



Software Engineering Department
ORT Braude College
Capstone Project Phase A – 61998

Taxi station management system.

Advisor: Zeev Barzily

Project code: 24-1-D-15

Student 1: Shorok Heib

ID: 315959429

E-mail 1: Shorok.heeb@e.braude.ac.il

Student 2: Emad Suliman

ID: 322386558

E-mail 2: emadsleman30@gmail.com

GIT Link: <https://github.com/emadsleman/Taxi-station-management-system.git>

Content

1. Introduction.....	3-4
2. Background and Related Work.....	4-6
3. Expected Achievements.....	7
3.1 Outcomes	7
3.2 Unique Features.....	7-8
3.3 Criteria for Success.....	8
4. Research / Engineering Process.....	9
4.1 Process.....	9
4.1.1 Research – Taxi station management.....	9
4.1.2 Methodology and Development Process.....	10
4.1.2.1 Planning.....	10
4.1.2.2 Implementation.....	10
4.1.2.3 Iterative development.....	10
4.2 Product.....	11
4.2.1 Requirements.....	11-13
4.2.2 UML Diagrams.....	14
4.2.2.1 Use Case Diagram.....	14
4.2.2.2 Class Diagram.....	15
4.2.2.3 Activity Diagram.....	16
5. Verification and Evaluation.....	17
5.1 Unit tests.....	17
5.2 Functional Tests.....	18
6. References.....	19

Abstract

The Taxi Station Management System is a comprehensive solution designed to streamline and enhance the operations of a taxi service. This system employs a hierarchical structure, managed by secretaries and a manager, to efficiently coordinate bookings, dispatching, and overall station management. The secretary role encompasses responsibilities such as booking management, customer information handling, and dispatching taxis to meet customer demands promptly. On the managerial side, the system provides oversight into the overall performance, user management, and strategic decision-making through detailed reports and analytics. Both roles benefit from shared features like secure authentication, real-time tracking, payment processing, and a feedback mechanism. The system aims to optimize taxi station operations, improve customer satisfaction, and provide a robust platform for effective management. Emphasizing user experience, scalability, and security, this Taxi Station Management System offers a comprehensive solution to address the diverse needs of a dynamic taxi service environment.

1.Introduction

In the rapidly evolving landscape of urban transportation, the efficient management of taxi services plays a pivotal role in ensuring a seamless and reliable experience for both customers and service providers. As the demand for convenient and on-demand transportation services continues to rise, the need for an organized and responsive taxi station management system becomes increasingly evident.

The proposed capstone project addresses this need by presenting a comprehensive Taxi Station Management System tailored to the specific requirements of a managerial and secretarial team. This system streamlines the day-to-day operations of a taxi station, facilitating smooth coordination between the

managerial personnel and the secretarial staff responsible for overseeing bookings and customer interactions.

With a focus on simplicity and functionality, the system aims to empower managers to efficiently handle driver and vehicle management, monitor bookings, and analyze key performance metrics. Simultaneously, secretaries can effortlessly manage customer bookings, assign available taxis, and maintain customer information.

The project encompasses secure authentication measures, real-time tracking, and an intuitive interface to enhance user experience. By incorporating features such as communication tools, reporting capabilities, and notifications, leading to enhanced operational efficiency.

As we delve into the development of this Taxi Station Management System, the objective is to provide a reliable and user-friendly platform that meets the unique requirements of a small-scale taxi station with a manager and secretary. The outcome of this capstone project aims to contribute to the evolution of urban transportation management systems.

2. Background and Related Work

In Israel, there are taxi companies operating in the public transportation sector. Each city and region can work with various taxi companies, providing taxi services to customers in different cities and settlements.

These taxi companies in Israel operate different vehicles, and sometimes they utilize computerized systems or applications aimed at improving service and providing more convenience to customers. In recent years, some taxi companies in Israel have launched applications that allow customers to read and order taxis through their mobile phones.

A taxi stand is an advertising space where taxis pick up passengers. Taxi drivers wait at the stand for potential customers. Taxi station management systems can assist in coordinating and improving processes at the stand, specializing in tracking vehicles and managing payments.

Using taxi services allows us to travel from one place quickly and efficiently to another. Unlike buses or trains, taxis do not have a predefined route and can take us from any location to any chosen destination at any time of the day.

