TAXI
BOOKING SYSTEM

Prepared by:
INFRAOPS TEAM 1

Submitted on:
29 APRIL 2024

Table of Content

Introduction

Project Overview

As the DigiLink InfraOps team we want to address the gap in the adoption of e-hailing technology within the long-distance travel sub-sector of the South African taxi industry. Despite the rise of platforms like Uber and Bolt, this sector has not fully embraced the convenience and efficiency offered by such technology. The project's goal is to introduce an online taxi booking platform that will benefit both passengers and taxi owners.

Problem Statement

The rise of e-hailing platforms such as Uber and Bolt has disrupted the taxi industry by leveraging technology to offer efficient and convenient ride-hailing services. Within the entire public transportation sector, trains and buses were able to adopt this method of booking tickets as it proved to be convenient for all stakeholders involved, especially their customers. However, even with 10 years of this technology's implementation in the South African market, it is still not fully adopted by the taxi industry. Taxis primarily transport many citizens, particularly low-income workers who need to get from townships to business centers and other parts of the city, where they most commonly work. Although most Uber users are middle-to-upper class due to the costs of fares, the are some instances where most citizens would prefer the use of the platform's service because of convenience and comfort. However, in this case the gap we want to address is within the long-distance travel sub-sector of the taxi industry, where utilizing such a system would have the best impact in terms of benefiting both passengers and taxi owners. The taxi owners would be able to earn more from the booking fees and passengers can have the convenience of not having to stand in long queues.

Purpose

This documentation is to specify the software requirements of the Online Taxi Booking platform. It will be a complete specification of what this system provides at both client and marshal end.

Project Vision

The project's aim is to introduce an automated system for passengers to book a seat for long distance taxis by means of a web app. The passenger will get to select the destination to which

they are going and preferred timeslot at which the taxi will be leaving then they will have the option of making a payment via a loyalty card issued by the taxi association or cash on arrival. This will ensure that taxis leave on time and are on schedule. The web app will collect and process information regarding the client to optimize quality of service. The web app will also collect and store data of registered users and taxis in the database to keep a travel log in case of an accident.

Project Goals

This project aligns with the digital transformation in the public transportation sector and aims to bring the benefits of e-hailing platforms to a wider demographic, particularly low-income workers who rely on taxis for their daily commute.

Our goals are as follows:

- Efficiency and convenience by developing a web app that will allow passengers to book seats for long-distance taxis, select their destination, and choose their preferred departure timeslot. This will eliminate the need for passengers to stand in long queues and ensure that taxis leave on time and according to schedule.
- Payment flexibility by providing passengers with the option to pay via a loyalty card issued by the taxi association or cash on arrival, offering greater flexibility and convenience.
- Quality of service because the web app will collect and process client information to optimize the quality of service provided.
- Safety and accountability by collecting and storing data of registered users and taxis in a database to maintain a travel log. This log will be crucial in the case of an accident, ensuring accountability and enhancing passenger safety.
- Economic benefit by earning more from booking fees, taxi owners stand to benefit economically from the implementation of this system.

Overall Description

Product Perspective

The taxi booking system is a mobile platform that will allow consumers to reserve a seat at any time and from any location. This will function similarly to a bus reservation platform, but with

various changes, such as a passenger being allowed to book only for one-way trips and the platform not allocating any seats. Not only will it benefit the customers with convenience, but the drivers will also earn an extra income as they will be charging a non-refundable booking fee. The taxi's takeoff time slots will be displayed by the system. The user will have the option to select the time that works best for them, but if that time slot is already taken, they will receive a notification informing them that the taxi is full

Product Features

- User-Friendly Booking Process
    - Select Destination
    - Select travel time slot
    - Payment for booking fee
- Real-time seat allocation/availability check (automated process)
- Secure Payment Options
    - Payment via cash
    - Pay via loyalty card
- Admin privileges for taxi marshal
    - Set and update booking fees
    - Add and delete destination taxis
    - Confirmation of bookings
    - Taxi Departure notifications

Operation Environment

The software is designed to be a cross-platform application. It requires a device with Android 10.0 and IOS 13 or higher with a stable internet connection and GPS access. It will be developed using C# using the Asp.NET framework utilizing REST APIs, SQL for databases and AngularJS for frontend.

