Software Design Document

Cab aggregator Web Application

Branch : B.Tech Computer Science

Semester & Section : Sem-5 'D'

Sl No.	Name of the Student	SRN
1.	G. Roshni	PES1UG21CS191
2.	Gaargi V	PES1UG21CS193
3.	Hansini KB	PES1UG21CS215
4.	Ishu Singh	PES1UG21CS242

Version 1.0

Printed: October 27th, 2023

1. Introduction

1.1 Purpose

The Software Design Document describes the architecture and system design for Grab a Cab, a cab aggregator website. Grab a Cab is designed to help travelers book cabs easily on the go. This document is intended for Project Managers, Software Engineers, and anyone else who will be involved in the implementation of the system.

1.2 Scope

This document describes the implementation details of the Grab a Cab (GAC) Web Application. GAC will consist of four major components: Database, Map, User, and Authentication. Each of the components will be explained in detail in this Software Design Document.

1.3 Definitions, Acronyms and Abbreviations

Acronym	Meaning
GAC	Grab A Cab
SDD	Software Design Document
OS	Operating System
API	Application Programming Interface

1.4 References

1. Google Maps JavaScript API
2. Bootstrap React

2. System Overview

MySQL Backend Database Implementation:

The MySQL backend database is intricately designed to manage user profiles, trip data, and payment information securely and efficiently. It utilizes database normalization and relational modeling to establish robust data relationships, guaranteeing data integrity and consistency.

Frontend Development using HTML and CSS:

HTML5 and CSS3 are utilized to design a visually appealing user interface. CSS preprocessors are employed to streamline stylesheet management and enable code reusability, thereby enhancing the maintainability and scalability of the frontend components.

Integration of Google Maps API for Location Services:

The Google Maps API integration offers dynamic mapping capabilities, enabling real-time tracking, precise geolocation, and interactive route visualization. Custom markers and overlays are incorporated to provide users with an intuitive and comprehensive view of available cabs and their estimated arrival times and optimized route navigation.