This service has many advantages, enabling people who are not interested in using private cars to enjoy all the benefits of private transportation without a significant financial commitment.

The high availability of taxis adds to the efficiency of the service – during most hours of the day, one can easily find a taxi on the street without planning in advance or ordering a taxi home. However, if you want to pre-order a taxi or need one at an unconventional hour, a small fee and a phone call to the taxi station will allow you to book a taxi in advance.

Taxi station management systems include various tools and processes aimed at managing and improving the daily operations of the taxi station. The system's functions and roles may include:

1. **Trip Management:** Efficiently locating, coordinating, and managing trips, including planning, and organizing trips for drivers.
2. **Driver Management:** Managing drivers, including personal data reception, driver's licenses, activity management, and scheduling for trips.
3. **Vehicle Management:** Managing and maintaining vehicles, tracking repairs and periodic maintenance.
4. **Scheduling and Shift Adjustments:** Consulting and adjusting shift schedules for drivers.
5. **Payments and Invoices Management:** Generating invoices, managing refunds, and financial updates.
6. **Information and Identification Systems:** Using computerized systems to record and manage information such as vehicle locations, bookings, and customer identification.

A taxi station's secretary role can assist in various tasks, including passenger registration, ticket sales, information classification, and customer service.

Secretaries can receive customer details, open orders, coordinate with drivers for trips, establish contact with customers, and match orders with available drivers.

The role of a taxi station manager includes managing and operating the station, managing the employee team, and being responsible for all operational and organizational aspects. This may involve planning and managing trips, financial management, customer service, human resource management, and overall strategic decision-making to promote the taxi station and improve service.

Taxi stations may use various algorithms for trip management and driver matching, such as:

1. **Matching Algorithms [1]:** Identifying the suitable driver for a trip based on parameters availability and expected travel time.
2. **Shift Adjustment Algorithms [2]:** Determining the order in which drivers will serve, improving their availability, and reducing customer waiting times.
3. **Pre-scheduled Trip Management [3]:** Allocating and coordinating trips in advance, enhancing service availability, and reducing waiting times.

The use of algorithms can be tailored to the specific needs of the taxi station and the local market conditions.

3. Expected Achievements

3.1 Outcomes

In this project, we aim to develop a comprehensive platform for managing a taxi station. Our platform will provide a set of features to make management easy and effective, yielding satisfactory results, including:

3.1.1 Improving Customer Experience:

Increasing customer satisfaction by speeding up booking and dispatch processes and reducing waiting times.

3.1.2 Facilitating Communication:

Improving internal operations organization and enhancing communication between drivers and the team.

3.1.3 Effective Booking Management:

Achieving accurate organization of booking operations.

3.1.4 Strategic Oversight:

Achieving effective supervision of all aspects of taxi station operations.

3.1.5 Data-Driven Decision-Making:

Utilizing reporting and analytics tools to make informed decisions, optimize routes, and identify areas for improvement, enhancing overall performance.

3.2 Unique Features

We will ensure that our system includes these unique features, with the help of which we will achieve a high success rate.

3.2.1 User Roles:

- Manager: Has access to overall system management, including, managing drivers, monitoring performance, and generating reports.
- Secretary: Handles day-to-day operations, including managing customer bookings, and handling customer inquiries.

3.2.2 Booking and Dispatch:

- Online booking for customers.
- Real-time taxi dispatching based on customer requests.
- Automated assignment of available taxis to incoming requests.

3.2.3 Vehicle Management:

- Tracking and management of the taxi fleet.
- Maintenance schedules and tracking.

3.2.4 Customer Management:

- Customer registration and profiles.

3.2.5 Communication:

- Internal messaging system for communication between managers, secretaries, and drivers.
- Notifications for booking confirmations, driver assignments, and other important events.

3.2.6 Reporting and Analytics:

- Generate reports on key performance indicators, such as the number of bookings, revenue, and customer satisfaction.

3.2.7 Security and Authentication:

- Role-based access control to ensure that managers and secretaries have the adequate authorizations.
- Secure login and data encryption.

3.2.8 Payment Integration:

- Integration with payment gateways for online transactions.
- Invoicing and billing features.

3.2.9 Mobile Application:

- A mobile app for drivers to receive and manage bookings.
- A customer-facing app for easy booking and tracking.