Design and Implementation Restraints

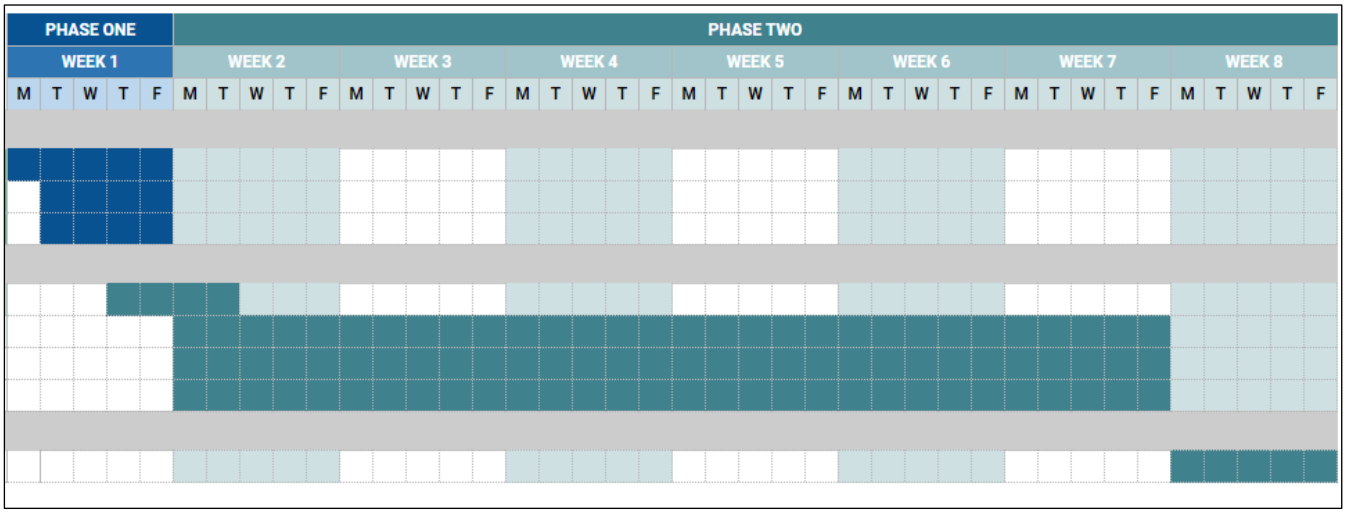
Unstable internet connection will interfere with the operation of the app.

Project Timeline

This is an 8-week project which must be completed by mid-June. It is divided into two phases which are documentation and development. We will employ the Agile methodology, which requires us to test every feature and provide feedback on areas where our system needs improvement before implementing the next feature.

8-week timeline is as follows:

- Documentation: 1week
    - Scope of work
    - Requirements
    - Analysis
- Development: 6 weeks
    - Database UML - 2 days
    - Design - 6 weeks
    - Development - 6 weeks
    - Testing - 6 weeks
- Final Testing and User Demo: 1 week



Project Methodology

As stated above, we will employ Agile methodology by implementing tasks in one-week sprint intervals. The reason we will be using this method is as follows:

- Customer Satisfaction - Agile prioritizes early and continuous delivery of software. By involving stakeholders throughout the development process, we will ensure that the product aligns with their needs and expectations.
- Flexibility – Being agile provides increased flexibility in terms of budget and deliverables. As our taxi booking system evolves, this will allow us to adapt to changing requirements and market conditions.
- Faster Delivery - We aim to deliver a working product within a shorter time frame of 1-week sprints. This delivery cycle will ensure that we can iterate and improve the product efficiently.
- Collaboration - We will be collaborating between business analysts and developers. Daily interactions and feedback help drive product improvements and enhance overall project success.
- Functional testing - Which will be done in each component and issues for that functionality will be solved immediately after testing.

