### GENBA SOPANRAO MOZE COLLEGE OF ENGINEERING BALEWADI, PUNE-411045.



**DEPARTMENT OF INFORMATION TECHNOLOGY**

### PROJECT REPORT ON

**“ANDROID APPLICATION FOR SMART PARKING SYSTEM”**

Submitted in the partial fulfilment of the requirement for BE in Information Technology in 2022-23

**Submitted by**

**Name: PRN:**

**Abhilash Kare 72163818L**

**Akash Dighe 72024843E**

**Swaraj Mirajkar 72163823G**

**Shubham Patil 72163830K**

**UNDER THE GUIDANCE OF**

**Prof. Sana Shaikh**

### GENBA SOPANRAO MOZE COLLEGE OF ENGINEERING

### BALEWADI, PUNE-411045.



**DEPARTMENT OF INFORMATION TECHNOLOGY**

### CERTIFICATE

**This is to certify that the Project entitled**

**“ANDROID APPLICATION FOR SMART PARKING SYSTEM”**

**Submitted by**

**Name: PRN:**

**Abhilash Kare 72163818L**

**Akash Dighe 72024843E**

**Swaraj Mirajkar 72163823G**

**Shubham Patil 72163830K**

It is a benefited work carried out by them under the guidance of **Prof. Sana Shaikh** and is approved for the partial fulfilment of the requirement of SPPU for the award of BE in Information Technology Engineering.

Date:…./…../

|  |  |  |
| --- | --- | --- |
| **Prof. Sana Shaikh** | **Prof. Sana Shaikh** | **Dr. Ratnarajkumar Jambi** |
| **(Project Guide )**  **Internal Examiner** | **(H.O.D)** | **(Principal)**  **External Examiner** |

**ACKNOWLEDGMENT**

This is a great pleasure & immense satisfaction to express our deepest sense of Gratitude & thanks to everyone who have directly or indirectly helped in Completing Project successfully. It gives us great pleasure in presenting the project report on:

**“ANDROID APPLICATION FOR SMART PARKING SYSTEM”**

We would like to take this opportunity to thank our guide **Prof. Sana Shaikh** for giving all the help and guidance that we needed. We are really grateful to them for her kind support. Her valuable suggestions were very helpful.We would also like to express our gratitude towards our parents for their kind co-operation and encouragement which helped us in completion of this project and We also appreciate the guidance given by our honorable Principal **Dr. Ratnarajkumar Jambi** sir & our beloved HOD, **Prof. Sana Shaikh**.

Last but not the least, many thanks to the Project Co-ordinator, **Prof. Kaveri Kari** who has invested her full effort in guiding the team for achieving the goal. We are also grateful to her dispensable support and suggestions.

Regards,

Abhilash Kare

Akash Dighe

Swaraj Mirajkar

Shubham Patil

**ABSTRACT**

Now-a-days, vehicle parking has become a major problem in urban areas with the shortage of parking spaces. it is very difficult and frustrating to find a parking space in most metropolitan areas, especially during the rush hours to solve this problem. The paper entitled smart parking system using android application, the major motivation of this paper is to reduce the traffic congestion in roads, multistoried buildings and malls due to unavailability of parking spaces. The proposed application provides an easy way for reservation of parking slot. In this application user can view various parking areas and also view whether space is available or not. If the booking space is available then he can book it for specific time slot. The paper displays the nearest empty slot if present with respect to user location. Our project aims to make efficient use of parking spaces. Also, this system provides an additional feature for user. To alleviate the parking problems, smart parking systems must be implemented. In this paper, the background on parking problems is introduced and relevant algorithms, systems, and techniques behind the smart parking are reviewed and discussed. This system gives a further feature of cancelling the bookings. User can cancel their booked area anytime. Users may even make price online primarily based totally at the time taken for the reserved area the quantity might be calculated and the person can make charge. This paper provides a good insight into the guidance, monitoring and reservations components of the smart parking and directions to the future development. A cloud-based smart parking application will enable real-time monitoring and booking of parking availability by providing enhanced services to the end users as well as reduce the workload of the parking administrator.

**INDEX**

**Acknowledgement**

Abstract

Table of contents

List of figures

List of tables

1. Introduction 1

[1.1 Overview………………………………………………………………... 2](#_TOC_250011)

[1.1.1 Motivation……………………………………………………… 2](#_TOC_250010)

[1.1.2 Objective](#_bookmark1)……………………………………………………….. 3

1. Literature Survey 4

[2.1 Study Of Research Paper………………………………………………... 4](#_TOC_250009)

* + 1. Paper 1 ..………………………………………………………. 4
    2. Paper 2…………………………………………………………. 5
    3. Paper 3………………………………………………………… 6

1. Requirement Specifications 7
   1. [Introduction……………………………………………………………. 7](#_TOC_250008)
      1. [User Classes Characteristics](#_bookmark2)………………………………… 7
      2. [Assumptions And Dependencies](#_bookmark3)……………………………… 8
   2. [External Interface Requirement..……………………………………… 9](#_TOC_250007)
      1. [User Interface](#_bookmark4)…………………………………………………. 9
      2. [Hardware Interfaces](#_bookmark5)…………………………………………... 9
      3. [Software Interfaces](#_bookmark6)…………………………………………… 9
   3. [Functional Requirement…………………………………………….… 10](#_TOC_250006)
   4. [Non-Functional Requirement………………………………………….. 11](#_TOC_250006)
   5. [System Requirements………………………………………………….. 12](#_TOC_250005)
      1. [Database Requirements](#_bookmark9)………………………………………. 12
      2. [Software Requirements……………………………………….. 12](#_TOC_250004)
      3. [Hardware Requirements………………………………………. 13](#_TOC_250003)
      4. [Analysis Models: SDLC Model to be applied……………… 13](#_TOC_250002)
      5. [System Implementation Plan](#_bookmark10)…………………………………. 15
2. Proposed system 16
   1. [System Architecture](#_bookmark11)…………………………………………………… 16
      1. [Module………………………………………………………… 16](#_TOC_250001)
      2. [Data Flow Diagram……………………………………………. 19](#_TOC_250000)
      3. UML Diagrams………………………………………………... 21
3. Project Implementation 26
   1. Overview Of Project Modules……………………………………...... 26
   2. Tools And Technologies Used……………………………………… 28

6 Software Testing 32

6.1 Types Of Testing………………………………………………………… 32

6.1.1 Unit Testing…………………………………………………….... 32

6.1.2 Regression Testing………………………………………………... 32

6.1.3 Smoke Testing……………………………………………………... 33

6.1.4 System Testing…………………………………………………… 33

7 Test Cases 34

1. Results And Screenshot 38
2. Conclusion 44

9.1 Conclusion………………………………………………………… 44

* 1. [Future scope](#_bookmark13)…………………………………………………………. 44

10 References 45

**LIST OF FIGURES**

* 1. [System Architecture](#_bookmark12) 20
  2. Data Flow Diagram 22
  3. Use Case Diagram 24
  4. Activity Diagram 25
  5. Sequence Diagram 26
  6. Class Diagram 27

8.1 Main UI……………………………………………………………...... 43

8.2 Login Page ……………………………………………………………. 43

8.3 Registration Page………………………………………………………. 44

8.4 Booking Time………………………………………………………….. 45

8.5 Parking Reservation………………………………………..…………... 45

8.6 Car Owner Interface……………………………………………………..46

8.7 Parking Owner Inteface………………………..……………………….. 46

8.8 Payment Gateway………………………………………………………..46

8.9 Parking History…………………………………………………………..47

**LIST OF TABLE**

3.1 System Implement………………………………………………….. 22

7.1 GUI Testing…………………………………………………………. 41

7.2 Registration Test Case……………………………………………… 39

7.3 Login Test Case …………………………………………………….. 40

**CHAPTER 1: INTRODUCTION**

# 

# Overview

The numbers of personal vehicles usage is increasing day by day. Due to this searching for a vacant parking area during peak hours is not only time-consuming but also results in wastage of fuel. The drivers keep searching for a proper parking lot that leads to increased traffic. Increasing volume of vehicular exhaust creates a negative impact on the environment. Hence reservation-based smart parking has become the need of the day.