3. System Components

3.1 Decomposition Description

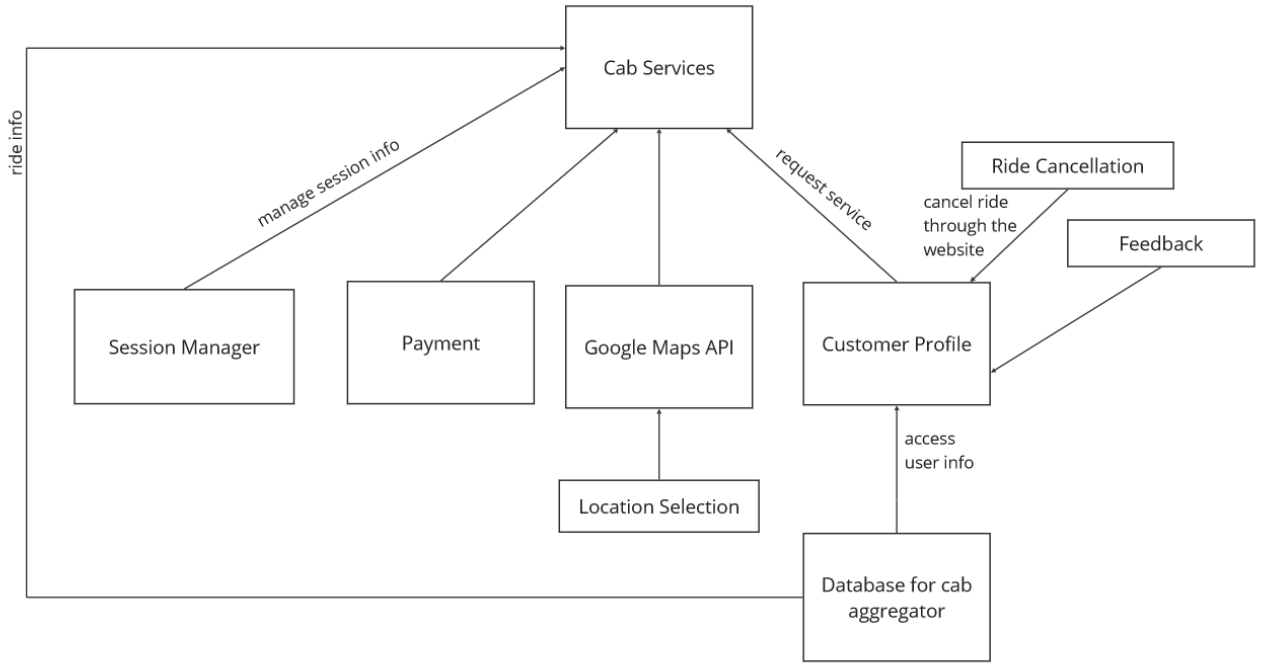


FIGURE 2

Figure 2 above shows a top down description of how the website is expected to work and how components will interact with one another. Cab Service is the main component. The cab service can store and extract data of the customer and the ride information from the database manager. The session manager is for logging in, logging out, and authenticating users. The google maps API can obtain real-time geolocation of customers and drivers. The payment can be done online. The database additionally stored the feedback from customers. The customer also has the option to cancel the ride.

