PEACE-WORK-FATHERLAND

MINISTRY OF HIHGER EDUCATION

UNIVERSITY OF BUEA

PAIX-TRAVAIL-PATRIE

MINISTERE DE L’ENSEIGNEMENT SUPERIEUR

UNIVERSITY OF BUEA



**UNIVERSITY OF BUEA**

FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER ENGINEERING

CEF 440: INTERNET PROGRAMMING AND MOBILE PROGRAMMING

Software requirements and requirements analysis 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

[I. TABLE OF CONTENT 1](#_Toc132399748)

[II. INTRODUCTION 2](#_Toc132399749)

[Purpose 2](#_Toc132399750)

[Scope 2](#_Toc132399751)

[III. REQUIREMENTS ENGINEERING 2](#_Toc132399752)

[Feasibility studies 2](#_Toc132399753)

[Product features 3](#_Toc132399754)

[Requirements gathering 3](#_Toc132399755)

[Software requirements specifications 3](#_Toc132399756)

[Project perspective 3](#_Toc132399757)

[Dataflow Diagram 3](#_Toc132399758)

[Use case diagram 4](#_Toc132399759)

[Design and implementation constraints 5](#_Toc132399760)

[Software requirements validation 6](#_Toc132399761)

[IV. SYSTEM REQUIREMENTS 6](#_Toc132399762)

[Functional Requirements 6](#_Toc132399763)

[Non-functional requirements 6](#_Toc132399764)

[V. DOMAIN REQUIREMENTS 7](#_Toc132399765)

[Software development tools that supports mobile application development 7](#_Toc132399766)

# INTRODUCTION

## Purpose

The purpose of collecting and analyzing requirements is to ensure that the expectations of the system are attained. These expectations involve the user expectations, business expectations and technical expectations of the system. This provides a basis to guide the design and implementation of the passenger positioning system.

## Scope

The requirements produced in this chapter will provide the basis to guide the design and implementation of a passenger positioning system. More specifically, to design and develop a simple and intuitive system, which shall cater the needs of the taxis drivers and the passengers in accordance to the municipal council

# REQUIREMENTS ENGINEERING

The process to gather the software from clients and analyze the is known as requirement engineering. The goal of requirement engineering is to develop and maintain sophisticated and descriptive software requirement specifications

requirement engineering process. These processes are as follows:

## Feasibility studies

This involves getting the ideology of the application from the client. That is, questioning the client to get out the information about the basic functionalities of the system that he requires. These functionalities are then analyzed by the analyst to verify if the idea can be **SMART** that is the idea is to be **S**pecific, **M**easurable, **A**chievable, **R**elevant and **T**ime bound.

Some of the features requested by the client can be viewed below:

### Product features

* + - * GPS Tracking: The system should be integrated with GPS (Global Positioning System) to track the real-time location of the vehicle.
      * Passenger console: The system should have a passenger console for passengers to book their rides, track the location of their ride, and communicate with the driver.
      * Driver Console: The system should have a driver console to enable drivers to accept passenger requests, view their trip details, and navigate to their destinations.
      * Routing and Navigation: The system should have routing and navigation functionality to offer optimized routes for drivers, reducing travel time and improving vehicle utilization.