3.3 Criteria for Success

The success criteria for our project include the number of customers seeking service from the taxi station and their satisfaction with the services provided to them through the secretary and station management, facilitated by the system we will develop.

4. Research / Engineering Process

4.1 Process

4.1.1 Research – Taxi station management

Regarding expanding our understanding of taxi station management, we focused on addressing the following main question:

- What are the most important factors influencing the decision of a taxi station manager to use our system?

Research focus on this question will help identify the specific needs of taxi station management and develop a system that meets those needs effectively.

We consulted a variety of materials, from academic articles to videos, to address these issues and increase our expertise. After reviewing these resources, we gathered to discuss our findings and identify key aspects to prioritize when developing the system.

Some of the conclusions we reached include the need to add a questionnaire after conducting an evaluation in the simulation. The questionnaire could include the following:

- Overall, how satisfied are you with the taxi station management system?
- How easy or difficult was it to use the system?
- What features or functionalities did you find most useful in the system?
- What features or functionalities would you like to see added to the system in the future?
- Was the system reliable and stable during use?
- Were you able to easily track and manage taxi operations using the system?
- How would you rate the speed and efficiency of the system?

If we find a correlation between the questionnaire responses and our final analysis, this will validate the taxi station management system.

4.1.2 Methodology and Development Process

To build a system that meets our objectives, we have constructed a work plan through which we organized all the necessary steps to achieve our goals, starting from the learning process and progressing through the system development. After conducting the necessary research, we found that the Agile approach is suitable for our project. This approach highlights the importance of working in stages with continuous planning and ongoing learning.

To gain a comprehensive and clear understanding of the strengths and weaknesses of current technologies in the market, we conducted a thorough research that helped us comprehend the matter. The aim of this research was to enhance the weaknesses of these technologies and create a more efficient system. This required a meticulous analysis of the current market landscape to identify areas where we could make a meaningful contribution.

Our process is divided into 3 sections:

4.1.2.1 Planning:

Initially, we created a list outlining the system requirements, drew a clear system diagram, and then identified the technologies we would use to develop both the front-end and back-end interfaces of the system.

4.1.2.2 Implementation:

This involved creating the main screens for users and building the fundamental infrastructure.

4.1.2.3 Iterative development:

We will gradually build the system until we achieve the desired result for the application. Each time, we will construct the system while improving weaknesses through continuous modifications made to the system.

4.2 *Product*

4.2.1 Requirements

Req. Num	Requirements description	Requirements type (FR or NFR)
1	The system allows to add data for new employees.	FR
2	The station manager can add data for new employees.	NFR
3	The customer can call the reception desk to order a taxi.	NFR
4	The system should provide functionality for customers to place taxi orders by calling the reception desk.	FR
5	The receptionists schedule the shifts for the drivers.	NFR
6	The system should ensure that the process of placing taxi orders by calling the reception desk	FR
7	The system allows the station manager to change employee details.	FR
8	The station manager can change employee details.	NFR
9	The driver receives payment from customers and provides a receipt.	NFR
10	The system should facilitate the process for drivers to receive payments from customers and issue receipts.	FR
11	The driver transfers the money to the taxi station.	NFR
12	the receptionists count the cash and handle all the invoices.	NFR
13	The system should enable	FR

	receptionists to accurately count cash received from drivers.	
14	The system allows the receptionists to enter the customer's name and order details.	FR
15	The receptionists can enter the customer's name and order details.	NFR
16	The system should provide functionality for receptionists to input the customer's name and order details accurately.	FR
17	The receptionists match a driver for the ride.	NFR
18	The system should enable receptionists to efficiently match a driver with each ride request.	FR
19	The driver can report a malfunction in the taxi.	NFR
20	The system should provide a feature that allows drivers to report any malfunctions in the taxi they are operating.	FR
21	The receptionists contact the garage to address the malfunction.	NFR
22	The system should enable receptionists to promptly contact the garage.	FR
23	The system allows to identify users by username and password.	FR
24	The username is unique for each user.	NFR
26	The system ensures that each user is assigned a unique username.	FR
27	User types include the station manager, receptionists, customer, and driver.	NFR
28	The system should distinguish	FR

	between user types, including station manager, receptionists, customer, and driver.	
29	The system allows identifying the user type immediately after logging in.	FR
30	The system allows changing the order status after handling it.	FR
31	Each reservation includes a time, destination, and origin.	NFR
32	The system allows the receptionists to remove the reservation.	FR
33	The receptionists can be removing the reservation.	NFR

