INTELLIGENT PARKING SYSTEM

Janhavi Paste (Roll No. 16102A0062)

Snehal Shinde (Roll No. 16102A0063)

Priyanka Mali (Roll No. 17102A2010)

Submitted in partial fulfillment of the requirements

of the degree of

Bachelor of Engineering in Computer Science

Under the Guidance of

Prof. Amit Nerurkar

Department of Computer Engineering

Vidyalankar Institute of Technology
Wadala(E), Mumbai-400437

University of Mumbai

2019-20

CERTIFICATE OF APPROVAL

This is to certify that the project entitled

"INTELLIGENT PARKING SYSTEM"

is a bonafide work of

Janhavi Paste(Roll No. 16102A0062)

Snehal Shinde(Roll No. 16102A0063)

Priyanka Mali(Roll No. 17102A2010)

submitted to the University of Mumbai in partial fulfillment of the requirement for the award of the degree of

Undergraduate in **"Computer Engineering"**.

Guide	Head of Department	Principal
Prof. Amit Nerurkar	Dr. Arun Chavan	Dr. S. A. Patekar

Project Report Approval for B. E.

This project report entitled ***INTELLIGENT PARKING SYSTEM*** by

1. ***Janhavi Paste (16102A0062)***
2. ***Snehal Shinde (16102A0063)***
3. ***Priyanka Mali (17102A2010)***

is approved for the degree of ***Bachelor of Engineering in Computer Engineering.***

Examiners

1.----------

2.----------

Date:

Place:

Declaration

I declare that this written submission represents my ideas in my own words and where others' ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Sr. No.	Name of Student	Roll No.	Signature
1.	Janhavi Paste	16102A0062	
2.	Snehal Shinde	16102A0063	
3.	Priyanka Mali	17102A2010	

Date:

ACKNOWLEDGEMENTS

I would like to take the opportunity to thank and express my deep sense of gratitude to my Guide Prof.Amit Nerurkar. I am greatly indebted for providing their valuable guidance at all stages of the study, their advice, constructive suggestions, positive feedback and encouragement, without which it would have not been possible to complete the project.

I owe my whole hearted thanks and appreciation to the entire staff of the company for their cooperation and assistance during the course of my project. I hope that I can build upon the experience and knowledge that I have gained and make a valuable contribution towards this industry in coming future.

I perceive this opportunity as a big milestone in my career development. I will strive to use gained skills and knowledge in the best possible way, and I will continue to work on their improvement, in order to attain desired career objectives. Hope to continue cooperation with all of you in the future

ABSTRACT

Parking is a problem for almost everyone today so there has to be a solution, which helps getting rid of problems arising due to the lack of a proper parking management system. Although various traditional PGIS (Parking Guidance Information System) exist, they can serve only a few users because it is difficult for such static systems to disseminate information on a wider scale. So the aim of this study is to provide a dynamic solution by introducing the concept of parking guidance system over the internet and also using one of the latest techniques available today i.e. the QR code for the user's ease. The system is basically designed for a parking which can further be extended as required. This system enhances the components of existing parking system available in the colleges. This system runs on a mobile phone platform and provides a visual display of parking lots available to the user so that the user can book or reserve a space. The Quick Response QR code is affixed at every parking space. The user can thus select the parking space from the visual display. The user needs to scan the QR Code while parking and unparking the vehicle. The action of the user is then reflected in the database. The android application was thus developed that can incur the parking information which was uploaded on the web map server. This system reduces the time which is involved in searching the parking space thus reducing the fuel consumption, user's frustration. It reduces vehicle travel time and parking time.

TABLE OF CONTENTS

INTRODUCTION

The number of vehicles is increasing day by day whereas the parking spaces are still limited. Vehicular population is shooting out the roof, no amount of space is sufficient to accommodate stationary vehicles. Management of parking has grown to large extent. Due to which in metro cities, Parking is a big challenging problem.

Daily, it is estimated that 30% of vehicles in the central area of cities travel around for finding parking slots, which takes an average of 7.8 minutes to find parking slots. Travelling around on streets for parking does not only waste time and fuel for drivers but also contributes to additional traffic congestion, gas emissions, and traffic accidents. Not only the scarcity of parking area but also the lack of well-designed reservation system for parking services is another parking problem. Since past few years, on-street and garage parking management problems attract attention from transportation science, operational management and computer science fields. Some researchers design and implement various smart parking systems. The automatic allocation or reservation approach is a key element in parking management systems. Many researchers combined parking reservation and pricing models to minimize the drivers' costs and maximize the resource utilization.

Now-a-days there is growing popularity and affordability of internet–enabled smartphones and because of data available online we can step to solve parking problem. Android smartphone enables user to virtually carry the internet with him.

A reservation system is used in our android application which will help the driver to reserve a parking spot ahead of time and will allocate a parking spot to the driver on its arrival to the parking spot. Hence, it is helpful to the drivers as well as the environment if we reserve a parking spot ahead of time before arrival to the destination.

1.1 Problem Statement

According to the traditional approach, Drivers have to waste time for finding a parking spot in a new area or a crowded area and finding an empty parking space in that selected parking slot is a tedious job. Not finding a parking space sometimes is indeed a critical issue and can cause congestion on streets. The number of vehicles is also increasing each day, adding to the parking issues faced at public places. Drivers in metro cities face difficulties on a daily basis to find a parking space especially during peak hours, the difficulty roots from lack of knowledge of where the parking spaces are available at the given time. Even if it is known, many vehicles may pursue a small number of parking spaces which in turn leads to traffic congestion. The traffic on roads and parking space has been an area of concern in majority of the cities. To avoid these problems, recently many new technologies have been developed that help in solving the parking problems to a great extent.

Availability of a parking spot is unknown to the drivers, hence after finding a parking lot it is uncertain whether the driver will get a parking spot or not, due to unavailability of parking spaces. This could result in wastage of time, money and fuel since travelling to such a parking lot will be useless. Sometimes while paying for the parking there is a shortage of exact change available at both the sides i.e. at the customer side and as well the parking manager side. So, it is a hectic job to find the exact amount or change required.

Due to the proliferation in the number of vehicles on the road, traffic problems are bound to exist. This is due to the fact that the current transportation infrastructure and car park facility developed are unable to cope with the influx of vehicles on the road. To alleviate the aforementioned problems, the smart parking system has been developed. With the implementation of the smart parking system, patrons can easily locate and secure a vacant parking space at any car park deemed convenient to them. Vehicle ingress and egress are also made more convenient with the implementation of hassle-free payment mechanism. With vehicle detection sensors aplenty on the market, the choices made may defer due to the different requirements in addition to the its pros and cons.

Subsequently, the various sensor systems used in developing the systems in addition to the recent research and commercial system on the market are examined as vehicle detection plays a crucial role in the smart parking system. The proposed system illustrates the counter for the cars

enter the park and give the order to the park gate never to open to any entered car when the number of cars in the car reach the highest number. In addition, the same counter made for the exit gate (where is the same the entrance gate in this project) count in reverse order for the cars exit from the park.

1.2 Aims and Objectives

To overcome all the problems of the traditional parking system, we have developed a system which will be helpful in dealing with these problems.

Primary Target:

Finding a parking lot in an unknown locality, by providing the user with a list of parking lots in his vicinity available to him. As well as providing the user with the status of available parking spots in the parking lot, and most importantly, reserving a parking spot beforehand.

Secondary Target:

Providing navigation to the parking spot, while showing the shortest and optimal path from user's current location to the parking lot which helps in saving time as well as fuel and makes the job of the driver easier. Including a wallet for online hassle-free payment.

OBJECTIVES:

- To minimise the effects of on street parking upon road safety and congestion.
- To reduce dependence on the car, particularly in town centres.
- To help maintain the vitality of town centres and to minimise out of town developments.
- To ensure that car parking provision and enforcement are broadly self-financing through the Decriminalisation of Parking Enforcement (DPE) process.
- To reduce, where possible, environmental damage caused by cars and car ownership, particularly in residential areas.
- To reduce, where possible, competition for road space, between residents and other groups.
- If through town centre regeneration and development more shoppers/visitors are attracted to the town centres surrounding roads will become increasingly congested. Some reduction in potential congestion may be achieved by encouraging through traffic to use other routes and by assessing the design and access to developments in terms of minimum access through the town centre core.
- To ensure that any on street parking close to town centres is of short duration to ensure reasonable customer turnover for shops and commercial premises and to prevent longer stay parking by non-users of town centre facilities. That the duration of stay is lengthened commensurate with the distance from the commercial core of the towns; to implement schemes to deal with local on street parking "issues", in an agreed priority order based upon objective criteria, for locations where genuine road safety, congestion or residents' problems occur.

- To monitor the effects of implemented schemes on adjacent areas with subsequent action only being taken where the same criteria are met.

- Where residents compete for road parking space in their own streets with other groups (local workers, commuters etc) new schemes will give greater priority to residents.
- Where local businesses compete for road parking space in their local industrial areas with other groups, (commuters etc) new schemes will give greater priority to local businesses.

1.3 Literature Survey

1. State-of-the-art Parking Management:

Traffic searching for parking comparison under different parking guidance strategies. Many parking guidance systems have been developed over the past decade .In this subsection, we study several existing parking guidance approaches and explain their limitations. Furthermore, we simulate these different parking management strategies under realistic traffic and parking Conditions, compare their performance, and show results.

2. Blind Search:

Blind searching is the simple strategy applied by users when there is no parking information. In this case, the drivers keep cruising for parking spaces within a certain distance to their destination. The drivers will stop searching until finding any available space. Otherwise, the drivers will extend the searching area and continuously look for vacant spaces in the neighbouring parking lots.

3. Parking Information Sharing (PIS):

This mechanism is commonly adopted by the current state of the smart Parking system design. After the smart parking system publishes the parking availability information to the drivers in certain area, the driver will decide their desired parking destination where the parking lot has available spaces, according to the obtained parking availability information. However, if the number of vacant spaces in a parking lot is very limited in busy hours, it is likely that the number of drivers in demand for these parking spaces, which is based on parking information. This phenomenon is called “multiple-car-chasing-singlespace”, which may cause severe congestion.

4. Buffered PIS (BPIS):

To address the problematic “multiple-car-chase-singleslot” phenomenon, some designers of smart parking systems modify the PIS mechanism. They intentionally reduce the number of vacant spaces, when publishing the live availability information to keep a buffer. Therefore, though there may be more drivers pursuing the limited available spaces, the system has some extra spaces to avoid the conflict. But it is difficult to determine the number of the buffer spaces. If the buffer is too small, the problem of “multiple-car-chase-single-space” will not be eliminated. If it is too large, the utilization of parking spaces will be low. As alluded to above, the blind search system is an open loop system, where users make decision without looking at the state of the system. The PIS and BPIS strategies allow drivers to make decisions based on the system state (e.g., parking

availability information). However, the phenomena of multiple car-chase-single-spaces cannot be fully eliminated. To reduce the traffic searching for parking, we suggest a reservation based system, where drivers make reservations through the parking management system. If a driver makes the reservation successfully, it guarantees an available parking space for him, and the driver can park at the reserved space without searching. The reservation-based system allows drivers to select the most convenient parking space under their budget constraints.

Existing Parking Systems

1. Vision Based Method:

Monitoring detection technology can be divided into two categories. The first estimates the number of remaining vacant spaces for the entire parking lot by counting incoming and outgoing vehicles. The second monitors the status of each individual space and can be used to guide a car to a vacant space. To detect the status of an individual parking space different methods have been utilized, such as ultrasonic sensors placed at each space (thus it requires many sensors), or surveillance cameras placed at a high position.

2. Sensor Based Method:

Another detection technology uses sensors to detect vacant spaces in a parking lot. Different factors play a role in choosing the proper sensor, including size, reliability, adaptation to environmental changes, robustness and cost. Sensors technologies are categorized as either intrusive or non-intrusive. Intrusive sensors need to be installed directly on the pavement surface, so digging and tunneling under the road surface are required. Non-intrusive sensors only require fixing on the ceiling or on the ground. Ultrasonic sensors are categorized as non-intrusive sensors. Ultrasonic sensors transmit sound waves between 25 kHz and 50 kHz. They use the reflected energy to analyze and detect the status of a parking space. Ultrasonic waves are emitted from the head of an ultrasonic vehicle detection sensor every 60 milliseconds, and the presence or absence of vehicles is determined by time differences between the emitted and received signals.

3. Two Tier Parking & Automatic Multilevel Car parking System:

Two Tier Car Parking System is ideally suited for people having 2 cars They can use parking space for a single car to park both their cars using the Two Tier Parking System one above the other. The system consists of a single platform which allows the car that is not used very frequently to be parked on the upper level and the one that is used frequently on the lower level. G offers 2 variants for the Two Tier Parking System - Hydraulic System and Electro-Mechanical System. Automatic Multilevel Car Parking Systems can be fully automatic or semiautomatic. They can be manned or unmanned systems (i.e. operated manually or using computers). These systems can be installed above or below the ground thereby making optimum use of available space. Another advantage in this case is that human intervention is not required for parking the car.

4. Performance Metrics:

In order to evaluate the performance of the strategies implemented in smart parking systems, we introduce the following metrics, which reflect the willingness of drivers, and our concerns on traffic congestion and environmental protection.