## Requirements gathering

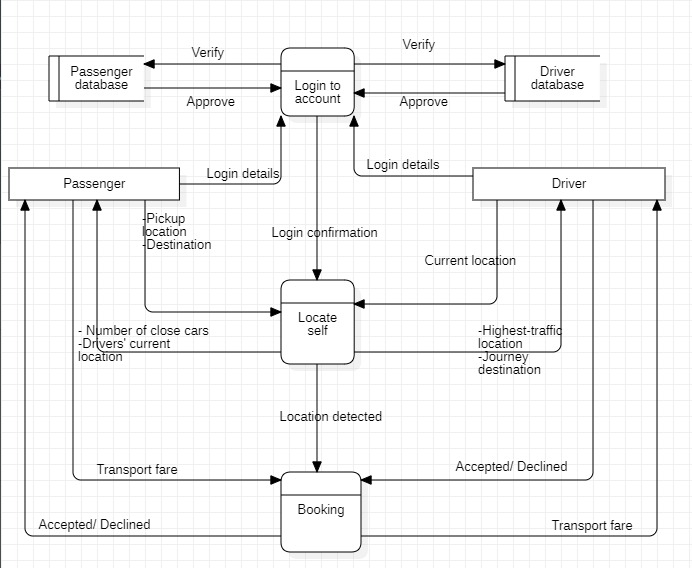
This involves two things which are the gathering of functional requirements from the client and brainstorming using various engineering techniques. A series of surveys was done to get what is expected of the application from the user and a large number of functionalities were requested by the users.

## Software requirements specifications

### Project perspective

* + The system is put in place to help reduce the passengers waiting time
  + There is access to more passenger requests for pickup and drop off, increasing opportunities for income with consistent daily work
  + There is access to route information, that is, the pick-up location and ride directions

### Dataflow Diagram

****

**Diagram description**

**Processes**

1. **Login to account**

**External entities**

1. **Driver:** Drivers log into their accounts by filling in the appropriate information.
2. **Passenger:** Passengers log into their accounts by filling in the appropriate information.

**Data stores**

1. **Driver database:** Verifies and confirms the authenticity of drivers.
2. **Passenger database:** Verifies and confirms the authenticity of passengers.
3. **Locate self**

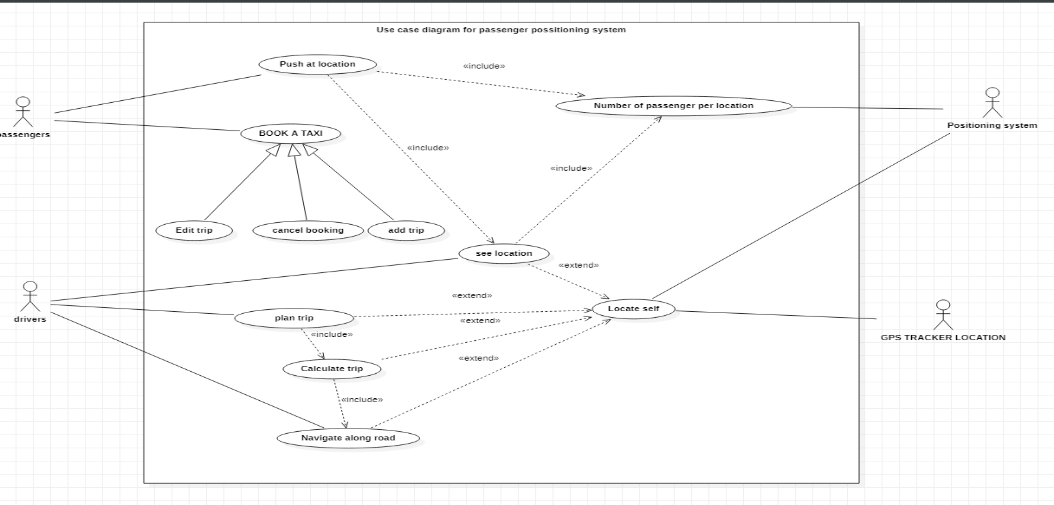
**External entities**

1. **Driver:** Drivers receive as output from the system locations with high number of passengers and their various destinations. They specify their current location in the system.
2. **Passenger:** Passengers input into the system their positions by specifying their pickup locations and their destinations. They receive as output the number of cars close to their location and the driver’s location.
3. **Booking**

**External entities**

1. **Passenger:** Passengers specify their fare which can be either accepted or declined by the available drivers.
2. **Driver;** Drivers receive passengers’ transportation fares and can either accept or decline.