3.2 Dependency Description

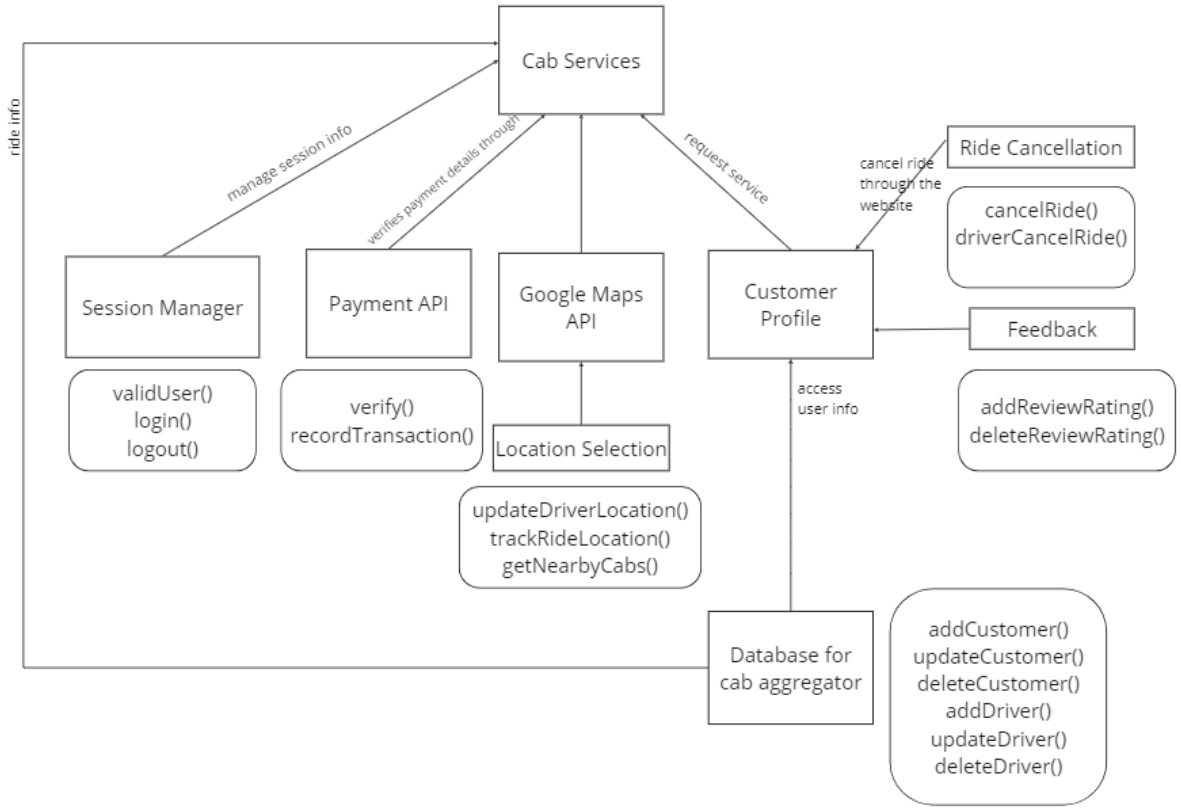


FIGURE 3

Figure 3 above represents the component diagram of our cab aggregator system and how each module is dependent on another for its functionality. Through the website, the user is expected to book the cab and proceed with the payment. Cab service system will use the Session Manager to authenticate the current customer. After being authenticated, user info will be retrieved from the database. If the user is part of the system, the system can retrieve their saved preferences required for the drop or pickup location.The default preferences can be changed according to the ride. If the user is not a member, they will be prompted to enter their location details. The preferences are stored within the database.

On collecting all the desired details, the cab booking system will progress through the ride finder

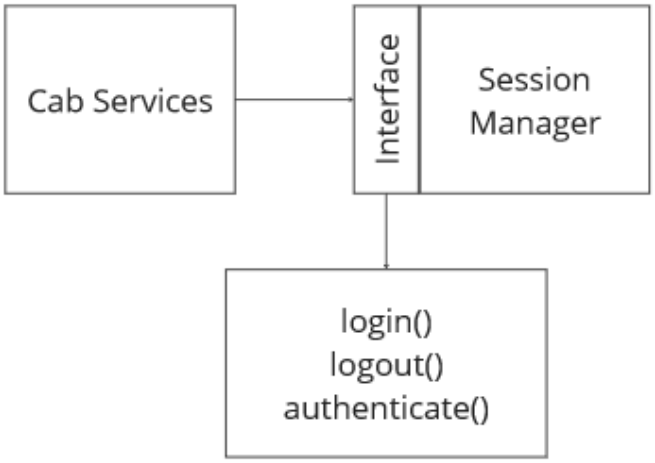
module which will generate an optimized route based on the user location preferences. The optimized routes will be stored into our database for future reference.

The booking system will pull the map data using the map API.

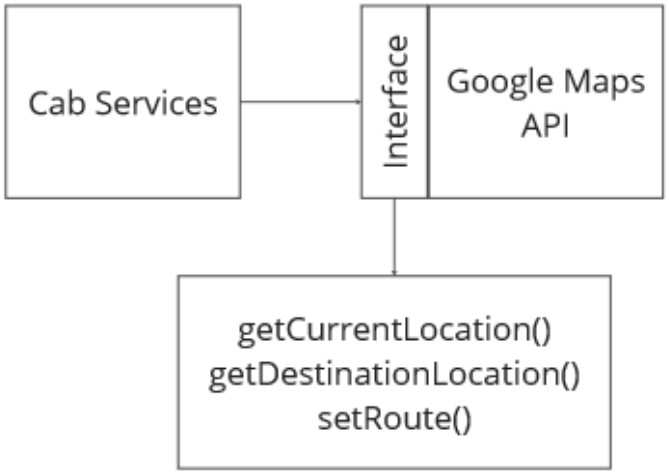
Most of the data from the cab user will be stored into our database as long as the user is an authorized member of the system which was confirmed .

