**Software Requirements Specification**

Inspection and Evidence control system

Prepared by: Asha Patel (15IT073)

Meet Patel (16MCA012)

Tapan Sheth (15CE)

Charotar University of Science and technology,

CSPIT, IT department

09-02-2018

**Table of Contents**

Table of Contents 1

1. Introduction 2

1.1 Purpose 2

1.2 Document Conventions 2

1.3 Intended Audience and Reading Suggestions 2

1.4 Product Scope 2

1.5 References 2

2. Overall Description 4

2.1 Product Perspective 4

2.2 Product Functions 4

2.3 User Classes and Characteristics 4

2.4 Operating Environment 4

2.5 Assumptions and Dependencies 4

3. Externam Interface Requirements 5

3.1 User Interfaces 5

3.2 Hardware Interfaces 6

3.3 Software Interfaces 6

3.4 Communications Interfaces 6

4. System Features 7

4.1 System Feature 1 7

5. Other Nonfunctional Requirements 8

5.1 Performance Requirements 8

5.2 Safety Requirements 8

5.3 Security Requirements 8

5.4 Software Quality Attributes 8

6.

Appendix A: Design Documentation 9

**CHAPTER:1**

**Introduction**

* This Software Requirements Specification (SRS) provides the overall general and detailed software requirements for the Inspection and evidence control system.
* This section provides an introduction to the INSPECTION AND EVIDENCE CONTROL SYSTEM project and the purpose, scope, and other details require fulfilling customers need.
  1. **Purpose**
* Inspection and evidence system specializing in management and controlling the proper flow of documents and task from point of origin to point of distribution
* Main purpose of this system is to eliminate the paper work to handle all documents. Management of proper service flow with less time and money by automating manual processes with the help of different technologies.
* The proposed system will feature different modules that will ensure the proper flow of documents and related task to all those who are concerned with a job.
* A system that would be able to manage users with authentication, their documents, verification of team members, forms for task, flow of entities from source to destination with proper validation, assets that are required to perform different task etc.
  1. **Document Conventions**

The following document conventions are followed in preparing this SRS.

* All key-words related to the academics are formatted in italics.
* SRS-Software Requirement Specification
* API-Application Programming Interface
* AWS-Amazon Web Service
  1. **Intended Audience and Reading Suggestions**
* This document is created for,
  + The Instructors of the course “Software Engineering‟ for their review and monitoring progress of the project.
  + The software development team for their use in analyzing the requirements.
  1. **Product Scope**

First user will have to do sign up to use the system. After proper verification of user’s detail such as email verification, contact number verification, One will be able to use the proposed system.

Different Module wise functionality is given below:

* **DASHBOARD**

It is provided to the users to view summary like structure of activities.

This sub module will display following details.

* + Number of jobs created by user.
  + Total added location
  + Number of reports generated by user
  + Space/Storage occupy by the documents stored by the user.
  + Information about jobs such as name, author, status (Pending, In progress, Closed).
* **PROFILE**

User information will be display in this sub modules.

It includes following details.

User Name

Email

Phone

Profile photo

Delete profile, Edit profile functionality

Change password functionality.

* **USER DOCUMENTS**

If any documents are uploaded by user, then the details such as job name in which documents are uploaded, description, total size, last modified by, and updated time and date would be display in this sub module.

* **JOB**
* Whenever this type of system will be introduced to user, it is necessary to have one most important feature that is able to manage or prepare single Job/Task.

Different working functionalities are given below.

