

# SMART PARKING WITH SMART TAP

A

Major Project Report

*Submitted in partial fulfillment of the requirement for the award of  
Degree of*

**BACHELOR OF ENGINEERING**

In

**COMPUTER SCIENCE**

Submitted to



**RAJIV GANDHI PRADYOGIKI VISHWA VIDHYALAYA,  
BHOPAL (M.P.)**

Submitted by

**Kajal Verma 0225cs151029**

**Simranjeet Kaur 0225cs151055**

**Richa Vishwakarma 0225cs151044**

**Adarsh Mehta 0225cs151003**

**Anurag Soni 0225cs151016**

Under the Supervision of

**Prof. Satpal Singh**

Assistant Professor, Computer Science  
GNCSGI, Jabalpur



**DEPARTMENT OF COMPUTER SCIENCE  
GLOBAL INSTITUTE OF ENGINEERING, JABALPUR  
Session 2015-2019**

# **GLOBAL INSTITUTE OF ENGINEERING, JABALPUR**



## **DEPARTMENT OF COMPUTER SCIENCE**

### **CERTIFICATE**

This is to certify that that **Kajal Verma 0225cs151029, Simranjeet Kaur-0225cs151055, Richa Vishwakarma 0225cs151044, Adarsh Mehta 0225cs151003, Anurag Soni 0225cs151016** of B.E Fourth Year, **Computer Science** have completed their minor project entitled **“Smart Parking With Smart Tap”** during the year 2018 under my guidance and supervision.

I approve the project for the submission for the partial fulfillment of the requirement for the award of degree **Bachelor of Engineering in Computer Science.**

Guided & Approved by:

**Prof. Sumit Nema**  
HOD, CS Department  
GNCSGI, Jabalpur

**Prof. Satpal Singh**  
Assistant Professor, Computer Science  
GNCSGI, Jabalpur

# **GLOBAL INSTITUTE OF ENGINEERING, JABALPUR**



**DEPARTMENT OF COMPUTER SCIENCE**

## **DECLARATION BY CANDIDATE**

I, **Kajal Verma, Simranjeet Kaur, Richa Vishwakarma, Adarsh Mehta, Anurag Soni** students of Bachelor of Engineering, Computer Science Branch, Global Institute of Engineering, Jabalpur hereby declare that the work presented in this Major project is outcome of our own work, is bonafide, correct to the best of my knowledge and this work has been carried out taking care of Engineering Ethics. The work presented does not infringe any patented work and has not been submitted to any University for the award of any degree or professional diploma.

**Kajal Verma-0225cs151029**

**Simranjeet Kaur-0225cs151055**

**Richa Vishwakarma-0225cs151044**

**Adarsh Mehta-0225cs151003**

**Anurag Soni-0225cs151016**

## **ACKNOWLEDGEMENT**

At the outset, we would like to thank our guide and advisor, **Prof. Satpal Singh, Assistant Professor, Computer Science** for giving us an opportunity to work on this challenging topic and providing us ample and valuable guidance throughout the Project.

We also greatly indebted to **Prof. Sumit Nema, CSE HOD, Computer Science, Global Engineering College**, without his encouragement and constant guidance we could not have finished this project. He was always a source of inspiration and motivator for innovative ideas during the entire span of this work.

We equally grateful to **Dr. Rajeev Khatri, Director, GNCSGI**, for providing all the necessary resources to carry out this Project work. We would like to thank all staff members of CS, GNCSGI. and friends, for their support.

We will be failing in our duty if we don't acknowledge the people behind this work to give us moral and psychological support. We hereby, would like to thank our parents for their endless care and constant support throughout our project work.

Kajal Verma-0225cs151029  
Simranjeet Kaur-0225cs151055  
Richa Vishwakarma-0225cs151044  
Adarsh Mehta-0225cs151003  
Anurag Soni-0225cs151016

# GLOBAL INSTITUTE OF ENGINEERING, JABALPUR



## DEPARTMENT OF INFORMATION TECHNOLOGY

### **CERTIFICATE OF APPROVAL**

The Project work entitled “**Smart Parking**” being submitted by **Kajal Verma-0225cs151029** **Simranjeet Kaur-0225cs151055**, **Richa Vishwakarma-0225cs151044**, **Adarsh Mehta-0225cs151003**, **Anurag Soni-0225cs151016** have been examined by us and is hereby approved for the award of degree “**Bachelor of Engineering Computer Science**”.

(Internal Examiner)

(External Examiner)

Date:

Date:

## **ABSTRACT**

Smart parking with smart tap is a concept, which will be going to solve number of problems related to parking (such as finding parking slot in time and slot booking in advance) and problem faced during accidents (such as unavailability of an information of a driver and his vehicle).

Smart parking with smart tap is a concept which is divided into two parts. In first part it will solve the problem of parking, by this user can see availability of parking slot and can also book parking slot in advance with the help of an app to save time and energy.

In second part it will look for problems occur during an accidents, we will offer each and every license holding citizen an RFID card which will have complete information of that user and his vehicle, therefore during an accident with a simple tap it will send alert with complete information of that user to the nearest police station and in registered mobile number.

It will minimize the searching time of a citizen and help to take action in time during accidents. Smart parking with smart tap will solve both the problems of parking and unavailability of an information of a driver during accidents. It will minimise the action time in both the cases efficiently.

# CONTENTS

## **1. Chapter 1 : Introduction**

1.1. Background

1.2. Objective

1.3. Purpose, Scope and Applicability

## **2. Chapter 2: Survey of Technologies**

## **3. Chapter 3: Requirement and Analysis**

3.1. Problem Definition

3.2. Identification of Need

3.3. Project Planning & Scheduling

3.4. Software Requirement Specification ( SRS)

## **4. Chapter 4 : System Design**

4.1

Control Flow Diagram

4.1.1. Data Flow Diagram

4.1.2. Entity Relationship Diagram

4.1.3. State Diagram

4.1.4. Class Diagram

4.1.5. Collaboration Diagram

4.1.6. Use Case Diagram

## **5. Chapter 5 : Implementation**

## **6. Chapter 6 :Testing Approach**

## **7. Chapter 7 : Results and Discussion**

## **8. Chapter 8 : Conclusions**

8.1. Limitations of the System

8.2. Future Scope and Further Enhancement of the Project

## **9. Chapter 9: References**

# CHAPTER 1 INTRODUCTION

## 1.1 Background

Moving towards smart city application, smart parking is a good example for a common citizen of how the Internet-of-Things (IoT) will be effectively and efficiently used in our daily living environments to provide different services to different users. Any citizen may use his mobile device, a computer having Internet can access website from anywhere in the world to find a free parking lot in the city and get to know which parking lot is still available. It provide efficient car parking management through remote parking slot localization and fast car retrieval and also it provides security.

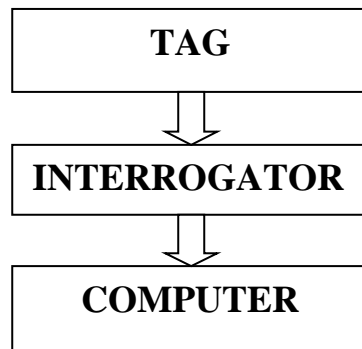
The process of aligning cars in a sequential form in a registered piece of land is called Parking. The Raspberry Pi is a low cost, credit-card sized computer that plugs into a computer monitor or TV, and uses a standard keyboard and mouse. It is a capable little device that enables people of all ages to explore computing, and to learn how to program in languages like Scratch and Python. It's capable of doing everything you'd expect a desktop computer to do, from browsing the internet and playing high-definition video, to making spreadsheets, word-processing, and playing games. The Raspberry pi has the ability to interact with the outside world.

An infrared sensor [2] is an electronic device that emits in order to sense some aspects of the surroundings. An IR sensor can measure the heat of an object as well as detects the motion

A servo motor [3] is an electrical device which can push or rotate an object with great precision. It is just made up of simple motor which run through servo mechanism.

Radio-Frequency Identification (RFID) [4] the use of radio waves (as shown in the fig 1.) to read and capture information stored on a tag attached to an object. A tag can be read from up to several feet away and does not need to be within direct line-of-sight of the reader to be tracked





## **1.2 Objective**

As population is increasing, number of vehicles are also increasing and the problem of proper management of vehicle parking also increases simultaneously the problem of late responses to a person when in need during emergency situation also increases. To overcome both the problem of improper management of parking of vehicle and providing help to a needy person in a correct time, we have introduced the concept of smart parking with smart tap.