3.3 Interface Description

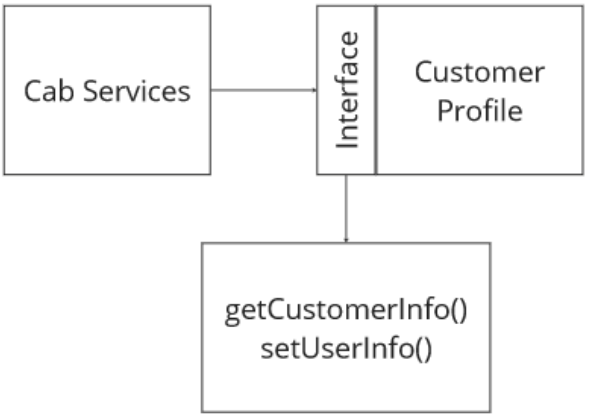
3.2.1 Cab Services to Session Manager Interface



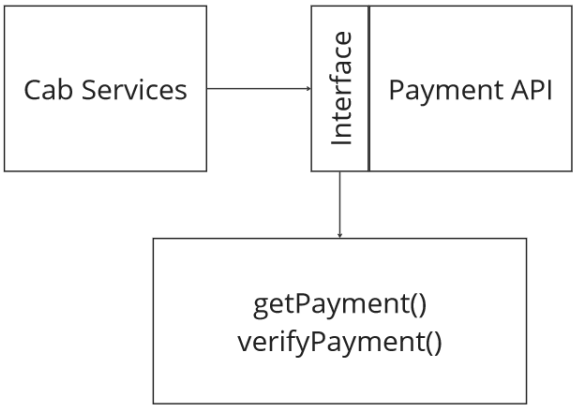
3.2.2 Cab Services to Google Maps API



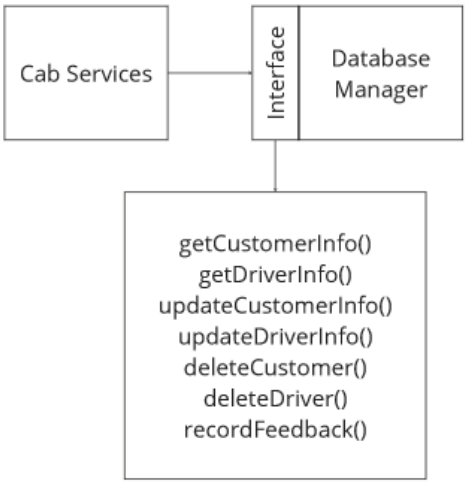
3.2.3 Cab Services to Customer Profile Interface



