PAIX-TRAVAIL-PATRIE

MINISTERE DE L’ENSEIGNEMENT SUPERIEUR

UNIVERSITY OF BUEA

PEACE-WORK-FATHERLAND

MINISTRY OF HIHGER EDUCATION

UNIVERSITY OF BUEA



**UNIVERSITY OF BUEA**

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER ENGINEERING

CEF 440: INTERNET PROGRAMMING AND MOBILE PROGRAMMING

Software design document of a passenger positioning system

Presented by:

EMADE ROSINE NSAH..........................................................FE20A032

ETUGE GIDEON......................................................................FE20A036

NOUGHUE LEMOUPA FRANK..............................................FE20A092

TAJOUEGO DJUIDJA ESTRELA............................................FE20A107

TAMAH JUSTENE....................................................................FE20A108

**Course Instructor**: Dr Nkemeni Valery

Table of content

Introduction

Implementation overview

Design and Implementation plan

Conclusion

# INTRODUCTION

Purpose

The aim of this project is to design a user friendly application for drivers and passengers of the Buea city. This application comes to solve the problem of time minimization for the passengers and lower the fuel consumption for Taxi drivers

# IMPLEMENTATION OVERVIEW

This project is put in place to bring the passengers and the drivers closer to each other.

According to our system, the drivers are of two categories. We have the freelance and the hired drivers

Freelance drivers:

These are drivers who function independently of the municipal council. That is, they are the sole decision makers of their routes. The decide on their payment fees and decide when and how to operate

Hired driver:

These are drivers that work under the council or a defined organization. The taxis are owned by the organization and the drivers work daily. They have no say in the price exchange as they receive their payments solely at the end of the month. These drivers can be hired to drive a particular passenger the whole day depending on the number of hours requested by the passenger

The freelance and the hired drivers as well as the passengers will be required to create accounts on the application. Once the accounts are created they can login any time they want to use the service

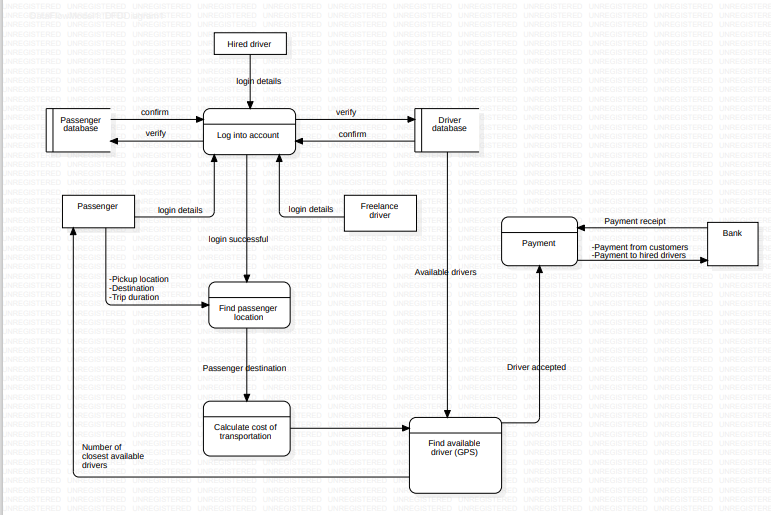
The interactions occurring between the driver and the passengers can be demonstrated in the diagram in the following section

# DESIGN AND IMPLEMENTATION PLAN

In this section will be the describing the system with the use of the different unified modeling diagrams