5. Walking Distance:

Walking distance is defined as distance from a driver's selected parking space to the destination. This important factor reflects the willingness of drivers when selecting parking spaces. The driver commonly wants to choose the most convenient parking space where it is closest to his destination. In the proposed model of SPSR, the drivers select the parking spaces depending on this factor, which indicates their satisfaction.

6. Traffic Volume:

In our proposed model, traffic volume is specifically defined as the amount of traffic generated by parking searching. This factor is not negligible and associated with the traffic congestion and air pollution. The proposed reservation-based smart parking system is design to reduce the traffic volume caused by parking searching, as well as satisfy the need of drivers. We investigate performance of the proposed smart parking system using these performance metrics.

Challenges faced:

Given the design objectives of smart parking systems that require the coordination among multiple parties, we summarize the main design considerations as follows:

1. **Fake Parking Requests:**

The system collects and stores the data about the performance metrics including the status of parking space, reservation time, parking location, driver's identity. All data stored by the system is at least stamped with time metadata. As the user is allowed to book only one parking space at a time from one id, it is a big challenge for us if user is trying to book one or many request at a time from one id. To overcome this problem we implemented a queue buffer which persistently checking the new request and compare it with the existing id requests. If any requesting id matches existing id then the request is directly discarded and a message showing 'no more parking spaces available is displayed.

2. **Users Identity Verification:**

Verifying users identity is a major security concern as users with no reservation can enter and occupy someone else's parking space. In our proposed system, the user can open the application and verify their identity via the received QR code. Reservation authority sends a QR code to the user as soon as user reserves the parking space. At the parking lot host identifies the user by scanning the QR code generated by authority management system.

3. **Delay in parking:**

User reserves a parking space for specific time duration (Starting time & ending time). What if the user is unable to reach the parking spot at the aforementioned time? So to overcome this challenge, we provided the time extend option to the user, but user can only extend the time by 15 min from the specified time. User has to pay some extra money to extend the time.

4. **Timer:**

As soon as reservation time is about to expire, user must be notified about this. To deal with this, SPSR gives prior notifications at regular intervals to the user. For example, if there are only 30 minutes left to reach the expiration time. SPSR will give notification when 20 minutes are elapsed and when 10 minutes are elapsed a final notification is given.

Tools and Technology:

1. **ASP.NET**

ASP.NET is more than the next version of Active Server Pages (ASP); it is a unified Web development platform that provides the services necessary for developers to build enterprise-class Web applications. While ASP.NET is largely syntax-compatible with ASP, it also provides a new programming model and infrastructure that enables a powerful new class of applications. You can migrate your existing ASP applications by incrementally adding ASP.NET functionality to them. ASP.NET is a compiled .NET Framework -based environment. You can author applications in any .NET Framework compatible language, including Visual Basic and Visual C#. Additionally, the entire .NET Framework platform is available to any ASP.NET application. Developers can easily access the benefits of the .NET Framework, which include a fully managed, protected, and feature-rich application execution environment, simplified development and deployment, and seamless integration with a wide variety of languages.

.net Framework:

The .NET Framework is Microsoft's Managed Code programming model for building applications on Windows clients, servers, and mobile or embedded devices. Microsoft's .NET Framework is a software technology that is available with several Microsoft Windows operating systems. In the following sections describes , the basics of Microsoft .Net Frame work Technology and its related programming models.

C# is a language for professional programming. C# (pronounced C sharp) is a programming language designed for building a wide range of enterprise applications that run on the .NET Framework. The goal of C# is to provide a simple, safe, modern, object-oriented, high-performance, robust and durable language for .NET development. Also it enables developers to build solutions for the broadest range of clients, including Web applications, Microsoft Windows Forms-based applications, and thin- and smart-client devices.

2. **ANDROID**

Android is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touchscreen mobile devices such as smartphones and tablets.

Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. In addition to touchscreen devices, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics. Android has the largest installed base of all operating systems (OS) of any kind. Android has been the best selling OS on tablets since 2013, and on smartphones it is dominant by any metric. Initially developed by Android, Inc., which Google bought in 2005, Android was unveiled in 2007 along with the founding of the Open Handset Alliance – a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. As of July 2013, the Google Play store has had over one million Android applications ("apps") published – including many "business-class apps"[1] that rival competing mobile platforms – and over 50 billion applications downloaded An April–May 2013 survey of mobile application developers found that 71% of developers create applications for Android,and a 2015 survey found that 40% of full-time professional developers see Android as their priority target platform, which is comparable to Apple's iOS on 37% with both platforms far above others. In September 2015, Android had 1.4 billion monthly active devices. Android's source code is released by Google under open source licenses, although most Android devices ultimately ship with a combination of open source and proprietary software, including proprietary software required for accessing Google services. Android is popular with technology companies that require a ready-made, low-cost and customizable operating system for high-tech devices. Its open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which deliver updates to older devices, add new features for advanced users or bring Android to devices originally shipped with other operating systems. The success of Android has made it a target for patent (and copyright) litigation as part of the so-called "smartphone wars" between technology companies.

FEATURES:-

Interface

Android's default user interface is mainly based on direct manipulation, using touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching, and reverse pinching to manipulate on-screen objects, along with a virtual keyboard. Game controllers and full-size physical keyboards are supported via Bluetooth or USB. The response to user input is designed to be immediate and provides a fluid touch interface, often using the vibration capabilities of the device to provide haptic feedback to the user. Internal hardware, such as accelerometers, gyroscopes and proximity sensors[61] are used by some applications to respond to additional user actions, for example adjusting the screen from portrait to landscape depending on how the device is oriented, or allowing the user to steer a vehicle in a racing game by rotating the device, simulating control of a steering wheel. Android devices boot to the homescreen, the primary navigation and information "hub" on Android devices that is analogous to the desktop found on personal computers. (Android also runs on regular personal computers, as described below). Android homescreens are typically made up of app icons and widgets; app icons launch the associated app, whereas widgets display live, auto-updating content, such as the weather forecast, the user's email inbox, or a news ticker directly on the homescreen. A homescreen may be made up of several pages, between which the user can swipe back and forth, though Android's homescreen interface is heavily customisable, allowing users to adjust the look and feel of the devices to their tastes.Third-party apps available on Google Play and other app stores can extensively re-theme the homescreen, and even mimic the look of other operating systems, such as Windows Phone.Most manufacturers, and some wireless carriers, customise the look and feel of their Android devices to differentiate themselves from their competitors.Applications that handle interactions with the homescreen are called "launchers" because they, among other purposes, launch the applications installed on a device. Along the top of the screen is a status bar, showing information about the device and its connectivity. This status bar can be "pulled" down to reveal a notification screen where apps display important information or updates, such as a newly received email or SMS text, in a way that does not immediately interrupt or inconvenience the user.Notifications are persistent until read by tapping it, which opens the relevant app, or dismissed by sliding it off the screen. Beginning on Android 4.1, "expanded notifications" can display expanded details or additional functionality; for instance, a music player can display

playback controls, and a "missed call" notification provides buttons for calling back or sending the caller an SMS message. Android provides the ability to run applications that change the default launcher, and hence the appearance and externally visible behaviour of Android. These appearance changes include a multi-page dock or no dock, and many more changes to fundamental features of the user interface.

Applications

Applications ("apps"), which extend the functionality of devices, are written using the Android software development kit (SDK) and, often, the Java programming language that has complete access to the Android APIs. Java may be combined with C/C++, together with a choice of non-default runtimes that allow better C++ support; the Go programming language is also supported since its version 1.4, which can also be used exclusively although with a restricted set of Android APIs. The SDK includes a comprehensive set of development tools, including a debugger, software libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Initially, Google's supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) plugin; in December 2014, Google released Android Studio, based on IntelliJ IDEA, as its primary IDE for Android application development. Other development tools are available, including a native development kit (NDK) for applications or extensions in C or C++, Google App Inventor, a visual environment for novice programmers, and various cross platform mobile web applications frameworks. In January 2014, Google unveiled an framework based on Apache Cordova for porting Chrome HTML 5 web applications to Android, wrapped in a native application shell.Android has a growing selection of third-party applications, which can be acquired by users by downloading and installing the application's APK (Android application package) file, or by downloading them using an application store program that allows users to install, update, and remove applications from their devices. Google Play Store is the primary application store installed on Android devices that comply with Google's compatibility requirements and license the Google Mobile Services software. Google Play Store allows users to browse, download and update applications published by Google and third-party developers; as of July 2013, there are more than one million applications available for Android in Play Store.As of July 2013, 50 billion applications have been installed. Some carriers offer direct carrier billing for Google Play application purchases, where the cost of the application is added to the user's monthly bill. Due to the open nature of Android,

a number of third-party application marketplaces also exist for Android, either to provide a substitute for devices that are not allowed to ship with Google Play Store, provide applications that cannot be offered on Google Play Store due to policy violations, or for other reasons. Examples of these third-party stores have included the Amazon Appstore, GetJar, and SlideMe. F-Droid, another alternative marketplace, seeks to only provide applications that are distributed under free and open source licenses.

Memory management

Since Android devices are usually battery-powered, Android is designed to manage processes to keep power consumption at a minimum. When an application is not in use the system suspends its operation so that, while available for immediate use rather than closed, it does not use battery power or CPU resources.Android manages the applications stored in memory automatically: when memory is low, the system will begin invisibly and automatically closing inactive processes, starting with those that have been inactive for longest. Lifehacker reported in 2011 that third-party task killers were doing more harm than good.

Virtual reality

At Google I/O on May 2016, Google announced Daydream, a virtual reality platform that relies on a smartphone and provides VR capabilities through a virtual reality headset and controller designed by Google itself.The platform will be built into Android starting with version N, differentiating from standalone support for VR capabilities. The software is available for developers, and will be widely released later in 2016.

3. Microsoft SQL Server

Relational database management:

A relational database management system uses only its relational capabilities to manage the information stored in its database.

Information Representation:

All information stored in a RDBMS is represented only by data item values, which are stored in the tables that makeup the database.

Logical Accessibility:

Every data item value stored in a relational database is accessible by stating the name of the table it is stored in, the name of the column under which it is stored and the value of the primary key that defines the row in which it is stored.

Representation of Null Values:

The database management system has a consistent method for representing null values. For example, null values for numeric data must be distinct from zero or any other numeric value and for character data it must be different from a string of blanks or any other character value

Catalog Facility:

The logical description of the relational database management system is represented in the same manner as ordinary data. This is done so that the facilities of the relational database management system itself can be used to maintain database description.

Data Language:

A relational database management system may support many types of languages for description data and accessing the database. However, there must be at least one language that uses ordinary character strings to support the definition of data, the definition of views, the manipulation of data, constraints on data integrity, information concerning authorization and the boundaries for recovery of units.

View Updating:

Any view that can be defined using combinations of base tables, that are theoretically updateable, is capable of being updated by the relational database management system.

Insert, Update and Delete:

Any operand that describes the results of a single retrieval operation is capable of being applied to an insert, update or delete operation as well.

Physical Data Independence:

Changes made to physical storage representations or access methods do not require changes to be made to application programs.

Logical Data Independence:

Changes made to tables, that do not modify any data stored in that table, do not require changes to be made to application programs.

Integrity Constraints:

Constraints that apply to entity integrity and referential integrity are specifiable by the data language implemented by the database management system and not by the statements coded into the application programs.

Database Distribution:

The data language implemented by the relational database management system supports the ability to distribute the database without requiring changes to be made application programs.

Not Subversion:

If the relational database management supports facilities that allow application programs to operate on the table's a row at a time, an application program using this type of database access is prevented from bypassing entity integrity or referential integrity constraints that are defined for the database.Business today demands a different kind of data management solution. Performance scalability, and reliability are essential, but businesses now expect more from their key IT investment.SQL Server 2005 exceeds dependability requirements and provides innovative capabilities that increase employee effectiveness, integrate heterogeneous IT ecosystems, and maximize capital and operating budgets. SQL Server 2005 provides the enterprise data management platform your organization needs to adapt quickly in a fast changing environment.

Benchmarked for scalability, speed, and performance, SQL Server 2005 is a fully enterprise-class database product, providing core support for Extensible Markup Language (XML) and Internet queries.

Easy-to-use Business Intelligence(BI) Tools:

Through rich data analysis and data mining capabilities that integrate with familiar applications such as Microsoft Office, SQL Server 2005 enables you to provide all of your employees with critical, timely business information tailored to their specific information needs. Every copy of SQL Server 2005 ships with a suite of BI services

Self-Tuning and Management Capabilities:

Revolutionary self-tuning and dynamic self-configuring features optimize database performance, while management tools automate standard activities. Graphical tools and performance, wizards simplify setup, database design, and performance monitoring, allowing database administrators to focus on meeting strategic business needs.

Data Management Application and Services:

Unlike its competitors, SQL Server 2005 provides a powerful and comprehensive data management platform. Every software license includes extensive management and development tools, a powerful extraction, transformation, and loading (ETL) tool, business intelligence and analysis services, and analysis service, and such as Notification Service. The result is the best overall business value available.

ANALYSIS

2.1 Process Model

For implementing our project, we are going to use Incremental model. Since the main feature of Incremental model is that it provides working model which becomes convenient in these ever-changing additional functionalities.

