
Use Case Name:	**Remove User**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	To remove the user from the system.
Preconditions:	**Manage User** must be triggered before the remove the user.
Description:	System staff allows removing some kind of users from the system. When unauthorized action is observed then staff members can remove such users. Eg:- remove the shop after the investigation when it has more complaints.
Exceptions:	None
Special Requirements:	Only privileged staff members can remove users from the system.
Postconditions:	**Remove from Login** use case trigger after some user removes from the system implicitly.
Assumptions:	None

Table 56:Logged System Staff - Remove User

Use Case ID:	57		
Use Case Name:	**Remove from Login**		
Created By:	S.S.D.S sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	Remove the login details of the user.
Preconditions:	Must activate the **View New Shop Registrations**
Description:	After removing some users from the system then logging details also need to remove.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 57:Logged System Staff - Remove from Login

Use Case ID:	58		
Use Case Name:	**Generate Report**		
Created By:	K.K.W. Vishwajith	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	Generate all types of reports in the system.
Preconditions:	None
Description:	Generate reports about Weekly, Monthly, or annual Sames of the system. Reports can be generated based on the shops, delivery, sales, etc. Different types of reports are generated by different system users according to their privilege level.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 58:Logged System Staff - Generate Report

Use Case ID:	59		
Use Case Name:	**View Report**		
Created By:	S.S.D.S Sandeepa	Last Update By:	
Date Created:	18/09/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	View all types of reports in the system.
Preconditions:	None
Description:	System staff can view all generated reports in the system. Like, order reports, sales reports, or any other report.
Exceptions:	If the system member does not accomplish the privileges, then they will not be able to view the relevant report.
Special Requirements:	Eligible to view system reports by various privileges levels of the staff member.
Postconditions:	None
Assumptions:	None

Table 59:Logged System Staff - View Report

Table 60:Logged System Staff - View Complaint

Use Case ID:	60		
Use Case Name:	**View Complaint**		
Created By:	K.K.W. Vishwajith	Last Update By:	
Date Created:	06/23/2021	Date Last Updated:	

Actors:	Logged System Staff
Purpose:	Staff users can view complaints that are given by customers.
Preconditions:	None
Description:	View complaints made by the user against orders made through the customer.
Exceptions:	None
Special Requirements:	None
Postconditions:	Attend to Complaint use case must be triggered when system staff attends to the complaint.
Assumptions:	None

Use Case ID:	61		
Use Case Name:	**Attend to a Complaint**		
Created By:	K.K.W. Vishwajith	Last Update By:	K.K.W. Vishwajith
Date Created:	06/23/2021	Date Last Updated:	06/23/2021

Actors:	Logged System Staff
Purpose:	Attend to a complaint that comes from the customers according to the order.
Preconditions:	Must activate **View Complaints**
Description:	System staff can view customer complaints as well as provide solutions or feedback to them by using the system like, • Update and keep tabs on the complaints
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 61:Logged System Staff - Attend to Complaint

Use Case ID:	62		
Use Case Name:	**Manage Item List**		
Created By:	S.S.D.S sandeepa	Last Update By:	K.K.W. Vishwajith
Date Created:	18/09/2021	Date Last Updated:	06/23/2021

Actors:	Logged System Staff
Purpose:	The main item list of the system manages by the system staff.
Preconditions:	None
Description:	System staff can manage all things about the item list. Accordingly, they can • Add item to the item list • Remove item from the item list • Update item (eg:- change MRP) Shop staff select items from their item list from this list.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 62:Logged System Staff - Manage Item List

Use Case ID:	63		
Use Case Name:	**Add Item**		
Created By:	S.S.D.S sandeepa	Last Update By:	K.K.W. Vishwajith
Date Created:	18/09/2021	Date Last Updated:	06/23/2021

Actors:	Logged System Staff
Purpose:	To add new items to the item list.
Preconditions:	Before the add item, **Manage Item List** must be triggered.
Description:	System staff can add new items to the system. When the system staff adds a new item compulsory item fields also fill with relevant information.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 63:Logged System Staff - Add Item

Use Case ID:	64		
Use Case Name:	**Remove Item**		
Created By:	S.S.D.S sandeepa	Last Update By:	K.K.W. Vishwajith
Date Created:	18/09/2021	Date Last Updated:	06/23/2021

Actors:	Logged System Staff
Purpose:	To remove an item from the item list
Preconditions:	Before the removed item, **Manage Item List** must be triggered.
Description:	System staff can remove the item from the item list. Then particular item cannot be select by the shop owner to display.
Exceptions:	None
Special Requirements:	None
Postconditions:	Removed item disappear from all shop item lists as well.
Assumptions:	None

Table 64:Logged System Staff – Remove Item

Use Case ID:	65		
Use Case Name:	**Update Item**		
Created By:	S.S.D.S sandeepa	Last Update By:	K.K.W. Vishwajith
Date Created:	18/09/2021	Date Last Updated:	06/23/2021

Actors:	Logged System Staff
Purpose:	System staff can update item details in the item list.
Preconditions:	Before the update item, **Manage Item List** must be triggered.
Description:	System staff allows making changes to the items in the item list. Only the fields that are allowed to be changed will be changed.
Exceptions:	None
Special Requirements:	None
Postconditions:	None
Assumptions:	None

Table 65:Logged System Staff - Update Item