At this time, most existing parking lots do not have a system in place. Most of them are managed by hand and are a bit ineffective. Every user's demand should be I. Should be more efficient ii. Users friendly iii. They should provide more security. The idea behind our Android Application- “valid spot” is to help the user for online parking booking. The Smart Parking Application aims at helping users to find the most suitable area for parking, make reservations and extend them, if required. In this application user can view various parking areas also he can select it to view whether parking slot is available or not. If the parking slot is available in parking, then user can book it for some specific time slot also, this system provides an additional feature of cancelling the bookings. It also utilizes the open ground for parking with security. Thus, it is going to solve the parking and traffic problem. In this case, it is not necessary to use an extra expensive camera and scanner for verification. The smart parking system based on slot reservation is implemented, utilizing the Android application. The app having the features of slot allocation, by using the slot allocation method, user can reserve their own lowest-cost parking slot. It is an effective way in resolving the parking issues, which helps for traffic congestions and also provide the automated payment billing process. This work gets extended as a fully automated system using multilayer parking method. "Smart Parking" system in their portable devices.

## Motivation

**`**

If the parking space is not available where customer wants to a park then this application helps the user to find the valid parking space nearby that location. This application helps the user to reserve the parking spot when they stay at home form not being frustrated of finding a parking spot. The interface is easy to use and allows the user to easily switch between other apps. The Android mobile operating system is used, which can be found on many mobile phones today.

## Objective

In Urban Areas :- user to easily switch between other apps. The Android mobile operating system is used, which can be found on many mobile phones today. The Android operating system uses a marketplace to sell applications for the phone. These places run with heavy load, hence the task of maintaining the data and record of the vehicle exits and entries becomes hard. We can overcome this problem with the help of technology which can help us in making this easier and help in reducing the issues with running a vehicle parking system.

In Rural Areas :- In rural areas there are lot of events which happen where many people from different areas come to visit those events and there are chances people not be able to get proper parking spots and during these times people with private spaces available can rent them to earn some money and use these simple to use apps to help manage those place for some days without needing an computer systems or somebody to get them trained as mobile apps are easy to use and not hard to learn.

In Private Places: - If someone has a private place where there is a lot of incoming and outgoing of vehicles it is important to have proper maintenance of the place and data for security purposes. Due to high load of the parking place using a system for the same is very useful in order to not get into trouble in any case on criminal offense as the owner might get into trouble

**CHAPTER 2: LITERATURE SURVEY**

* 1. **Study Of Research Paper**

1. **Paper Name:** Android application for Smart Parking System

# Author: Pranjali D. Jambhulkar, Sejal R. Thaware

**Abstract:** Now-a-days, vehicle parking has become a major problem in urban areas with the shortage of parking spaces. it is very difficult and frustrating to find a parking space in most metropolitan areas, especially during the rush hours to solve this problem. The paper entitled smart parking system using android application, the major motivation of this paper is to reduce the traffic congestion in roads, multistoried buildings and malls due to unavailability of parking spaces. The proposed application provides an easy way for reservation of parking slot. In this application user can view various parking areas and also view whether space is available or not. If the booking space is available then he can book it for specific time slot. The paper displays the nearest empty slot if present with respect to user location.

1. **Paper Name :** Android-based Booking Application for Smart Parking

System.

**Author:** [Rosalyn R Porle](https://ieeexplore.ieee.org/author/37428652800); [Nursyafiqah Nabilah Mohd Saiful](https://ieeexplore.ieee.org/author/37089211374)

**Abstract:** The Android parking slot booking application enables end users (drivers) to save time and money while avoiding parking lot congestion. Most of the parking lots operate on a manual parking system, in which vehicles enter the lot, obtain a ticket from a machine at the entrance gate, look for an available slot, pay for parking at a counter, and insert the paid ticket into a machine at the exit gate. These actions can be carried out using a mobile application. Numerous studies have analysed the methodologies employed in smart parking systems; however, the majority of research has not addressed mobile applications in detail. The purpose of this paper is to demonstrate how to construct a mobile application using Android Studio. The end user must register and log in to this programme, select a vacant slot to reserve, specify the period of his or her parking, and make a payment.

**3. Paper Name:** Android Based Smart Car Parking System

**Author:** Harshala A. Waghmare, Aishwarya S. Tole , Swati R. Shilimkar

**Abstract:**-The purpose of this system is to computerize the parking space reservation. Its talk about undertaking which introduces a miniature model of car parking that can direct and manage the number of cars that can be parked in given space at any given time based on availability of parking space after doing the registration by user using android application on his smart phone. Automated parking is a strategy for parking and leaving cars utilizing detecting device i.e., sensors. The entering to or leaving from parking lot is also commanded by an android based application. This provide users to book parking spaces online in advance for given location and then park the vehicle with minimal fees. We have concentrated on some current systems and it shows that the current systems are not totally automated and require a certain level of human interference and communication with the system. The difference between our system and existing systems is that we intend to make our system as less human independent by automating whole parking area.

**CHAPTER 3: REQUIREMENTS SPECIFICATION**

* 1. **Introduction**

The purpose of this document is to provide a detailed specification of the Smart Parking System, outlining the functional and non-functional requirements of the system. The Smart Parking System is designed to efficiently manage and optimize the parking process by providing real-time information to drivers about available parking spaces and facilitating parking spot reservations.

## 3.1.1 User Classes Characteristics

This application use in government level to detect the criminal and to overcome the crime graph in India or in world. This project can be used for surveillance in public places. Through the visual surveillance, human activities can be monitored in sensitive and public areas such as bus stations, railway stations, airports, banks, shopping malls, school and colleges, parking lots, roads, etc. to prevent terrorism, theft, accidents and illegal parking, vandalism, fighting, chain snatching, crime andother suspicious activities.

## Assumptions And Dependencies

Assumptions and dependencies for an Android application for a smart parking system can vary depending on the specific implementation and requirements. However, here are some common assumptions and dependencies to consider:

**1. Assumptions:**

- Availability of a reliable network connection for real-time communication with the parking system server.

- Users have compatible Android devices with the required minimum specifications.

- GPS functionality is available and accurate for location tracking.

- Parking lots are equipped with the necessary hardware, such as sensors or cameras, to detect parking space availability.

- The parking system has a centralized server or cloud infrastructure to store and process parking data.

- Adequate security measures are in place to protect user data and prevent unauthorized access to the system.

**2. Dependencies:**

- Integration with a backend server or cloud-based infrastructure to store and retrieve parking data, manage user authentication, and handle transaction processing.

- Access to real-time parking space availability information from sensors or cameras installed in parking lots.

- Integration with payment gateways or systems for handling parking fee transactions.

- Utilization of location-based services and GPS functionality for accurate positioning and navigation to available parking spaces.

- Integration with external APIs or services for additional features like traffic data, navigation, or third-party integrations.

# 3.2 EXTERNAL INTERFACE REQUIREMENT

## 3.2.1 User Interface:

* Application Based On Android application for smart parking system.

## 3.2.2 Hardware Interfaces:

* Hardware : intel i5
* Speed : 2.80 GHz
* RAM : 8GB
* Hard Disk : 500 GB
* Key Board: Standard Windows Keyboard

## 3.2.3 Software Interfaces:

* Operating System: Windows 10
* Programming Language : Android

# 3.3 FUNCTIONAL REQUIREMENT

**3.1 User Registration and Authentication**

The system shall allow users to register and create accounts. Users shall be able to authenticate themselves to access the system.

**3.2 Parking Space Management**

The system shall provide an administrative interface for managing parking spaces. Parking spaces should be categorized and labeled for easy identification. The system shall allow adding, modifying, and removing parking spaces.

**3.3 Real-time Parking Space Availability**

The system shall integrate with parking sensors to provide real-time parking space availability information. Users shall be able to view the availability of parking spaces in a specific area.

**3.4 Parking Spot Reservation**

The system shall allow users to reserve parking spots in advance. Users shall be able to specify the desired parking duration and time. Once a parking spot is reserved, it should be marked as unavailable for other users.

