Software Requirement Specification

BTBT

BEST TIME, BEST TRANSIT
COURIER SERVICE

Best time Best transit (BTBT)

(Courier service management system)

Prepared by:

Name: Abeer Mathur

Registration No: 21BCE3630

Document Version Control:

Full Name	Date	Change Comments	New version	Previous version
Abeer Mathur	16.01.24	Initial version	V1.0	NA
Abeer Mathur	23.01.24	Work breakdown structure added	V1.1	V1.0
Abeer Mathur	28.02.24	Class diagram, Use case diagram, ER diagram and Activity selection diagram added	V1.2	V1.1
Abeer Mathur	18.03.24	Data flow diagram (level 0,1,2), sequence diagram, state chart diagram, collaboration diagram, and user interface diagram added	V1.3	V1.2

Requirements Version Control:

Full Name	Date	Change Comments	New version	Previous version
Abeer Mathur	16.01.24	Initial specification.	V1.0	NA
Abeer Mathur	23.01.24	Improving initial requirements following the criteria of good requirements.	V1.1	V1.0
Abeer Mathur	28.02.24	Verified initial requirements.	V1.2	V1.1
Abeer Mathur	18.03.24	Solution oriented requirement has been added.	V1.3	V1.2

Table of contents

1. **Introduction**

1.1Purpose:

The BTBT COURIER SERVICE application is designed to provide a comprehensive solution for courier services. This document serves as a detailed design description, encompassing aspects such as diagrammatic representations for login and signup processes, descriptions of static and dynamic behaviors of the application, an overview of the application's features and the design constraints, identification of the stakeholders who will benefit from this document, and pictorial representations highlighting both Functional and Non-Functional requirements of the application.

1.2 Scope

The BTBT COURIER SERVICE, an Android-based application, is crafted to revolutionize the way items are transferred from one location to another. Key aspects include time-efficient and cost-effective transfer of items, considering variables such as travel date, source, and destination, providing users with multiple carrier options, ensuring comfort and flexibility in choice, and a design aimed at reducing time, cost, and effort in courier services.

1.3 Definitions and Acronyms

This document adheres to specific formatting rules for clarity and consistency: Font is Times New Roman, with size 12 for regular text and size 14 for headings, formatting conventions include the app's name in bold and subheadings underlined and bold, all text in black, 'Shall' to indicate mandatory functionalities, and labeling conventions include FE (Feature), POB (Project Objectives), HC (Hardware Constraints), and SC (Software Constraints), among others.

2. **System Overview**

2.1 Project Background

The BTBT COURIER SERVICE app is a response to the limitations and high costs associated with traditional courier services. This mobile app solution is designed to provide a more efficient and cost-effective method for sending materials and to overcome the delays and constraints of existing courier services.

2.2 Project Scope

The application stands out by offering a user-friendly platform for connecting individuals traveling to similar destinations for courier purposes and emphasizing ease of use, cost-effectiveness, and time efficiency in courier services.

2.3 Project Objectives

The project's objectives are to save users' time by negating the need to physically visit courier offices (POB-1), ensure reasonable pricing for all types of courier services (POB-2), and provide services that are faster than traditional courier options (POB-3).

2.4 Stakeholders

The primary stakeholders of this application include the General Public (carrier), who are individuals willing to transport materials and can post their travel plans on the app, and Sender/Receiver, any user in need of sending or receiving items.

2.5 Operating Environment

The application operates under the following conditions: Hardware requires Android devices with an internet connection and camera functionality, and Software is compatible with Android OS and uses MySQL Lite database, designed for Android version 4.3 Lollipop or higher.

2.6 System Constraints

The system operates within these constraints: Hardware Constraints depend on a smartphone with internet and a working camera, and Software Constraints require an Android operating system, version 4.3 Lollipop or above.

3. **Process Model**

3.1 Overview

The BTBT COURIER SERVICE application development follows the **Spiral Model**, a comprehensive, risk-driven process model. This model is preferred due to its emphasis on iterative development and risk management, crucial for creating a user-centric courier service application.

3.2 Phases of the Spiral Model

The development process encompasses four distinct phases: the Planning Phase involves gathering requirements and defining project objectives with a thorough risk analysis to foresee potential challenges, the Risk Analysis Phase identifies and analyzes potential risks with strategies developed to mitigate these risks, the Engineering Phase is where actual development and testing of the software occur, including the creation of a small-scale prototype or mock-up of the application, and the Evaluation Phase where stakeholders and customers evaluate the current progress of the project with feedback crucial for planning the next iteration of the development cycle.

3.3 Suitability of the Spiral Model

The Spiral Model's focus on early risk identification and mitigation is particularly advantageous for the BTBT COURIER SERVICE project, allowing for the development of a robust, efficient, and user-friendly application, aligning with the project's objectives and stakeholder expectations.

4. **Process Scheduling**

4.1 Overview

January 2024

- Jan 7 - Jan 14: Project Kickoff and Initial Planning
  - Setup project infrastructure.
  - Begin UI/UX design sketches.

- Jan 15 - Jan 21: Requirements Analysis
  - Finalize authentication and authorization requirements.
  - Define search functionality parameters.

- Jan 22 - Jan 28: System Design
  - Design booking and payment system.
  - Outline notification system architecture.

- Jan 29 - Feb 4: Initial Development Sprint
  - Start development of application infrastructure.
  - Implement base UI components.

February 2024

- Feb 5 - Feb 11: Development Continuation
  - Develop search functionality.
  - Implement booking and payment modules.

- Feb 12 - Feb 18: Mid-Project Evaluation
  - Initial reporting and analytics setup.
  - Develop customer support framework.

- Feb 19 - Feb 25: Feature Development and Testing
  - Continue UI development and trip management features.
  - Test matching and scheduling algorithms.

March 2024

- Mar 1 - Mar 7: System Integration and Testing
  - Integrate tracking and monitoring systems.
  - Refine settings and preferences.

- Mar 8 - Mar 14: User Acceptance Testing (UAT) and Feedback
  - Conduct UAT for user ratings and feedback module.
  - Begin documentation and marketing planning.

- Mar 15 - Mar 21: Final Development Sprint
  - Address testing feedback.
  - Finalize marketing and promotion strategies.