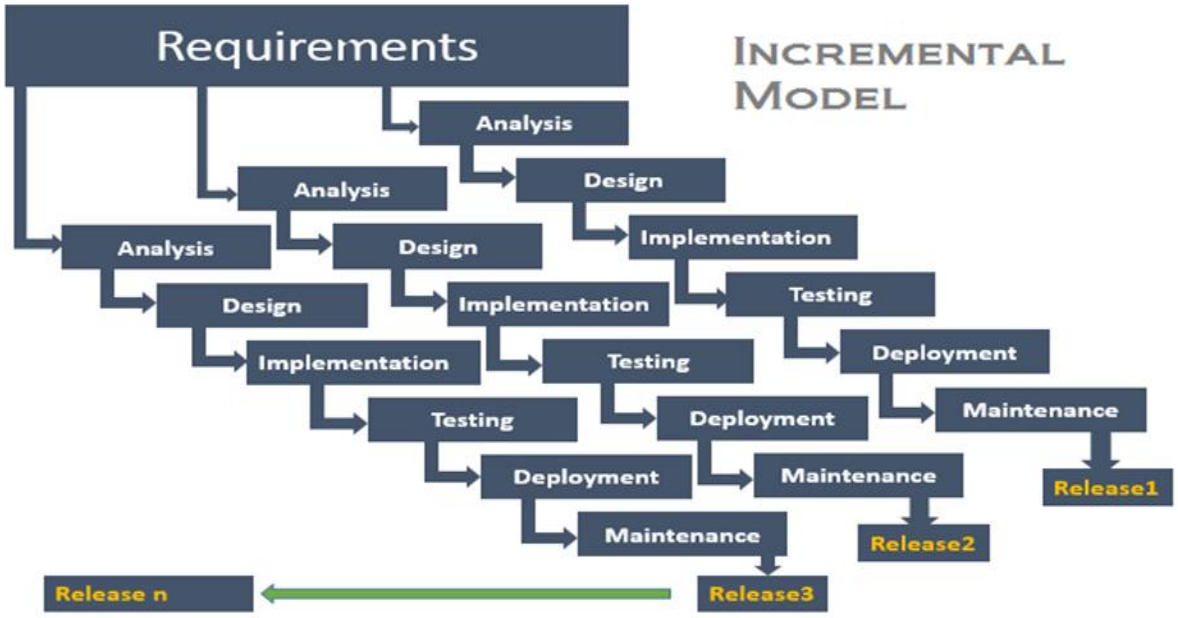

Fig 2.1.1 Process Model (Incremental Model)

The incremental build model is a method of software development where the product is designed, implemented and tested incrementally (a little more is added each time) until the product is finished. It involves both development and maintenance. The product is defined as finished when it satisfies all of its requirements. This model combines the elements of the waterfall model with the iterative philosophy of prototyping. The product is decomposed into a number of components, each of which is designed and built separately (termed as builds). Each component is delivered to the client when it is complete. This allows partial utilization of the product and avoids a long development time. It also avoids a large initial capital outlay and subsequent long waiting period. This model of development also helps ease the traumatic effect of introducing a completely new system all at once. There are, however, several problems with this model.

Advantages-:

1. After each iteration, regression testing should be conducted. During this testing, faulty elements of the software can be quickly identified because few changes are made within any single iteration.
2. It is generally easier to test and debug than other methods of software development because relatively smaller changes are made during each iteration. This allows for more targeted and rigorous testing of each element within the overall product.

2.2 Feasibility Study

Feasibility study is totally depending upon the preliminary investigations and requirements of the system. Hence we have to determine the system requested is feasible or not. This helps us to check technical, operational and financial feasibility of requested system against the current system. The data collection done at preliminary stage examines that the system, which we are developing, will be beneficial to user. Various types of feasibility study we have done are user, operational, resource, technical and financial which are equally important.

User Feasibility:

The proposed software will provide ease to the user as they can reserve a parking spot before arrival so they are assured that they will get a parking spot guaranteed even at a peak time.

Technical Feasibility:

Proposed system is an android application with firebase at the backend. Various resource allocation algorithms will be used for allocating a parking a spot to the user at given time slot. Since, the technology needed for the proposed system is available and it is feasible.

Financial and Economic Feasibility:

The cost of software is feasible, as it requires investments at the system that is only computer and mobile is required. Thus, our project is economically feasibleand possible to complete this project till the given deadline.

Operational feasibility:

The proposed project is an improvement over an existing system. The sequence is suitable for the implementation and development of the project. The proposed system is sustainable and reliable.

Resource feasibility:

There is sufficient time available to build the new system, it can be built, and it does not interfere with normal business operations, type and amount of resources required, dependencies, and developmental procedures.

2.3 Cost Estimation:

Since the team size required is adequately small, the problem is well understood and has been solved in the past and also the team members have a nominal experience regarding the problem, hence, the software project is said to be an organic type. Considering organic value, the basic COCOMO can be calculated as follows:

Effort estimation is required to find the number of person working on project, number of duration and lastly cost. The effort estimation of our project is as follows.

Software Project	a_b	b_b	c_b	d_b
Organic	2.4	1.05	2.5	0.38
Semi-Detached	3.0	1.12	2.5	0.35
Embedded	3.5	1.20	2.5	0.32

1. **Efforts**

   $E= a_b(KLOC)^{bb}$

   $E= 2.4(3.2)^{1.05}$

   E= 8.14 person per month ~ 8.

2. **Project Duration**

   $D= c_b(E)^{db}$

   $D= 2.5(8)^{0.38}$

   D= 5.509 months ~ 5.

3. **Number of Person**

   N= E/D

   N= 8.14/5.50

   N=1.51 ~ 2 persons.

Since, major project is software based only the cost required to complete the project is the effort required to design and code the system. If we assume cost per person monthly is Rs: 250.

Estimate cost = Efforts *250* number of person

Estimated cost =8.04*250*3=6030

DESIGN

3.1 UML Diagrams

UML (Unified Modelling Language) is a standard language for specifying, visualizing, constructing, and documenting the artefacts of software systems.It was initially started to capture the behaviour of complex software and non-software system and now it has become an OMG standard.UML is not a programming language but tools can be used to generate code in various languages using UML diagrams. UML has a direct relation with object oriented analysis and design.There are a number of goals for developing UML but the most important is to define some general purpose modelling language, which all modelers can use and it also needs to be made simple to understand and use.UML diagrams are not only made for developers but also for business users, common people, and anybody interested to understand the system. The system can be a software or non-software system. Thus it must be clear that UML is not a development method rather it accompanies with processes to make it a successful system.UML can be described as the successor of object-oriented (OO) analysis and design.UML is powerful enough to represent all the concepts that exist in object-oriented analysis and design.

In conclusion, the goal of UML can be defined as a simple modelling mechanism to model all possible practical systems in today's complex environment.Here, we present how our application will be structured, what functions and modules it has and how we will implement efficient data transfer between devices.

First, we will look at the structure of our application with the help of below given UML Diagrams.

3.1.1 Use Case Diagram

To model a system, the most important aspect is to capture the dynamic behaviour. Dynamic behaviour means the behaviour of the system when it is running/operating. Only static behaviour is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML, there are five diagrams available to model the dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature, there should be some internal or external factors for making the interaction. These internal and external agents are known as actors. Use case diagrams consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system. Hence to model the entire system, a number of use case diagrams are used.

The purpose of use case diagram is to capture the dynamic aspect of a system. However, this definition is too generic to describe the purpose, as other four diagrams (activity, sequence, collaboration, and State chart) also have the same purpose. We will look into some specific purpose, which will distinguish it from other four diagrams.

The purposes of use case diagrams can be said to be as follows

- ❑ Used to gather the requirements of a system.
- ❑ Used to get an outside view of a system.
- ❑ Identify the external and internal factors influencing the system.
- ❑ Show the interaction among the requirements are actors.

Admin Use Case:

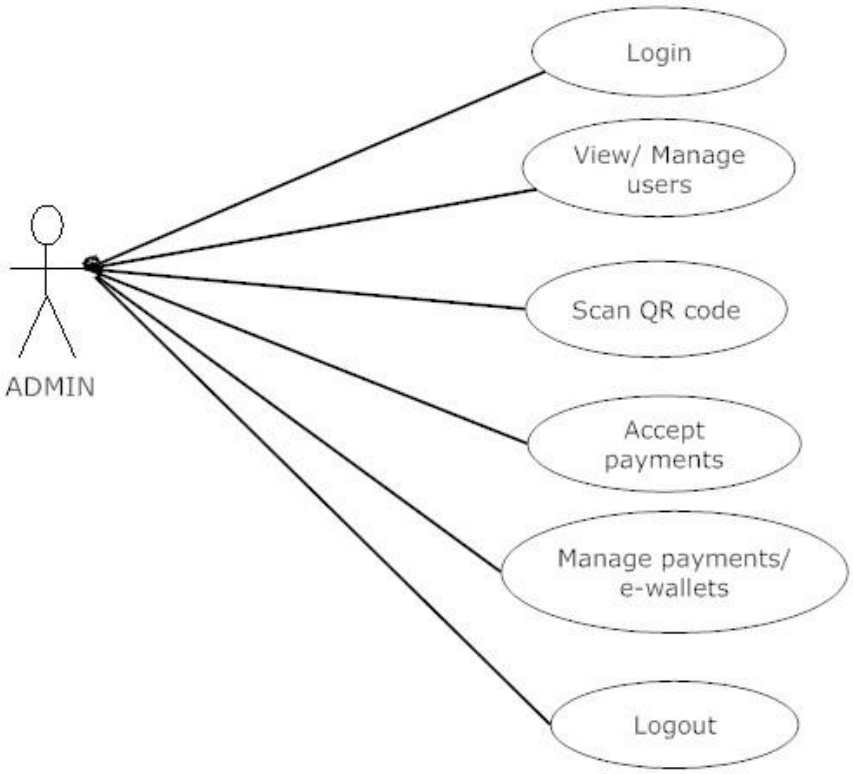

Fig. 3.1.1.1 Admin Use Case

User Use Case:

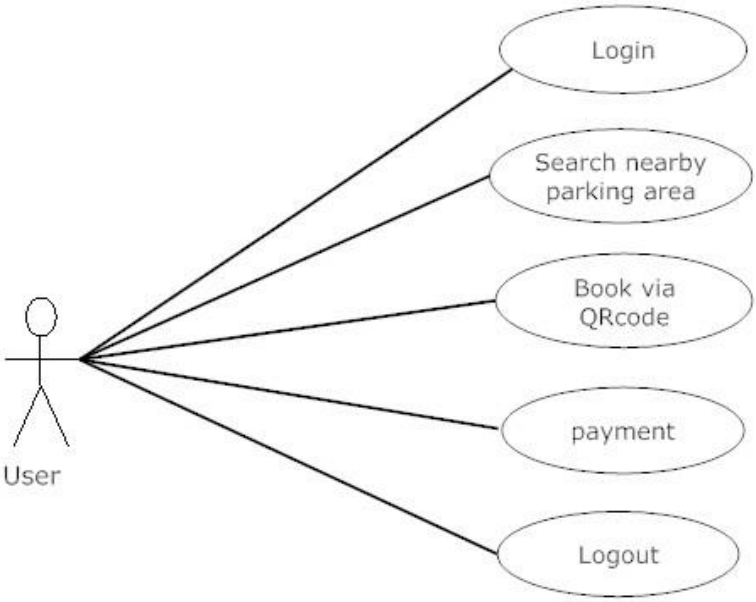

Fig. 3.1.1.2 User Use Case

3.1.2 Activity Diagram

Activity diagram is another important diagram in UML to describe the dynamic aspects of the system.Activity diagram is basically a flowchart to represent the flow from one activity to another activity. The activity can be described as an operation of the system.The control flow is drawn from one operation to another. This flow can be sequential, branched, or concurrent. Activity diagrams deal with all type of flow control by using different elements such as fork, join, etc. Activity diagrams are mainly used as a flowchart that consists of activities performed by the system. Activity diagrams are not exactly flowcharts as they have some additional capabilities. These additional capabilities include branching, parallel flow, swimlane, etc.

Before drawing an activity diagram, we must have a clear understanding about the elements used in activity diagram. The main element of an activity diagram is the activity itself. An activity is a function performed by the system. After identifying the activities, we need to understand how they are associated with constraints and conditions.The basic purposes of activity diagrams is similar to other four diagrams. It captures the dynamic behaviour of the system. Other four diagrams are used to show the message flow from one object to another but activity diagram is used to show message flow from one activity to another.Activity is a particular operation of the system. Activity diagrams are not only used for visualizing the dynamic nature of a system, but they are also used to construct the executable system by using forward and reverse engineering techniques. The only missing thing in the activity diagram is the message part.

The purpose of an activity diagram can be described as follows

- ❑ Draw the activity flow of a system.
- ❑ Describe the sequence from one activity to another.
- ❑ Describe the parallel, branched and concurrent flow of the system.