Smart parking with smart tap is basically divided into two parts. First part consist of hardware interfacing with an android application and in second part it is solving the problem of late responses to a person when in need during emergency situation with the help of the above mentioned application alone interfaced with global tracking system provided by Google itself.

As discussed above first part is a hardware module interfaced with an android application, consists of following hardware components and they are:- a raspberry pi 3, infrared sensors, wires for connectivity, light bulbs for demonstrating status of particular slots in real world, servo motor to control barricade and a RFID system( which includes both RFID receiver and RFID tag).

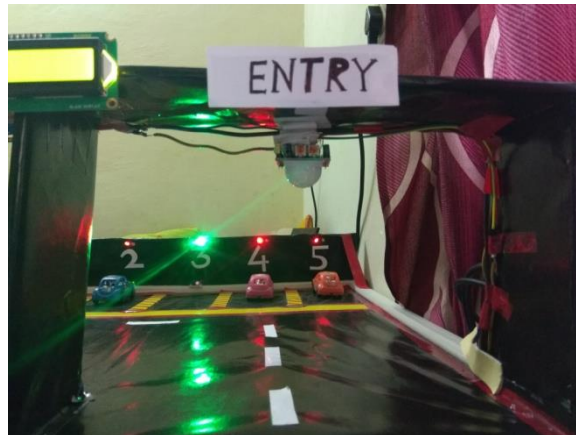
And in second module we are using an android application using Google tracking system.

## **1.3 Purpose,scope and availability**

Smart parking with smart tap is basically divided into two parts. First part consist of hardware interfacing with an android application and in second part it is solving the problem of late responses to a person when in need during emergency situation with the help of the above mentioned application alone interfaced with global tracking system provided by Google itself.

As discussed above first part is a hardware module interfaced with an android application, consists of following hardware components and they are:- a raspberry pi 3, infrared sensors, wires for connectivity, light bulbs for demonstrating status of particular slots in real world, servo motor to control barricade and a RFID system( which includes both RFID receiver and RFID tag).

And in second module we are using an android application using Google tracking system.



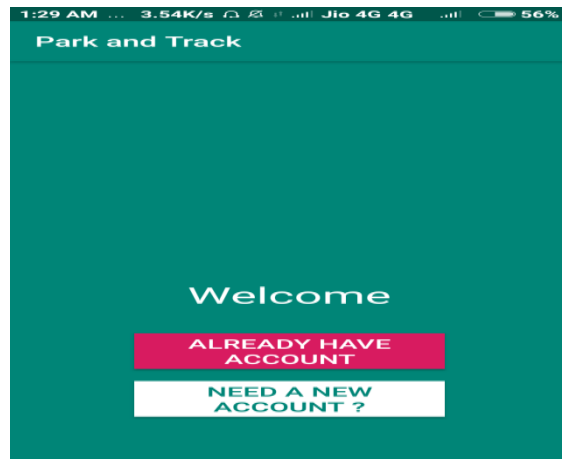
We have divided the whole process that is task in five steps, they are as follows:-

Hardware implementation: - firstly (as shown in the fig 2.) the hardware model is made. All the connections to the breadboard are made with the help of jumpers and these connections then are fed as an input to GPIO pins of raspberry pi which then gives the required output.

Sense the intruder: - The next we have used IR sensors which are used to sense the object placed in the parking slot. For demonstration we have used few parking slots. If the car comes in the parking slot the IR senses the object and the output is displayed by red light bulb that means if the car is parked red light gets activated and if not green light is activated.

Build an android application: - with the help of an android studio and firebase an app is developed. There are total seven pages.

The first page is the home page (fig 3.), where user can either create an account or login into an account..



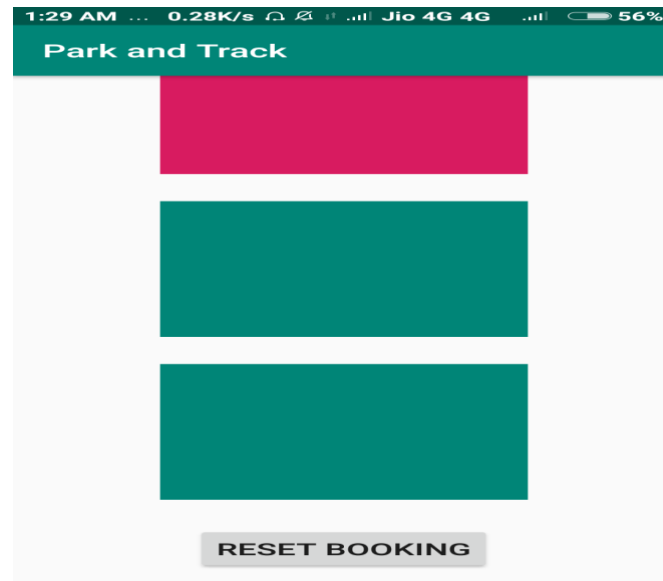
Home page of an android application

If a user does not have his account, he will create an account first with the help of an option “need a new account

Registration form

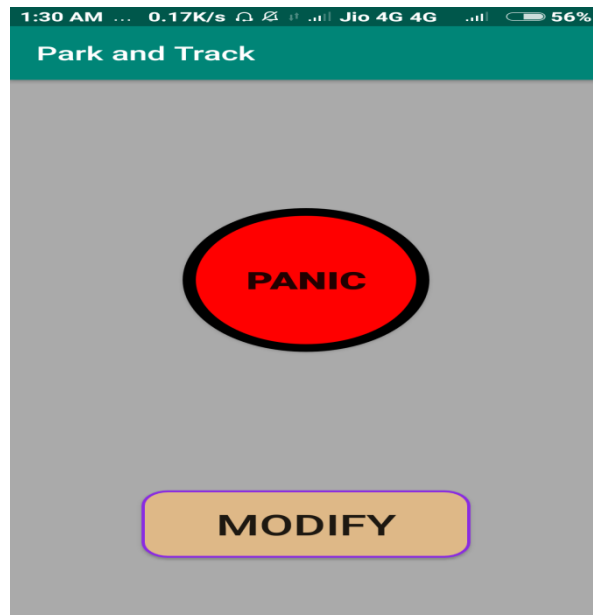
If the user has an account already then he will login directly. When the user login, his/her information get stored in the database. Then after logging in the user moves to the next page where user gets two options first is panic and second is parking. When user taps on parking, user moves to the status page where

the status of the whole parking system is displayed in the parking slots. The slots which are filled they will be shown in red colour while those which are empty that means will be shown in green colour (fig 5.). And by just a single tap upon parking slot user can also book slot before reaching to a parking slot. You can also change or cancel your booking just by tapping on a reset button provided on the same page on an android app.



Demonstration of parking slot

Second module: - second module is prepared to solve the problem of late response to the user when in need during an emergency situation. If we have put a panic button (fig 6.) in our app. This panic button will work to help us in the right moment on trouble



User interface of panic function

As soon as we press the panic button .It sends alert on our three registered number as well as nearest police station that we are in trouble.

A screenshot of the "Park and Track" app's registration form. The form has a light gray background. At the top, there is a teal header bar with the text "Park and Track". Below the header, the form contains the following fields: "Your Name" with a text input field containing the placeholder "Enter your name..."; "Contact No." with three separate text input fields labeled "First No.", "Second No.", and "Third No."; and a teal "SAVE" button at the bottom. The top status bar shows the time as 1:29 AM, a data speed of 0.00K/s, and a battery level of 56%.

Registration form to perform panic functionality

For the first time if you are using the app then you have to fill the form.The form is made to register the numbers. This will be the number you can contact at the time of the trouble.

