Software Requirements Document for Ride-Booking Feature of Uber

1. Introduction

This document presents a comprehensive description of the software requirements for the ride-booking feature of the Uber application. User wants to get a quick ride by Bike, CNG or Sedan whenever they will need it. This ride booking feature helps a user to select a ride according to his schedule, comforts, availability, affordability. This document provides a detailed account of the system's functionalities and specifies the requisite standards for development and implementation.

1.1 Purpose

The purpose of this Software Requirement Document (SRD) is to ensure the development of the ride-booking feature according to the specified requirements, providing a continuous and efficient experience for users wishing to book rides through the Uber app.

1.2 Scope

The scope surrounds the entire process of booking a ride, from selecting a destination to confirming the ride with a driver.

1.3 Definitions, Acronyms, and Abbreviations

- **ETA**: Estimated Time of Arrival
- **UI**: User Interface
- **UX**: User Experience

1.4 References

- Uber App Design Guidelines
- Google Maps API Documentation

2. Overall Description

2.1 Product Perspective

This feature is an integral component of the Uber application ecosystem, interfacing with the user's mobile device and the Uber backend system to facilitate ride booking.

2.2 Product Functions

- Selection of ride destination
- Choice of vehicle type
- Viewing of fare estimates
- Confirmation of ride booking
- Real-time updates on driver status

2.3 User Classes and Characteristics

- **Regular Users**: Users who book rides for personal transportation.
- **Business Users**: Users who book rides for professional purposes.
- **Uber Drivers**: Users who provide the transportation service.

2.4 Operating Environment

The ride-booking feature will function within the existing Uber application on both iOS and Android platforms.

2.5 Design and Implementation Constraints

- The application should be developed in Java for Android and Swift for iOS.
- The UI must be intuitive and accessible, conforming to accessibility standards.

3. Stakeholders

- **Passengers**: Individuals using the app to book rides.
- **Drivers**: Individuals providing transportation services to passengers.
- **Uber Operations Team**: Responsible for overseeing the platform's operation and ensuring system efficiency and reliability.
- **Software Developers and Engineers**: Tasked with the development, deployment, and maintenance of the ride booking feature.
- **Product Managers**: Focus on the feature's alignment with market needs and user satisfaction.
- **Legal and Compliance Officers**: Ensure that the feature complies with all relevant laws and regulations.
- **Customer Support Representatives**: Provide assistance and support to both passengers and drivers.
- **Security Specialists**: Safeguard the application against cyber threats and protect user data.

4. System Features

4.1 Ride Booking Process

4.1.1 Home Page

Description: Users begin the ride-booking process from the home page.

Functional Requirements:

- **Functional Requirement 1**: The app shall display a list of ride options.
- **Functional Requirement 2**: Promotional content must be displayed without obstructing ride selection.

4.1.2 Destination Setting

Description: Users must be able to set a destination for their ride.

Functional Requirements:

Functional Requirement 3: Users shall be able to input a destination manually.

Functional Requirement 4: The app shall offer a list of recent destinations.

4.1.3 Ride Option Selection

Description: Users select from various ride options available.

Functional Requirements:

Functional Requirement 5: The system shall display all available vehicle types (i.e. Uber Moto, UberX, Uber XL, Uber CNG) along with their estimated fares.

Functional Requirement 6: The fare estimates shall be calculated based on current location, destination, and traffic data.

4.1.4 Ride Confirmation

Description: Users confirm their ride and receive details about their driver.

Functional Requirements:

Functional Requirement 7: Upon confirmation, the app shall display the driver's details, including name, vehicle type, and ETA.

Functional Requirement 8: The system shall allow users to cancel a booking within a specified time frame without a penalty.

5. External Interface Requirements

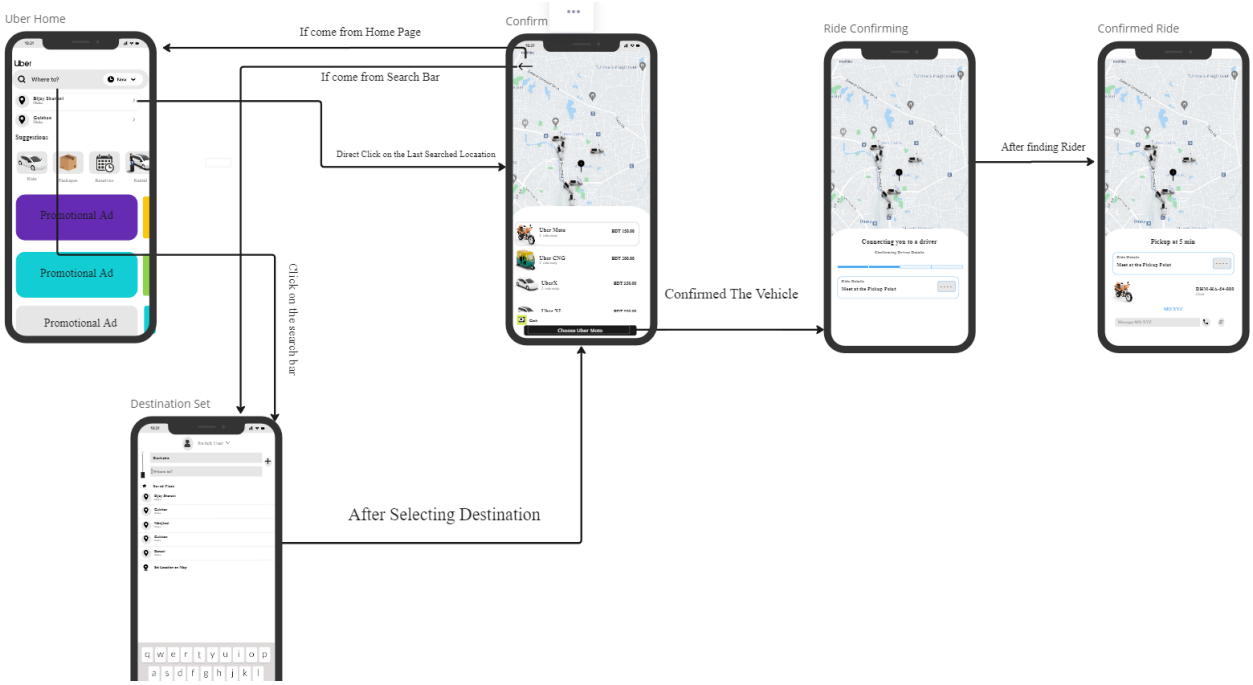
5.1 User Interfaces

The feature's UI must be consistent with the existing Uber design language and provide a responsive experience across various device resolutions.

5.2 Hardware Interfaces

The application will interface with the device's GPS module for location services and the network module for data transmission.

5.3 Mock-up:



Mock-up of Ride Booking Feature

For better quality please Visit: [Uber Ride Share Mok-up]

5.4 Software Interfaces

- The application will integrate with Google Maps for mapping services.
- The payment processing system for handling transactions.

6. Other Non-Functional Requirements

6.1 Performance Requirements

- The feature must support up to 1 million concurrent users.
- The system should process a ride request within 5 seconds 99% of the time.

6.2 Security Requirements

The application must comply with GDPR and other relevant data protection regulations.

7. Documentation and Help

7.1 User Documentation

A user manual shall be provided within the app, including FAQs and troubleshooting guides.

7.2 Testing Documentation

A detailed test plan will be created, covering all functional and non-functional requirements.