Admin Acitivity:

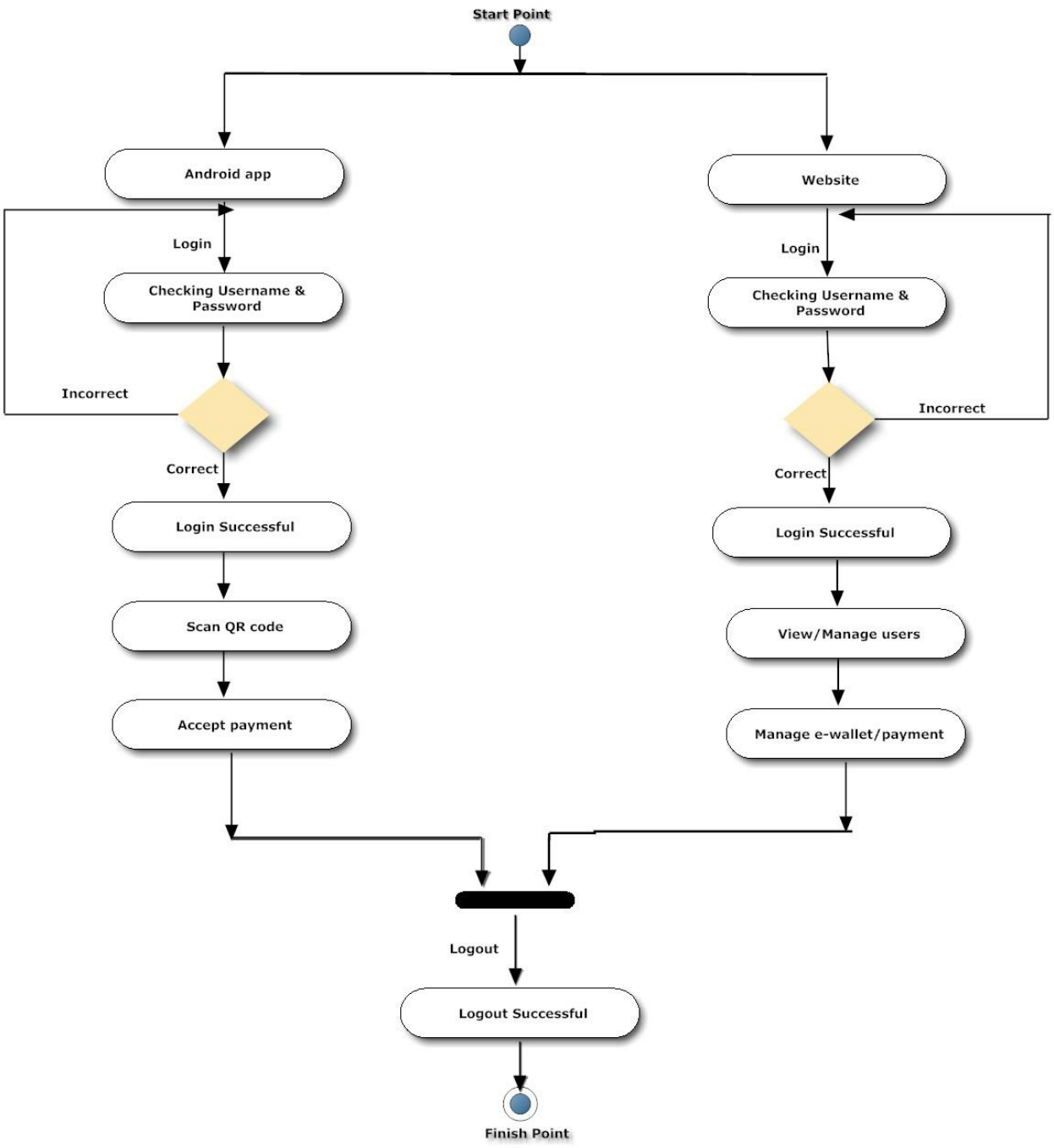

Fig. 3.1.2.1 Admin Activity Diagram

User Acitivity:

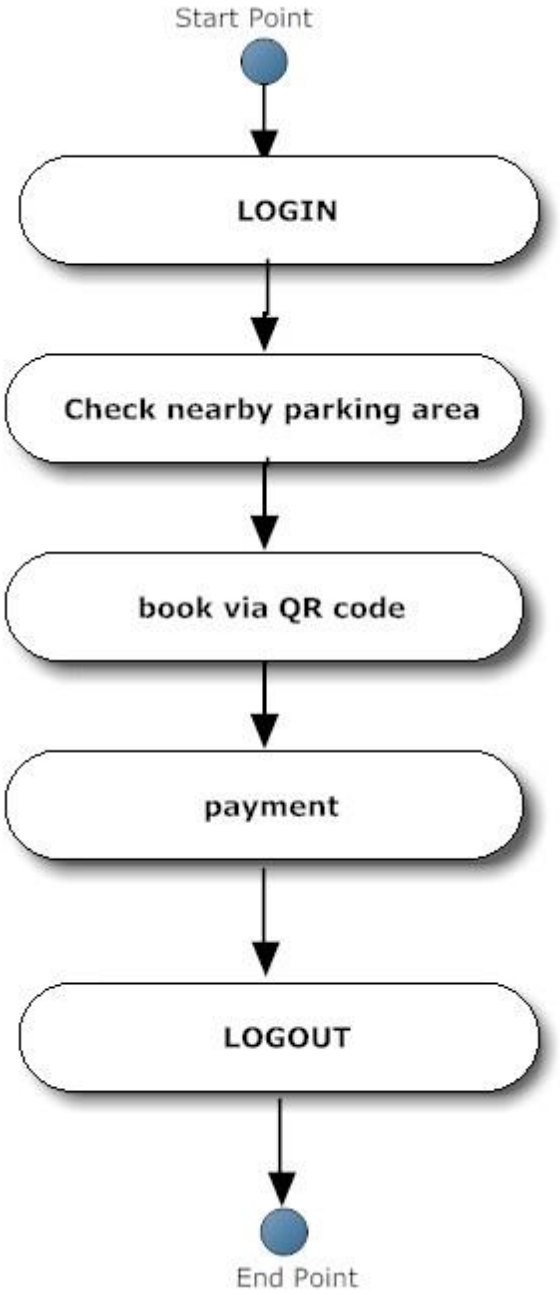

Fig. 3.1.2.2 User Activity Diagram

3.1.3 Sequence Diagram

Sequence diagrams present events during specific intervals shown chronologically along a line. These events may be historic, related to a specific criminal case or business development, or they may be important milestones in a project.Sequences are designed to provide a broad overview of a sequence of events in time. They don't go into detail, but links to events, information and images may be added as needed. A sequence diagram consists of a horizontal bar or line representing time progressing from left to right. This bar is marked with events or steps to indicate when they should or did happen.In project management, sequences are most useful for showing important milestones and deadlines.

A sequence provides a visual representation of events that helps you better understand history, a story, a process or any other form of an event sequence arranged in chronological order and displayed along a line (usually drawn left to right or top to bottom). It explains what happened during a certain period or to a particular person, starting with the earliest event and moving forward through time. Increasingly, sequences are illustrated in infographics combining text and graphic images for a better presentation.

A sequence is useful to document for any type of development, providing an easy-to-understand history and helping viewers to understand past and ongoing quickly.In project management, knowing how to create a project sequence is one of the most essential skills a project manager, as a sequence are most useful for showing important milestones, deadlines and other significant dates and events over the lifecycle of the project. Building comprehensive, accurate sequences will help a manager get every project off on the right foot.

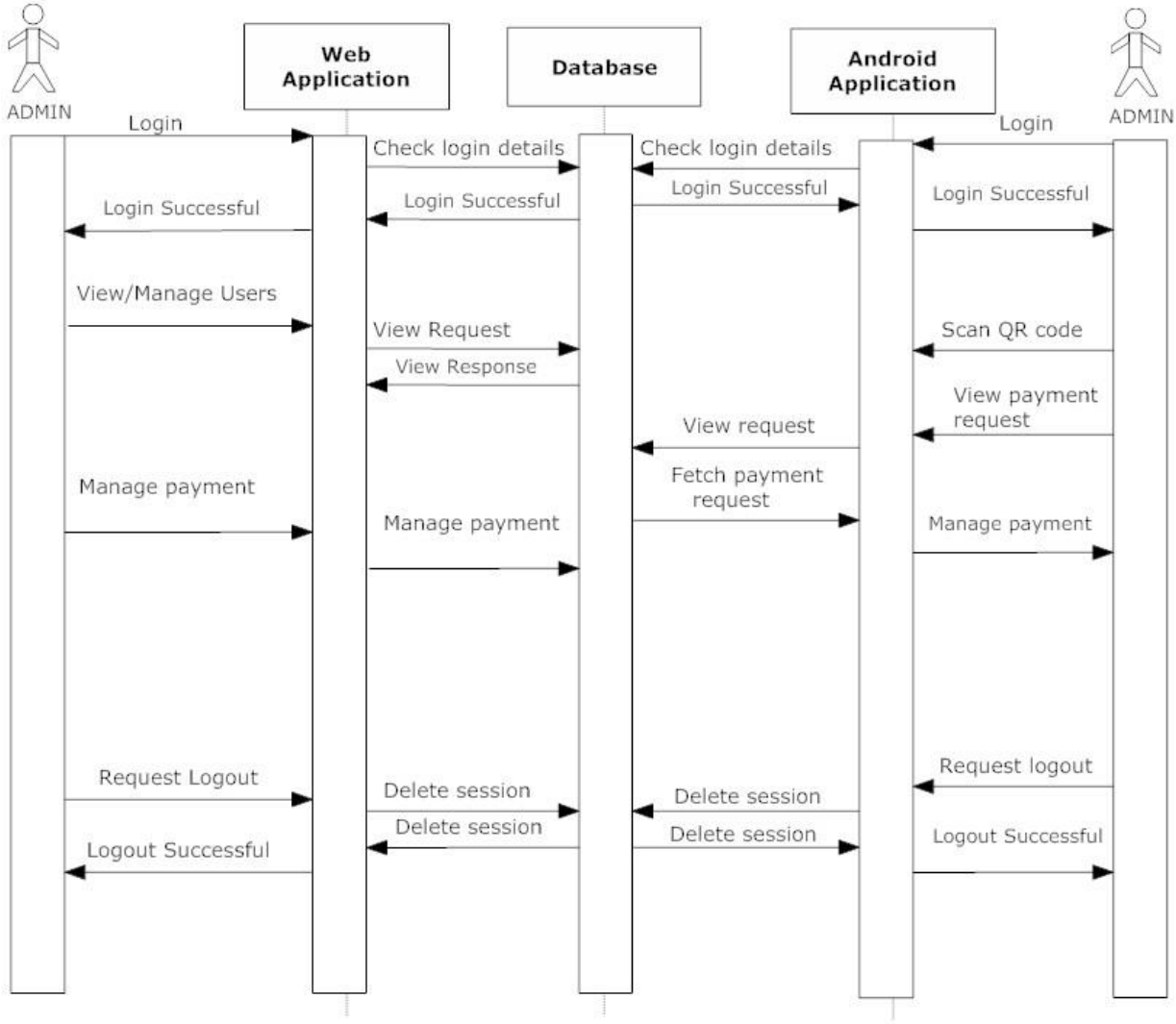

Fig. 3.1.3.1 Sequence Diagram

3.2 Data Flow Diagram

Data flow diagram is graphical representation of flow of data in an information system. It is capable of depicting incoming data flow, outgoing data flow and stored data. The DFD does not mention anything about how data flows through the system.There is a prominent difference between DFD and Flowchart. The flowchart depicts flow of control in program modules. DFDs depict flow of data in the system at various levels. DFD does not contain any control or branch elements.A data-flow diagram (DFD) is a way of representing a flow of a data of a process or a system (usually an information system). The DFD also provides information about the outputs and inputs of each entity and the process itself. A data-flow diagram has no control flow, there are no decision rules and no loops.

Depending on the methodology, DFD symbols vary slightly. However, the basic ideas remain the same. There are four basic elements of a data flow diagram: processes, data stores, external entities, and data flows.

Data Flow Diagrams are either Logical or Physical.

- **Logical DFD** - This type of DFD concentrates on the system process, and flow of data in the system.

E.x,In a Banking software system, how data is moved between different entities.

- **Physical DFD** - This type of DFD shows how the data flow is actually implemented in the system. It is more specific and close to the implementation.

The data-flow diagram is part of the structured-analysis modelling tools. When using UML, the activity diagram typically takes over the role of the data-flow diagram. A special form of data-flow plan is a site-oriented data-flow plan.Data-flow diagrams can be regarded as inverted Petri nets, because places in such networks correspond to the semantics of data memories.