* + **Create job, Edit job, Delete job, Share job**
* After implementation of the solution, user will be able to create Job with job details (Title , Description, location, overview, conclusion).
* In overview and conclusion tab, user will be able to use an existing forms or can create form as per the requirements.
* System will provide dynamic form building functionality to satisfy their needs. Different attachment options would be there to attach images or documents in different format.
* Create job functionality will also allow to add task that has to be done, agents who will manage task, assets require to complete job etc.
* User will be able to share job with others who are responsible to actively participate in job.
  + **Report Generation.**
* After job is created successfully, user can generate report as per his format as there would be different formatting options available mentioned below.
  + - Basic options:
      * Print quality (Grayscale, Colorful)
      * Image options (All images, Select Images)
      * Layout options (Full, Compact)
      * Thumbnail size (Normal, 2x)
      * Additional details (Full size media, Print attachments, Download links )
    - Advanced options:
      * Choose sections (Check In, Check List, Images, Audio , Video, Job, Attachment, Email details, Conclusion history)
      * Grid line options (Horizontal, Vertical)
      * Table header options (Left-to-right, Top-to-bottom)
* Mail will be sent to user by which he can easily download report.
* **DATA IMPORT**
* **INSTACOUNT**
* **FORMS**
* **ASSETS**
* **CONTACTS**
* **CAB SEARCH**
* User can search cab for a particular location here.
* User required to enter Source, Destination, and Place where he want to go.
* **LOGIN PROCESS:**
* In this the customer has to give out the login details i.e. user id/email and password and then only he can be logged on.
* The user id/email and password given by the customer are checked from the data stored in the database.
* **REGISTRATION PROCESS:**
* User must be registered before booking a cab.
* Proper validations will be provided to keep only authenticated users i.e. those users who will provide correct information. All the data supplied by the user will be stored in database and it will be used for further validation and authentication.
* During registration, user has to give login and password of their choice.
* Login names and passwords will be stored in the database so that the user can directly login without registering again and again.
  1. **References**
* <https://www.spaceotechnologies.com/taxi-booking-app-outstation/>
* <https://www.spaceotechnologies.com/uber-clone-script/>
* https://www.tutorialspoint.com/php/

**CHAPTER:2**

**Overall Description**

**2.1 Product Perspective**

* The web pages are present to provide the user interface on customer client side. Communication between customer and server is provided through HTTP/HTTPS protocols.
* The Client Software is to provide the user interface on system user client side and for this TCP/IP protocols are used.
* Database server is for storing the information.
* To provide online cab without any information about area or perfect location.

**2.2 Product Functions**

* Registration
* Login
* Online Booking Car
* Services
* Payment

**2.3 User Classes and Characteristics**

* The client should have the basic idea to operate (use) the system and he already has the experience to work in the internet (browser). Default Language is English.
* Some of the users identified for this system through use case analysis are listed below:

User

Driver

**2.4 Operating Environment**

1. Hardware Requirement:

* Phone with internet access.

1. Software Requirement:

* Android version: 5.0.2
* User should have an application to use cab booking system.

**2.5 Assumptions and Dependencies**

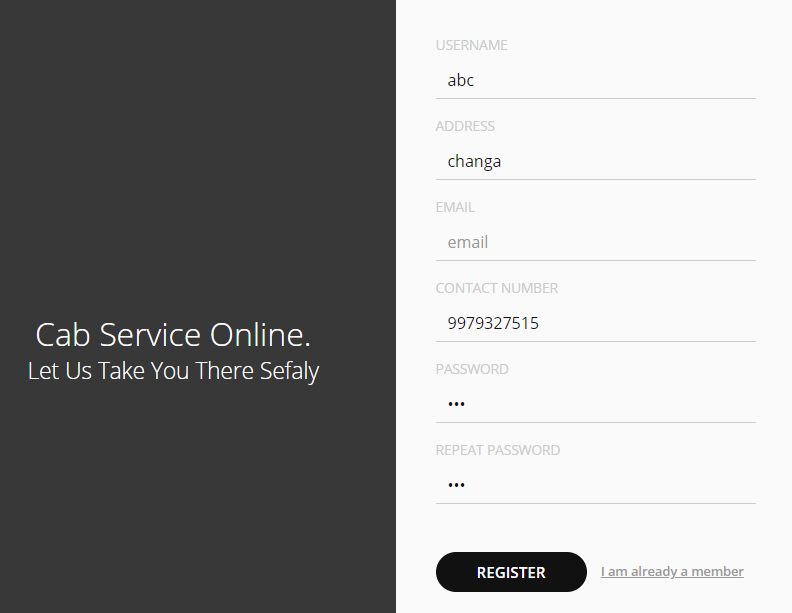
* Vehicles are already purchased and available for use.
* Roles and responsibilities are already established.
* Administrator is already created.