As a result we can see current status a parking slot from anywhere with the help of our app as shown in fig 8. The slots which are filled or booked they will be shown in red colour while those which are empty that means will be shown in green colour

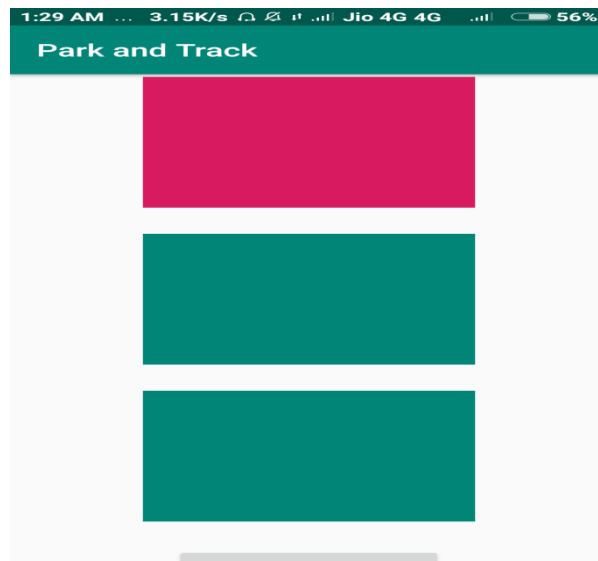
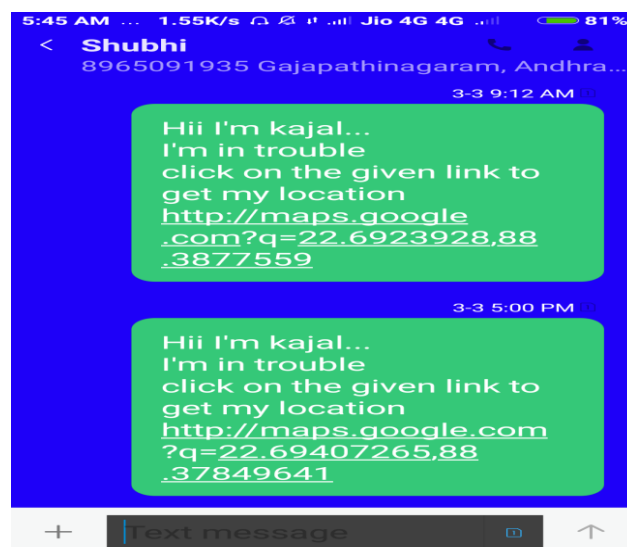


Fig 8. Status of parking slot

As soon as you press the panic button it sends the message to the three registered number. The message would look something like this



Result of panic alert

## Chapter 2 Survey of technologies

Internet of Things (IOT) plays a vital role in connecting the surrounding environmental things to the network and made easy to access those un-internet things from any remote location. It's inevitable for the people to update with the growing technology. And generally people are facing problems on parking vehicles in parking slots in a city. In this study we design a Smart Parking System (SPS) which enables the user to find the nearest parking area and gives availability of parking slots in that respective parking area. And it mainly focus on reducing the time in finding the parking lots and also it avoids the unnecessary travelling through filled parking lots in a parking area. Thus it reduces the fuel consumption which in turn reduces carbon footprints in an atmosphere. Our system is a Raspberry pi based parking system and the implementation is through infrared sensors connected to the GPIO pins of the raspberry pi to detect the empty parking spaces and sends this data to server, this stored data is accessed by users. This enhances the user to check the status/availability of parking spaces before setting their journey. Here the challenge is to use the existing resources in optimum level to reduce the searching time, traffic congestion in the city. The Raspberry Pi 3 is here. The latest version of the credit card-sized computer comes a whole year after the launch of its predecessor, the Raspberry Pi 2 and four years after the launch of the original Raspberry Pi, has its own set of upgrades. Here's the highlights: it's faster, it has the exact same form-factor, and yes, it has wireless on it.

The Raspberry Pi is a series of small single-board computers developed in the United Kingdom by the Raspberry Pi Foundation to promote the teaching of basic computer science in schools and in developing countries.

Several generations of Raspberry Pis have been released. All models feature a Broadcom system on a chip (SoC) with an integrated ARM compatible central processing unit (CPU) and on-chip graphics processing unit (GPU).

Processor speed ranges from 700 MHz to 1.4 GHz for the Pi 3 Model B+; on-board memory ranges from 256 MB to 1 GB RAM. Secure Digital (SD) cards are used to store the operating system and program memory in either SDHC or MicroSDHC sizes. The boards have one to four USB ports. For video output, HDMI and composite video are supported, with a standard 3.5 mm phono jack for audio output. Lower-level output is provided by a number of GPIO pins which support common protocols like I<sup>2</sup>C. The B-models have

an 8P8C Ethernet port and the Pi 3 and Pi Zero W have on-board Wi-Fi 802.11n and Bluetooth. Raspberry Pi 3 Model B was released in February 2016 with a 64 bit quad core processor, and has on-board WiFi, Bluetooth and USB boot capabilities. On Pi Day 2018 model 3B+ appeared with a faster 1.4 GHz processor and a 3 times faster network based on gigabit ethernet (300 Mbit / s) or 2.4 / 5 GHz dual-band Wi-Fi (100 Mbit / s). Other options are: Power over Ethernet (PoE), USB boot and network boot (an SD card is no longer required). This allows the use of the Pi in hard-to-reach places (possibly without electricity).

The Raspberry Pi Foundation recommends the use of Raspbian, a Debian-based Linux operating system. Other third-party operating systems available via the official website include Ubuntu MATE, Windows 10 IoT Core, RISC OS and specialised distributions for the Kodi media centre and classroom management. Many other operating systems can also run on the Raspberry Pi.

## **CHAPTER 3 REQUIREMENT AND ANALYSIS**

### **3.1 Problem definition**

Internet of Things (IOT) plays a vital role in connecting the surrounding environmental things to the network and made easy to access those un-internet things from any remote location. It's inevitable for the people to update with the growing technology. And generally people are facing problems on parking vehicles in parking slots in a city. In this study we design a Smart Parking System (SPS) which enables the user to find the nearest parking area and gives availability of parking slots in that respective parking area. And it mainly focus on reducing the time in finding the parking lots and also it avoids the unnecessary travelling through filled parking lots in a parking area. Thus it reduces the fuel consumption which in turn reduces carbon footprints in an atmosphere. Our system is a Raspberry pi based parking system and the implementation is through infrared sensors connected to the GPIO pins of the raspberry pi to detect the empty parking spaces and sends this data to server, this stored data is accessed by users. This enhances the user to check the status/availability of parking spaces before setting their journey. Here the challenge is to use the existing resources in optimum level to reduce the searching time, traffic congestion in the city.



### 3.2 Identification of need

There is a need of this project for future development of enhancing the resources .The need for this project is basically in hospitals where there is urgent need of the parking areas.Also due to unorganized parking space related problems may arise which is a disadvantage for the place.Also Due to uneven parking there may be lot of accidents .

### 3.3 Planning and scheduling

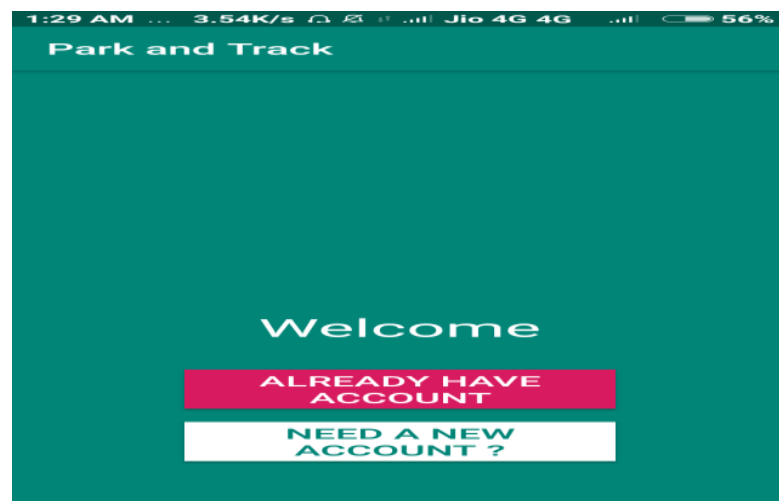
We have divided the whole process that is task in five steps, they are as follows:-

Hardware implementation: - firstly (as shown in the fig 2.) the hardware model is made. All the connections to the breadboard are made with the help of jumpers and these connections then are fed as an input to GPIO pins of raspberry pi which then gives the required output.

Sense the intruder: - The next we have used IR sensors which are used to sense the object placed in the parking slot. For demonstration we have used few parking slots. If the car comes in the parking slot the IR senses the object and the output is displayed by red light bulb that means if the car is parked red light gets activated and if not green light is activated.