### Use case diagram



**Use case description**

|  |  |  |
| --- | --- | --- |
| Use cases | Description | Pre-condition |
| Push at location | A use case that will be use by users to tell the drivers where they are, and to know where they are if they are missing. | A button must be clicked |
| Book a taxi | A use case that allows users to book a taxi for a day. | The taxi must indicate its availability, and the taxi must be a taxi on call |
| Plan trip | A use case that use by the drivers to determine which direction he should go | This is don after the system see the drivers location and the number of passenger at different locations, calculate the shortest path with more passengers and display the road with high chance of getting profit |
| Number of passengers per location | A use case that will let the drivers know the number of passenger that is available at the particular location | This will be done after the user has clicked a location button in the app, the system detects the user location and the increment the number of users at that location |
| See location | A use case that allows drivers and passengers to know exactly where they are, in case they do not know. | Can only happen when the user click a button indicating that they want to see |

### Design and implementation constraints

The design and implementation of the passenger positioning system face several constraints that can affect its functionality, usability, and effectiveness. The following are some of the design and implementation constraints of the passenger positioning system:

* + - Technical Constraints: The technical constraints include the selection of a suitable platform for the system, software and hardware limitations, and network connectivity. The system's performance and functionality can be affected by constraints such as processing power, memory, and storage capacity.
    - Scalability constraints: The road passenger positioning system needs to be scalable to accommodate an increasing number of passengers and drivers. Scalability constraints such as database management, data storage, and server capacity can affect the system's ability to accommodate large volumes of data.
    - Security Constraints: As the system deals with sensitive and personal data such as passenger details, driver information, and location tracking data, security constraints are significant. Constraints such as data encryption, firewalls, and access controls must be considered during the system design and implementation.
    - User Interface Constraints: The user interface design must be intuitive, user-friendly, and accessible across different devices, operating systems, and screens.
    - Legal Constraints: Legal constraints such as data protection laws and privacy regulations, licensing, and intellectual property rights must be adhered to.

## Software requirements validation

After a thorough investigation of the requirements, they were processed and found to be legal, practical and well interpreted following a set of requirement elicitation techniques. These techniques include:

* Task analysis
* Domain analysis
* Brainstorming
* observations

# SYSTEM REQUIREMENTS

## Functional Requirements

* Passengers should be able to specify their locations and destinations in the app
* Drivers should be able to view the location of passengers in the app
* The app should provide direction to easiest path in reaching the passengers
* The app should be able to determine the most profitable point for the drivers
* The application should get the drivers location and be able to approximate his arrival time
* Future Passenger Should be able to use the application to book a taxi
* Future Passengers Should be able to cancel a booked taxi
* Application should be able to differentiate between hired taxi and freelance taxi
* Application Should be able to communicate with a GPS tracker
* The Application should be able to run on various platforms

## Non-functional requirements

* **Performance:** The passenger positioning system must be able to perform efficiently and effectively, with minimal lag or delay in response times.
* **Availability:** The passenger positioning system must be available for use at all times, without any downtime or disruptions.
* **Maintainability**: The passenger positioning system must be easy to maintain and update, with minimal disruption to the system’s operation.
* **Compatibility:** The passenger positioning system must be compatible with different operating systems, devices, and software versions.
* **Security:** The passenger positioning system must have robust security measures in place to protect user location and data
* **Reliability**: The passenger positioning system must be reliable and consistent in its operation, without any errors or glitches.
* **Usability:** The passenger positioning system must be easy to use and navigate, with clear instructions and intuitive design.
* **Scalability**: The passenger positioning system must be able to handle increasing amounts of data and users as the system grows over time.
* **Interoperability**: The passenger positioning system must be able to integrate and communicate with other systems and devices.
* **Regulatory Compliance:** The passenger positioning system must comply with relevant regulations and standards for data privacy and security.

# DOMAIN REQUIREMENTS

Domain requirements reflect the environment in which the system operates, so when we talk about an application domain we mean environments for operating

and building the application.

## Software development tools that supports mobile application development

* Development framework
* Programming language
* Integrated Development environment
* Database