4.2.2 UML Diagrams

4.2.2.1 Use Case Diagram

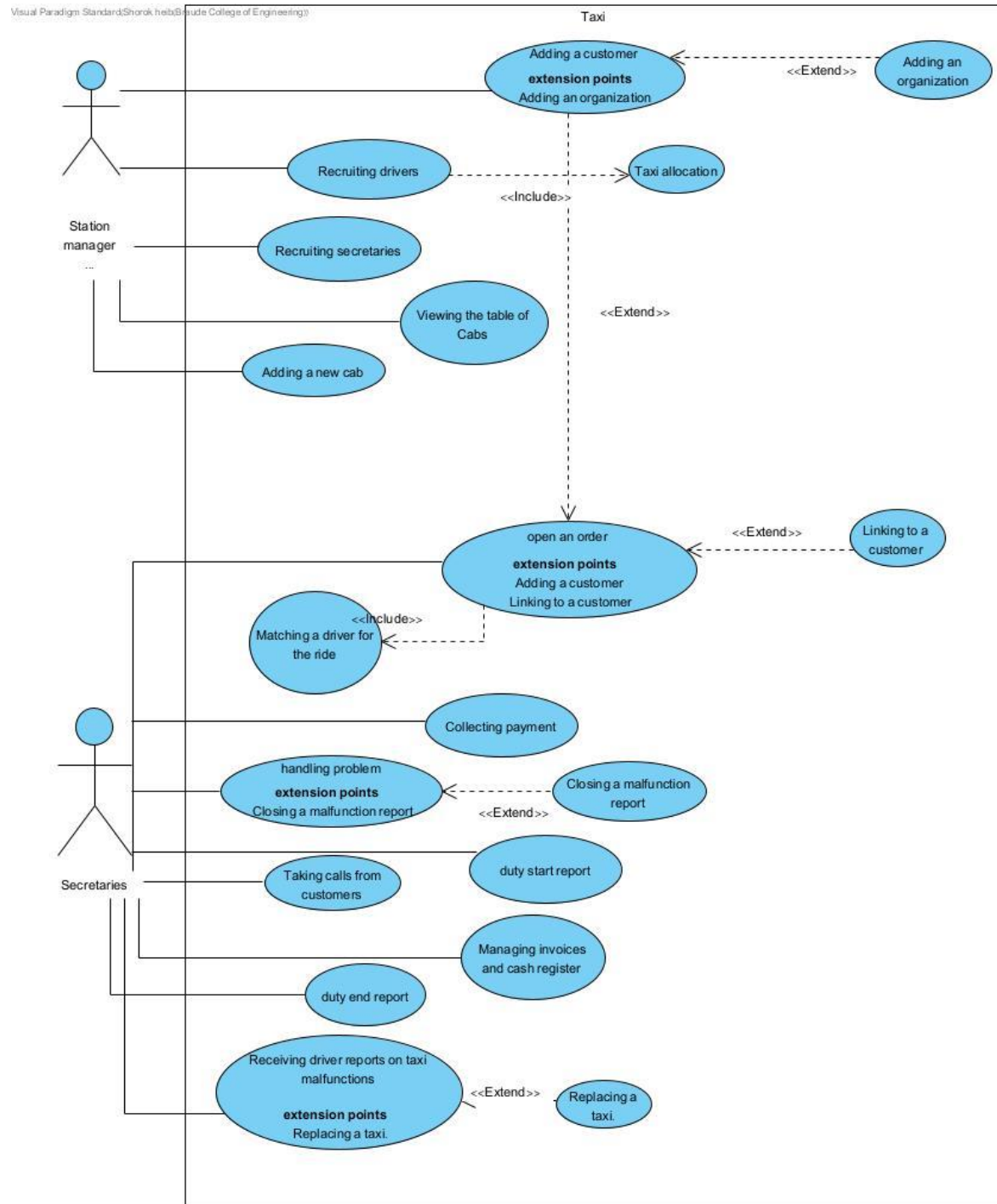


Fig1: Use Case Diagram

4.2.2.2 Class Diagram

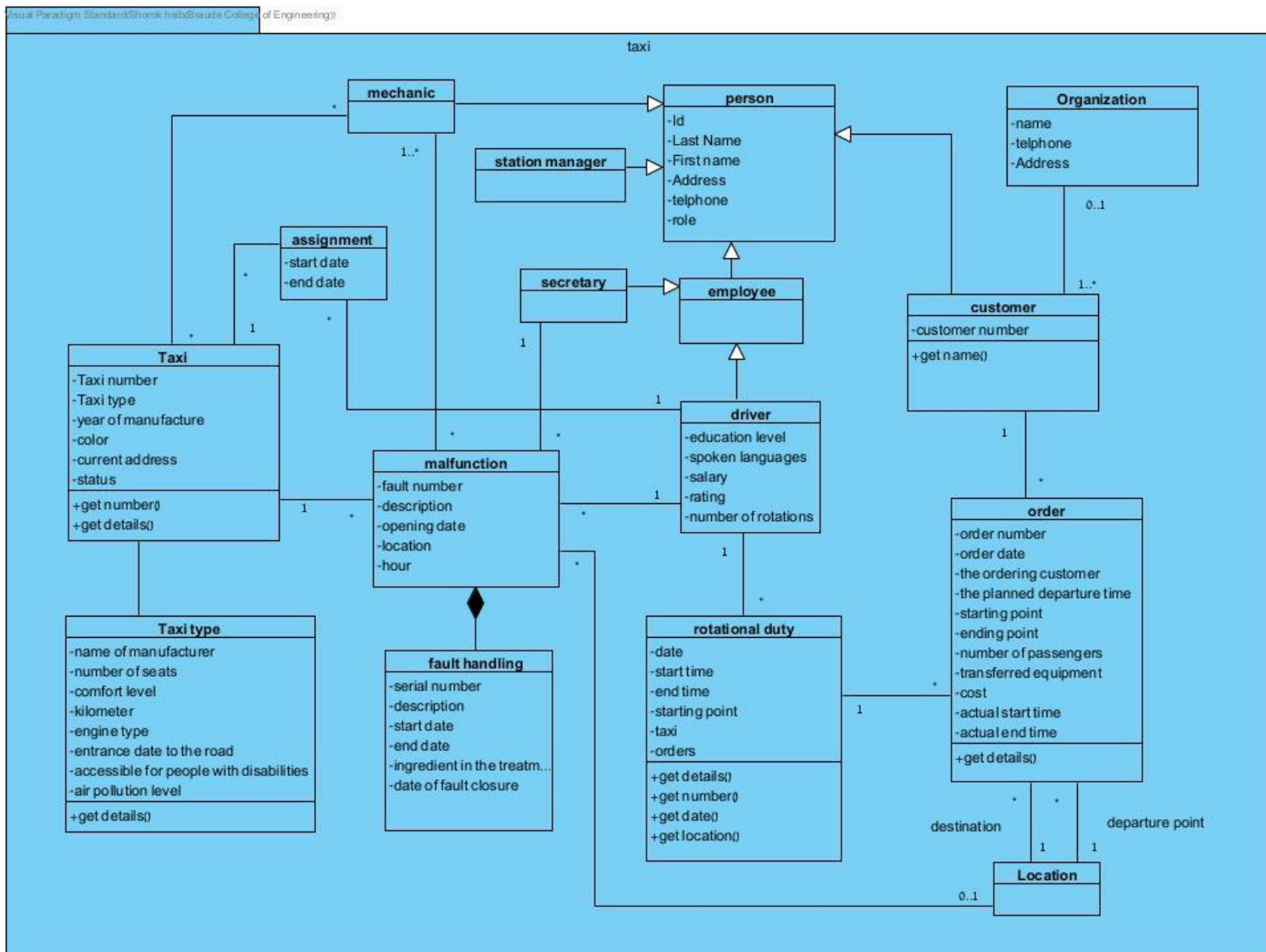


Fig2: Class Diagram

4.2.2.3 Activity Diagram

Ordering a taxi through the reception desk.

1. The secretary types the origin and destination.
2. The secretary asks for the customer's name.
3. If the customer is a member, she links the order to the customer.
4. If the customer is not a member, she types in the customer's name.
5. The secretary contacts the driver available during the order.
6. If the driver is not assigned to the task while placing the order, she checks how long it will take for him to reach the customer's location. If it takes a long time, she returns to step 5, and if it takes a short time, she assigns the order to the customer and notes the order process.
7. If the driver is not available for assignment, the secretary checks if there is another available driver with availability.
8. If there is another driver available during the order, the secretary proceeds to step 5.
9. If there is no additional driver available, the secretary informs the customer that there is no taxi available for their order.