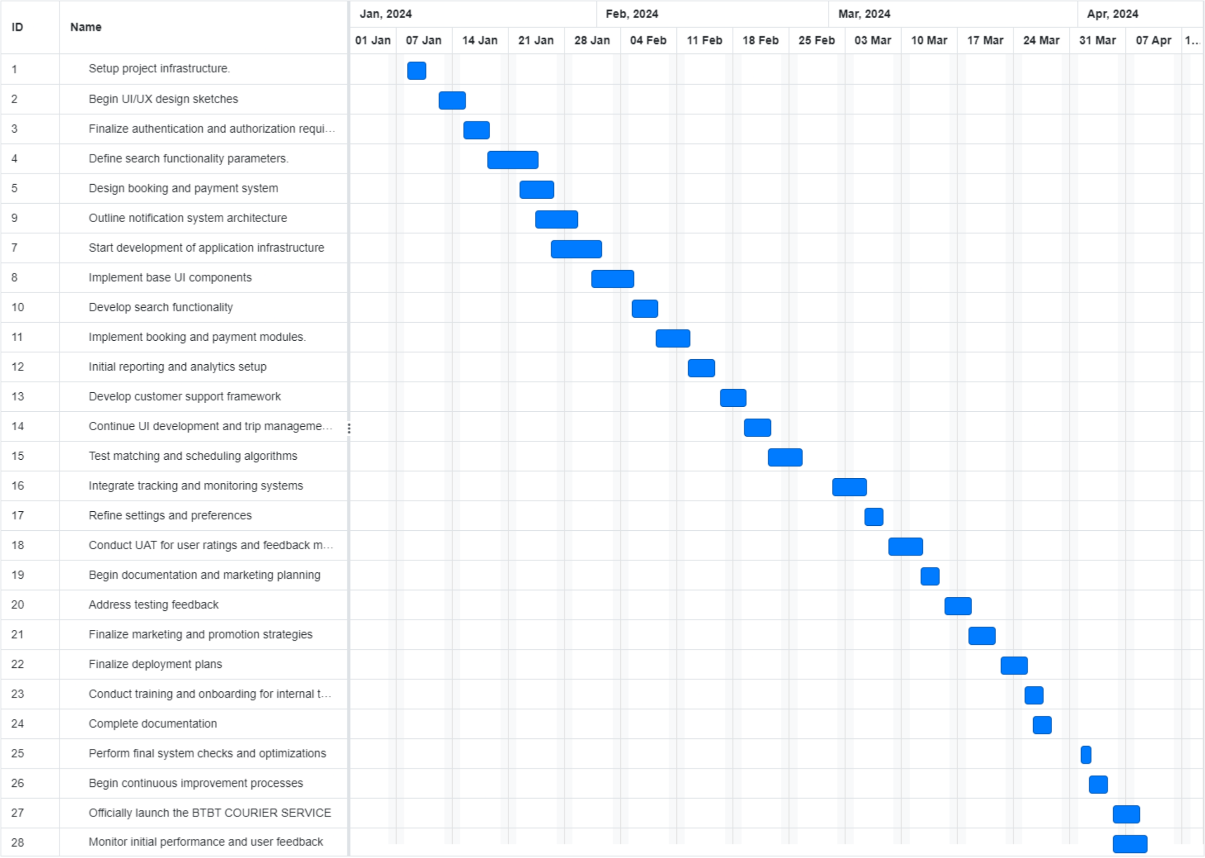
- Mar 22 - Mar 28: Pre-Launch Preparations
  - Finalize deployment plans.
  - Conduct training and onboarding for internal teams.
  - Complete documentation.

April 2024

- Apr 1 - Apr 4: Launch Readiness Review
  - Perform final system checks and optimizations.
  - Begin continuous improvement processes.

- Apr 5 - Apr 7: Project Launch
  - Officially launch the BTBT COURIER SERVICE.
  - Monitor initial performance and user feedback.

4.2 Gantt Chart



ID	Name
1	Setup project infrastructure.
2	Begin UI/UX design sketches
3	Finalize authentication and authorization requi…
4	Define search functionality parameters.
5	Design booking and payment system
9	Outline notification system architecture
7	Start development of application infrastructure
8	Implement base UI components
10	Develop search functionality
11	Implement booking and payment modules.
12	Initial reporting and analytics setup
13	Develop customer support framework
14	Continue UI development and trip manageme…
15	Test matching and scheduling algorithms
16	Integrate tracking and monitoring systems
17	Refine settings and preferences
18	Conduct UAT for user ratings and feedback m…
19	Begin documentation and marketing planning
20	Address testing feedback
21	Finalize marketing and promotion strategies
22	Finalize deployment plans
23	Conduct training and onboarding for internal t…
24	Complete documentation
25	Perform final system checks and optimizations
26	Begin continuous improvement processes
27	Officially launch the BTBT COURIER SERVICE
28	Monitor initial performance and user feedback

5. **Work breakdown structure**

5.1 Overview

The following provides the outline for the process of various components and aspects of the application:

1. Application Infrastructure: This section might detail the core technical architecture, including cloud services, databases, and server setups that support the application's functioning.

2. Authentication and Authorization: This area likely covers security mechanisms of the app, focusing on how users sign in and out and how permissions are managed.

3. Search Functionality: This part of the mind map appears to describe how users can search within the app, including the search algorithms, filters, and how results are presented.

4. Booking and Payment: This branch pertains to the features that allow users to book courier services and handle financial transactions, including interfaces with payment gateways and transaction history.

5. Notification System: This section indicates the system through which the app communicates with users, which might include push notifications, email alerts, and SMS.

6. Reporting and Analytics: This segment might detail the backend tools used for data analysis, tracking app usage, and generating reports.

7. Customer Support: This branch outlines the support mechanisms in place for users, which might include FAQs, support chat, and ticketing systems.

8. Deployment: This section outlines the processes involved in deploying the app, which might cover strategies for deployment, version control, and update mechanisms.

9. Training and Onboarding: This might refer to the educational materials and processes designed to train users or staff in using the application.

10. Continuous Improvement: This suggests that the application is under continuous development, with processes for incorporating user feedback, updating features, and maintaining quality.

11. User Interface (UI) Development: This involves the design of the app's front end, focusing on the visual elements, user experience, and navigation.

12. Trip Management: This indicates the features related to organizing and managing courier trips, which might include tools for scheduling, optimizing routes, and tracking shipments.

13. Matching and Scheduling: This could be related to how the app aligns courier service requests with providers and manages the timing of pickups and deliveries.

14. User Ratings and Feedback: This implies a system where users can rate the services and provide feedback, which is important for maintaining and improving service quality.

15. Tracking and Monitoring: This likely refers to the features that allow for the real-time monitoring of courier services, such as GPS tracking and delivery status updates.