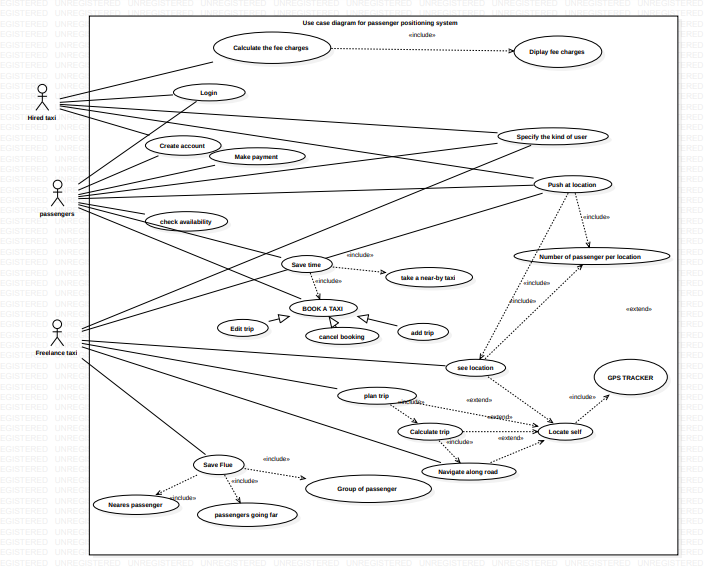
## Dataflow diagram

We provide the flow information from one object to another and how they inter relate



## Use case diagram

The use case diagram provides a high level view of the interactions between the actors which could be the users of the external systems and the systems being designed. It shows the various use cases (tasks) that the system will perform to meet the needs of the drivers and the passengers. With this diagram, we can identify the actors involved, the goals the each want to achieve and the interactions between them. With this information, we get to identify the system requirements, prioritize the development efforts and ensure that the system meets the needs of the users. Below is a sample use case diagram for the passenger positioning system.



**Diagram description**

Actors of the system include:

* **Hired Taxi:** A Taxi employed by a municipal council or some other organization which acts only on demand

His interface is both visual and vocal permitting him to get and input sensitive data like location and Destination

* **Free Lance Taxi:** A Taxi that acts by itself placing profit above all

His interface is mainly vocal, allowing to get sensitive data without being distracted.

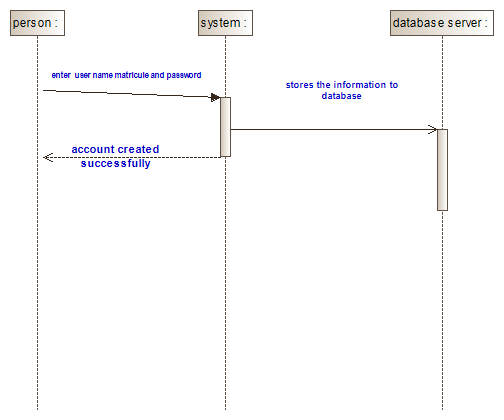
* **Passenger:** A visual interface allowing him to accurately enter needed information to get a taxi

A detailed explanation of the different use cases involved with the system can be demonstrated in the series of sequence diagrams below

## Sequence diagram

These diagrams provide information about the interactions between the different objects in the system. It portrays the flow of messages between these objects in a particular scenario of a use case. With this information, we are able to understand the flow of control and data through the passenger positioning system, identify potential performance issues and validate the correctness of the systems behavior. Below are a set of diagrams that have been designed per use case

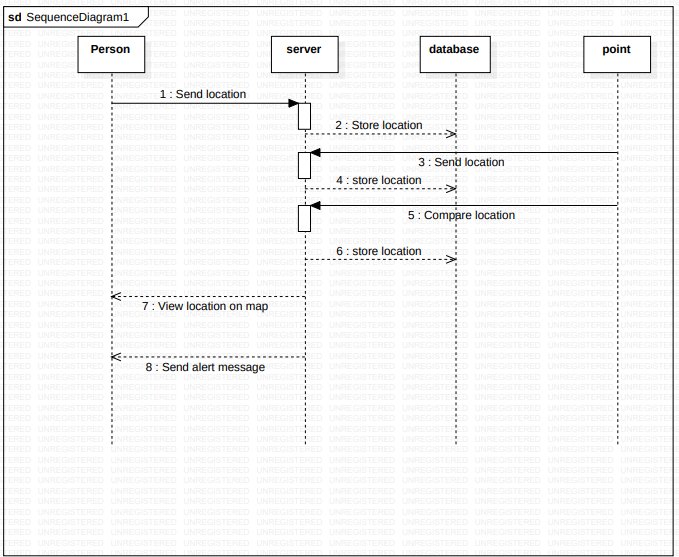
1. **Create account and login sequence diagram:**