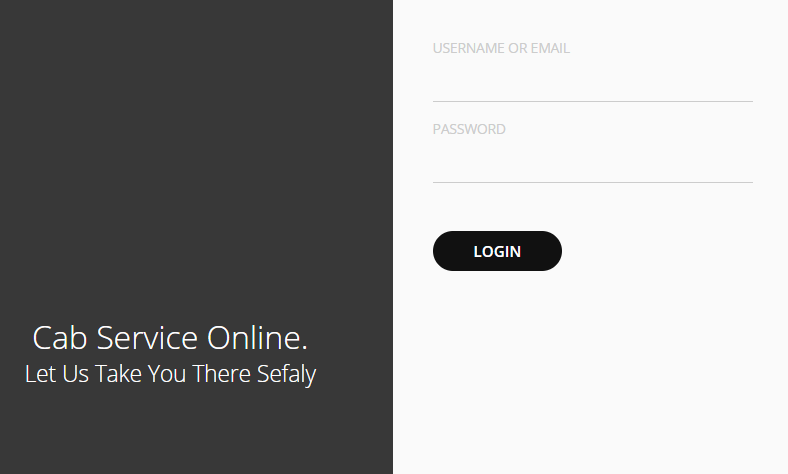
**CHAPTER:3**

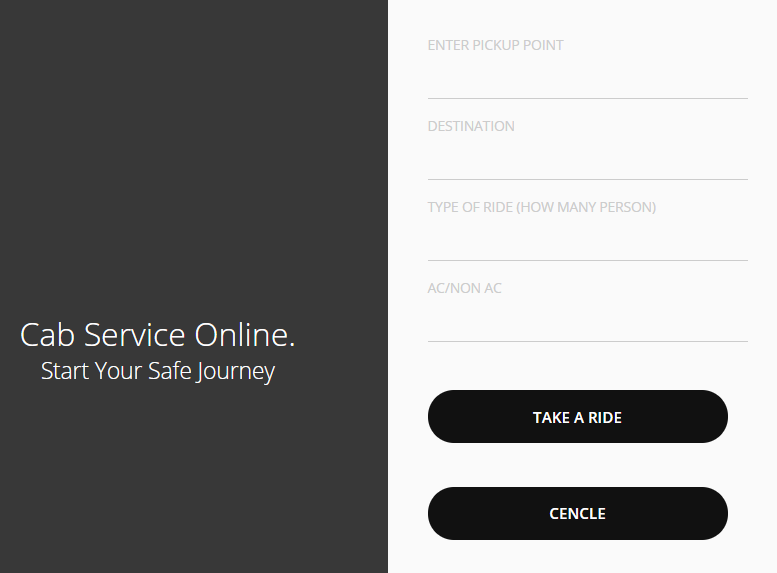
External Interface Requirement

**3.1 User Interfaces**

**GUI design:**







**3.2 Hardware Interfaces**

* Processor: - Intel core i3
* Phone with internet access.
* RAM: - 4GB
* Hard Disk: -20 GB or above.

**3.3 Software Interfaces**

* Android version: 5.0.2
* Front End: HTML,PHP,CSS
* Back end: MySQL

**3.4 Communication Interfaces**

* The communication interface used by Cab booking is an email service for contacting customers, GPS modules for reaching to the customer and PHP for back-end communication.

**CHAPTER:4**

System Features

**4.1 System Feature**

Some of the features are identified for the software. They are listed below:

* **View Available Vehicles:**

The client must able to see all details about the available vehicles without any constraints.

* **Calculate Fare:**

The client must be available to check the fare they should pay for the vehicles.

* **Feedback:**

The administrator can able to see the feedback given by each client so that he can take appropriate actions for future improvement.

* **Report Generation:**

The system supports generation of reports based on different criteria.

* **Record maintenance:**

The system also must keep track the statistical reports of daily activities of the online booking.

* **Discount Offer:**

The admin can create discount codes and the client can get discount on fares using the codes.