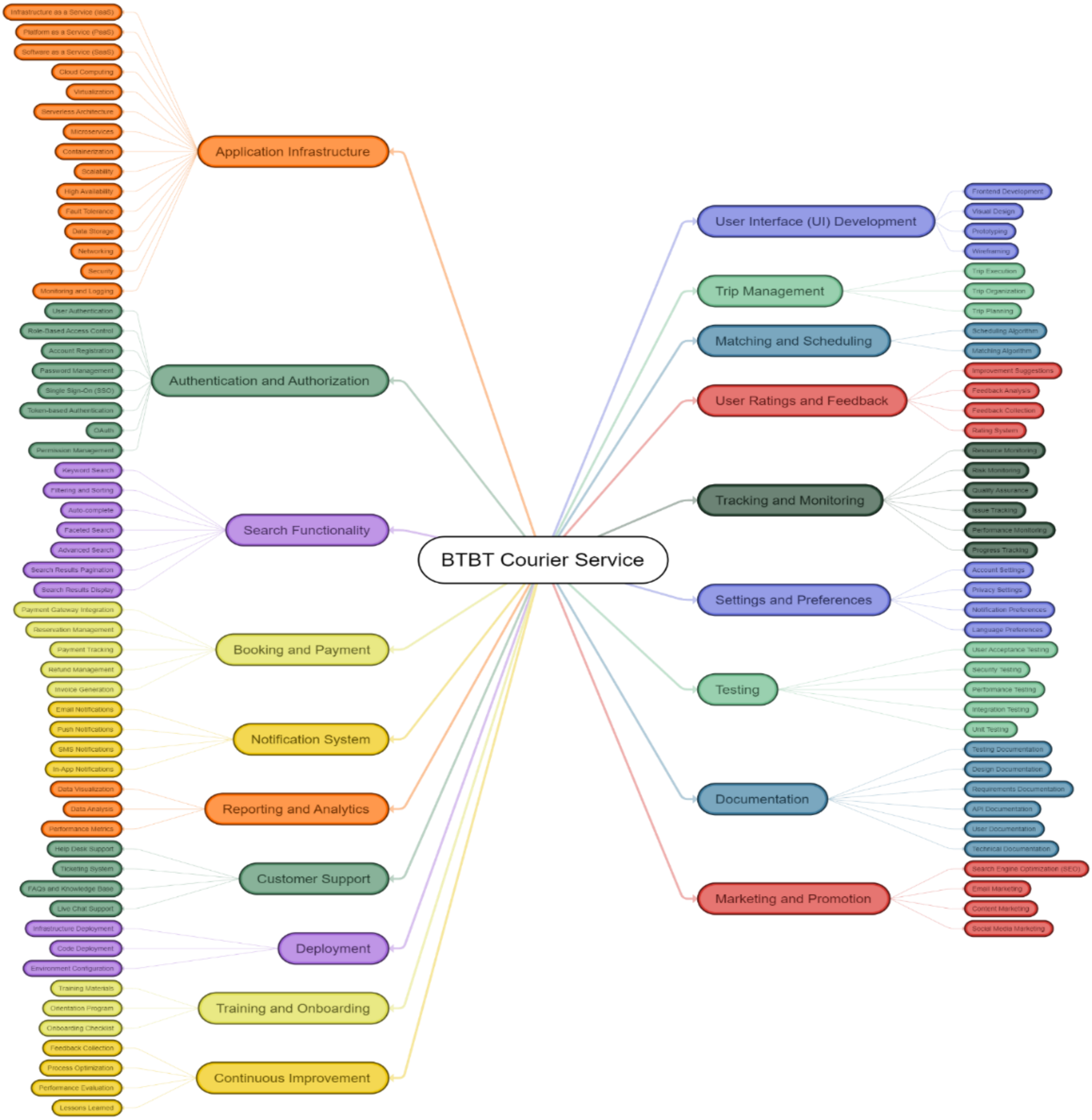
16. Settings and Preferences: This represents the features that allow users to customize their app experience, such as setting notification preferences or delivery options.

17. Testing: This section covers the different types of testing the app needs to go through, which may include tests for usability, performance, and security.

18. Documentation: This includes the written materials that support the software, such as API documentation, technical documentation, and user manuals.

19. Marketing and Promotion: This section encompasses the strategies and channels used to market the app, potentially including search engine optimization, content marketing, and social media campaigns.

5.2 WBS Diagram



6. **Use case diagram**

6.1 Overview

The following is the outline of the use case diagram which would be used for BTBT Courier Service.
The diagram includes three types of actors: Member, Visitor, and Server_Boy.

Member: This actor has several use cases associated with them:

SignUp: The member can create an account, which seems to be a prerequisite for the "Login" function, indicated by an "include" relationship.
Login: Allows the member to access their account.

Delete Offer: The member can remove an offer, which could be a service or product they previously provided.

Search for Carrier: Members can look for a carrier, presumably to ship items.

View Detail: Members can view details of a certain element, which could be offers, carriers, or other members' profiles. This use case extends to "Contact Members".

Contact Members: Members can contact other members, which further branches into two sub-use cases:

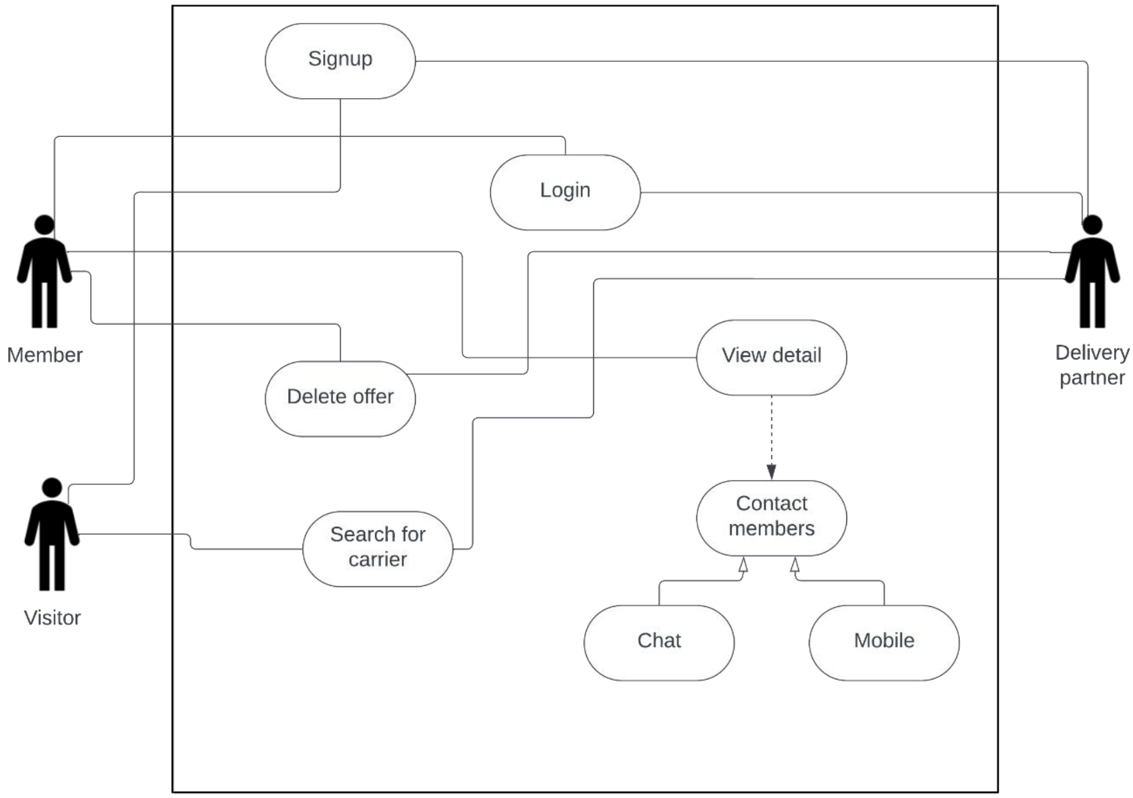
> Chat: Indicates the member can contact others through chat.
> Mobile: Indicates the member can contact others through mobile, which might mean a call or text message.
> Visitor: This actor has one use case associated with them:

Search for Carrier: This implies that even non-members (visitors) can search for a carrier within the application.

Server_Boy: This actor is connected to "View Detail" and "Contact Members", suggesting they play a role in these functionalities, but the diagram does not specify their exact actions.

The "extend" relationship from "View Detail" to "Contact Members" suggests that viewing details may optionally lead to contacting members.

6.2 Use case diagram



7. **Class Diagram**

7.1 Overview

User: This class represents the generic user of the courier service system. Since it does not have any attributes or methods defined in the diagram, it serves as a placeholder for more specific types of users.

Authenticate: This class is a definite part of the User class, as indicated by the composition relationship. It contains two attributes, Username and Userpass, which are used for logging in to the system. The Authenticate() method is used to verify user credentials, and the Register() method is used to create new user accounts in the system.

Visitor: This class is associated with the Authenticate class, indicating that visitors of the courier service website or system can authenticate to access more features. The absence of attributes or operations for Visitor implies that it is a role played by non-registered users who have limited access to the system.