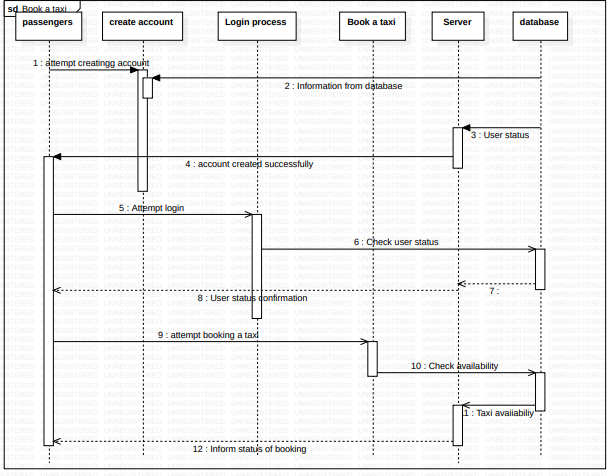
**Diagram description**

**Create an account:**  
This functionality involves all users. Before they reach to create an account page, they must first specify the kind of user they are. They enter their information, and the information is checked on the server for correctness and existence. If the information is correct but does not yet exist in the database, the account will be created.  
 **Login**:  
The login page comes after an account has been created. The users involves enters their credentials and the system checks in the database for existence. If the information is correct, the user then views the dashboard

1. **Upload location sequence diagram(plan trip):**



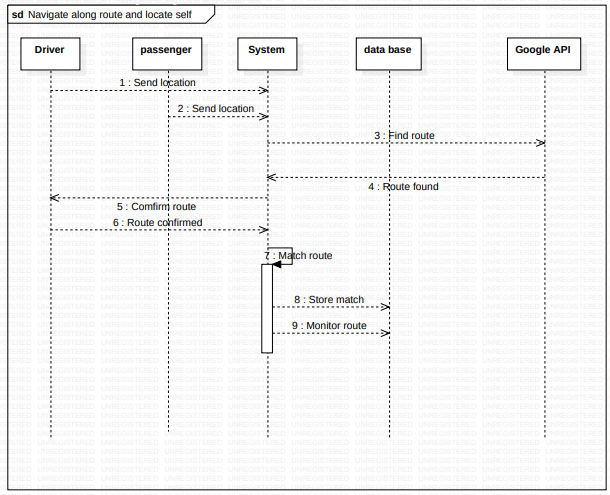
1. **Booking sequence diagram:**



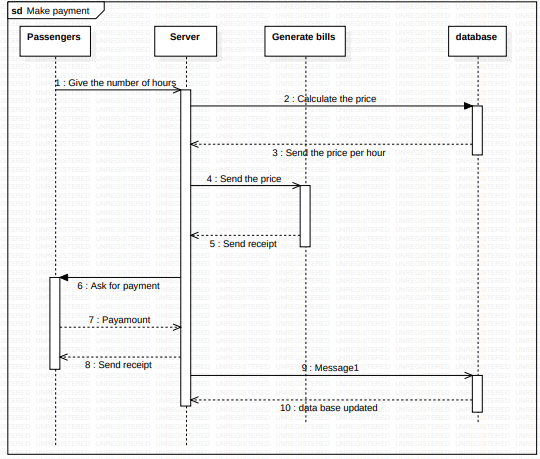
**Diagram description:**

It is used by the passenger. When the passenger enters the system, the system asks what kind of user he is. After the kind of user is specified by the person, the person enters the dash board. The passenger then sends a request to book a taxi. The system then asks the user to either create an account or login. For new users, they will create an account and then log in. For old users, they will login to their account. After logging in, they select make payment, and the system asks them the number of hours they will want the taxi to work with them. After specifying the number of hours, the system calculates the price and sends it. The user then sends a request to the server, and the server requests the card information. After input and validation, the user receipt is displayed and ready for download.

1. **Location sequence diagram:**  
   This use case is used by all the users of the system. The person involved sends a request to the server for the location. The server then sends the information to the database. The Google API now sends the point to the server, and the server sends the information to the database. The API now compares the position and sends it back to the server. The server then stores it in the database. The server also sends the information to the person on Google Maps, and an alert message is given so the person can check.