**3.5 Navigation and Directions**

The system shall provide navigation and directions to available parking spots. Users should be able to view the shortest or most convenient route to their reserved parking spot.

**3.6 Payment Integration**

The system shall integrate with payment gateways to facilitate parking fee payments. Users shall be able to make payments for their parking reservations through the system.

# 3.4 NON-FUNCTIONAL REQUIREMENTS

**4.1 Performance**

The system should respond to user requests within an acceptable time frame. The system should handle multiple concurrent users without significant performance degradation.

**4.2 Security**

User authentication and data transmission should be secured using encryption protocols. The system should enforce access control to ensure that only authorized users can perform administrative tasks.

**4.3 Reliability**

The system should be highly available and minimize downtime. Data backups and disaster recovery mechanisms should be in place to prevent data loss.

**4.4 Usability**

The user interface should be intuitive and easy to use. The system should provide clear and concise instructions to users for parking spot reservations and navigation.

**4.5 Scalability**

The system should be designed to accommodate a growing number of parking spaces and users. The architecture should support horizontal scaling to handle increased traffic and data load.

**4.6 Compatibility**

The system should be compatible with popular web browsers and mobile devices. APIs should be available to allow

# SYSTEM REQUIREMENTS

## 3.4.1 Database Requirements

Firebase is a comprehensive mobile and web development platform provided by Google. It offers a wide range of tools and services that help developers build high-quality and scalable applications with ease. Firebase provides a real-time database, authentication services, cloud storage, hosting, cloud functions, and many other features that simplify the development process and enable seamless integration with other Google services.

## 3.4.2 Software Requirements

**Android Studio –**

Android Studio is the official Integrated Development Environment (IDE) for Android app development, provided by Google. It is built on the popular IntelliJ IDEA platform, offering a rich set of features specifically designed for Android development. With a user-friendly interface, Android Studio simplifies the process of creating, debugging, and testing Android applications. Its powerful code editor comes equipped with intelligent code completion, syntax highlighting, and code analysis tools, enhancing developer productivity. Developers have the flexibility to choose between programming languages like Java and Kotlin, as Android Studio supports both. The IDE includes a comprehensive set of tools for designing user interfaces, allowing developers to create visually appealing and interactive layouts. Testing apps is made easy with the advanced emulator, enabling developers to test their apps on virtual devices with different configurations and Android versions. Android Studio seamlessly integrates with other popular development tools, such as the Android Software Development Kit (SDK), Gradle build system, and Firebase services. The IDE also offers extensive debugging capabilities, aiding in identifying and fixing issues quickly through features like breakpoints and real-time performance monitoring.

## Hardware Requirements

## 

## 3.4.4 Analysis Models: SDLC Model to be applied

The software development cycle is combination of Different phases such as designing, implementing and deploying the project. These different phases of the software development model are described in this section. The sdlc model for the project development can be understood using the following figure the chosen sdlc model is the waterfall model which is easy to follow and fits bests for the implementation of this project.

**Requirements analysis**: at this stage, the business requirements, definitions of use cases are studied and respective documentations are generated.

**Design:** in this stage, the designs of the data models will be defined and different data preparation and analysis will be carried out.

**Implementation:** the actual development of the model will be carried out in this stage. Based on the data model designs and requirements from previous stages, appropriate algorithms, mathematical models and design patterns will be used to develop the agent’s back-end and front-end components.

**Testing:** the developed model based on the previous stages will be tested in this stage. Various validation tests will be carried out over the trained model.

**Deployment:** after the model is validated for its accuracy scores its ready to be deployed or used in simulated scenarios.

**Maintenance**: During the use of the developed solution various inputs/scenarios will been countered by the model which might affect the models overall accuracy. Or with passing time the model might not fit the new business requirements. Thus, the model must be maintained often to keep its desired state of operation.

## System Implementation Plan

The system implementation plan table, shows the overall schedule of tasks compilation and time duration required for each task.

|  |  |  |  |
| --- | --- | --- | --- |
| **Sr.**  **No.** | **Name/title** | **Start date** | **End date** |
| 1 | Preliminary survey | 12/08/2022 | 18/08/2022 |
| 2 | Introduction AND problem Statement | 20/08/2022 | 20/08/2022 |
| 3 | Literature survey | 07/09/2022 | 13/09/2022 |
| 4 | Project statement | 14/09/2022 | 14/09/2022 |
| 5 | Software requirement and  Specification | 15/09/2022 | 16/09/2022 |
| 6 | System design | 04/10/2022 | 08/10/2022 |
| 7 | Partial report submission | 12/10/2022 | 15/10/2022 |
| 8 | Architecture design | 08/11/2022 | 24/11/2022 |
| 9 | Implementation | 18/11/2022 | 29/12/2022 |
| 10 | Deployment | 14/01/2023 | 01/02/2023 |
| 11 | Testing | 13/03/2023 | 26/03/2023 |
| 12 | Paper Publish | 12/05/2023 | 16/05/2023 |
| 13 | Report submission |  |  |

**CHA****PTER 4: PROPOSED SYSTEM**

* 1. **System Architecture**

A picture containing text, diagram, screenshot, line

Description automatically generated

**Fig 4.1: System Architecture**

## 4.1.1 Module

## Client Side:

## Start the application: The person desires to put in the application on his Android primarily based totally device. After installation, the icon of the app will function at the Home Screen of the person’s device. App welcome display might be flashed to the person on beginning the software.

## 2. Registration: Initially, the person has to sign in his information with the application for the primary time. This is a one-time registration. The person has to go into information like person name, gender, phone number and email- id. All these records can be saved on server. Booking for slots mandatory must be carried out a day before to arrival. On server aspect the parking owner additionally needs to sign in the number of parking slots available and for what sort of cars and the amount that needs to be paid.

## 3. Login: Once the user registers, he can use his email id and phone number to login in future. This authenticates the user.

## 4. Selection of location for parking: The user is provided with multiple parking locations. User has to select one of the locations provided where he desires to park the vehicle

## 5. Select vehicle type: After selecting the location, options for the vehicle type are provided i.e., 2-wheeler or 4- wheeler alongside the rate chart for parking charges is prompted.

## 6. Availability: Status of the slots based on the type of vehicle selected availability of the empty slots will be displayed along with the total slots reserved for that vehicle type. Color coding is used to indicate empty v/s reserved slots. Grey indicates empty slots and Red indicates that currently there are no empty slots for reservation.

## 7. Payment: On availability of empty slot, the consumer can verify his reserving of his preferred slot. After booking a specific slot, the use can continue to the charge alternative in any other case terminate the complete process. The system requires full payment in advance /Hence, the person needs to provide all his card info to book his preferred slot. After a success payment he gets a slot number, each to his cellular and mail. After usage of a specific slot, he can circulate out of the parking place by clearing his payment.

## 8. Confirmation to user: On a success of reservation, a confirmation page with user information is proven that's editable and Green is indicated to show user's reserved parking slot.

## 9. Parking Dashboard: Parking dashboard gives more efficient distribution of parking slots and with the aid of using the use of this

## dashboard the parking owner can control their parking slots. the allocation and de-allocation of slot is done by parking owner. The dashboard additionally suggests reserved slots which may be allotted while the corresponding user verifies his information and confirms the selected slot.

## Server Side:

## Initially the administrator logins the application through the use of his username and password. The administrator has authority to feature new customers and stores their info in the database which might be used for in addition purpose. On receiving the request at server aspect through user, the administrator indicates all to be had locations on the nearest requested destination.