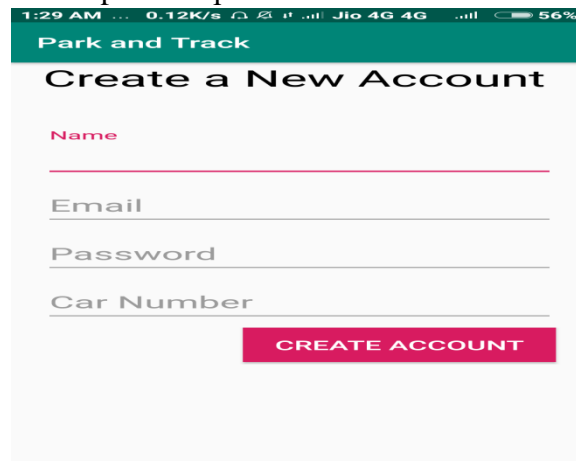
Build an android application: - with the help of an android studio and firebase an app is developed. There are total seven pages.

The first page is the home page where user can either create an account or login into an account..



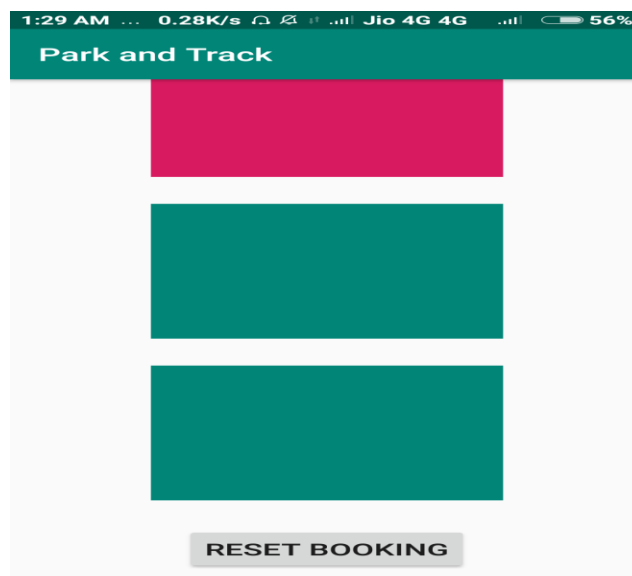
. Home page of an android application

If a user does not have his account, he will create an account first with the help of an option “need a new account

A screenshot of an Android application interface. At the top, a green header bar contains the text "Park and Track". Below the header, the title "Create a New Account" is displayed in bold black text. The form consists of four input fields: "Name", "Email", "Password", and "Car Number", each with a corresponding label above it. A pink "CREATE ACCOUNT" button is positioned to the right of the "Car Number" field. The status bar at the top shows the time as 1:29 AM, a data speed of 0.12K/s, and a battery level of 56%.

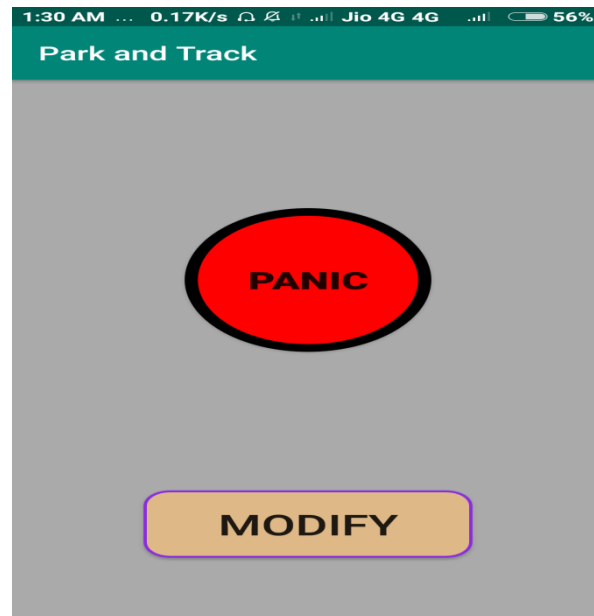
Registration form

If the user has an account already then he will login directly. When the user login, his/her information get stored in the database. Then after logging in the user moves to the next page where user gets two options first is panic and second is parking. When user taps on parking, user moves to the status page where the status of the whole parking system is displayed in the parking slots. The slots which are filled they will be shown in red colour while those which are empty that means will be shown in green colour . And by just a single tap upon parking slot user can also book slot before reaching to a parking slot. You can also change or cancel your booking just by tapping on a reset button provided on the same page on an android app.

A screenshot of an Android application interface. At the top, a green header bar contains the text "Park and Track". Below the header, there are three rectangular slots representing parking status. The top slot is pink, and the two slots below it are green. At the bottom of the screen, there is a grey button labeled "RESET BOOKING". The status bar at the top shows the time as 1:29 AM, a data speed of 0.28K/s, and a battery level of 56%.

## Demonstration of parking slot

Second module: - second module is prepared to solve the problem of late response to the user when in need during an emergency situation. If we have put a panic button in our app. This panic button will work to help us in the right moment on trouble



User interface of panic function

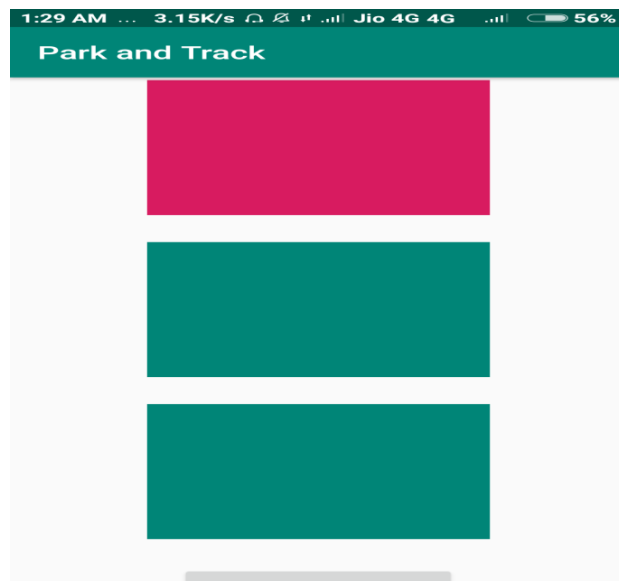
As soon as we press the panic button (shown in fig 6.). It sends alert on our three registered number as well as nearest police station that we are in trouble.

A screenshot of a mobile application interface titled "Park and Track". The background is a light gray. It contains a registration form with the following fields: "Your Name" with a text input field containing "Enter your name..."; "Contact No." with three separate text input fields labeled "First No.", "Second No.", and "Third No.". At the bottom of the form is a green button with the word "SAVE" in white capital letters. The top status bar shows the time as 1:29 AM, data speed as 0.00K/s, and battery level as 56%.

. Registration form to perform panic functionality

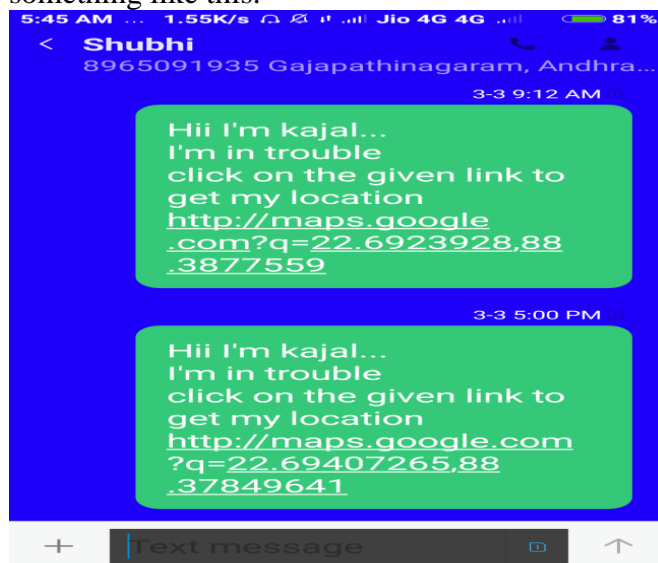
For the first time if you are using the app then you have to fill the form. The form is made to register the numbers. This will be the number you can contact at the time of the trouble.

As a result we can see current status a parking slot from anywhere with the help of our app as shown in fig 8. The slots which are filled or booked they will be shown in red colour while those which are empty that means will be shown in green colour



Status of parking slot

As soon as you press the panic button it sends the message to the three registered number. The message would look something like this.



### **3.4 Software requirement specification**

#### **3.4.1 User Interface**

##### **Android Application**

Our android application consists of 3 modules:

##### **Login Form:**

1. Firstly the user has to fill a form which consists of login details such as Username, Phone No. , Address.
2. The user also needs to give three contact numbers ,it can belong to any of their 3 relatives or friend.
3. The location of the user can be sent to all the three contacts in case of emergency.