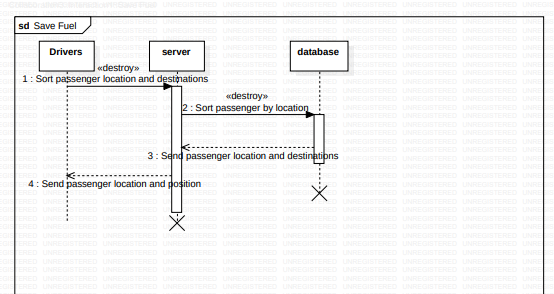
1. **Payment sequence diagram:**



**Diagram description:**

This is done by the passenger. The passenger enters and clicks on book a taxi, the system then ask for the number of hours he will want to work with the taxi. The system then calculates the amount that the passenger is supposed to pay, and the price is display to the passenger. The passenger now chooses a payment method of payment, enter his/her credentials and validate the transaction. The system after receiving the payment then gives a receipt to the passenger. The system then updates the database

1. **Saving fuel sequence diagram:**

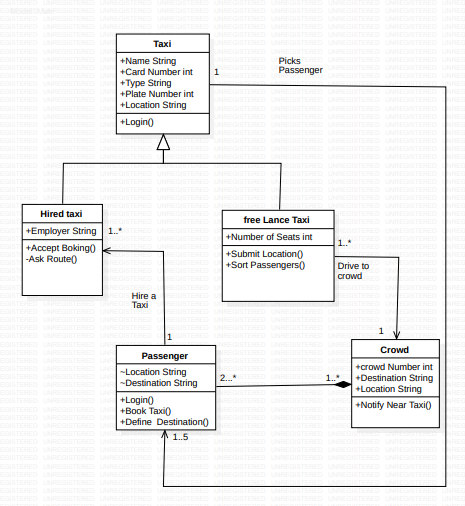


**Diagram description**

The drivers use this use case. The driver sort’s passenger location and destination from the server, and the server asks the database for this information; the database gives the information to the server. The server then counts the number of passengers by location and displays it to the drivers. The driver then bases his decision on the population per location, the population going in the same direction, and the shortest distance from his location.

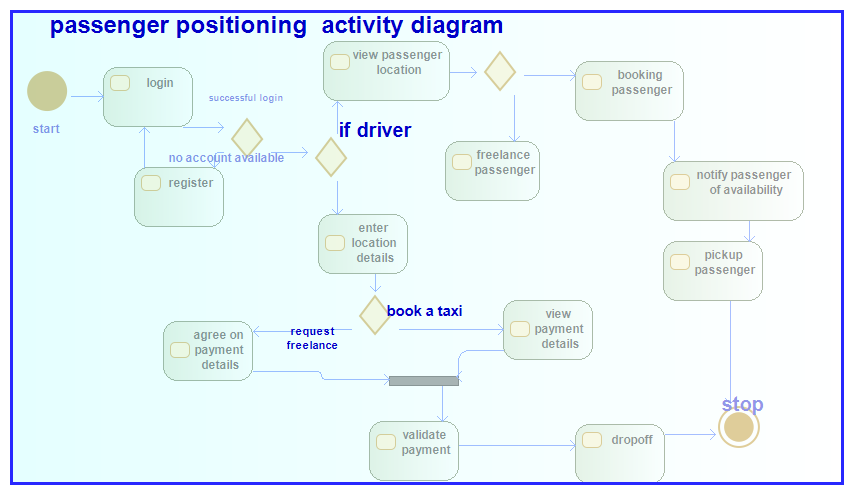
## Class diagram

This is a visual representation of the classes, interfaces and relationships in the system. We are provided wit the detailed view of the systems structure, including the attributes and methods of each class and the relationships between them. Below you are provided with the different objects that we use in the passenger positioning system, their properties and their behaviors which is used to design the systems architecture, define the systems data model and ensure that the system is scalable and maintainable



## Activity diagram

In this section, we communicate the information about the flow of activities in the system including the decisions, loops and concurrency of the passenger positioning system. Here you get the sequence of actions and their dependencies, as well as the conditions and constraints that govern them



**Conclusion**

As a team, we feel that this project has great potential to very useful to our community. It will enable the passengers and drivers to come closer to each other with having a benefit. The passenger gains time and the driver save fuel and time as well.

.