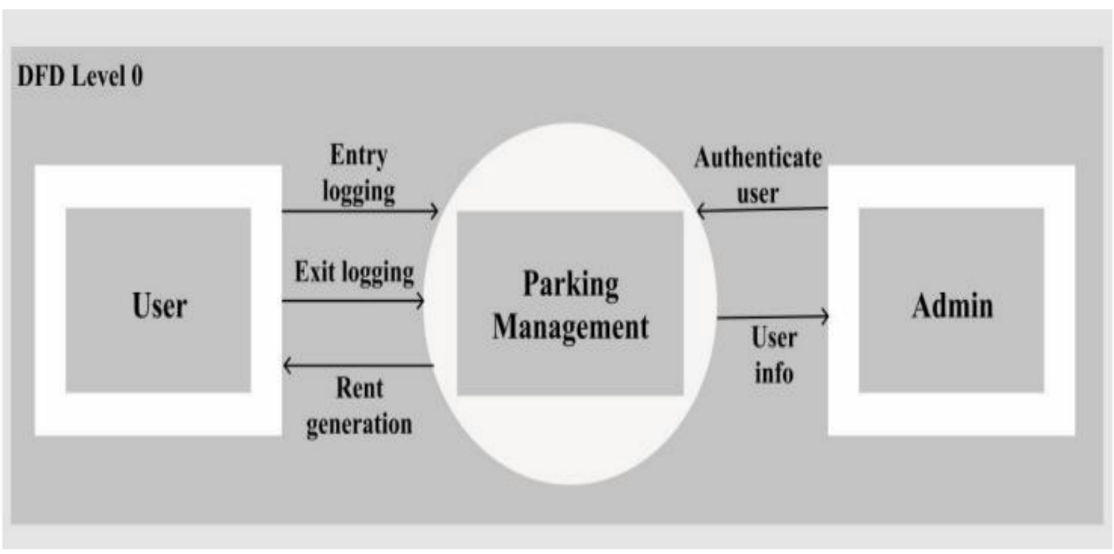
## 1. Login: The administrator can login to the application through giving e-mail and password. If the administrator receives successfully login, then the administrator is stated to be authorized. After getting login to the application the administrator can perform many responsibilities such as: - Adding Parking Locations - View Parking Locations - View All Users - View All Bookings.

## 2. Add and view Parking Locations: The administrator can add distinct locations in which parking slots are available. The person can pick out any area that is nearest to his destination. The administrator also can delete the places if he wishes. The administrator can view distinct places in which parking slots are available and also can take a look at the status of various parking slots. View All Users and Respective.

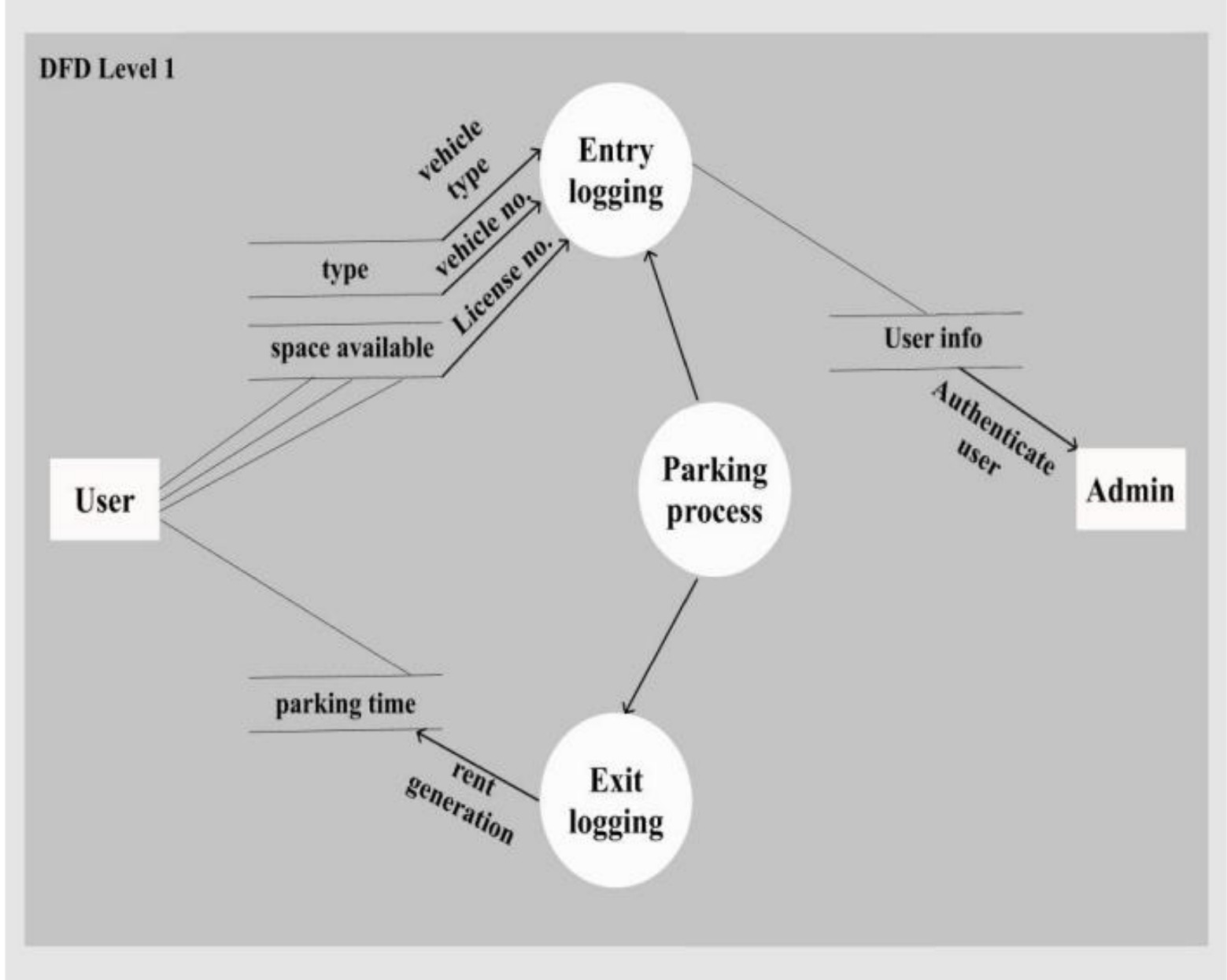
## 3. Booked Slots: The administrator can view all of the users who're the use of the application and also can test the reserving information consisting of the time and date at which the person requires a slot, number of hours a person is the use of the allotted slot, at which area he requires a slot etc., The administrator can view all of the reserved slots of all registered users. The administrator takes this as a reference for similarly allocation.

## Data Flow Diagram

Data Flow Diagram,we Show that flow of data in our system in DFD0 we show that base DFD in which rectangle present input as well as output and circle show our system,In DFD1 we show actual input and actual output of system input of our system is text or image and output is rumor detected like wise in DFD of user as well as admin



**Fig:4.2 data flow diagram Level 0**



**Fig:4.2 data flow diagram Level 1**

## UML diagrams

Unified Modeling Language is a standard language for writing software blueprints. The uml may be used to visualize, specify, construct and document the artifacts of a soft-ware intensive system is process independent, although optimally it should be used in process that is use case driven, architecture-centric, iterative, and incremental. The Number of UML Diagram is available.

Use Case Diagram

Activity Diagram

Sequence Diagram

Class Diagram

A picture containing diagram, sketch, line, drawing

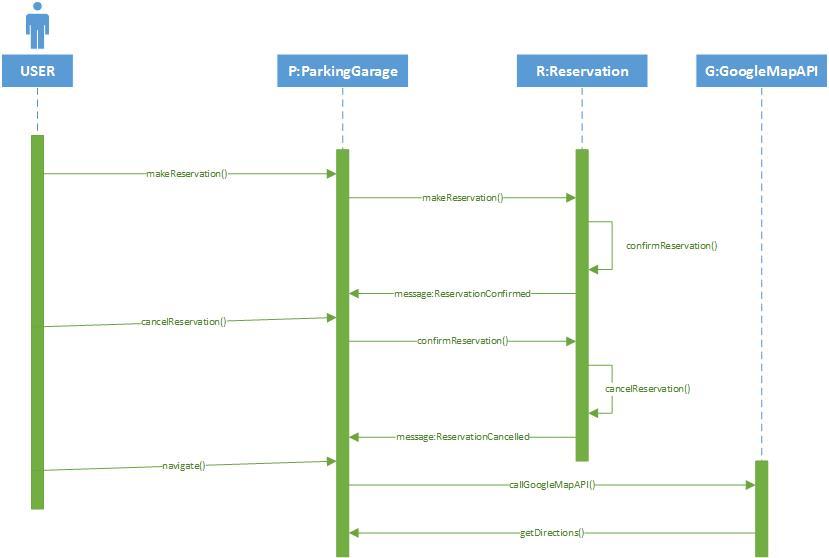
Description automatically generated

**Fig 4.3 Use Case Diagram**

A picture containing text, screenshot, diagram, line

Description automatically generated

**Fig 4.4 Activity Diagram**



**Fig 4.5 Sequence Diagram**

A picture containing text, diagram, screenshot, line

Description automatically generated

**Fig 4.6 Class Diagram**

**CHAPTER 5: PROJECT IMPLEMENTATION**

**5.1 Overview of Project Modules**

## Client Side:

## 1. Start the application: The person desires to put in the application on his Android primarily based totally device. After installation, the icon of the app will function at the Home Screen of the person’s device. App welcome display might be flashed to the person on beginning the software.