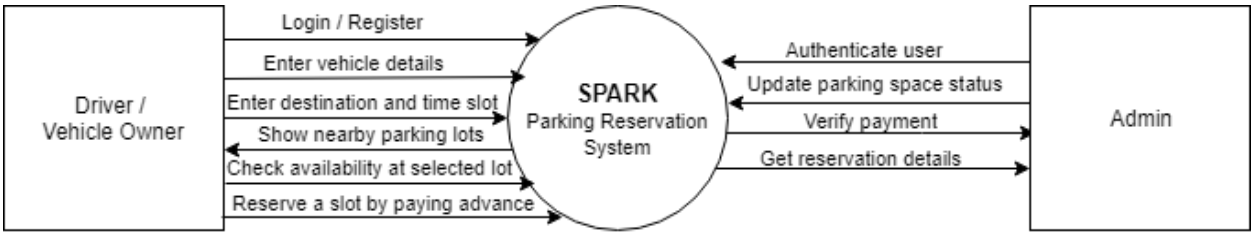

Fig. 3.2.1 DFD Level 0

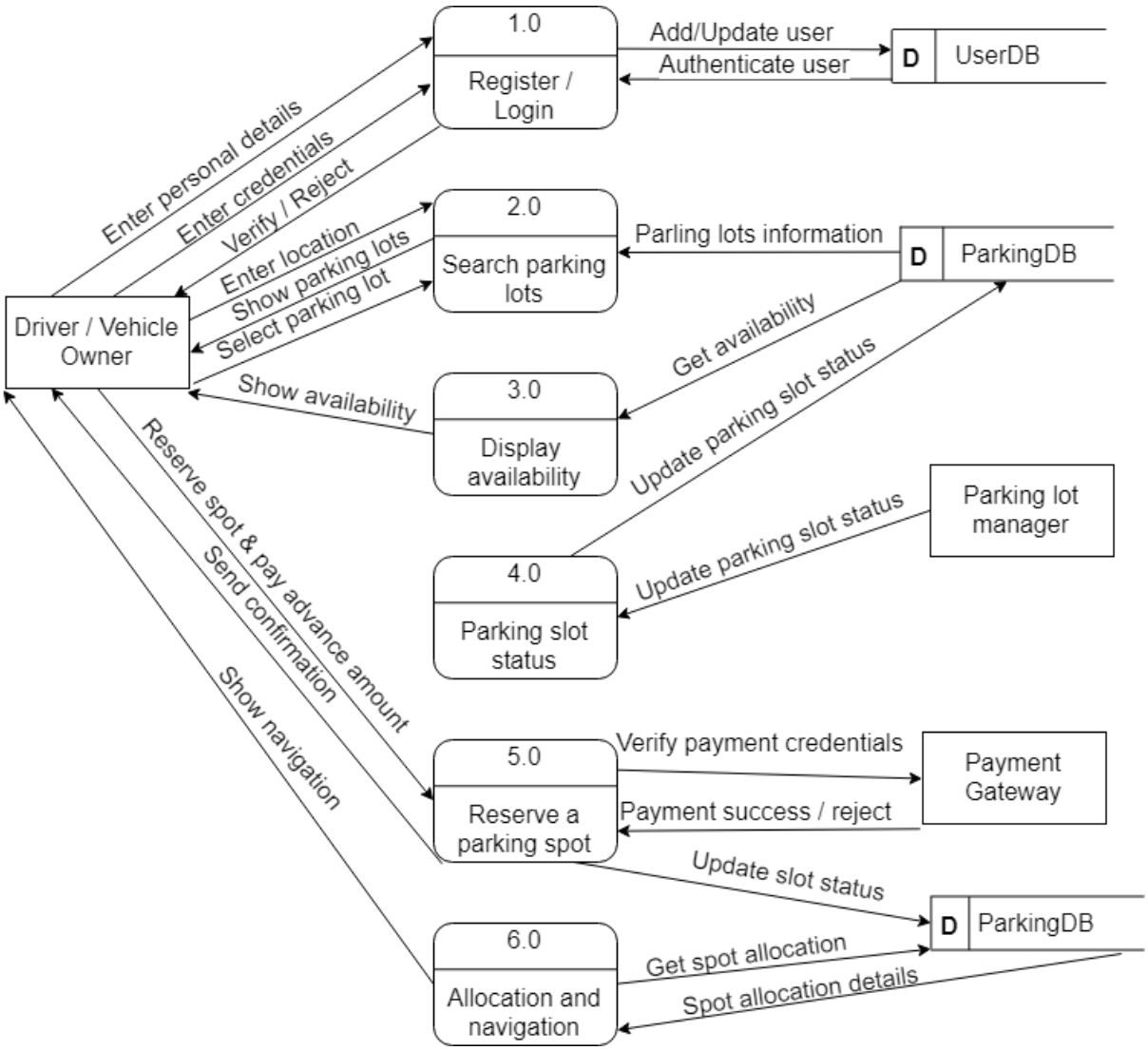

Fig. 3.2.1 DFD Level 1

IMPLEMENTATION

4.1. Proposed System

The proposed system is a parking reservation system that provides customers an easy way of reserving a parking space online. It overcomes the problem of finding a parking space in commercial areas that unnecessarily consumes time. The system is divided into two subsystems based on role of the user- an admin sub-system for management of the parking lot by the admin (or parking lot manager) and a customer subsystem where actual user (i.e. Drivers, Car owners, etc.) will use the application to reserve the parking spot.

Users would register with their basic information and contact details along with the their vehicle information in detail whether they have a two/four wheeler, if it's a four wheeler does it belong to the micro/mini/sedan segment to determine the empty space going to be required for its parking. Users will get a list of nearby parking lots available to them (or near the destination they are visiting), they can select one parking lot and the availability status of the parking spaces would be displayed to the users. Then according to their preference and vehicle type they can reserve a spot for the specific time. Users would be shown the estimated amount to be paid during checkout, users have to pay a small amount of money in advance to reserve a spot and the rest of the amount can be paid during their vacating of the parking spot allotted to them according to the time of usage of the parking spot. A QR code will be generated and sent to the user which will be scanned by the admin (or manager) at the parking lot for authentication of the user and spot can be accommodated by the user after verification. Verification is done by the admin system which will be used by the admin (or manager) and it will update the status of the given parking spot as 'occupied'.

When the user has to exit the parking lot he has to pay the amount excluding the amount paid during reservation of parking spot and including the penalty (if applicable, i.e. user will be penalized if he occupies the parking space above the specified time) after successful transaction user will get a new QR code which will be used to scan and exit at the checkout. User can pay the amount using cash or else he can pay using the online payment portal of the application. After successful scan of QR code, the status of the parking spot will be updated to 'available' again.