Sprint Process Implementation

The sprint process will be key in our project management approach. Below is a more detailed step-by-step of how we will use it for developing, testing, deploying and maintaining our system.

1. Doing Sprint Planning Weekly: We will hold a planning meeting to decide which functionalities from the product backlog we will tackle in the upcoming sprint and the complexity of each functionality will determine the time allocated for its completion.
2. Having Daily Scrum Sessions: We will dedicate a specific number of hours to system development every day. This focused work period will be crucial for maintaining momentum and progress where all team members will be concentrating exclusively on the system, setting aside other assignments.
3. Having Daily Reviews: We will be conducting a review session as it will be an opportunity to assess our progress, identify any issues encountered, and brainstorm solutions. These daily reviews will ensure that we stay on track and address problems promptly, preventing them from escalating or hindering our progress.

4. Sprint Review: We will hold a sprint review to demonstrate the completed functionalities to stakeholders and gather feedback. This will allow us to adjust our plans and priorities based on stakeholder input and project realities.

This approach will help us deliver high-quality functionalities in a timely manner, while also maintaining the flexibility to adapt to changing requirements or circumstances. The sprint technique in agile project management is not just about speed, but also about maintaining a sustainable pace of development that ensures the overall quality and success of the project.

Roles and responsibility

Our roles and responsibilities as a team are follows:

- Malebo Moleleki, our project manager, will work on the system's front end.
- Simphiwe Mabuza and Vhutshilo Mavhandu will work on the system's backend.
- Testing will be handled by the entire team.

User Stories

- User story 1:
  As a customer I want to be able to select my destination from a list of suggested locations, so that I can easily and accurately specify where I want to go.
- User story 2:
  As a taxi marshal (admin), I want to be able to add, update and delete the list of suggested destinations, so that we can improve the accuracy and relevance of locations.
- User story 3:
  As a customer, I want to be able to select a specific time slot for my taxi booking, so that I can plan my journey and schedule my day more effectively.
- User story 4:
  As a taxi marshal (admin), I want to be able to add time slots and scheduling, so I can reduce waiting time.
- User story 5:
  As a customer, I want to be able to pay for my taxi ride using a secure and convenient payment method, so that I can easily complete my transaction.

- User story 6:

  As a taxi marshal (admin), I want to be able to add prices and confirm payments, so that I can monitor and optimize our business operations.
- User story 7:

  As a taxi marshal (admin), I want to be able to log in to the system with a unique username and password, so that I can securely administratively functions and manage our fleet.

Requirements

Functional requirements

- Authentication:
  - Secure signup and login process
- Booking process:
  - Schedule a ride in advance for a specific date and time slot.
  - Schedule a ride based on destination
  - Schedule a ride instantly
  - Provide the ability to save frequently visited destination
- Payments:
  - Secure payment methods
  - Ability to save payment method such as loyalty cards
  - Ability to book for cash payment.
- Landing Page:
  - Explain user manual guide for rider and a driver
- Dashboard:
  - Display booking details

Non-functional requirements

- Usability:
  - Simple user interface.
  - Easy navigation and clear menus.
  - Minimal steps for booking a ride.

- Performance:
    - Reliable and responsive.
- Security:
    - Encryption of user data and payment information.
- Scalability:
    - Ability to handle many users.
- Additional Requirements:
    - Promotions and discounts.
    - Booking cancellation.

Analysis

Cross-Platform Development

We have decided to create a cross-platform application. It is strategic because it will allow us to run the software as a web application. This will save the development team time and effort compared to building separate native apps for each platform.