**CHAPTER:5**

Other Nonfunctional Requirements

**5.1 Performance Requirements**

* The program or application should run at such a speed that the user can move at his/her own space, without noticing interruption due to processing.

**5.2 Safety Requirements**

* This software is intended to communicate over an internal network , therefore security is of little concern.
* The user will have to enter the username and password so the program can connect to the database server.
* The username and password will not be stored because encryption of such information is outside the scope of the project.

**5.3 Security Requirements**

Some of the factors that are identified to protect the software from accidental or malicious access, use, modification, destruction, or disclosure are described below.

* The system, at any time, should be accessed only by the authenticated users.
* Network communications should use cryptographic protocols such as SSL.
* Automated responses should be restricted using CAPTCHA.
* The system is required to end the session automatically, when an open session is not used for a specific period of time.
* Utilize certain cryptographic techniques
* Keep specific log or history data sets
* Assign certain functions to different modules
* Restrict communications between some areas of the program
* Check data integrity for critical variables

**5.4 Software Quality Attributes**

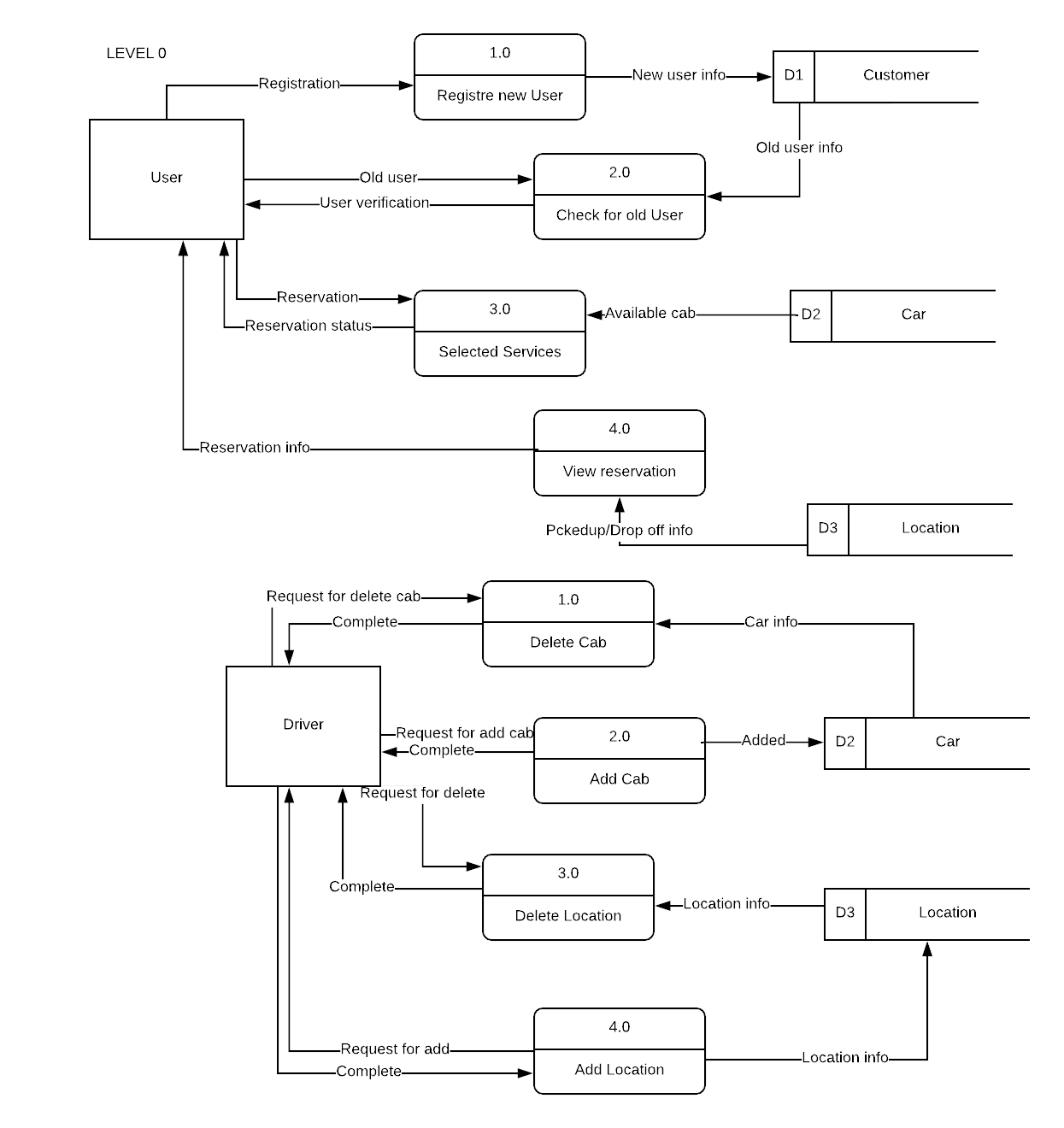
* This application is an application which is portable as it is an android application and it is in the smart phones so which is portable and the application can be used with or integrated with other systems also so it is reusable.
* There is no maintenance for this application, the application is durable and reliable as it stores its data on the cloud in the database.