##### **Check Status & Book a slot:**

1. In this module as the name suggests the user can perform two things.
2. He/she can check the parking status of a particular place.
3. They can also book a specific slot for themselves using the app.

##### **Panic Button/Security Feature:**

1. The app consists of a panic button which can be used at times of emergency.
2. By pressing this button a message will be sent with location of the user on the three contact numbers.

#### **3.4.2. Operating Environment**

Raspbian is a Debian-based computer operating system for Raspberry Pi. There are several versions of Raspbian including Raspbian Stretch and Raspbian Jessie. Raspbian is highly optimized for the Raspberry Pi line's low-performance ARM CPUs.. An operating system is the set of basic programs and utilities that make your Raspberry Pi run. However, Raspbian provides more than a pure OS: it comes with over 35,000 packages, pre-compiled software bundled in a nice format for easy installation on your Raspberry Pi.

#### **3.4.3 Product Functions**

Smart parking is a vehicle parking system that aid drivers to identify empty parking lots . The smart parking system also includes the means of calculating the space in the parking lot. The idea behind such arrangement is that, the system allocates a parking space.

Functionality-

Our project works in four steps:

1.As the vehicle enters the parking slot, the IR sensor sense the vehicle and generates the output using LEDs.

2.The RED LED indicates that the parking slot is empty. The GREE LED indicates that the parking slot is occupied.

3. The user can use the website .First he/she needs to login in the page by entering their username and password and then go to the next page to see the parking status.

4. The details of the user are stored in the database for future references and security purpose.

#### **3.4.4 Design and Implementation Constrains**

Since we were not very familiar with IOT and RASPBERRY PI it took us a lot of time to understand the functionalities of PI and how to use it in a model.

We faced a lot of problem during the connectivity of Raspberry Pi and PHP server due to slow/lost internet connection.

#### **3.4.5 Hardware Interface**

1. Raspberry Pi
2. Mobile phone
3. RFID reader and tags

#### **3.4.6 Communication Interface**

Internet Connectivity is one of the main things to keep in mind while developing any Internet of Things (IOT) project.The major source of connectivity of IOT is the INTERNET. Internet is the basic requirement for our project as we are using Raspberry Pi which doesnot work without internet.

The other use of internet in our project is for opening the webpage to see parking status.

#### **3.4.7 System Failure**

The only failure we faced in our project was to connect our second webpage “STATUS PAGE” with Raspberry Pi. We are right now unable to show the status of the parking slot in the website. We make sure that we complete the project as soon as possible.

### **3.4.8 Software System Attributes**

#### **Reliability**

This system can work for 24 hours if provided continuous internet connection. Smart Parking system is reliable as every person at some or the other place had to face the problem of parking their vehicle. Therefore this project in some ways could help people to find their parking slot in no time. By knowing the parking status a person can identify whether he/she wants to go to the particular place or not. Therefore it can be concluded that this project is reliable to use.

#### **Availability**

Availability of a system is typically measured as a factor of its reliability – as reliability increases, so does availability. Availability of a system may also be increased by the strategy of focusing on increasing testability, diagnostics and maintainability and not only on reliability. Improving maintainability during the early design phase is generally easier than reliability (Testability & diagnostics). Maintainability estimates are also generally more accurate. However, because the uncertainties in the reliability estimates (and also in diagnostic times) are in most cases very large, it is likely to dominate the availability (and the prediction uncertainty) problem, even while maintainability levels are very high.

Availability of this project depends on the factor that it works properly and the customers already using it are satisfied with the results.

#### **Security**

Security is freedom from, or resilience against, potential harm from external forces. Beneficiaries of security may be persons and social groups, objects and institutions, ecosystems, and any other entity or phenomenon vulnerable to unwanted change by its environment.

Security mostly refers to protection from hostile forces, but it has a wide range of other senses: for example- absence of harm (if one car is already placed in a particular area, the other person cannot park their vehicle in the same place even by mistake as they already have the website to see the status, therefore no harm to the already parked vehicle can be made.), As resilience against potential harm or damage, as secrecy (every vehicle would have its own slot therefore the fact of secrecy is maintained), As containment.

## **Maintainability**

In engineering, maintainability is the ease with which a product can be maintained in order to:

- Correct defects or their cause.
- Repair or replace faulty or worn-out components(in case the IR sensors or LEDs gets damaged) without having to replace still working parts.
- Prevent unexpected working condition( the LEDs can change their working in case it is disturbed ).
- Maximize a product's useful life(the life of the project depends on the life of the resources used i.e. Raspberry, sensors, LEDs, cables).
- Maximize efficiency, reliability, and safety.
- Meet new requirements(some requirements can be added i.e. to add the feature of online parking booking system).
- Make future maintenance easier, or
- Cope with a changed environment.

## **Portability**

Portability in high-level computerprogramming is the usability of the same software in different environments. The pre-requirement for portability is the generalized abstraction between the application logic and systeminterfaces. When software with the same functionality is produced for several computing platforms, portability is the key issue for development cost reduction. The “Smart Parking System” can be made portable if we use wireless sensors and wi-fi connectivity instead of using LAN cables for providing internet connection.

## **Performance**

Performance is completion of a task with application of knowledge, skills and abilities. In work place, performance or job performance means good ranking with the hypothesized conception of requirements of a task role, whereas citizenship performance means a set of individual activity/contribution that supports the organizational culture.



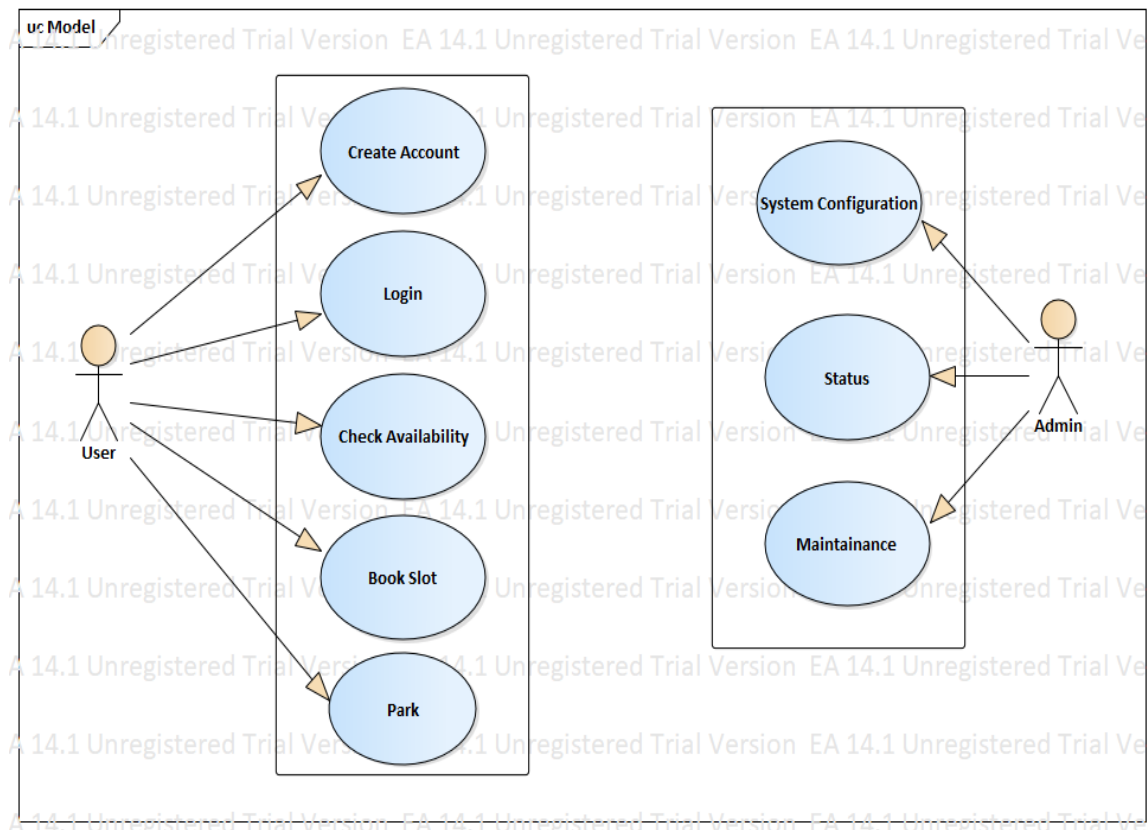
Action	SOLUTIONS
	Check that Raspberry is working properly
	Check all network connections are installed properly
	Checking Database connectivity properly
	Checking of the correct login and registration

## CHAPTER 4 SYSTEM DESIGN

System design can be explained through various software process model .They are explained as follows.