Technology Stack

- *C# and ASP.NET* - We will be developing using C# as the primary programming language. The ASP.NET framework will be used for building the backend of the application. It provides tools and libraries for creating web services, handling requests, and managing data.
- *REST APIs* - We will be using REST APIs for communication between the frontend and backend. It is lightweight, scalable, and widely adopted for building web services.
- *SQL for Databases* - Relational databases will be best suited for managing structured data, such as passenger profiles, bookings, and other relevant information.
- AngularJS for Frontend - We will be building the front end of the web application with AngularJS, which is great at creating dynamic and responsive user interfaces.

Advantages of the Chosen Stack

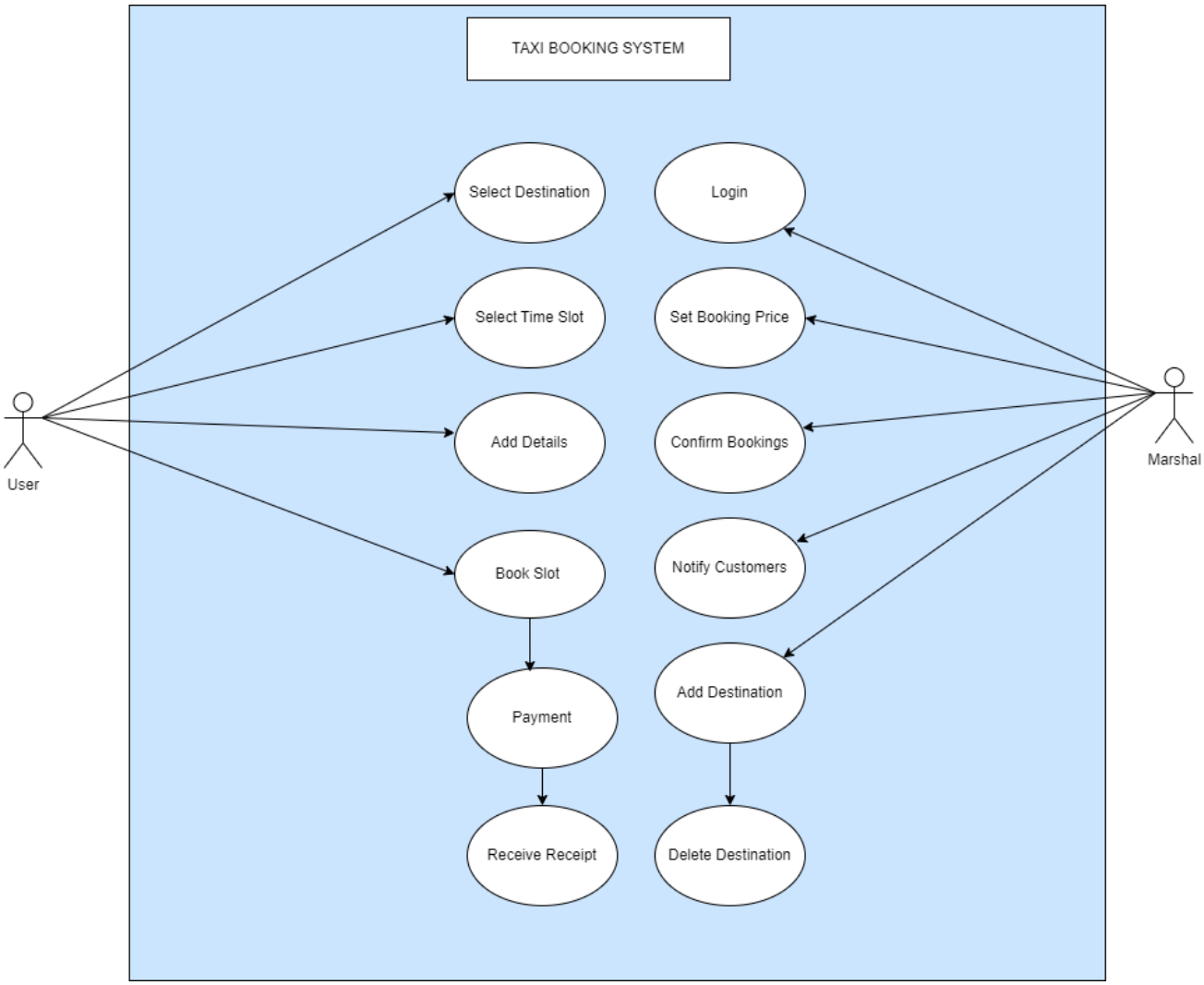
- Consistent User Experience

- Scalability
- Relational Data Management
- Responsive Frontend

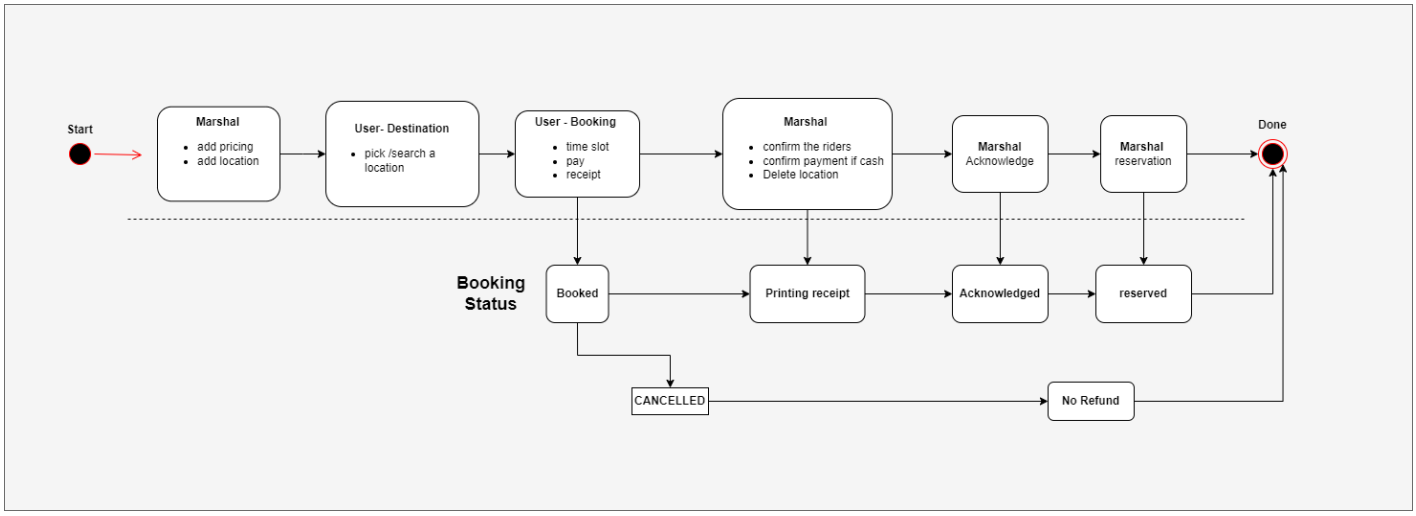
The chosen technology stack aligns well with our goal of building a platform that will be accessible to most smartphone users who will be utilizing the booking service, having a user-friendly experience for passengers and taxi marshals.