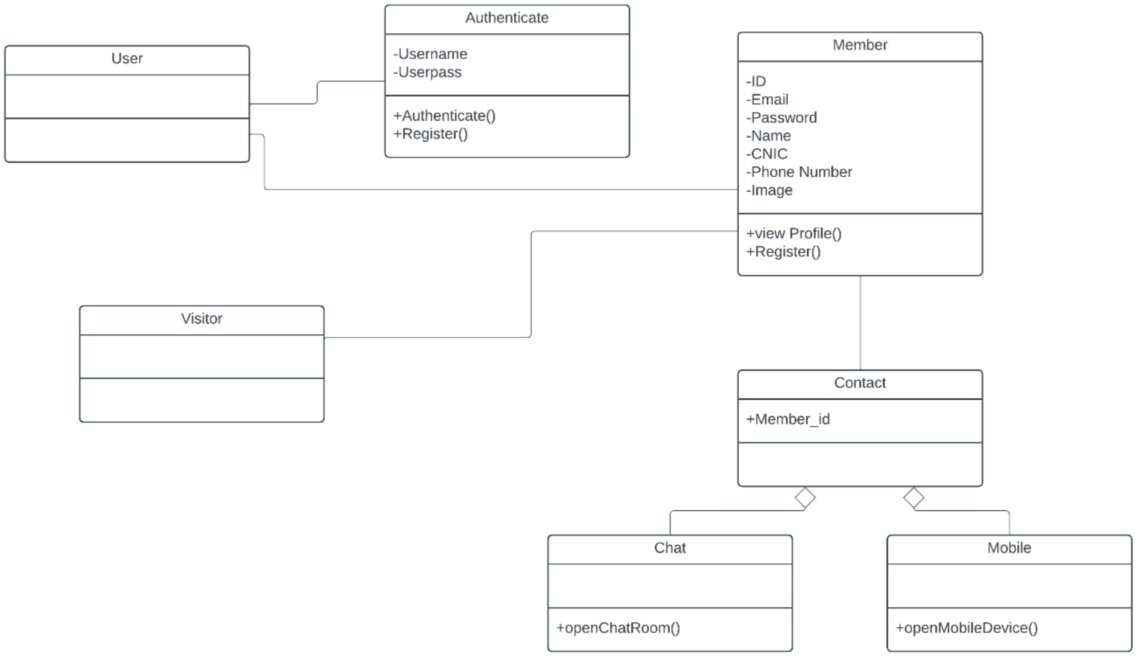
Member: This class is directly associated with the Authenticate class, suggesting that members are authenticated users with additional privileges in the system. The Member class includes personal details relevant to the courier service, like ID, Email, Password, Name, CNIC (which could be the customer’s national identity card number), Phone Number, and Image. Members can viewProfile(), which may involve checking their details related to courier transactions, and can Register(), possibly for courier services or updating their information.

Contact: This class holds the Member_id attribute, signifying a one-to-one correspondence with a member. It represents the contact information that is crucial for the courier service operations like sending notifications, tracking shipments, and providing customer service. It is a part of the Member class, meaning that each member's contact information is intrinsic to their profile.

Chat: The Chat class is associated with the Contact class, indicating that chat functionality is based on the member’s contact information. The openChatRoom() method likely enables real-time communication between the courier service members and support or between members themselves for service-related inquiries.

Mobile: Similarly associated with the Contact class, the Mobile class includes the openMobileDevice() method, implying functionality that allows members to interact with the courier service through mobile devices, such as tracking shipments, receiving updates, or accessing mobile-exclusive features.

7.2 Class Diagram



8. **Entity Relation model**

8.1 Overview

system that seems to manage users, customers, shipments, items, admins, and delivery persons. Here is a breakdown of the entities and their relationships:

User: This entity represents the users of the system. It has attributes like ID, Username, Password, Email, and Status. The User entity has a "Login" relationship with itself, indicating that users can log in, and it "Has" a relationship with the Customer entity, suggesting that a user can be associated with a customer.

Customers: This is the central entity that is related to Users, Shipments, Items, Admins, and Delivery persons. Attributes include ID. Customers can "Add item" and "Remove item," indicating actions they can perform within the system. They also "Manage" Admins and "Tracks" Shipments, showing they have oversight and tracking capabilities.

Admin: This entity represents the administrators of the system with attributes such as Name, Mobile no, ID, Office ID, and Status. The admin can "Manage" Customers and has a "Has" relationship with Delivery persons, meaning the admin can assign or be associated with delivery persons.

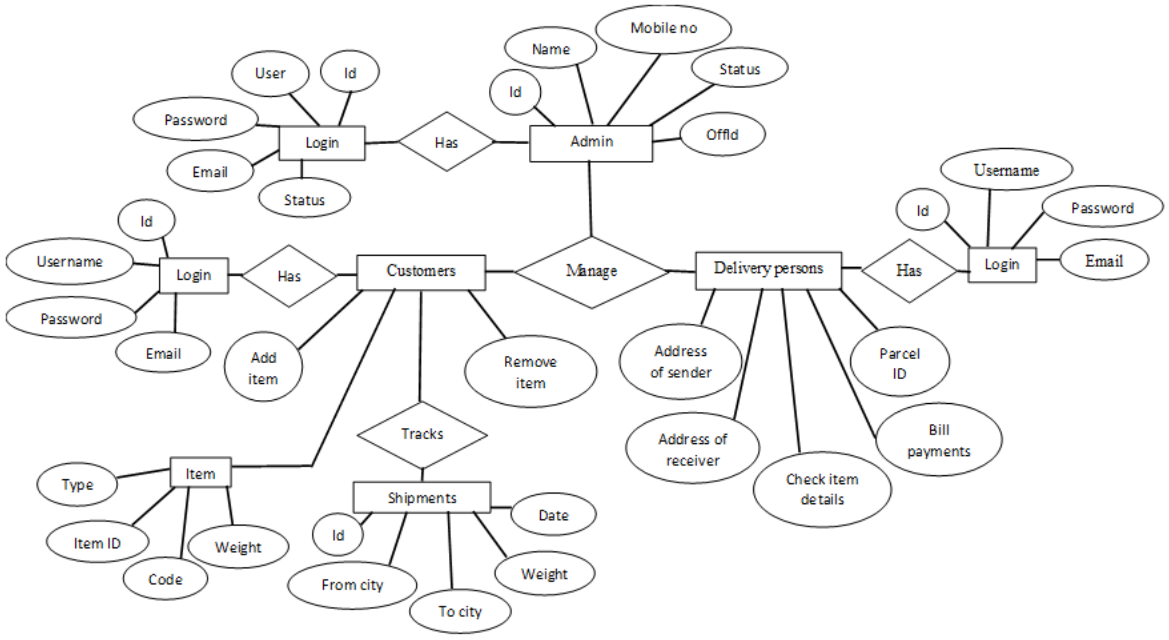
Delivery persons: This entity has relationships with Admins, suggesting that they are managed by Admins. They have attributes like Address of sender, Address of receiver, and can "Check item details" relating to parcels, indicating their role in the delivery process.

Shipments: This entity is related to Items and Customers. It includes attributes such as ID, Date, Weight, From city, and To city, providing details on the shipment logistics.

Item: The entity that represents the goods or parcels being shipped with attributes such as Type, Item ID, Weight, and Code. It has a "Has" relationship with Shipments, indicating that each shipment contains items.

The relationships are mainly "Has" and "Manage," suggesting ownership and administrative control. The diagram also shows actions such as "Add item," "Remove item," "Check item details," which seem to be operations that can be performed within the system.