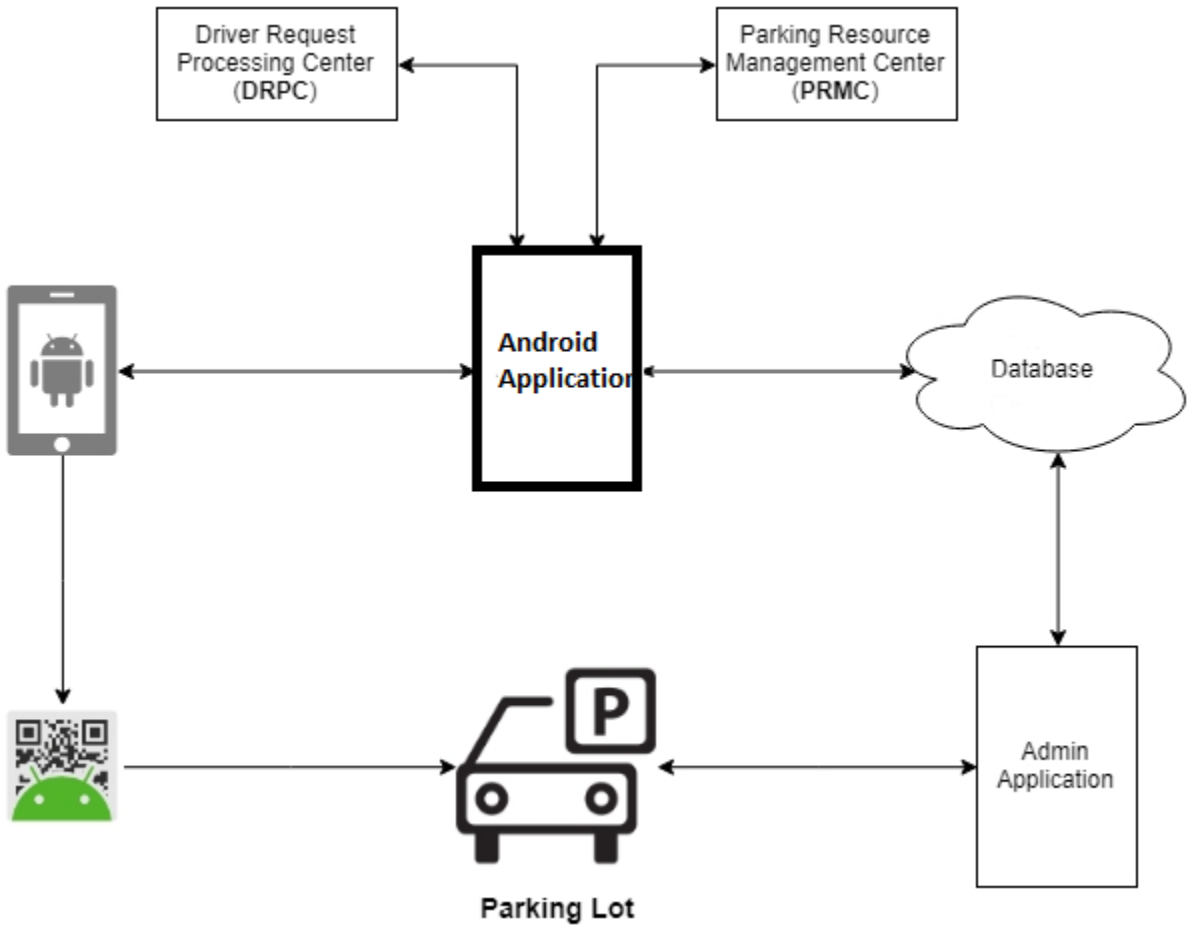

Fig. 4.1.1 Block Diagram of Proposed System

4.2 User Flow Diagram

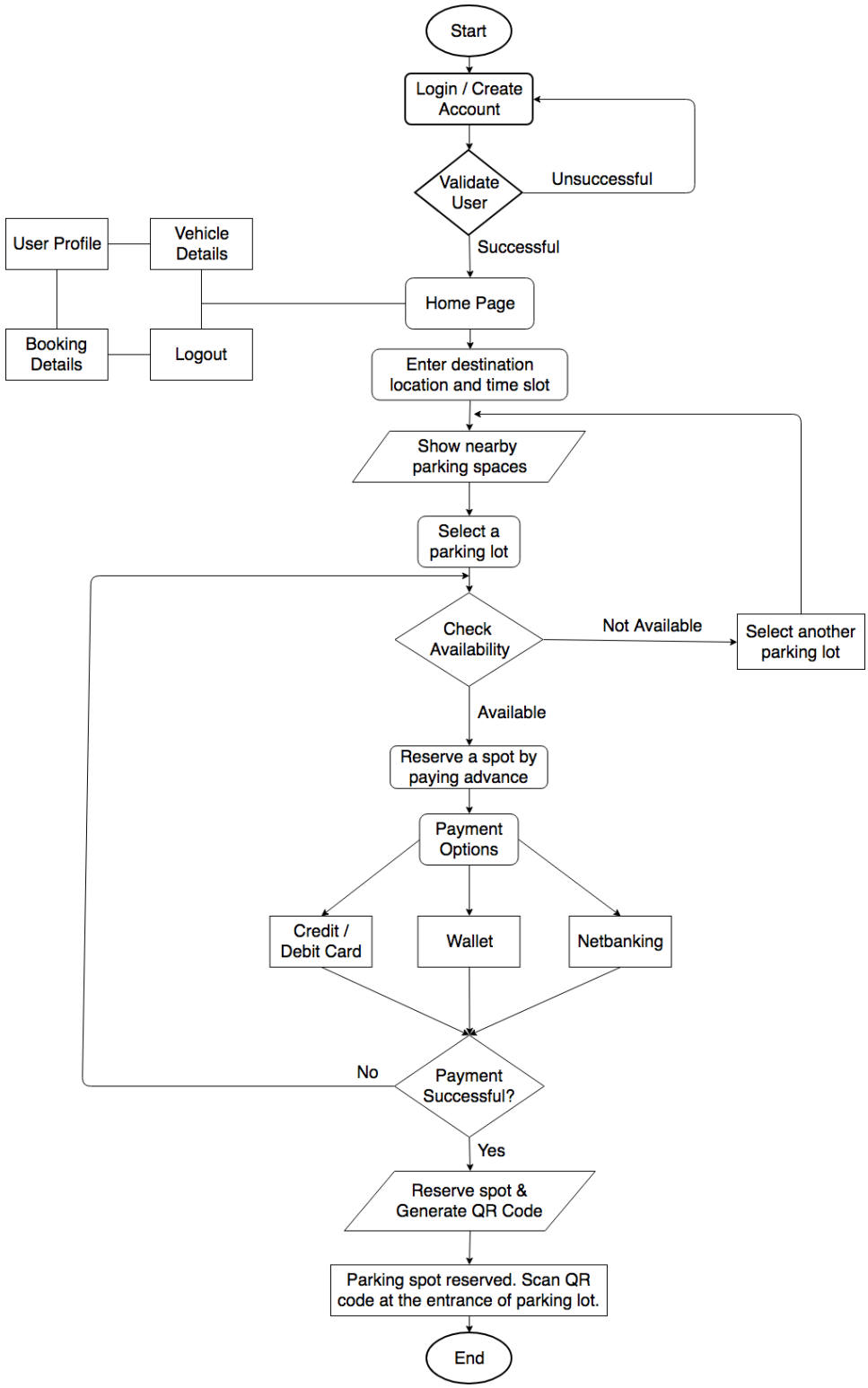

Fig. 4.2.1 User Flow Diagram

WORKING

5.1 System Architecture

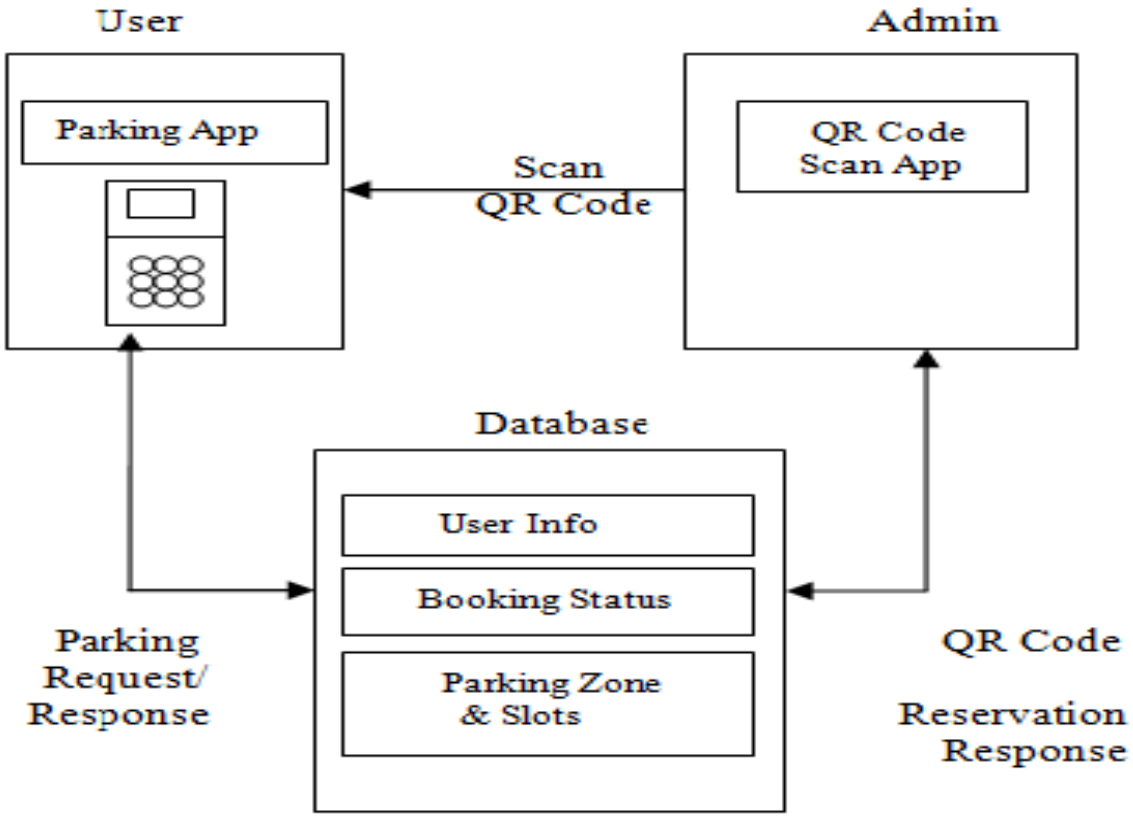

Fig. 5.1.1 System Architecture

Above figure shows three components in the parking model, including parking zones, users and the database. The management system determines the parking prices and broadcast lives parking availability information to users (also drivers). Upon receiving parking information, the user selects desired parking lot and reserves a space. As soon as user reserves a parking space, System generates a unique QR code and sends it to the user. As a result, the state of parking resources is changed by users parking decisions. The parking lot consists of a group of parking spaces. The state of a parking lot is the number of occupied spaces versus total spaces. Every parking lot has access to the Internet to communicate with the management system and users, and share parking information with other parking lots. In each parking lot, the reservation authority is deployed for authenticating the individual user's identity and reservation request.

The system is divided into two subsystems based on the role of the user which are:

1. Admin Subsystem (Used and Managed by Parking Lot manager / worker)
2. User / Customer Subsystem (Used by End-Users which are Drivers / Vehicle Owners who actually want to reserve a parking spot)

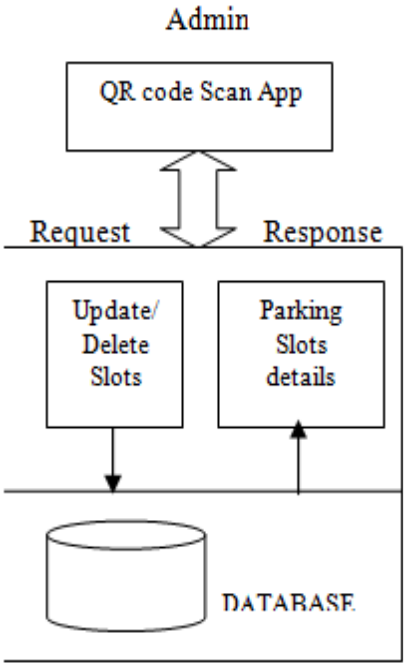

Fig. 5.1.2 Admin API

The Admin sub-system contains various modules as follows:

- Login

  It is an android base application, in this application the parking lot manager registers and enters various essential details related to parking lot. The managers will get their respective user id and password for accessing the module.

- Updating Status of Parking Lot

  When a vehicle enters into the parking and is being allocated a parking spot, the status of that parking spot is updated by the manager using the application so as to keep real-time status of the parking spot.

- Scanning the QR code

  When the customer (i.e. User using the Customer module) enters the parking lot, the manager scans the QR code for authenticating the user.

  Another QR code scan is done while the user exits the parking lot for confirmation of payment.

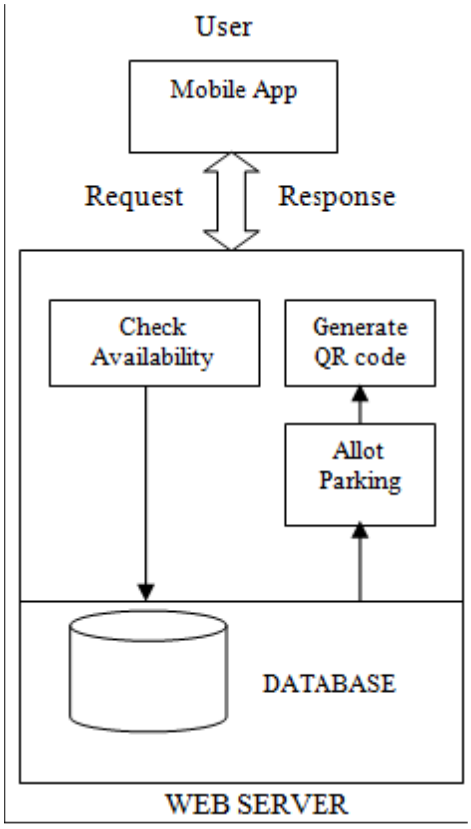

Fig. 5.1.3 User / Customer API

The Customer sub-system contains various modules as follows:

- User Registration / Login

  It is an android base application, in this application user need to register with user details like username, password, email Id, mobile number. After registration, user can be logging in the application by our registered mobile number and password.

- User profile:

  The profile of the user will be displayed such as Name, username, email, contact no, etc. with an option to make any changes if required.

- Parking availability check:

  User can click on the parking lot to view the availability. If a spot is available for the specified time slot given by the user then, user can be able to reserve a spot. Otherwise, another parking lot near the current parking lot would be suggested to the user.

- Book the parking spot

  The user can select any empty spot from available parking spots. Users will be given a list of parking lots available nearby current location or the location where the user is visiting. The reservations are done on the basis of time of arrival & time of departure (No. of hours parking lot is to be reserved for).
- Automated pricing model:

  Based on the time for which the user reserved the spot for is used to calculate the total cost for parking.
- Reservation status:

  After successful reservation of a parking spot the status and details of the reserved spot are displayed to the user.
- Navigation:

  This provides a guidance for user to navigate to the parking lot.
- QR Code Generation:

  After successful reservation, user will get a confirmation for their reserved parking spot and a QR code will be sent to user which will be used for authentication.
- QR Code Scan:

  User has to scan the QR code at parking lot to authenticate and accommodate the reserved spot as well as while exiting the parking lot.

5.2 Graphical User Interface

The graphical user interface (GUI) is a form of user interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, instead of text-based user interfaces, typed command labels or text navigation. GUIs were introduced in reaction to the perceived steep learning curve of command-line interfaces which require commands to be typed on a computer keyboard. The actions in a GUI are usually performed through direct manipulation of the graphical elements. Beyond computers, GUIs are used in many handheld mobile devices such as MP3 players, portable media players, gaming devices, smartphones and smaller household, office and industrial controls.

Our system interacts with its users through an Android Application hence we have added some of the screenshots of the application's GUI for better understanding.