Diagrams

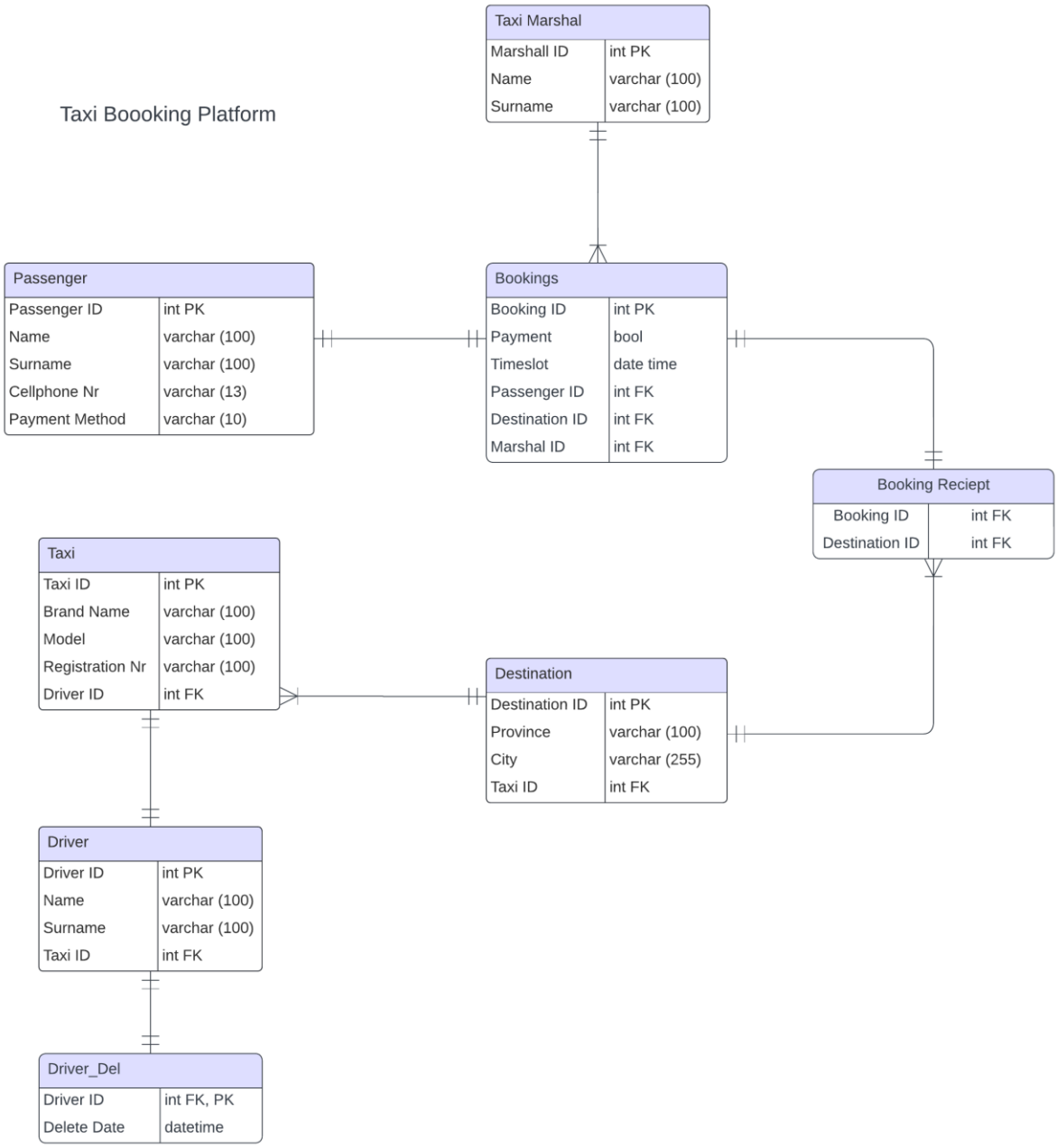
Use Case



State Chart Diagram



ERD Diagram


Taxi Boooking Platform
Taxi Marshal
Marshall ID int PK
Name varchar (100)
Surname varchar (100)
Passenger
Passenger ID int PK
Name varchar (100)
Surname varchar (100)
Cellphone Nr varchar (13)
Payment Method varchar (10)
Bookings
Booking ID int PK
Payment bool
Timeslot date time
Passenger ID int FK
Destination ID int FK
Marshal ID int FK
Booking Reciept
Booking ID int FK
Destination ID int FK
Taxi
Taxi ID int PK
Brand Name varchar (100)
Model varchar (100)
Registration Nr varchar (100)
Driver ID int FK
Destination
Destination ID int PK
Province varchar (100)
City varchar (255)
Taxi ID int FK
Driver
Driver ID int PK
Name varchar (100)
Surname varchar (100)
Taxi ID int FK
Driver_Del
Driver ID int FK, PK
Delete Date datetime