8.2 ER Diagram



9. **Activity Selection Diagram**

9.1 Overview

Customer Activity:
The customer initiates the process by choosing to send a courier.
The system then presents different options to the customer, such as types of delivery services.
The customer selects the desired options provided by the system.
Next, the customer selects a path, presumably the delivery path or method.
Alongside, the customer also selects specific requirements for the courier, possibly including package size, weight, or special handling instructions.

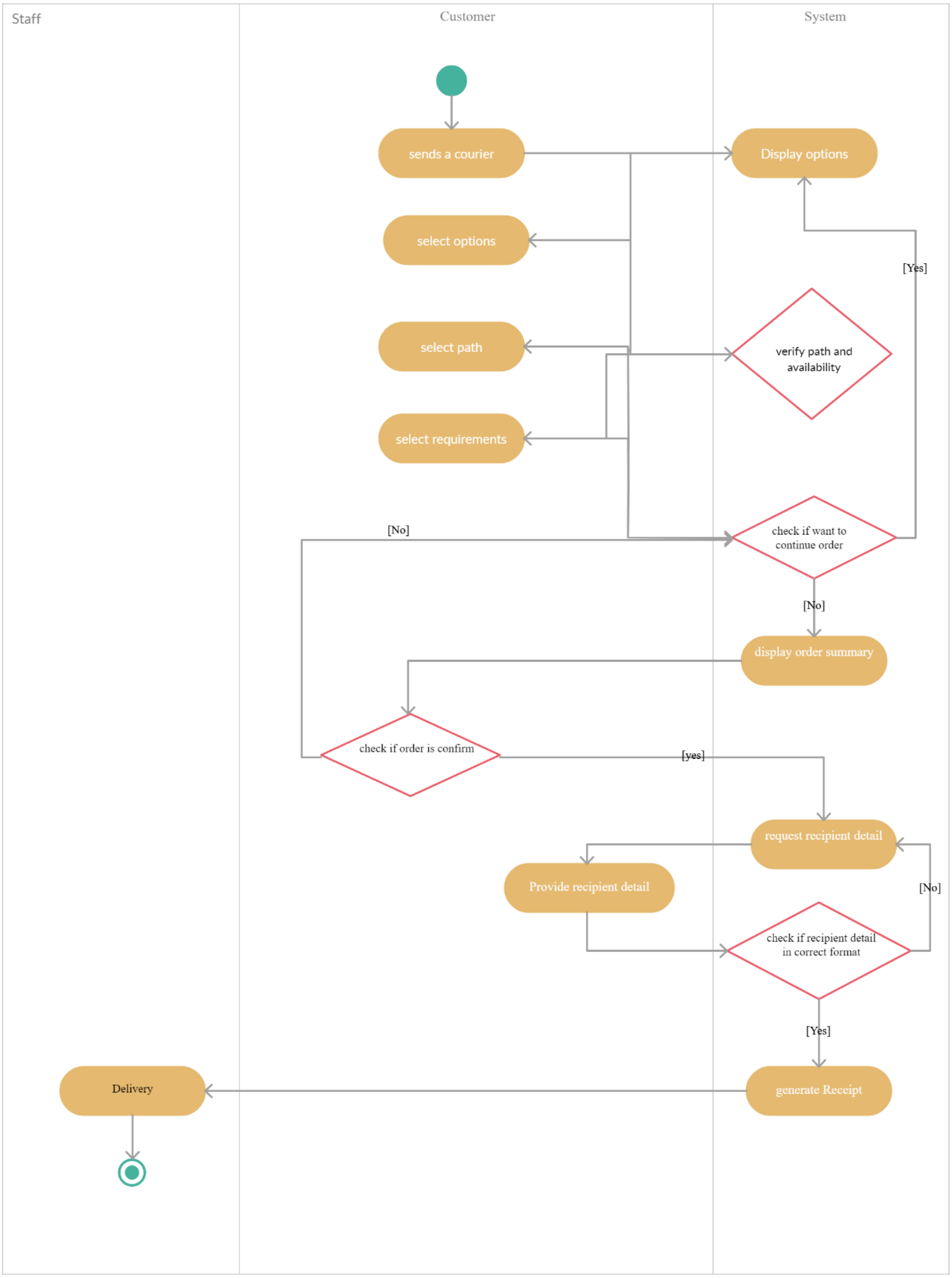
System Activity:
The system verifies the path and availability based on the customer's selection.
If the path and availability are confirmed ('Yes'), the system proceeds. If not ('No'), the system checks if the customer wants to continue the order.
If the customer decides not to continue ('No'), the process ends here.
If the customer decides to continue, the system displays an order summary.

Decision Points:
The system checks if the order is confirmed by the customer.
If 'Yes', the system requests recipient details from the customer.
If 'No', the process likely ends or loops back for the customer to make changes.

Customer Provides Details:
The customer provides recipient details, and the system checks if these details are in the correct format.
If the details are not in the correct format ('No'), the system presumably prompts the customer to re-enter them correctly.
Once the recipient details are confirmed to be in the correct format ('Yes'), the system generates a receipt for the customer.

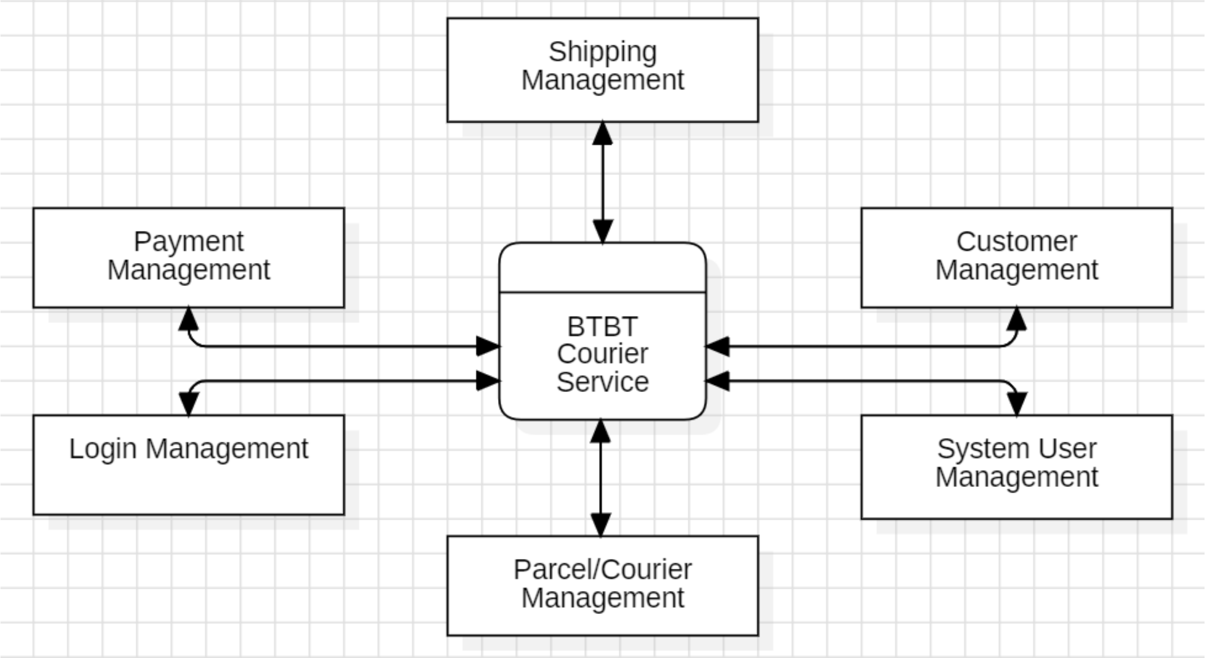
Completion:
The process concludes with the courier being sent out for delivery.