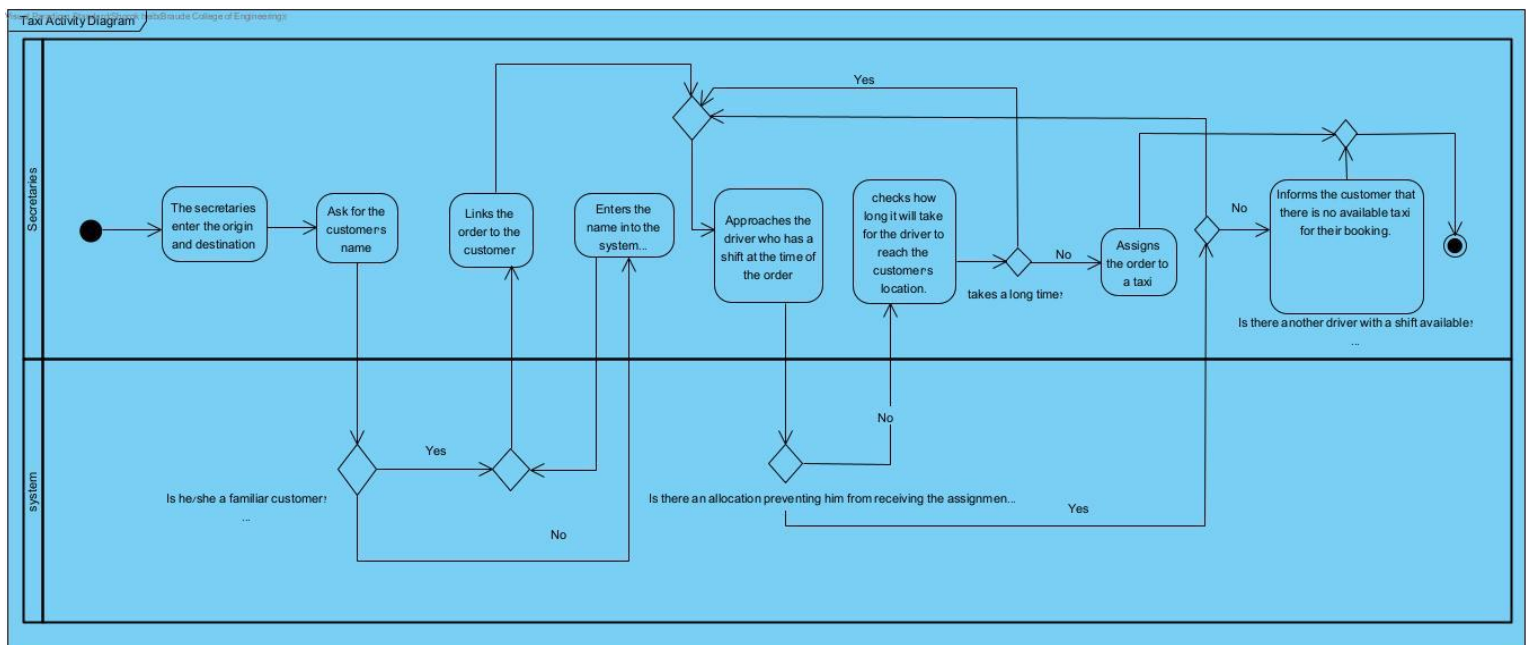


Fig3: Activity Diagram

5 Verification and Evaluation

5.1 Unit tests

No	Test Subject	Expected result
Login form		
1	Enter an empty username or password.	Error displays a message: "Please fill in the fields".
2	Log in with a 'frozen' status username and password	Display message: "User is Frozen".
3	Enter a valid username and password	Display message: "Login succeed".
4	Enter the wrong username or password.	Error displays a message: "Invalid details, try again" We don't want to give information about which field is invalid.
Add new driver		
5	Enter empty details (one or more)	Error displays a message: "Please fill in the fields".
6	Enter an existing id.	Display a message: "This driver is already in the system".
7	Enter an existing email address	Display a message: "This email address is already in use".
8	Enter an existing phone number	Display a message: "This phone number is already in use".
9	Enter an existing username	Display a message: "This username is already in use".
10	Enter valid details	Display a message: "driver registration succeeds, he can now log in with the username and password".
Add new secretary		
11	Enter an existing id.	Display a message: "This driver is already in the system".
12	Enter new email address	Display a message: "This email address not in use".
13	Enter an existing username	Display a message: "This username is already in use".
14	Enter valid details	Display a message: "driver registration succeeds, he can now log in with the username and password".
15	Enter empty details (one or more)	Error displays a message: "Please fill in the fields".
16	Enter an existing phone number	Display a message: "This phone number is already in use".
Add new customer		
17	Enter an existing id.	Display a message: "This driver is already in the system".
18	Enter an existing phone number	Display a message: "This phone number is already in use".
19	Enter an existing email address	Display a message: "This email address is already in use".
20	Enter valid details	Display a message: "customer registration succeeds, he can now log in with the username and password".
Open new order		
21	Enter an existing name and location	Error displays a message: "This order is already existing"
22	Enter empty details (one or more)	Display message: "Please fill in the fields".
23	Enter valid details	Display a message: "Registration succeed, please wait for approvement".

5.2 Functional Tests

No#	Test Case	Steps	Expected result
Login form			
1	Navigation from the login screen to the dashboard of the manager screen after successful login.	Open the system, enter valid login credentials, and click on the 'login' button.	Move to the manager form.
2	Login with a valid secretary username and password		Move to the secretary form.
Add new driver			
3	Navigation from the screen dashboard of the manager screen to the list of driver screen.	Open the system, click on the 'requitement of the drivers' button.	The system should navigate to the 'list of drivers' screens