### 4.1 USECASE DIAGRAM

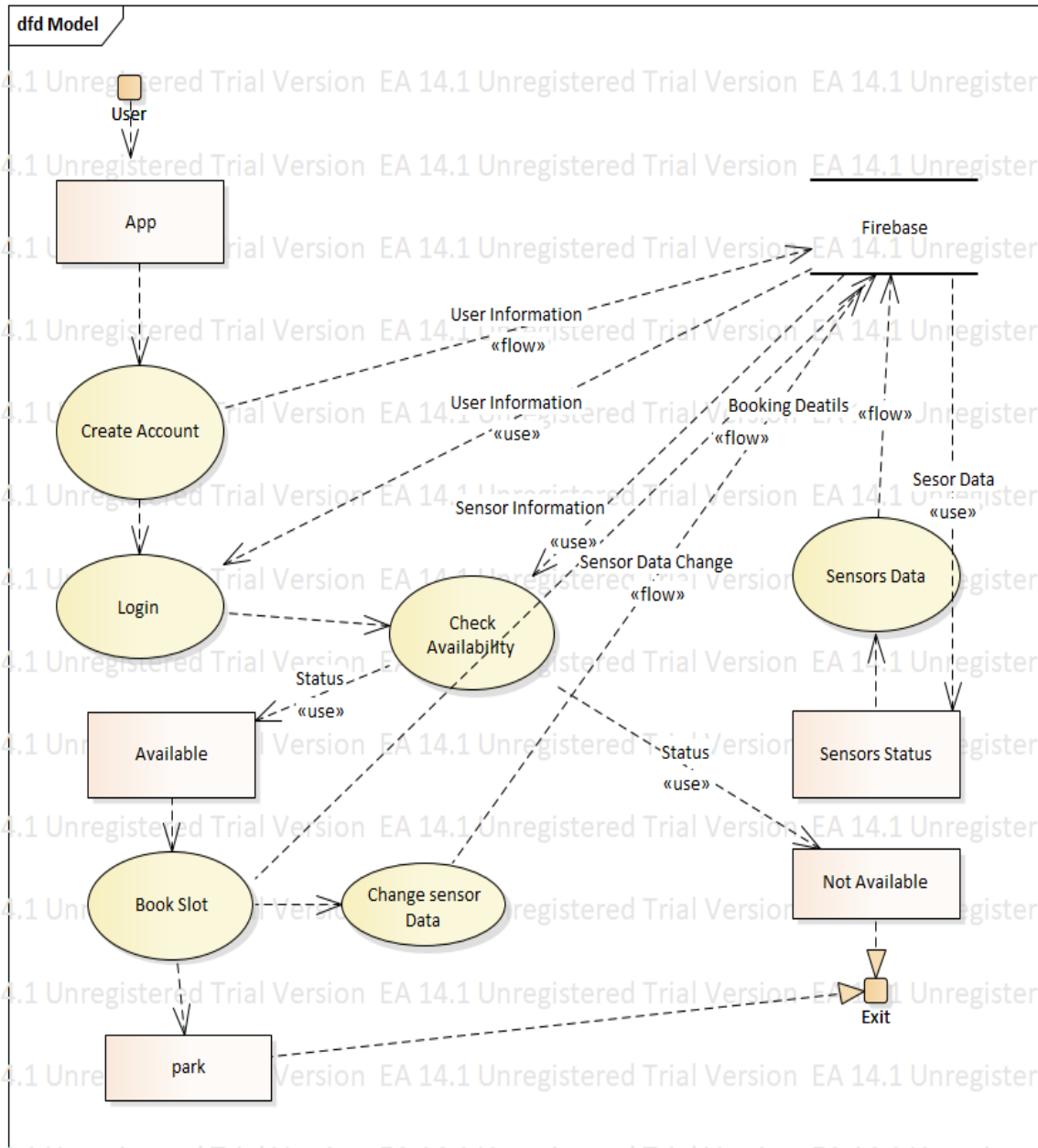
A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.



## 4.2 DATA FLOW DIAGRAM

A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects. A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).

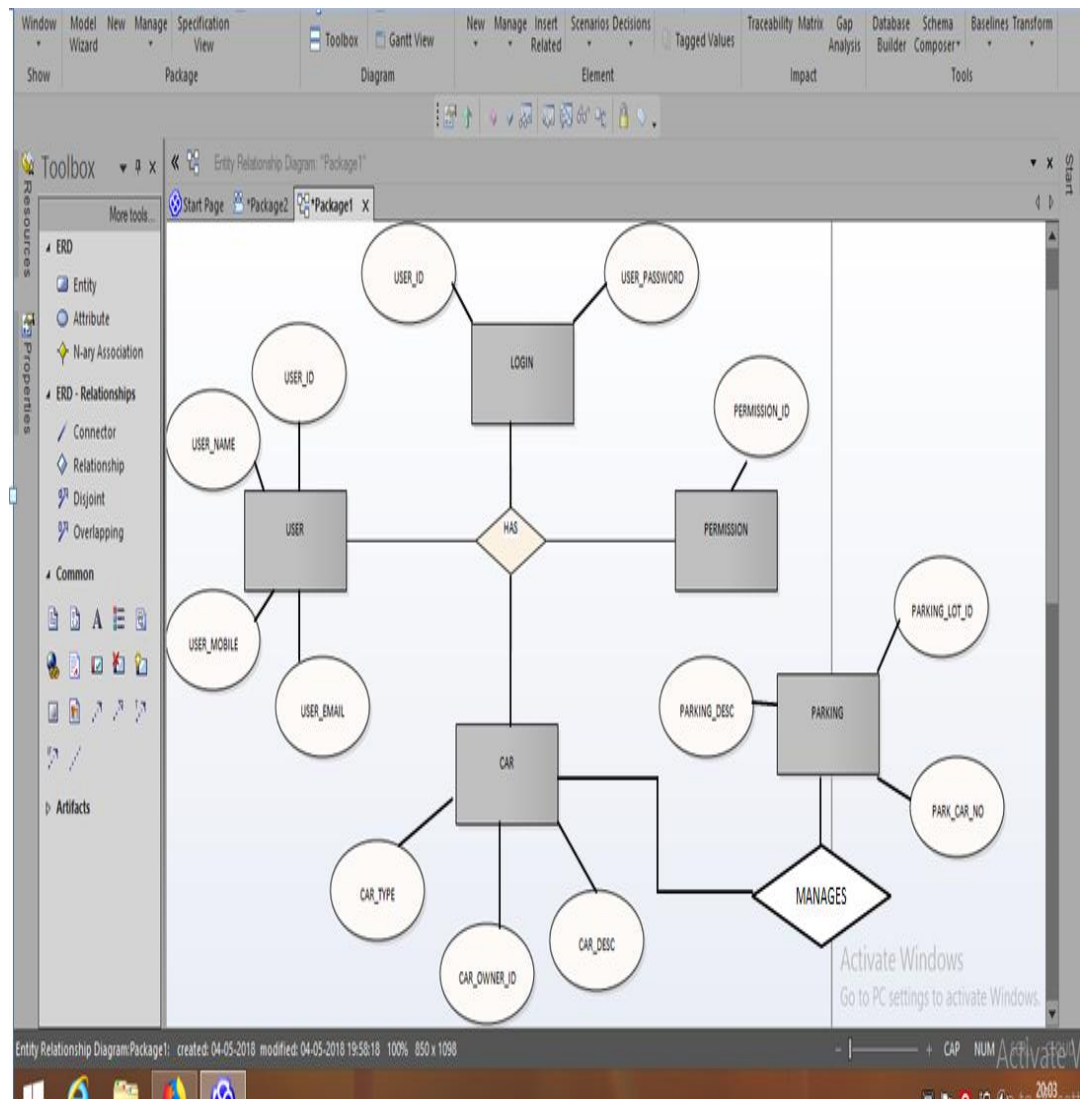
A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored



### 4.3 ENTITY RELATIONSHIP DIAGRAM

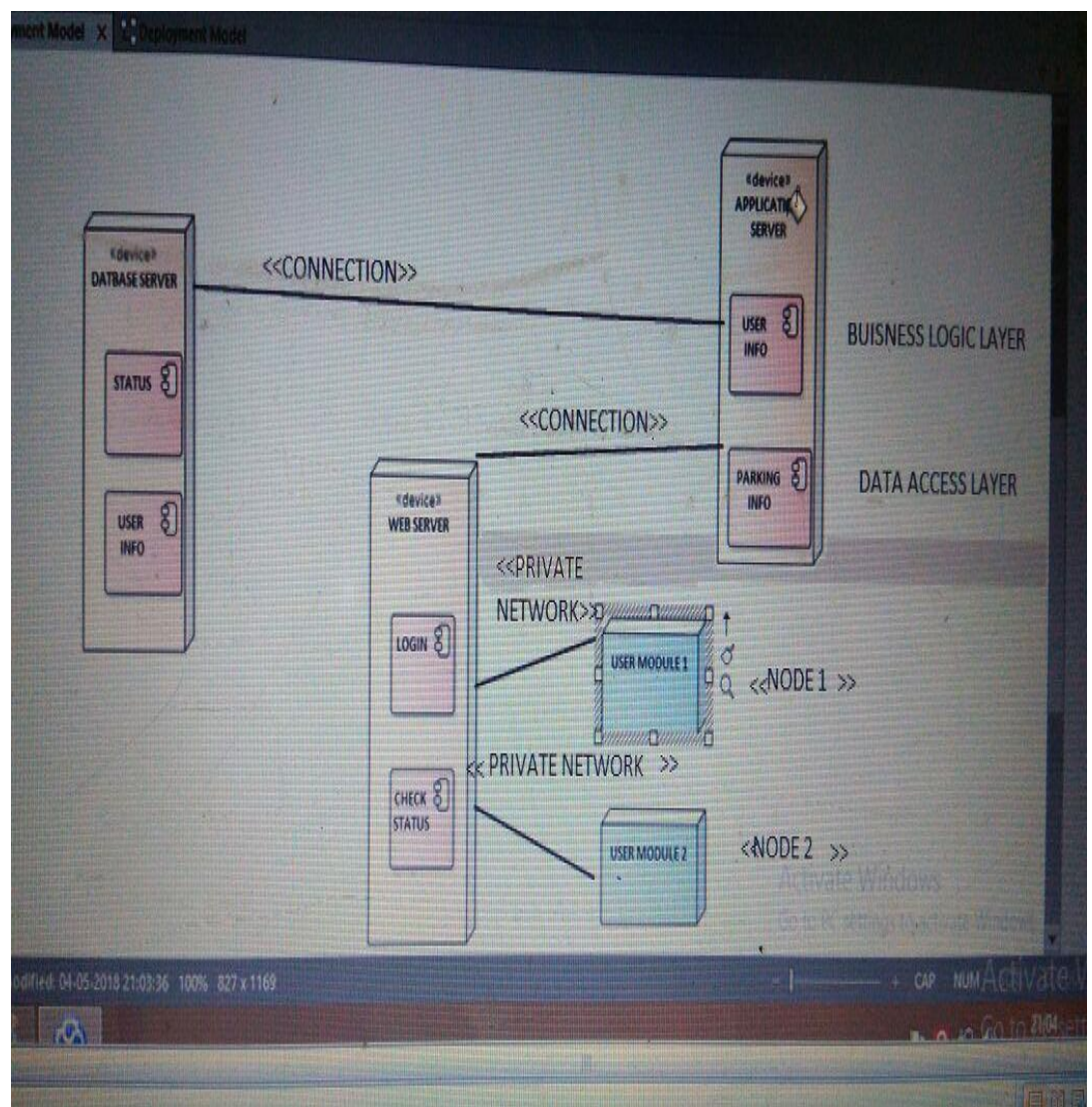
An entity–relationship model (ER model for short) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types (which classify the things of interest) and specifies relationships that can exist between instances of those entity types..

In software engineering, an ER model is commonly formed to represent things that a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model, that defines a data or information structure which can be implemented in a database, typically a relational database.



## 4.4 DEPLOYMENT DIAGRAM

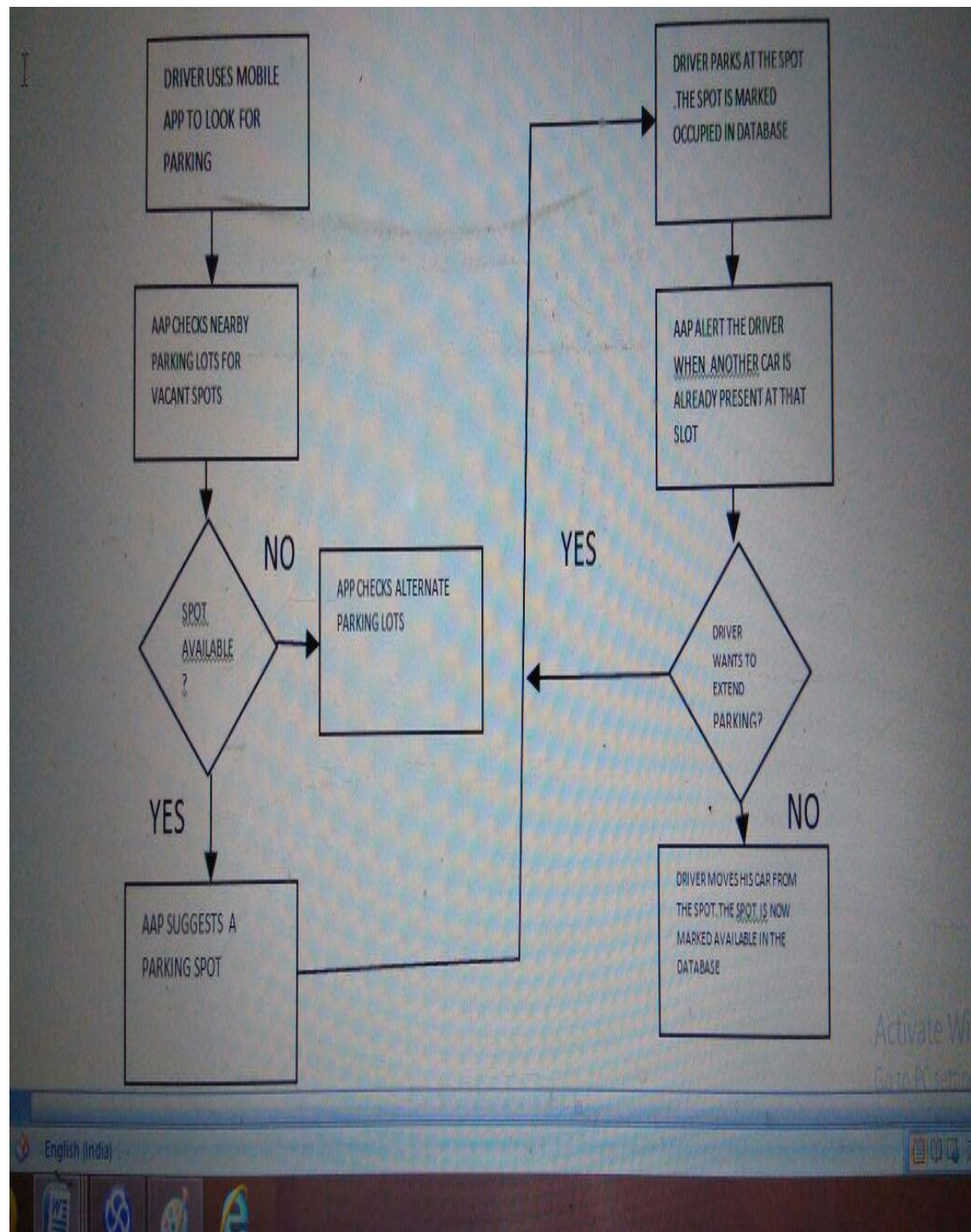
A deployment diagram in the Unified Modeling Language models the physical deployment of artifacts on nodes.<sup>[1]</sup> To describe a web site, for example, a deployment diagram would show what hardware components ("nodes") exist (e.g., a web server, an application server, and a database server), what software components ("artifacts") run on each node (e.g., web application, database), and how the different pieces are connected (e.g. JDBC, REST, RMI). The nodes appear as boxes, and the artifacts allocated to each node appear as rectangles within the boxes. Nodes may have subnodes, which appear as nested boxes. A single node in a deployment diagram may conceptually represent multiple physical nodes, such as a cluster of database servers.