## 2. Registration: Initially, the person has to sign in his information with the application for the primary time. This is a one-time registration. The person has to go into information like person name, gender, phone number and email- id. All these records can be saved on server. Booking for slots mandatory must be carried out a day before to arrival. On server aspect the parking owner additionally needs to sign in the number of parking slots available and for what sort of cars and the amount that needs to be paid.

## 3. Login: Once the user registers, he can use his email id and phone number to login in future. This authenticates the user.

## 4. Selection of location for parking: The user is provided with multiple parking locations. User has to select one of the locations provided where he desires to park the vehicle

## 5. Select vehicle type: After selecting the location, options for the vehicle type are provided i.e., 2-wheeler or 4- wheeler alongside the rate chart for parking charges is prompted.

## 

## 6. Availability: Status of the slots based on the type of vehicle selected availability of the empty slots will be displayed along with the total slots reserved for that vehicle type. Color coding is used to indicate empty v/s reserved slots. Grey indicates empty slots and Red indicates that currently there are no empty slots for reservation.

## 7. Payment: On availability of empty slot, the consumer can verify his reserving of his preferred slot. After booking a specific slot, the use can continue to the charge alternative in any other case terminate the complete process. The system requires full payment in advance /Hence, the person needs to provide all his card info to book his preferred slot. After a success payment he gets a slot number, each to his cellular and mail. After usage of a specific slot, he can circulate out of the parking place by clearing his payment.

## 8. Confirmation to user: On a success of reservation, a confirmation page with user information is proven that's editable and Green is indicated to show user's reserved parking slot.

## 9. Parking Dashboard: Parking dashboard gives more efficient distribution of parking slots and with the aid of using the use of this dashboard the parking owner can control their parking slots. the allocation and de-allocation of slot is done by parking owner. The dashboard additionally suggests reserved slots which may be allotted while the corresponding user verifies his information and confirms the selected slot.

## Server Side:

## Initially the administrator logins the application through the use of his username and password. The administrator has authority to feature new customers and stores their info in the database which might be used for in addition purpose. On receiving the request at server aspect through user, the administrator indicates all to be had locations on the nearest requested destination.

## 1. Login: The administrator can login to the application through giving e-mail and password. If the administrator receives successfully login, then the administrator is stated to be authorized. After getting login to the application the administrator can perform many responsibilities such as: - Adding Parking Locations - View Parking Locations - View All Users - View All Bookings - Users Feedback

## 2. Add and view Parking Locations: The administrator can add distinct locations in which parking slots are available. The person can pick out any area that is nearest to his destination. The administrator also can delete the places if he wishes. The administrator can view distinct places in which parking slots are available and also can take a look at the status of various parking slots. View All Users and Respective.

## 3. Booked Slots: The administrator can view all of the users who're the use of the application and also can test the reserving information consisting of the time and date at which the person requires a slot, number of hours a person is the use of the allotted slot, at which area he requires a slot etc., The administrator can view all of the reserved slots of all registered users. The administrator takes this as a reference for similarly allocation.

**5.2 Tools And Technologies Used**

Android is an open-source mobile operating system developed by Google. Since its initial release in 2008, Android has experienced remarkable growth and has become the most widely used mobile operating system globally. With a significant market share, Android has captured the attention of users and manufacturers alike. Its versatility and compatibility have enabled it to power a vast range of devices, including smartphones, tablets, smart TVs, and wearables.

Android follows a layered architecture, with the Linux kernel forming the core. Above the kernel, there are native libraries, the application framework, and the user interface layer, all working together to provide a seamless user experience.

One of Android's standout features is its extensive app ecosystem. The Google Play Store hosts millions of apps covering diverse categories, offering users a wide selection of applications to cater to their needs and interests. Developers benefit from the robust Android development tools, with Android Studio being the official Integrated Development Environment (IDE) for Android app development. Supporting multiple programming languages, such as Java and Kotlin, Android Studio provides a comprehensive set of features to streamline the development process.

Android offers a range of platform features that enhance the functionality and usability of applications. These include notifications, background services, connectivity options, location services, and support for various hardware capabilities. Security and privacy are paramount in the Android ecosystem, with app sandboxing, permission-based model, and regular security updates to protect users' data and privacy.

The integration of Android with Google services further enhances its capabilities. Seamless integration with Google Maps, Google Drive, Google Pay, and Google Assistant provides users with a seamless experience across devices. Developers can leverage powerful APIs and tools provided by Google to create innovative and feature-rich applications.

**Android Studio:** Android Studio is the official Integrated Development Environment (IDE) for Android app development, provided by Google. It is built on the popular IntelliJ IDEA platform, offering a rich set of features specifically designed for Android development. With a user-friendly interface, Android Studio simplifies the process of creating, debugging, and testing Android applications. Its powerful code editor comes equipped with intelligent code completion, syntax highlighting, and code analysis tools, enhancing developer productivity. Developers have the flexibility to choose between programming languages like Java and Kotlin, as Android Studio supports both. The IDE includes a comprehensive set of tools for designing user interfaces, allowing developers to create visually appealing and interactive layouts. Testing apps is made easy with the advanced emulator, enabling developers to test their apps on virtual devices with different configurations and Android versions. Android Studio seamlessly integrates with other popular development tools, such as the Android Software Development Kit (SDK), Gradle build system, and Firebase services. The IDE also offers extensive debugging capabilities, aiding in identifying and fixing issues quickly through features like breakpoints and real-time performance monitoring. Regular updates ensure that Android Studio stays up to date with the latest tools and technologies, providing developers with an efficient environment for Android app development.

The Layout Editor in Android Studio offers a visual interface for designing user interfaces, allowing developers to drag and drop components and create responsive layouts. Code templates and snippets are available to assist developers in writing code more efficiently, providing pre-built patterns for common tasks.

The built-in debugger and profiler aid in the debugging and optimization of apps. Developers can set breakpoints, step through code, and analyze performance metrics to identify and resolve issues. Additionally, the Android Virtual Device Manager allows for the creation and management of virtual Android devices, enabling thorough app testing across various configurations.

To ensure app quality, Android Studio supports testing frameworks like JUnit and Espresso, facilitating unit testing and UI testing. It also includes Instant Run, a feature that enables developers to see code changes reflected in the running app without needing a complete restart.

Android Studio is regularly updated to incorporate the latest tools and technologies for Android development. Its user-friendly interface, along with its extensive range of features and tools, makes it an invaluable resource for developers to efficiently build, test, and optimize high-quality Android applications.

**CHAPTER 6: SOFTWARE TESTNG**

Testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation. Test techniques include, but are not limited to, the process of executing a program or application with the intent of finding software bugs. Software testing can also be stated as the process of validating and verifying that a software program or application or product: 1. Meets the business and technical requirements that guided 2. Works as expected; 3. Can be implemented with the same characteristics

**6.1 TYPES OF TESTING**

**6.1.1 Unit Testing**

It focuses on smallest unit of software design. In this we test an individual unit or group of inter related units.

**6.1.2 Regression Testing**

The objective is to take unit tested components and build a program structure that has been dictated by design, integration testing is testing in which a group of components are combined to produce output.

**6.1.3 Smoke Testing**

Very time new module is added leads to changes in program. This type of testing make sure that whole component works properly even after adding components to the complete program.