**CHAPTER:6**

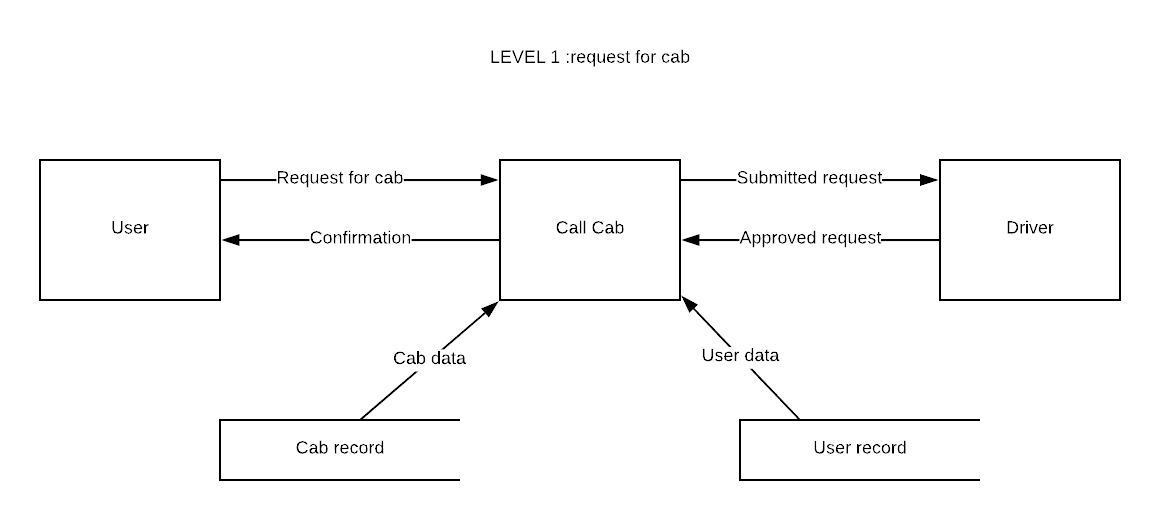
**Appendix A: Design Documentation**

1. **Data Flow Diagram**