9.2 Activity selection diagram

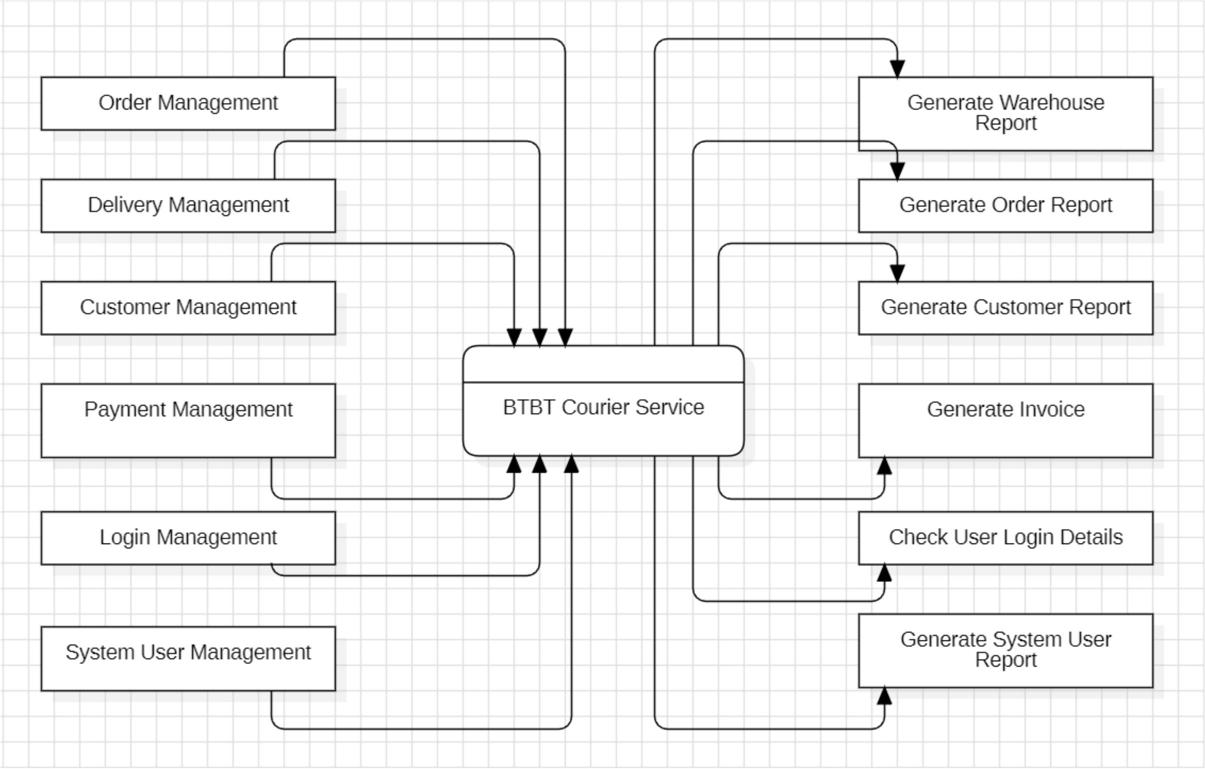


10. Data Flow Diagram

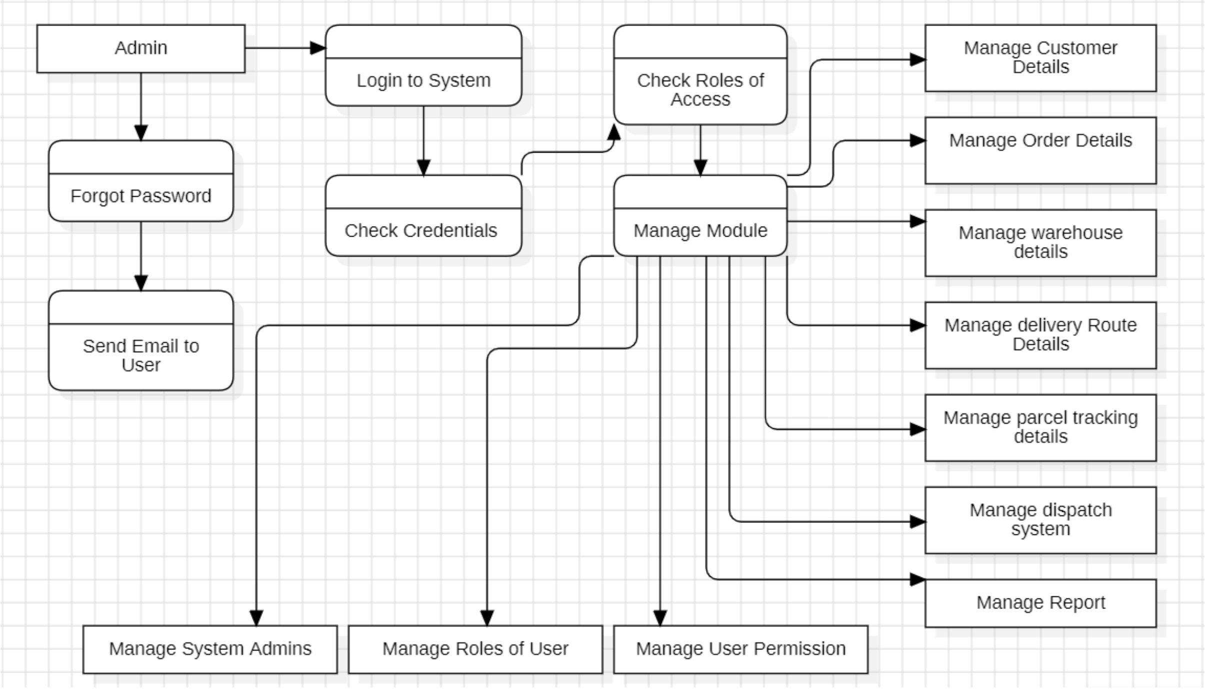
10.1 Level 0



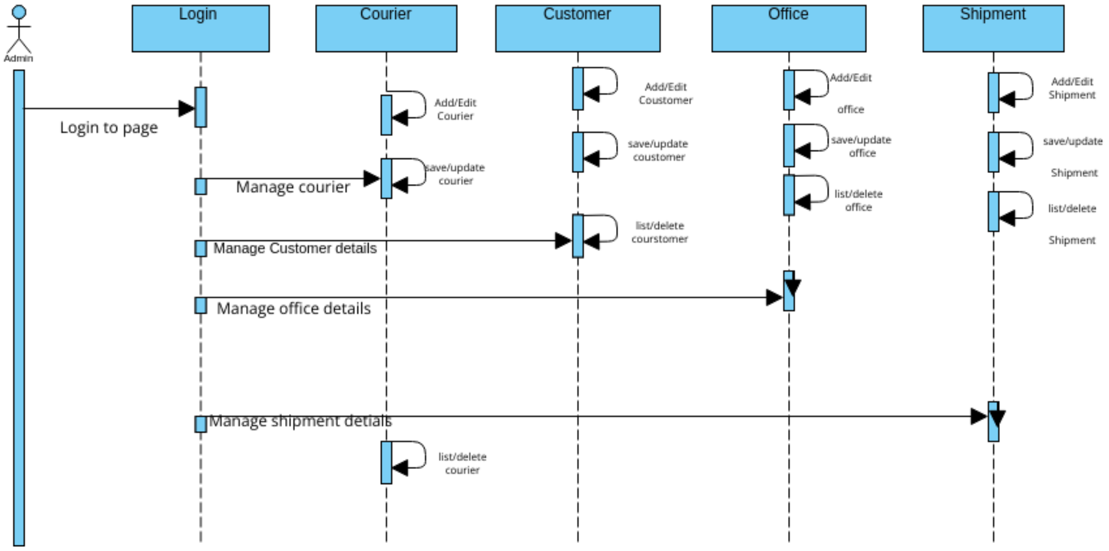
10.2 Level 1



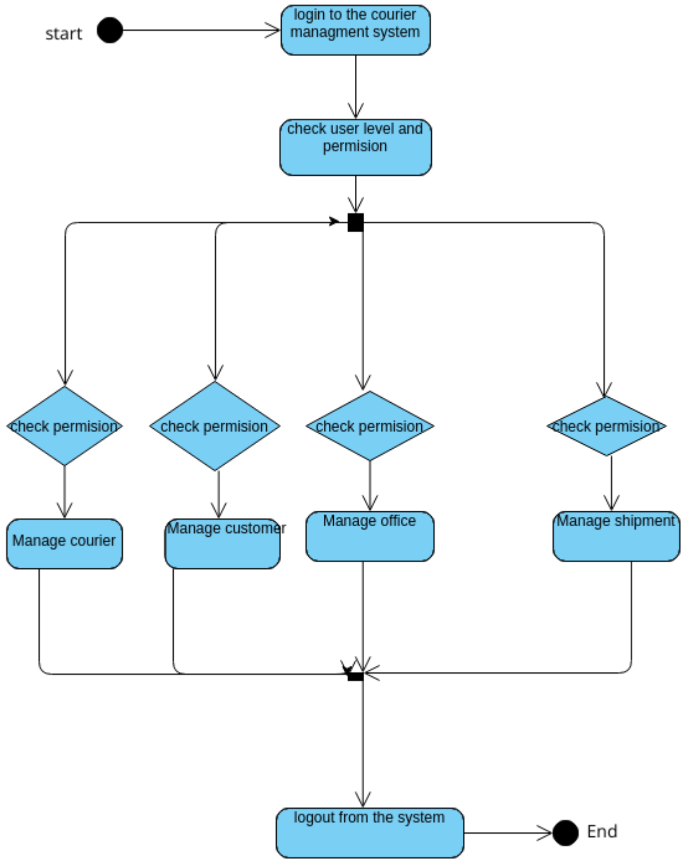
10.3 Level 2

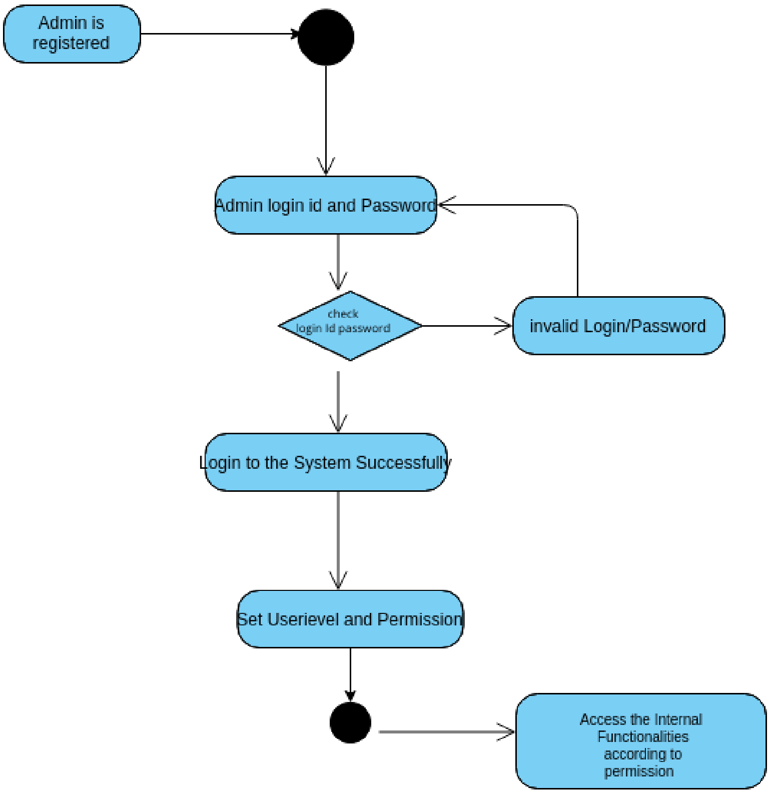


11. Sequence Diagram

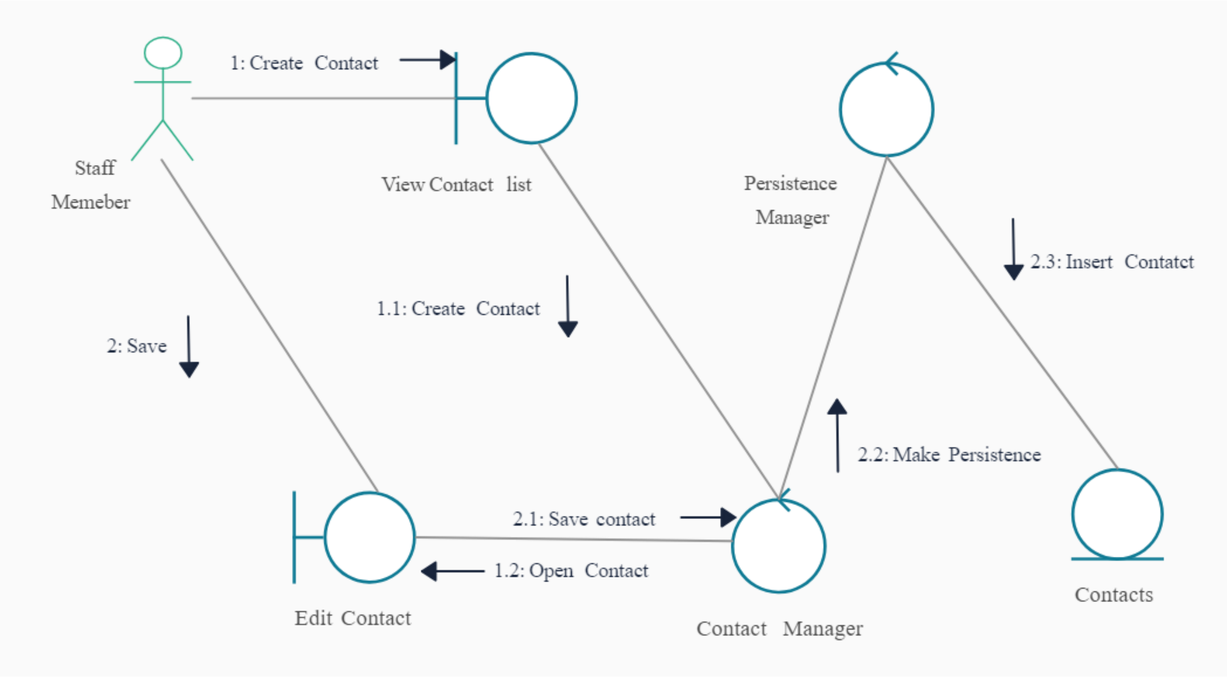


12. State Chart Diagram



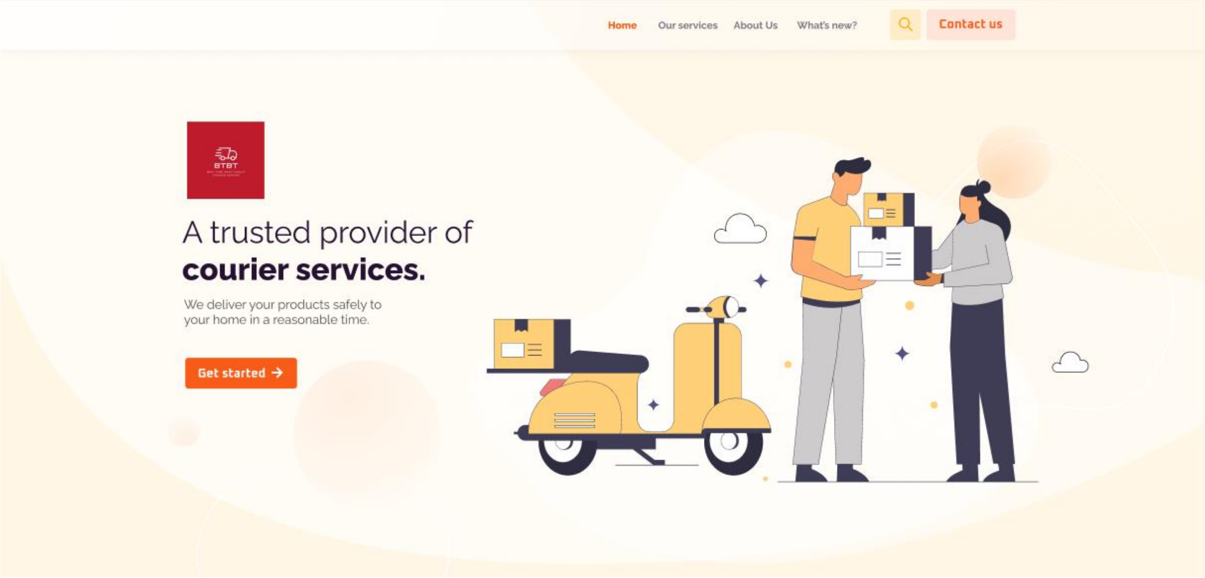

Admin is registered
Admin login id and Password
check login Id password
invalid Login/Password
Login to the System Successfully