**6.1.4 System Testing**

In this software is tested such that it works fine for different operating system.it is covered under the black box testing technique.

**CHAPTER 7: TEST CASE**

**7.1 GUI Testing**

|  |  |
| --- | --- |
| **Test case** | **Login screen-sign up** |
| Objective | Clicks on sign up button then check all required mandatory fields with leaving all field blank |
| Expected results | All required mandatory fields should display with symbol instruction line fields are mandatory should be displayed |
| Test case | Create a password>>text box  Confirm password>> text box |
| Objective | Check the validation message for password field |
| Expected results | Correct validation message should be displayed accordingly or password and confirm password should be same in place of password mismatch |

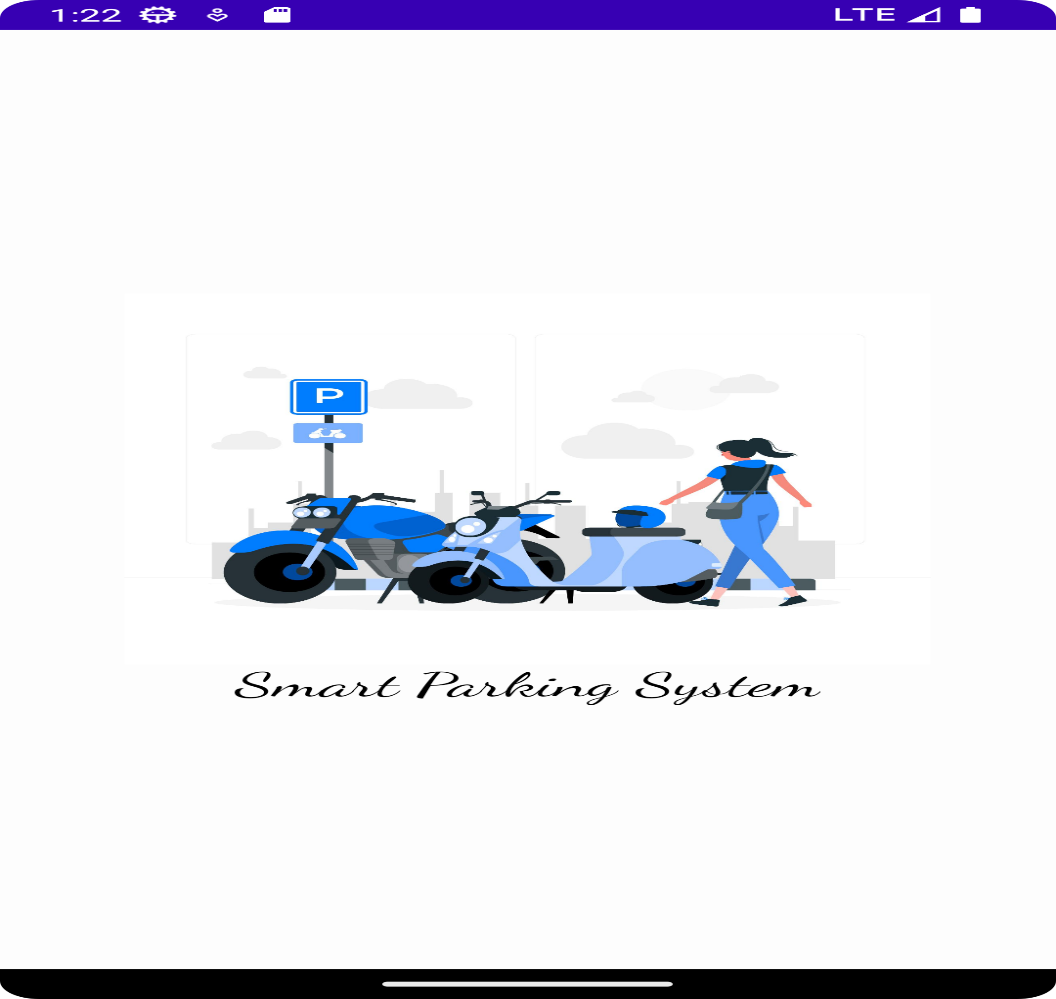
**7.2 Registration Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Test CASE ID** | **Test CASE** | **Test CASE I/P** | **Actual**  **Results** | **Expected**  **Results** | **Test CASE**  **Criteria(P/F)** |
| 001 | Enter the number in username, middle name, last name field | Number | Error comes | Error should  Comes | P |
| 001 | Enter the character in username, middle name, last name field | Character | Accept | Accept | P |
| 002 | Enter the invalid  Email id format in email id field | Kkgmail.com | Error comes | Error should comes | P |
| 002 | Enter the valid email id format  In email id field | Kk@gmail.com | Accept | Accept | P |
| 003 | Enter the invalid digit no in phone no field | 99999 | Error comes | Error should  Comes | P |
| 003 | Enter the 10 digits no in phone no field | 9999999999 | Accept | Accept | P |

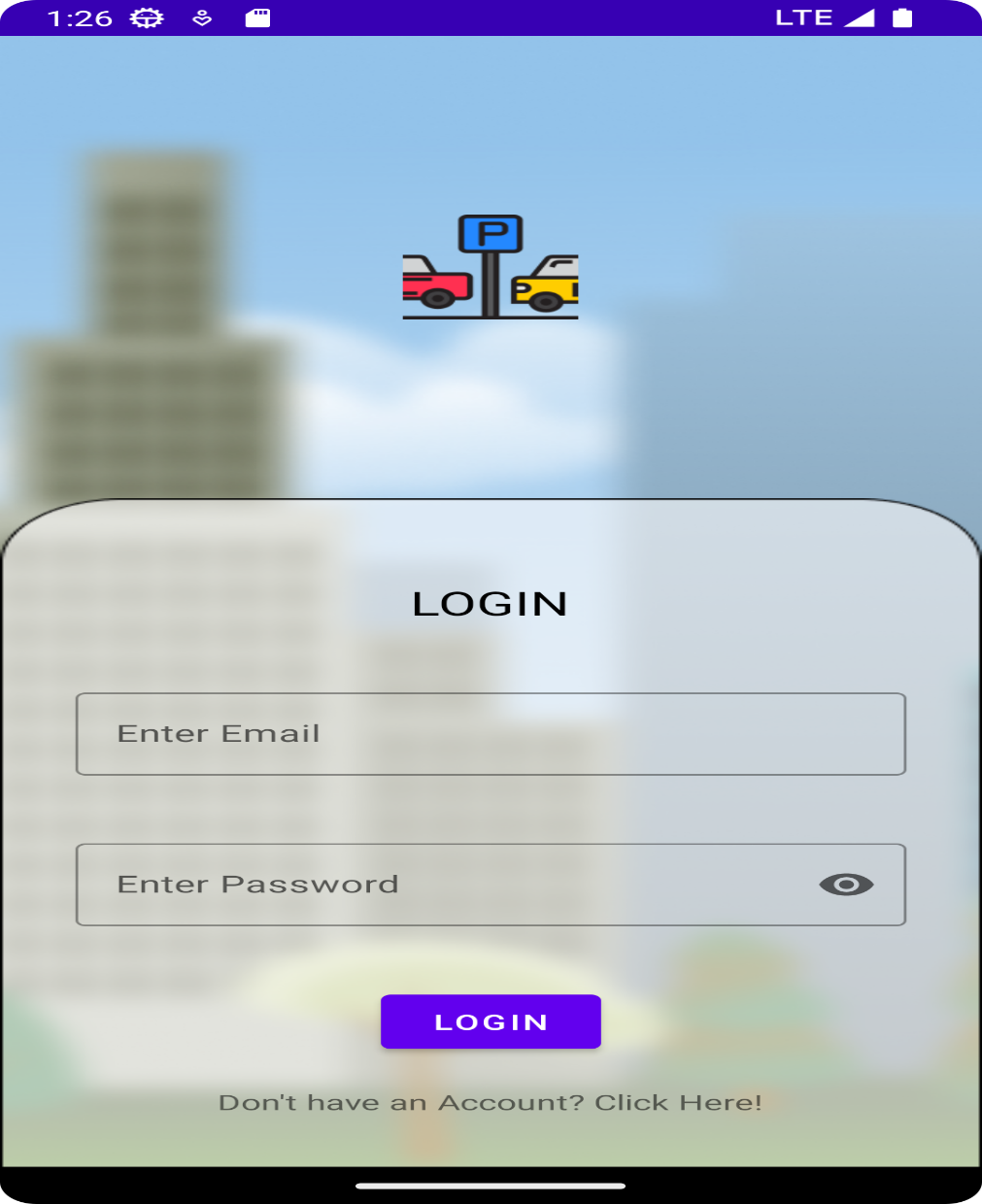
**7.3 Login Test Case**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test case id | Test case | Test case i/p | Actual results | Expected result | Test case  Criteria(p/f) |
| 001 | Enter the wrong  Username or password click on submit button | Username or  Password | Error comes | Error should come | P |
| 002 | Enter the correct username and password click on submit button | Username and passsword | Accept | Accept | P |