SCREENSHOTS

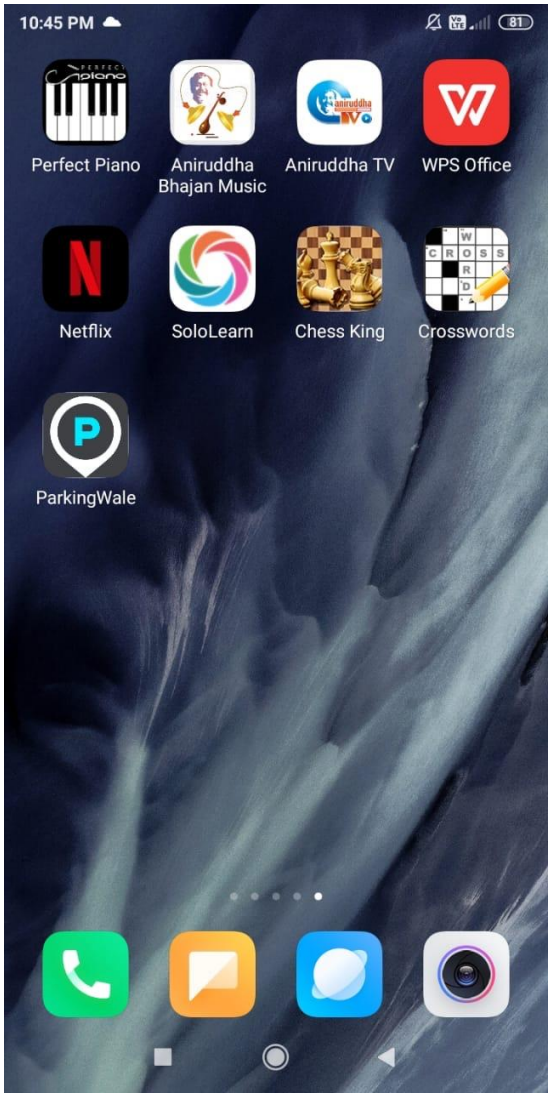

Fig. 5.2.1 HOME SCREEN

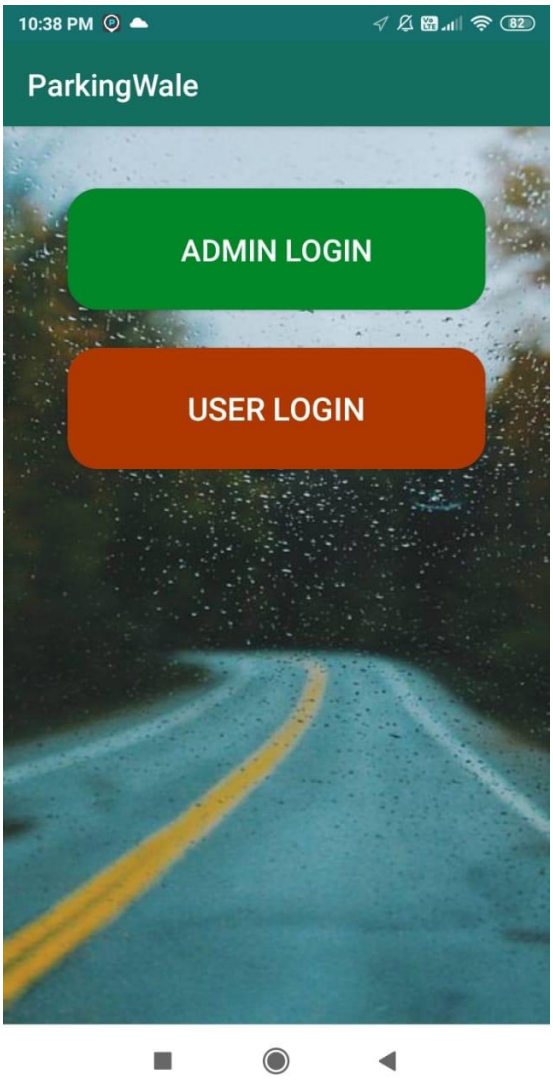

Fig. 5.2.2 MAIN SCREEN

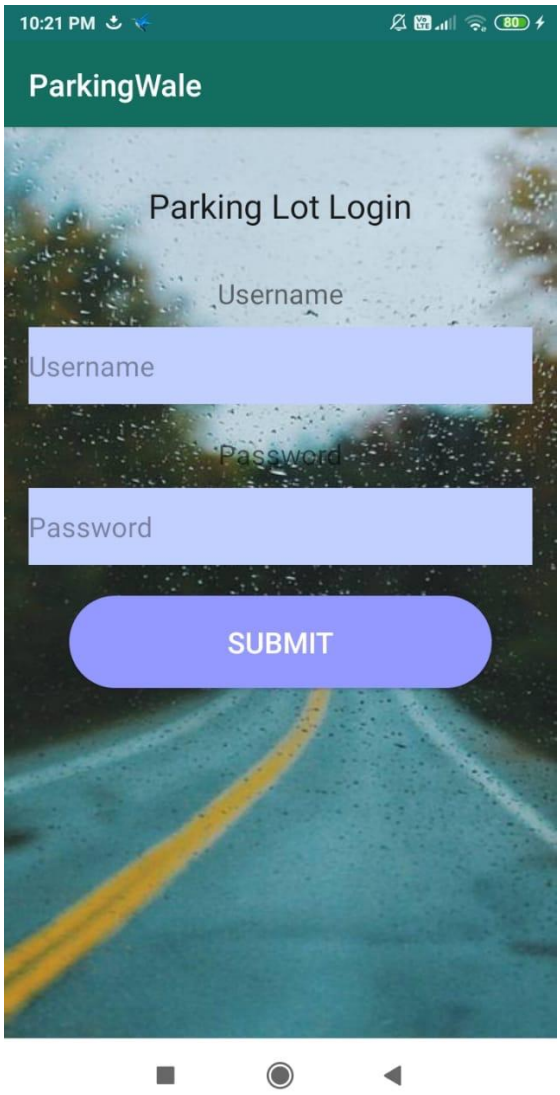

Fig. 5.2.3 Admin can login here using credentials

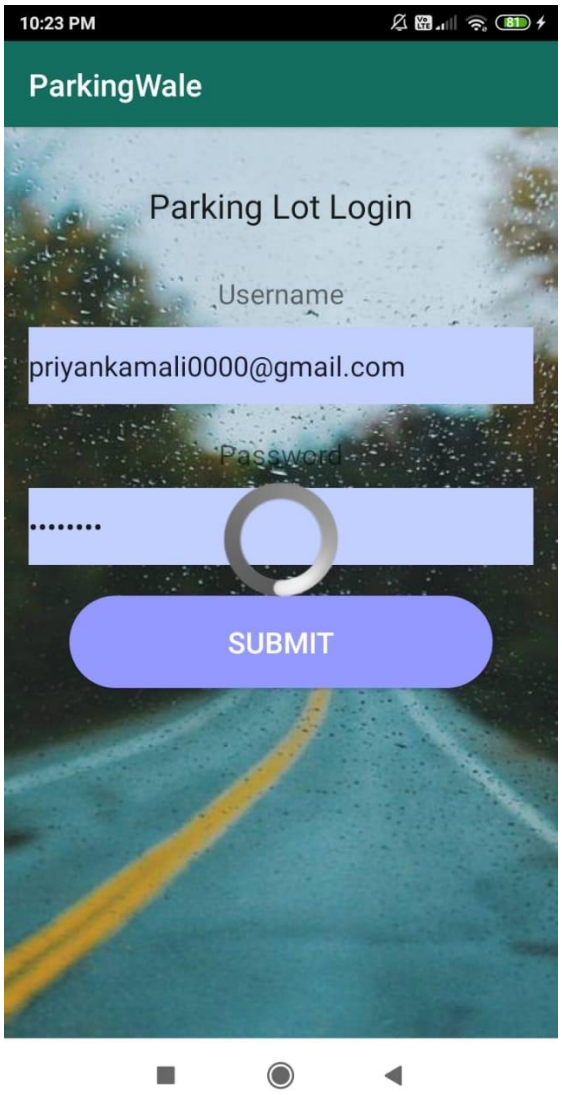

Fig. 5.2.4 Admin trying to login

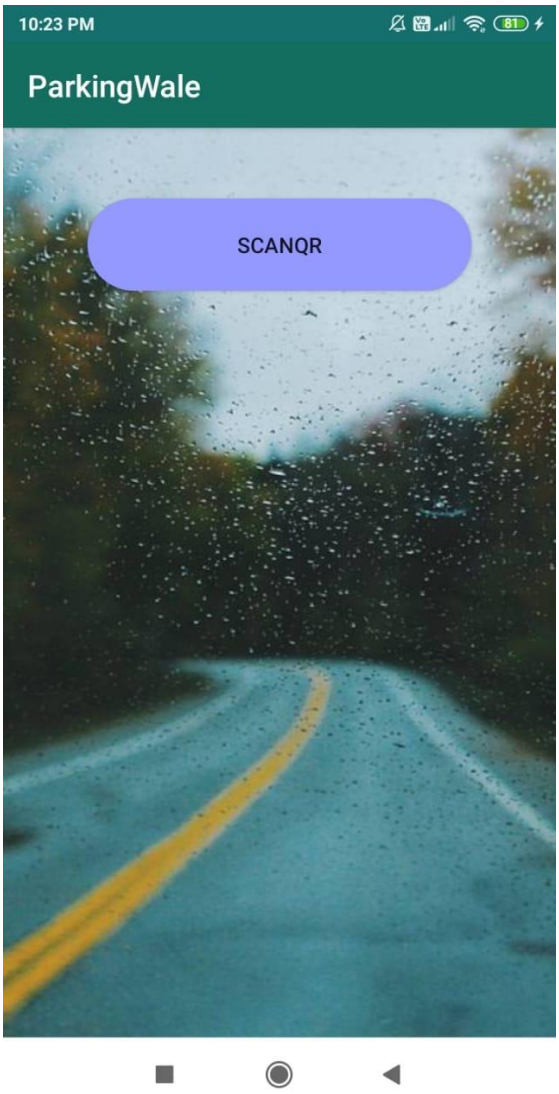

Fig. 5.2.5 Admin can scan the users QR code by clicking on scan QR

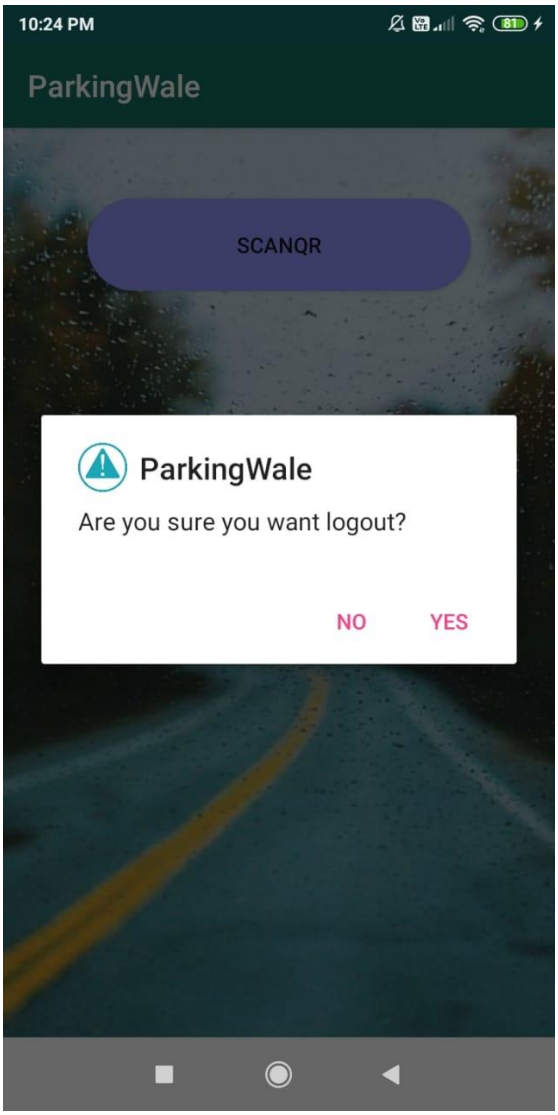

Fig. 5.2.6 Admin can logout after scanning the QR code

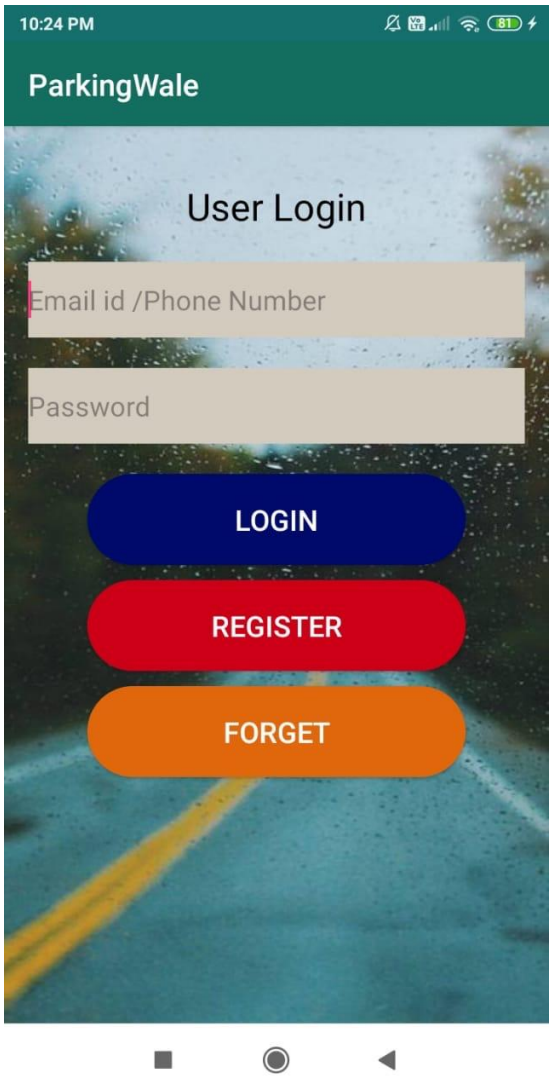

Fig. 5.2.7 User can login here and book the slot

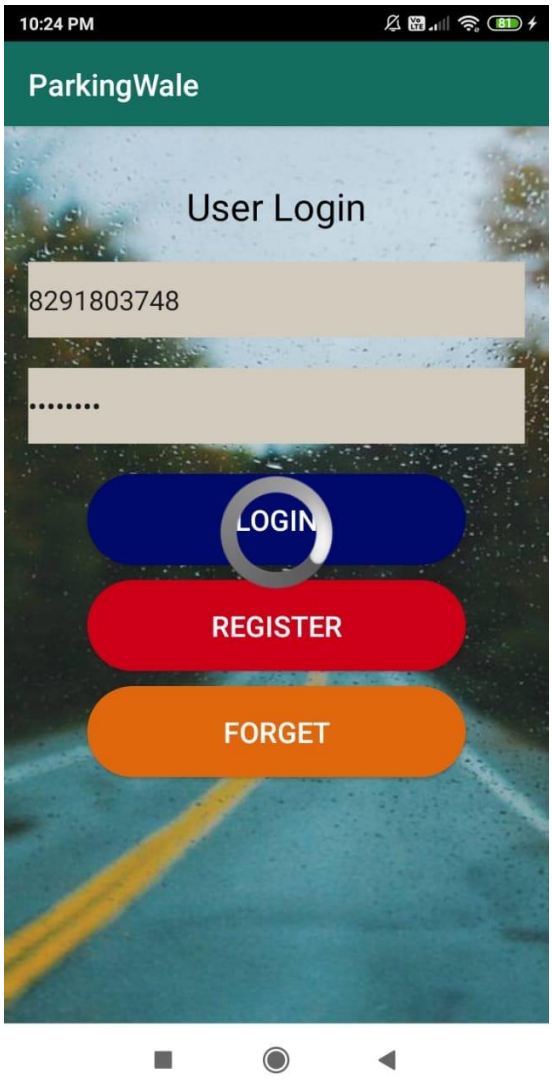

Fig. 5.2.8 User trying to login

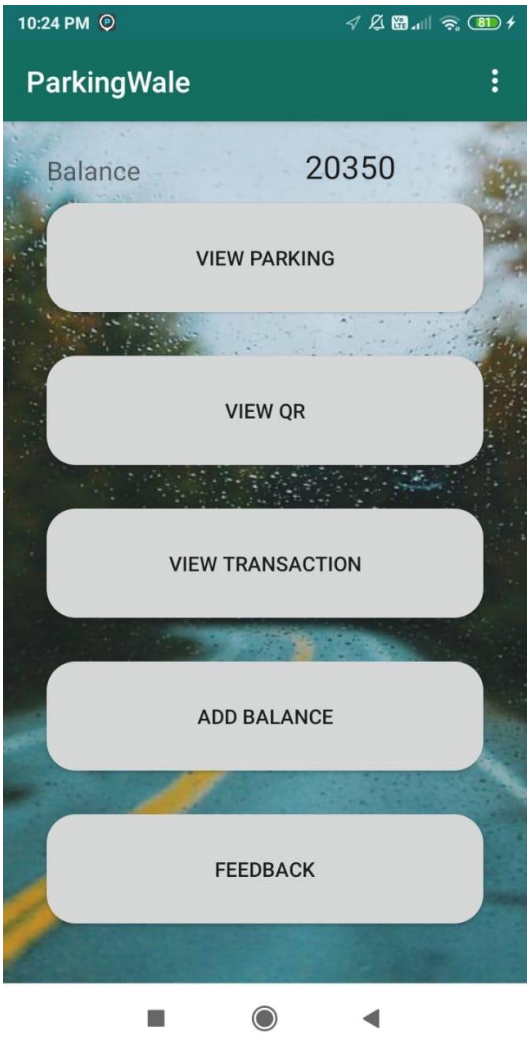

Fig. 5.2.9 User can book their slot and can add balance, view qr and transaction and can give feedback