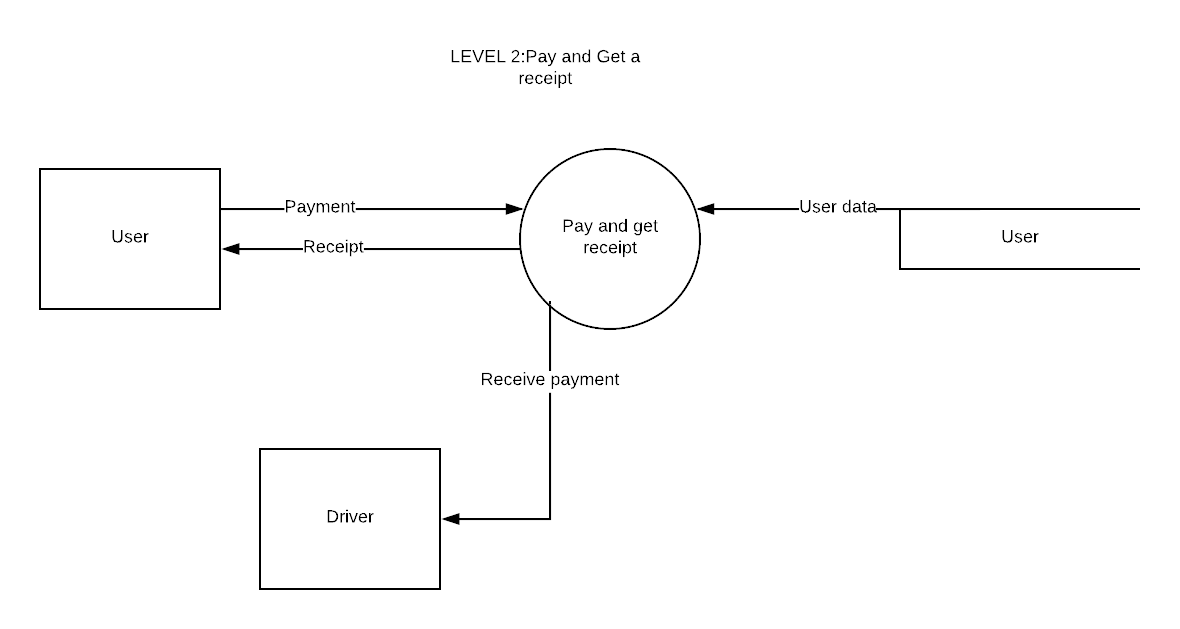
**Level 0:**

****

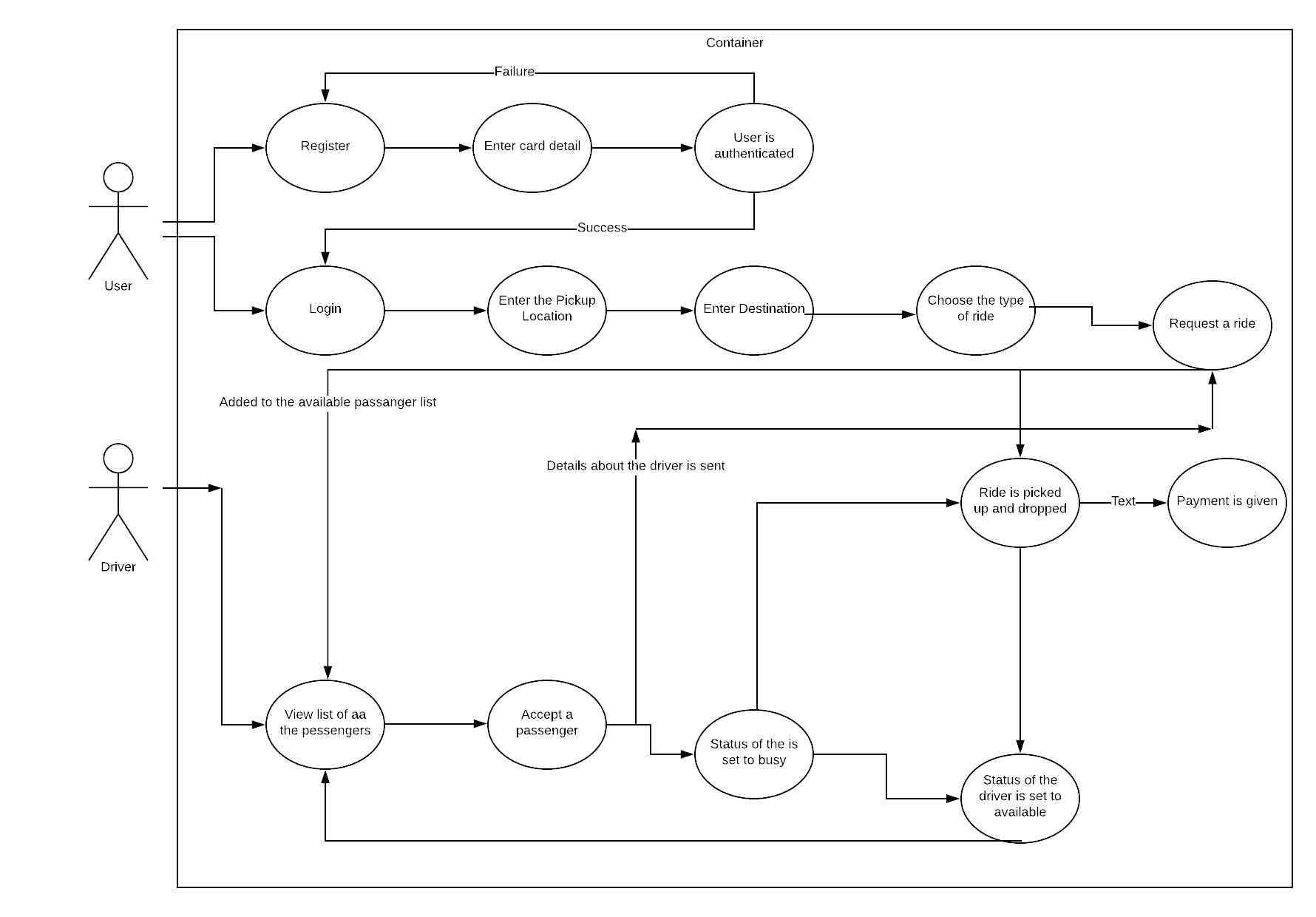