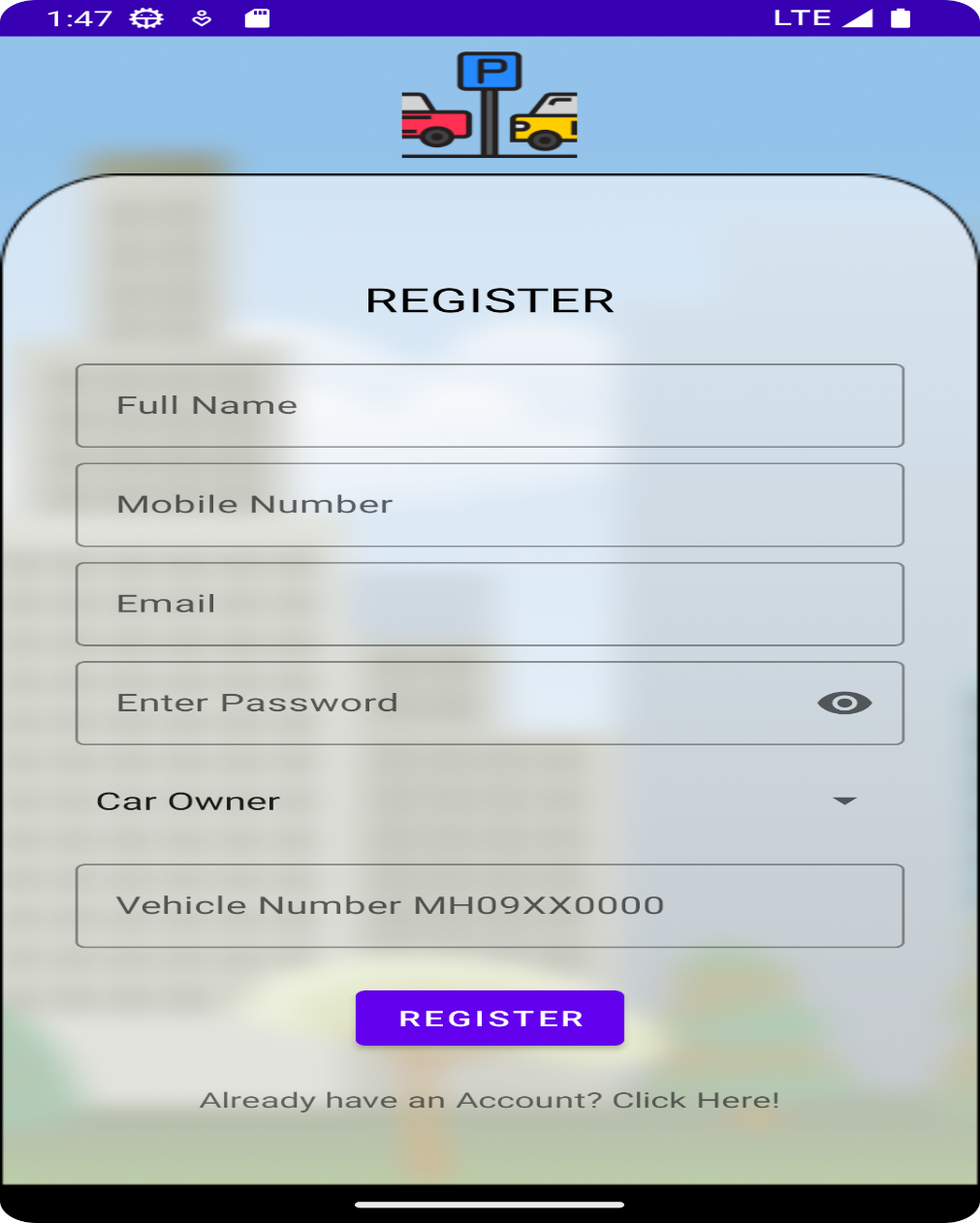
**CHAPTER 8: RESULTS AND SCREENSHOT**

****

**Fig: 8.1 Main UI**

****

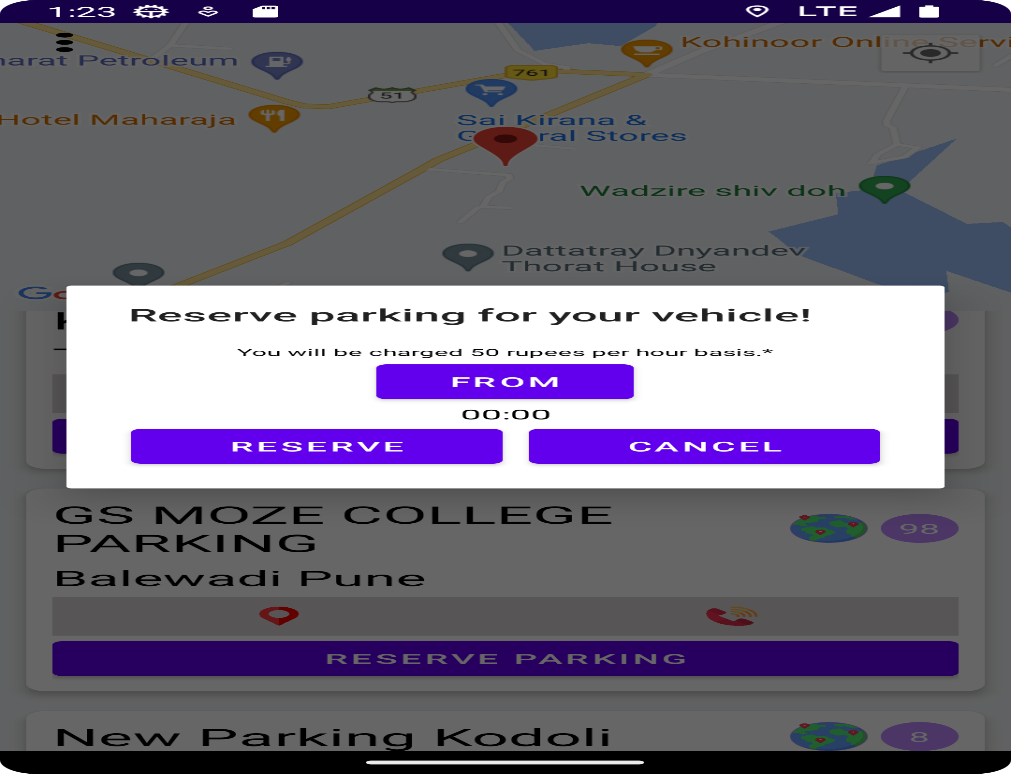
**Fig: 8.2 Login Page**

****

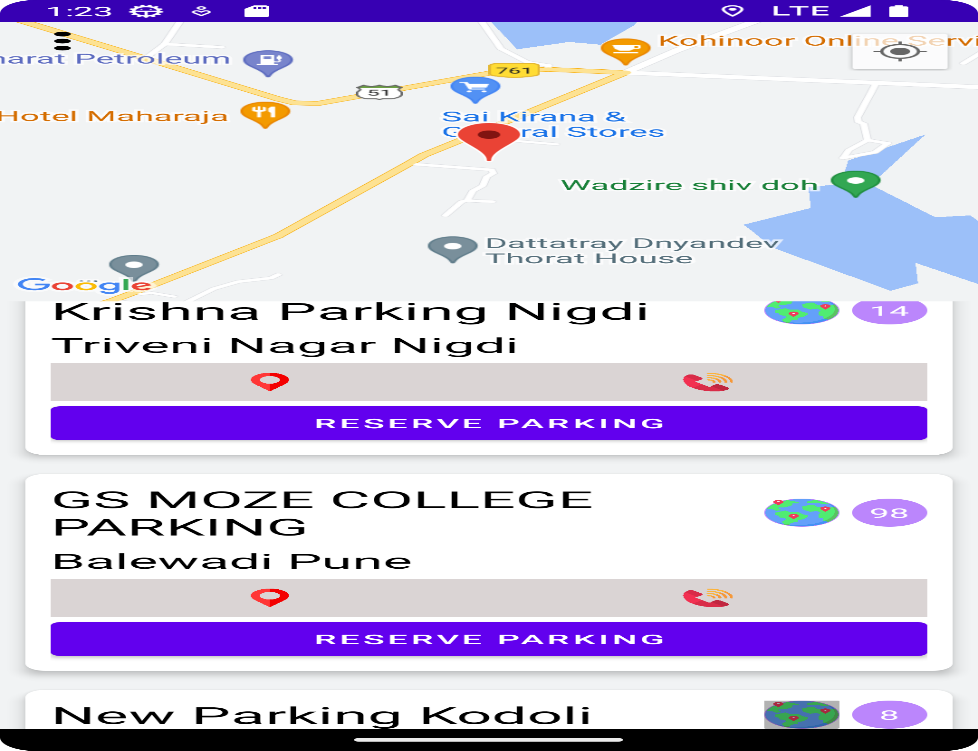
**Fig: 8.3 Registration Page**

****

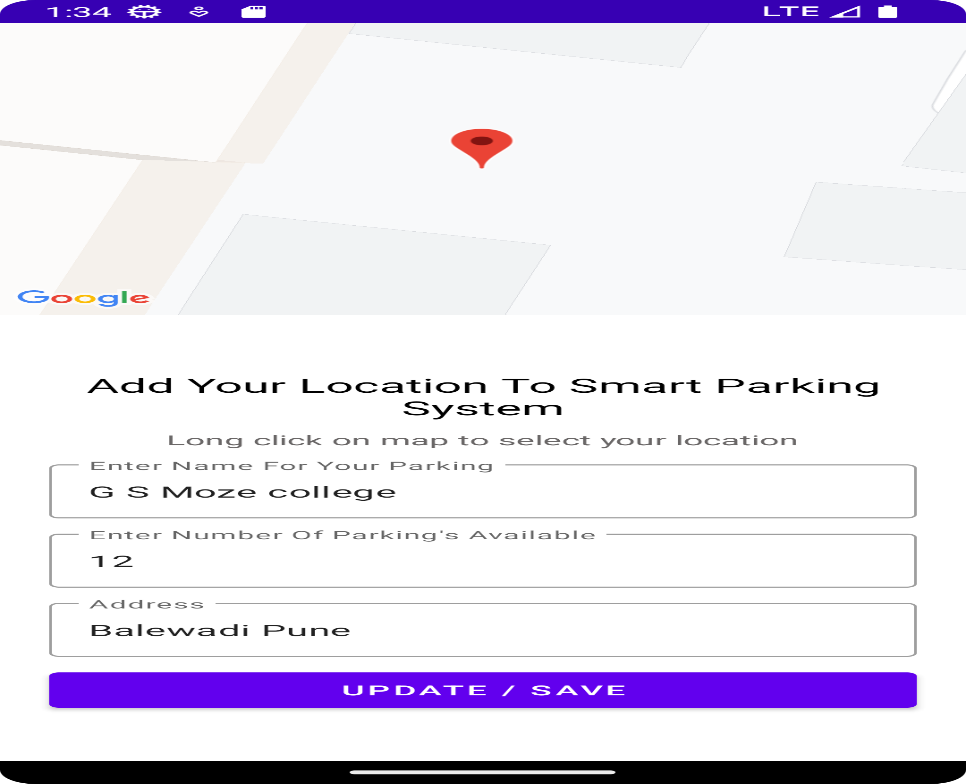
**Fig: 8.4 Booking Time**

****

**Fig: 8.5 Parking Reservation**

****

**Fig: 8.6 Car Owner Interface**

****

**Fig: 8.7 Parking Owner Interface**

****

****

**Fig: 8.8 Payment Gateway**

****

**Fig: 8.9 Parking History**

**CHAPTER 9: CONCLUSION**

**9.1 Conclusion**

From this we can conclude that with the growing use of vehicles and smartphones being a very accessible device most of the places it becomes greatly convenient to solve the problem of managing a parking place with the help of a mobile application. Android has the largest user base and hence helps us get access to much more customers than any other platform. An Android app will be of great help to manage any parking place on the go.

Android apps being comparatively cheaper and more accessible, it becomes easier for the user to have access to the app. This app will provide the user with most of the basic requirements such as logging in details of the vehicles and rent generation. Further, it also helps the user by indicating him statistics about his operations, which will be of great help in monitoring the working of the parking place and take some data backed decisions. From this we can conclude that with the growing use of vehicles and smartphones being a very accessible device most of the places it becomes greatly convenient to solve the problem of managing a parking place with the help of a mobile application. Android has the largest user base and hence helps us get access to much more customers than any other platform. An Android app will be of great help to manage any parking place on the go.

**9.2 Future Scope**

This "valid spot" app is a small step to make city a 'smart city'. This can be developed in future for wide area so that it can help people on large scale. In future this application can be implemented on the existing operation systems like ios and windows. And also it would be viable to add some extra features like extend the time period of booked slot and should be possible to send message to user before expired time period.