3.2.4 Cab Services to Payment API



3.2.5 Cab Services to Database Manager

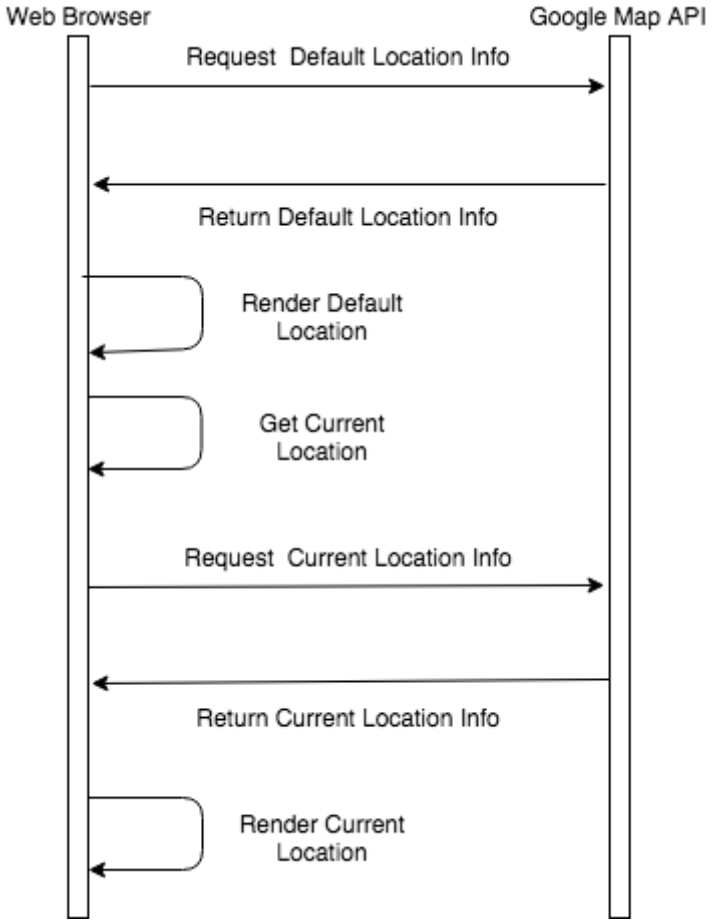


4. Detailed Design

4.1 Module Detailed Design

4.1.1 Mark Current Location

Sequence Diagrams



Pseudocode

Browser shows default location on google maps

- Request default location information with google's api
- Google returns default location

Uses browser to get current location

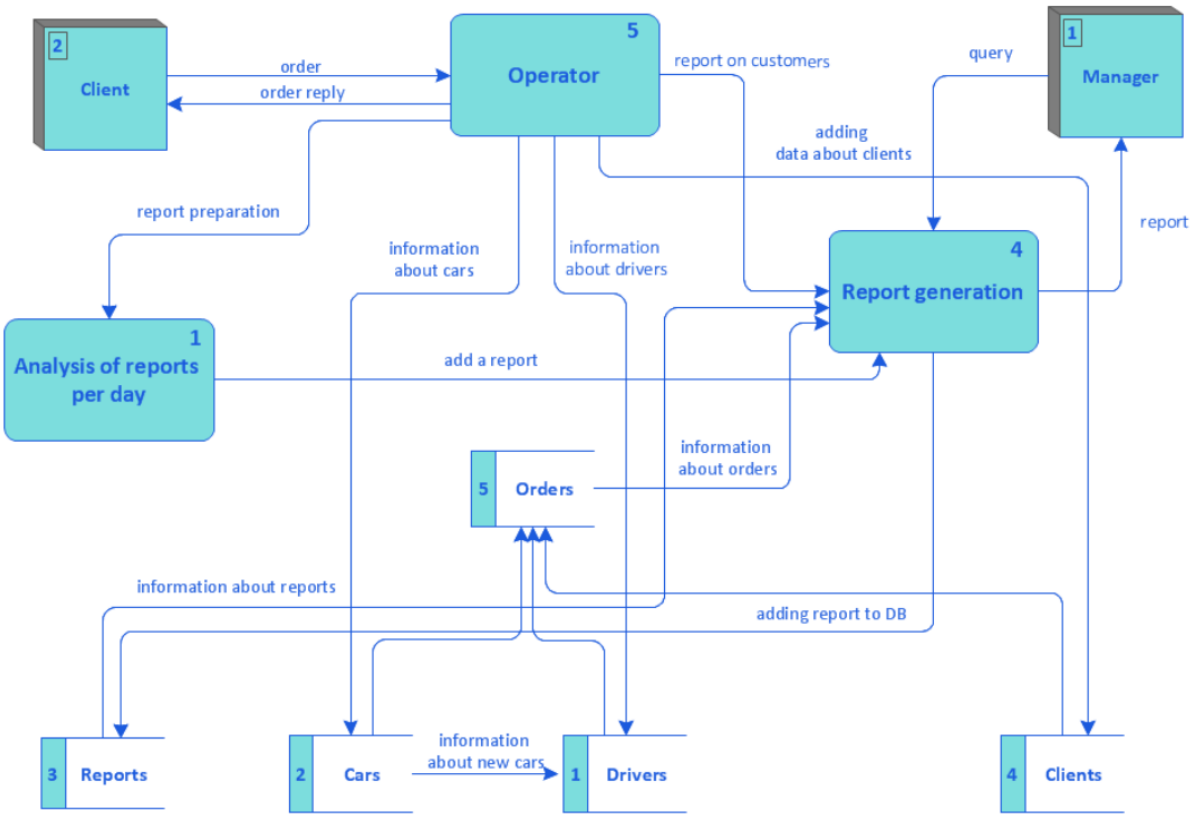
Request current location information from google api