**Level 1:Request for cab**

****

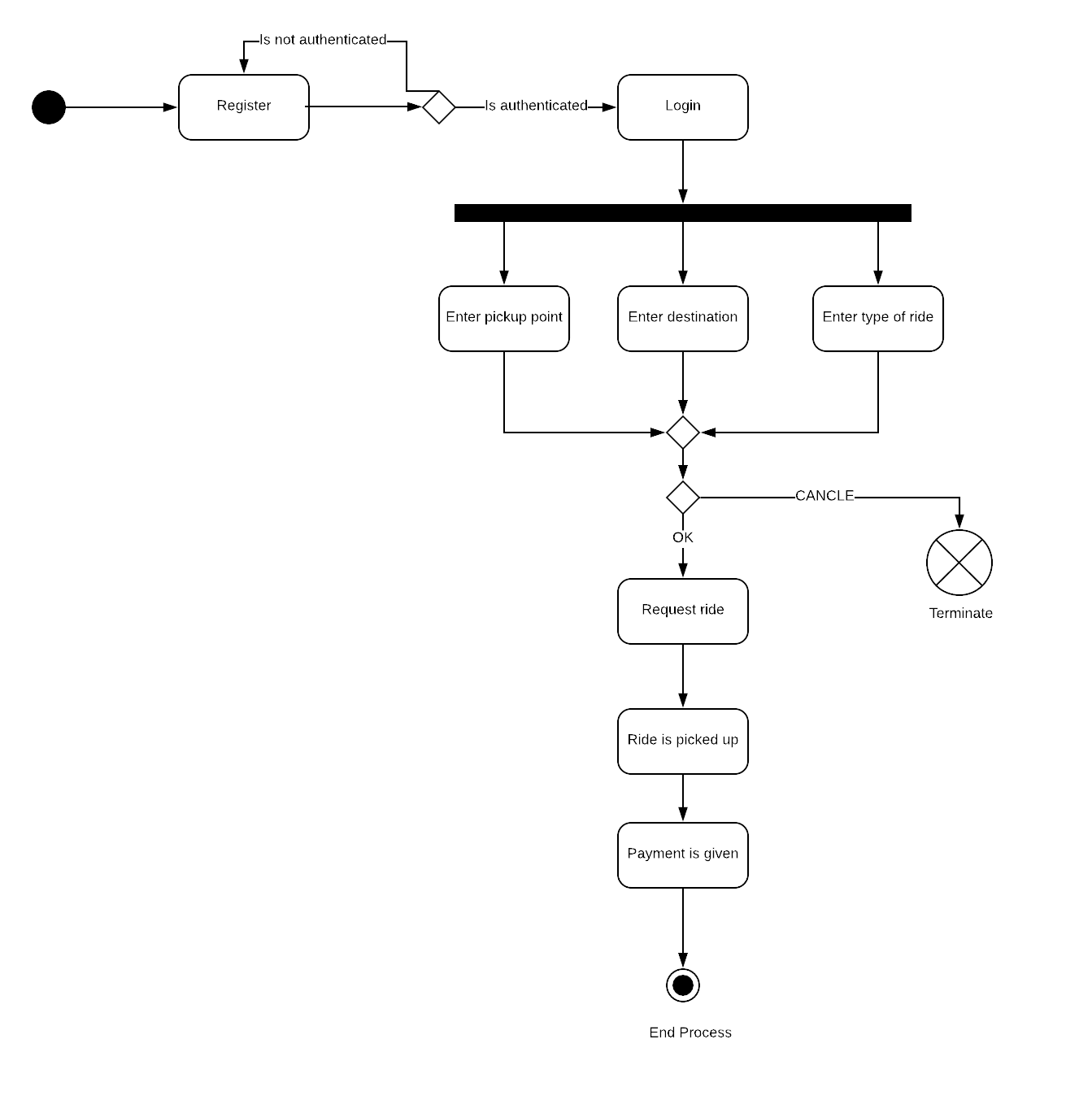
**Level 2:Pay and Get recepit**

****