Set Userievel and Permission
Access the Internal Functionalities according to permission

13. Collaboration Diagram



14. User Interface diagram

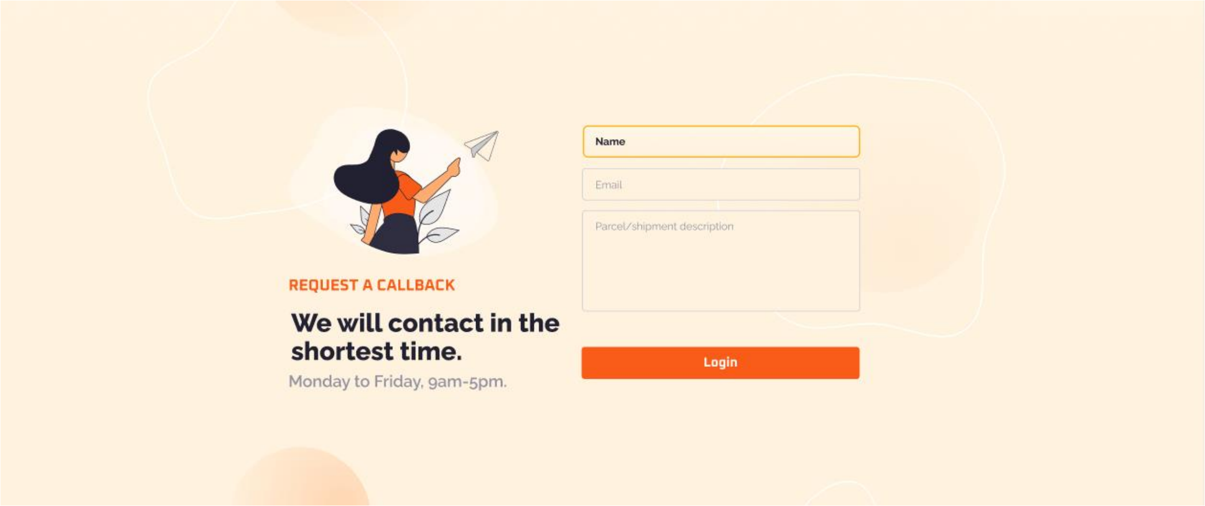
Landing Page:



A trusted provider of **courier services.**

We deliver your products safely to your home in a reasonable time.

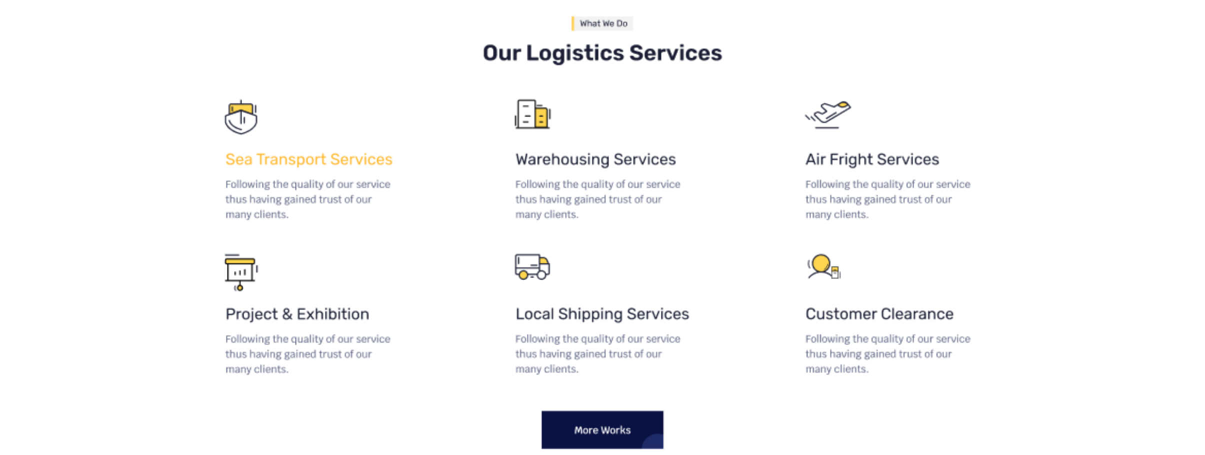
User onboarding page –



REQUEST A CALLBACK

We will contact in the shortest time.

Monday to Friday, 9am-5pm.



Our Logistics Services

Sea Transport Services

Following the quality of our service thus having gained trust of our many clients.

Warehousing Services

Following the quality of our service thus having gained trust of our many clients.

Air Fright Services

Following the quality of our service thus having gained trust of our many clients.

Project & Exhibition

Following the quality of our service thus having gained trust of our many clients.

Local Shipping Services

Following the quality of our service thus having gained trust of our many clients.

Customer Clearance

Following the quality of our service thus having gained trust of our many clients.