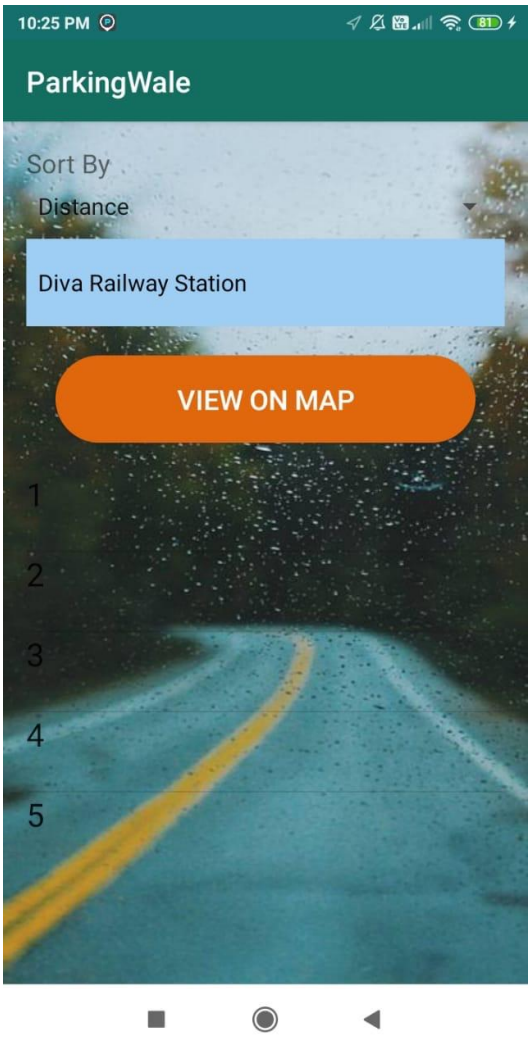

Fig. 5.2.10 User can select the slot and can view the route by clicking on view map

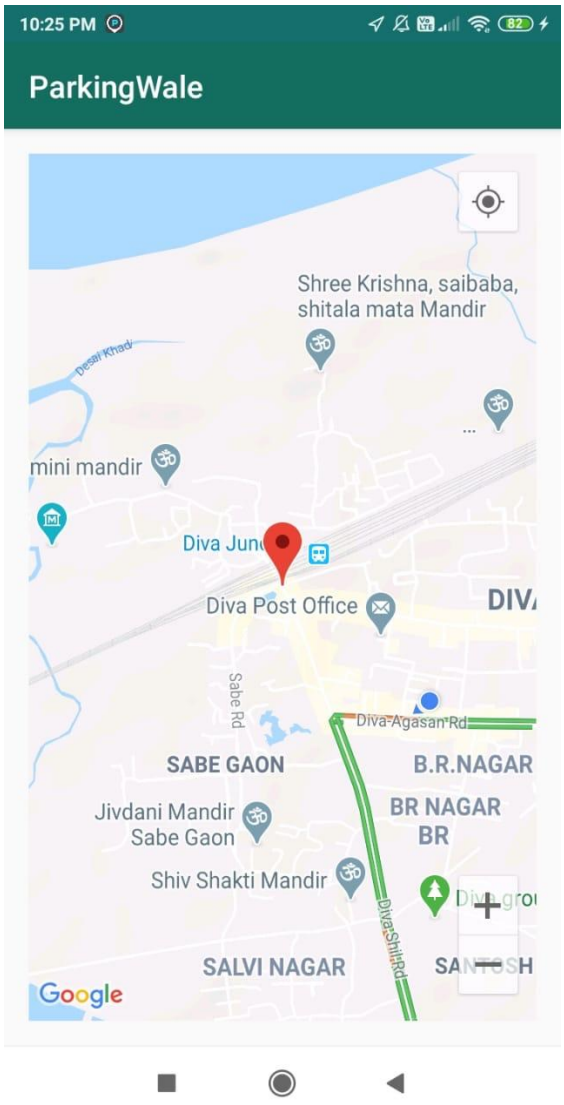

Fig. 5.2.11 MAP VIEW OF SLOT

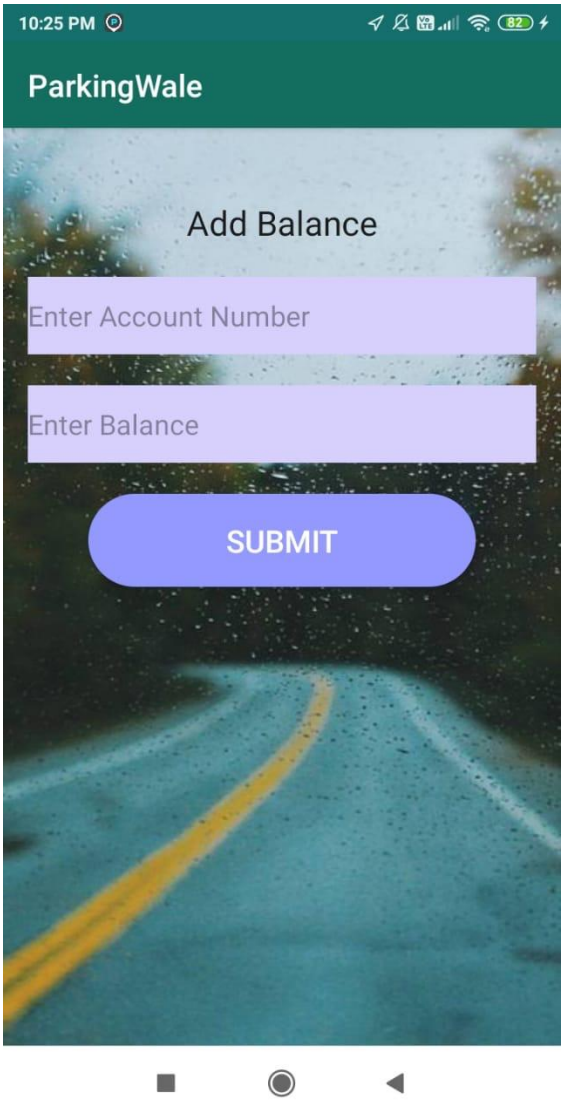

Fig. 5.2.12 User can add balance to their account

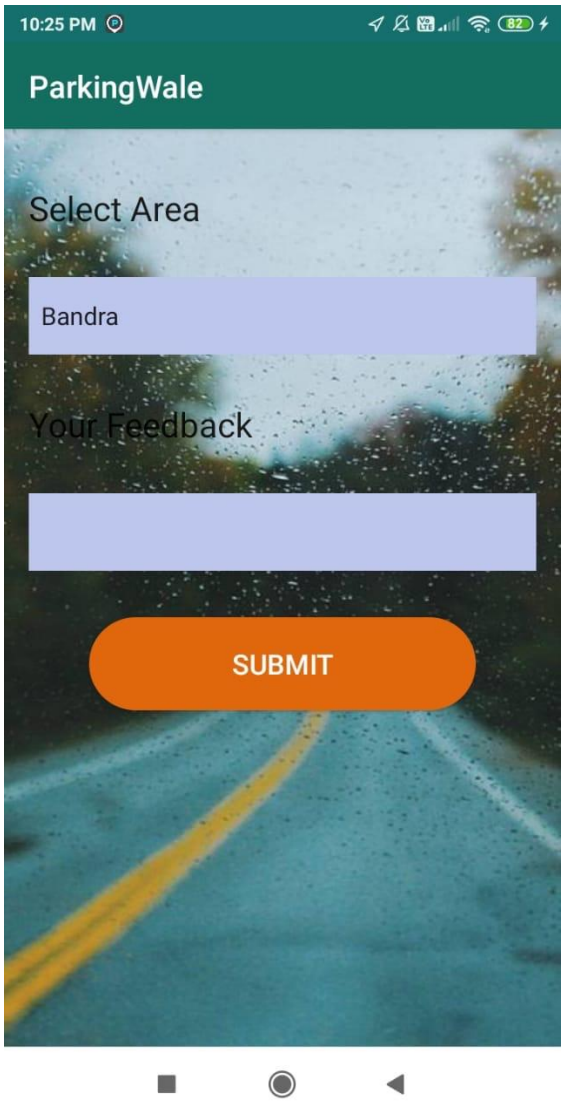

Fig. 5.2.13 User can give feedback about Application

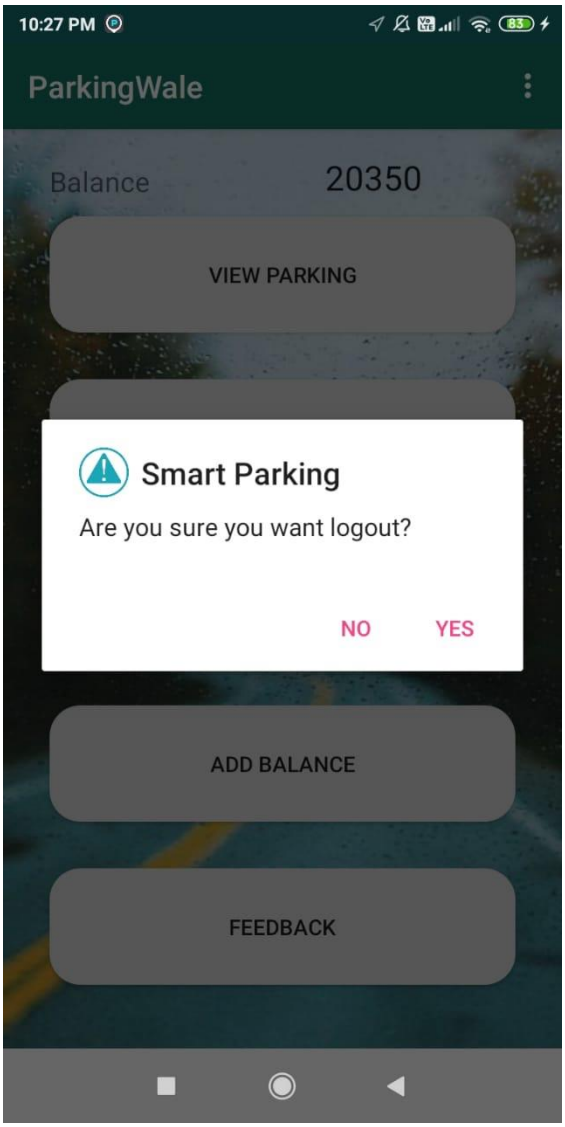

Fig. 5.2.14 User can successfully logouts after booking the slot

CONCLUSION

In this project, we have developed a new system of parking reservation. In this system, we implement parking reservation policy to balance the benefit of service providers and requirements from the users. Moreover, we have presented the detailed design, implementation and evaluation of the system. Based on the obtained results from our simulation study, we conclude that the proposed reservation-based smart parking system can alleviate traffic congestion caused parking searching and reduce the amount of traffic volume searching for parking.

The application provides a solution to overcome the problem of finding a parking space. The system implements a dynamic allocation of parking slots to the users as well as gives a pricing model based on the time slot required for parking. User authentication and payment authentication can be formulated using QR code present in the system. Due to real-time nature of database and use of cloud functions, the system has a real-time and updated status of parking lots. This system helps user to not only find and reserve a parking space but also to save time, fuel and energy.

Due to simplicity of GUI, portability of smartphones & android application and system being online, any registered user can login and use the system from anywhere at his/her own convenience.

REFERENCES

[1] R. Arnott, T. Rave, and R. Schöb, "Alleviating urban traffic congestion," MIT Press Books, The MIT Press, vol. 1, January 2005.

[2] Y. Geng, and C. Cassandras. "A new 'smart parking' system infrastructure and implementation," Procedia-Social and Behavioral Sciences, vol. 54, pp. 1278-1287, October 2012.

[3] H. Yang, W. Liu, X. Wang and X. Zhang, "On the morning commute problem with bottleneck congestion and parking space constraints," Transportation Research Part B: Methodological, vol. 58, pp. 106-118, December 2013.

[4] B. Zou, N. Kafle, O. Wolfson and J. Lin, "A mechanism design based approach to solving parking slot assignment in the information era," Transportation Research Part B: Methodological, vol. 81(Part 2), pp.631-653, November 2015.

[5] Z. Chen, Y. Yin, F. He, and L. Jane, "Parking reservation for managing downtown curbside parking," Transportation Research Record: Journal of the Transportation Research Board, vol. 2498, pp. 12-18, June 2015.

[6] F. Caicedo, C. Blazquez, and P. Miranda, "Prediction of parking space availability in real time," Expert Systems with Applications, 2012, vol. 39, pp. 7281-7290, June 2012.

[7] Y. Geng and C. Cassandras, "New smart parking system based on resource allocation and reservations," IEEE Trans. Intell. Transp. Syst., vol. 14, no. 3, pp. 1129–1139, Sep 2013.

[8] M. Karthi and Preethi Harris "Smart Parking with Reservation in Cloud Based Environment" e-ISBN: 978-1-5090-4573-0 PoD ISBN: 978-1-5090-4574-7 IEEE.

[9] "Solutions for Improving City Operations," http://www.streetlinenetworks.com/site/index.php

[10] https://www.researchgate.net/publication/224500383_SPARK_A_new_VANET-based_smart_parking_scheme_for_large_parking_lots