Google's API returns current locations info

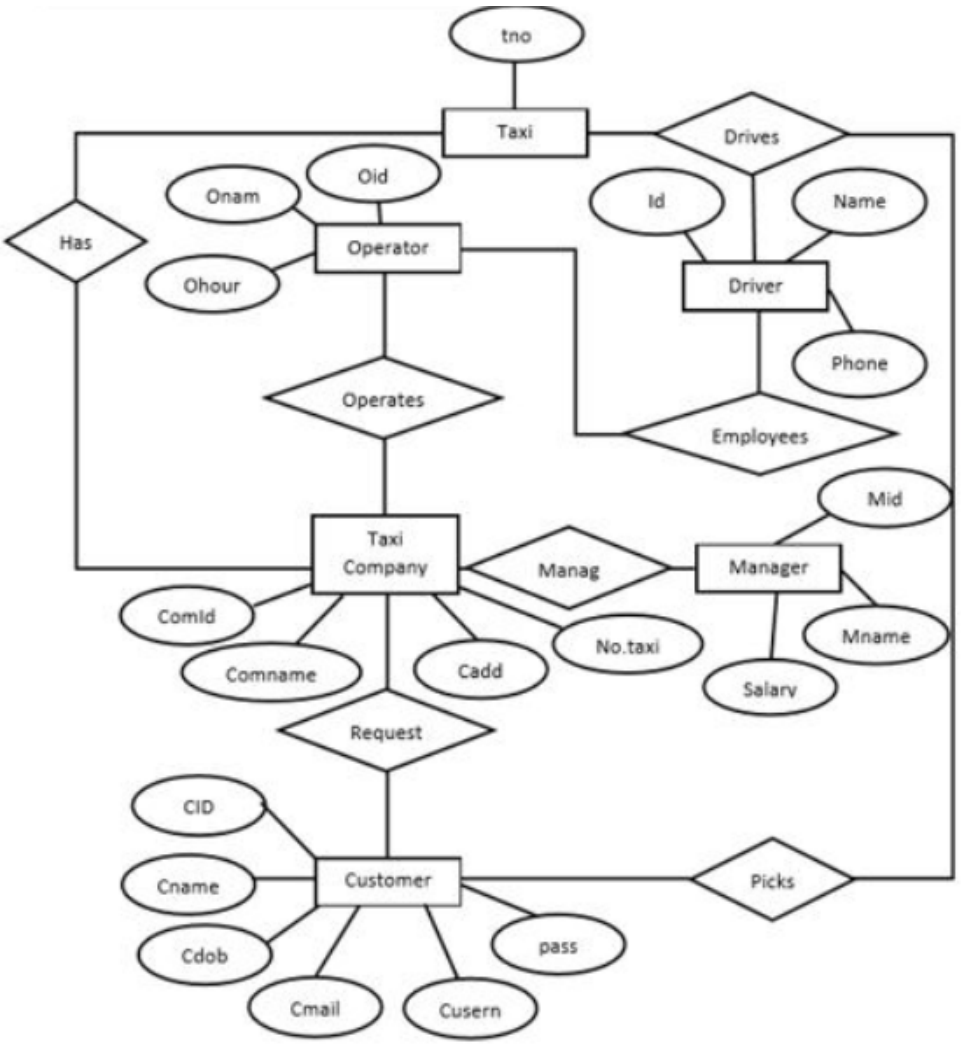
Browsers renders current location

4.2 Data Detailed Design

Data flow diagram:



ER Diagram:


tno
Taxi
Drives
Oid
Onam
Id
Name
Has
Operator
Ohour
Driver
Phone
Operates
Employees
Mid
Taxi
Company
Manag
Manager
ComId
No.taxi
Mname
Comname
Cadd
Salary
Request
CID
Cname
Customer
Picks
pass
Cdob
Cmail
Cusern