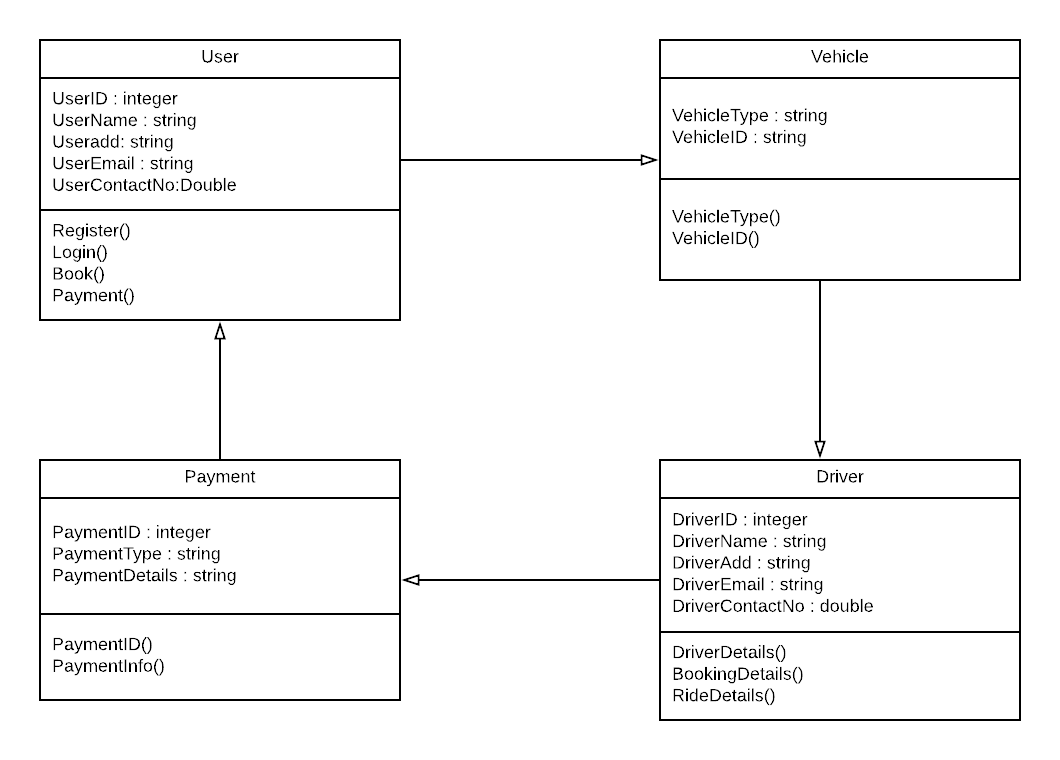
1. **Use Case Diagram:**

****

1. **Activity Diagram:**

****

1. **Class Diagram:**

****