**CHAPTER 10: REFERENCES**

[1] M. A. R. Sarkar, A. A. Rokoni, M. O. Reza, M. F. Ismail, “Smart parking system with image processing facility”, I. J. Intelligent System and Application, 41-47, 2012.

[2] D. J. Bonde “Automated car parking system commanded by android application” in Proc. IEEE Conf.,03- 05, Jan 2012.

1. R. Yusnita, FarizaNorbaya, and Norazwinawati Basharuddin “Intelligent Parking

Space Detection System Based on Image Processing”, Internationl Journal of Innovation, Management and Technology, 232-253, 2012.

1. Yanfeng Geng, Christos G. Cassandras, “A new “Smart Parking” system Infrastructure and Implementation”, Science Direct, Social and Science Behavioral sciences, 1278-1287 ,2012
2. M. M. Rashid, A. Musa, M. AtaurRahman, and N. Farahana, A. Farhana “Automatic

Parking Management System and Parking Fee Collection Based on Number Plate Recognition” International Journal of Machine Learning and Computing, 93-98, 2012.

1. Tejal Lotlikar Minla Chandrahasan, Ankita Mahadik, Madhusmita Oke, Anjali Yeole

“Smart Parking Application September 2016 International Journal of Computer Applications 149(9):32- 37DOI:10.5120/ijca2016911529

1. Zhanlin Ji, Ivan Ganchev1, Máirtín O’Droma and Xueji Zhang, “A Cloud-Based

Intelligent Car Parking Services for Smart Cities” 2014 XXXIth URSI General Assembly and

Scientific Symposium (URSI GASS) Telecommunications Research Centre (TRC),

University of Limerick, Ireland.

1. Sangwon Lee, Dukhee Yoon, Amitabha Ghosh Autonomous Networks Research Group Ming Hsieh Department of Electrical Engineering Intelligent Parking Iot Application Using Wireless Sensor Networks.

1. S. V. Srikanth, Pramod P. J, Dileep K. P, Tapas S, Mahesh U. Patil, Sarat Chandra

Babu N “Design and Implementation of a Prototype Smart PARKing (SPARK) System Using Wireless Sensor Networks”

1. Amir O. Kotb, Yao-chun Shen, and Yi Huang “Smart Parking Guidance, Monitoring and Reservations” Senior Member, IEEE, Department of Electrical Engineering and Electronics, University of Liverpool, Liverpool, L69 3GJ UK.