4	Navigation from the screen list of drivers screen to add new driver screen.	Open the system, click on the 'Add new+' button.	The system should navigate to the 'Add new driver' screen.
Add new customer			
5	Navigation from the screen dashboard of the manager screen to the list of customer screen.	Open the system, click on the 'requitement of the customer' button.	The system should navigate to the 'list of customers' screens
6	Navigation from the screen list of customer screen to add new customer screen.	Open the system, click on the 'Add new+' button.	The system should navigate to the 'Add new customer' screen.
Monthly updates			
7	A driver has X orders for the last month.		The driver will receive a bill that includes the amount of payment he has to pay "price per order * X".
8	number of customer orders has decrease.		The customer status will be changed from 'active' to 'inactive'
9	The cab has broken down		The status of the cab has changed (active/inactive)
10	A driver has retired.		The driver will remove from the list.
Add new secretary			
11	Navigation from the screen dashboard of the manager screen to the list of secretary screen	Open the system, click on the 'requitement of secretary' button.	The system should navigate to the 'list of secretaries' screens
12	Navigation from the screen list of drivers screen to add new driver screen.	Open the system, click on the 'Add new+' button.	The system should navigate to the 'Add new secretary' screen.

6. References

[1] Matching Algorithms:

Chen, X., Xu, M., & Wang, X. (2017). A driver-task matching model for online ride-hailing systems. *Transportation Research Part E: Logistics and Transportation Review*, 100, 1-19.

Ke, J., Zheng, H., Yang, H., & Chen, X. M. (2017). Short-term forecasting of passenger demand under on-demand ride services: A spatio-temporal deep learning approach. *Transportation Research Part C: Emerging Technologies*, 85, 591-608.

[2] Shift Adjustment Algorithms:

Moreira-Matias, L., Ferreira, J., Mendes-Moreira, J., & de Sousa, J. F. (2015). An incremental vector autoregressive model for transportation service demand forecasting. *IEEE Transactions on Intelligent Transportation Systems*, 16(4), 1967-1978.

Nourinejad, M., & Roorda, M. J. (2016). A dynamic carsharing decision support system. *Transportation Research Part E: Logistics and Transportation Review*, 86, 18-30.

[3] Pre-scheduled Trip Management:

Dong, Z. S., Lam, A. Y., & Leung, V. C. (2018). A critical review of academic literature on ride-sharing optimization: Models, methods, and future research directions. *Transportmetrica A: Transport Science*, 14(7), 654-685.

Furuhata, M., Dessouky, M., Ordóñez, F., Brunet, M. E., Wang, X., & Koenig, S. (2013). Ridesharing: The state-of-the-art and future directions. *Transportation Research Part B: Methodological*, 57, 28-46.