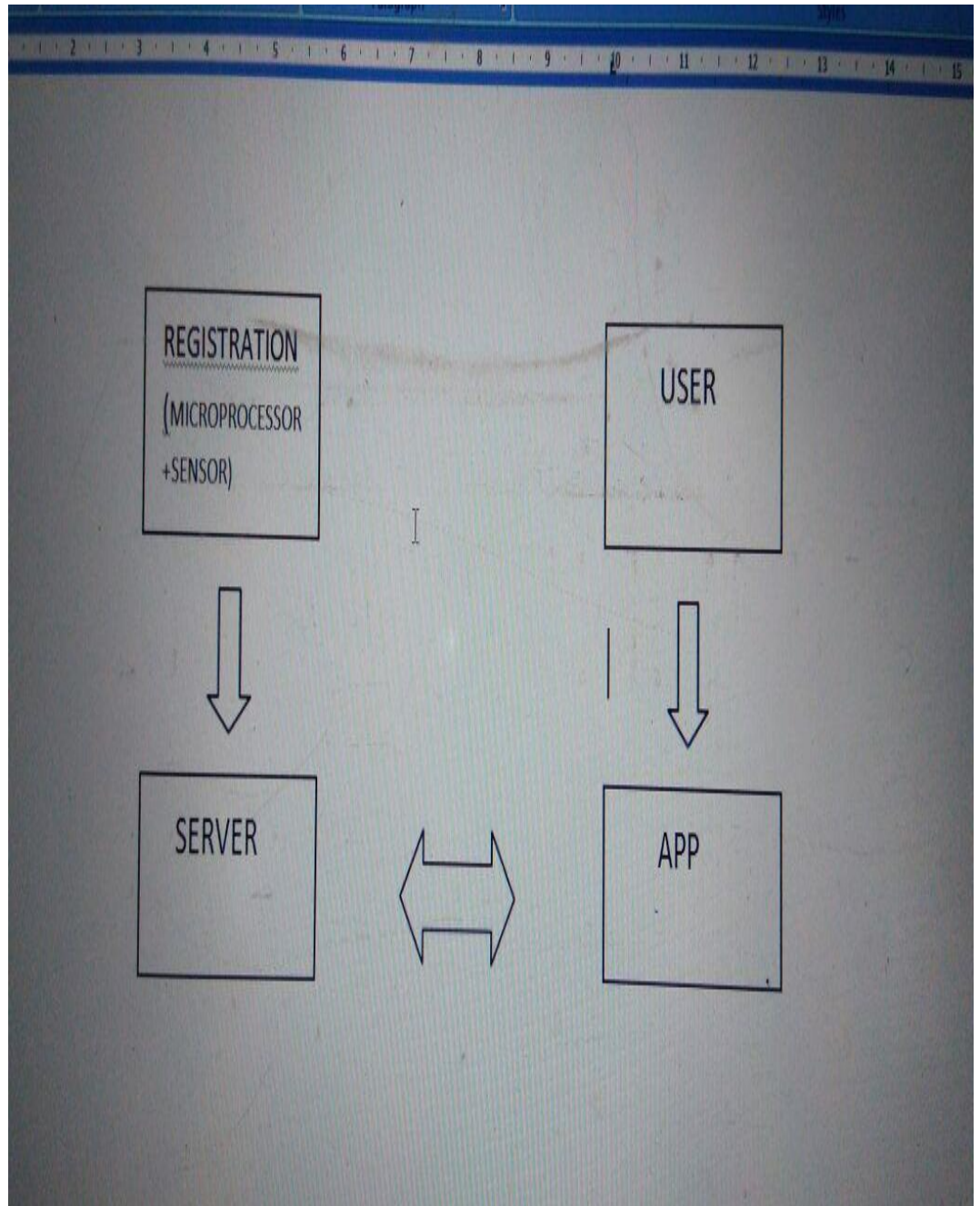
## 4.5 PROCESS DIAGRAM

A process flow diagram (PFD) is a diagram commonly used in chemical and process engineering to indicate the general flow of plant processes and equipment. The PFD displays the relationship between major equipment of a plant facility and does not show minor details such as piping details and designations. Another commonly used term for a PFD is a flowsheet.



## 4.6 ARCHITECTURAL VIEW

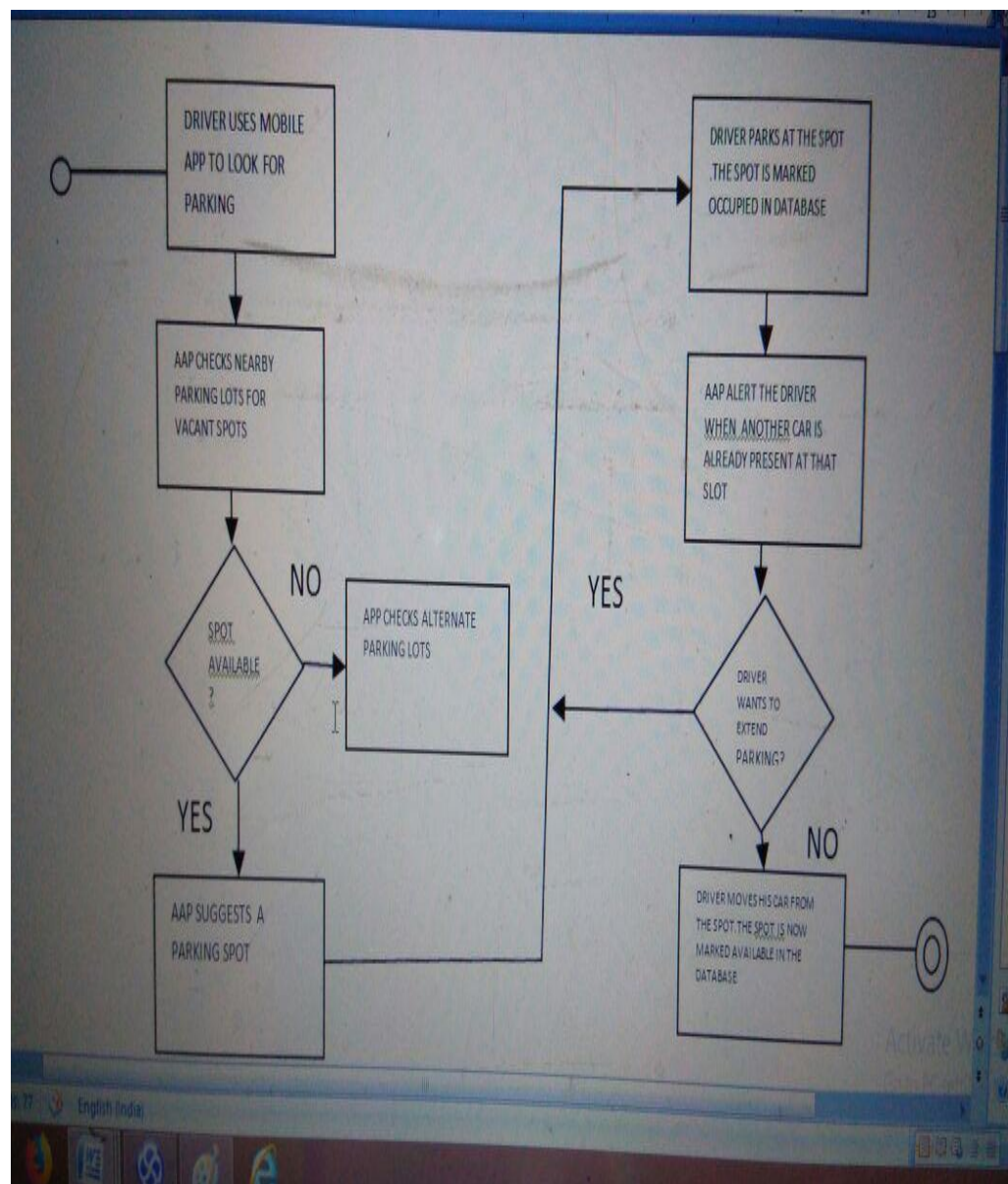
A concept is an idea, a theory or notion, but in architecture we could also describe a concept as 'an approach' to the design. When we think of an architectural concept, we think of an abstract idea, one that is unchanging throughout the design process.





## 4.7 ACTIVITY DIAGRAM

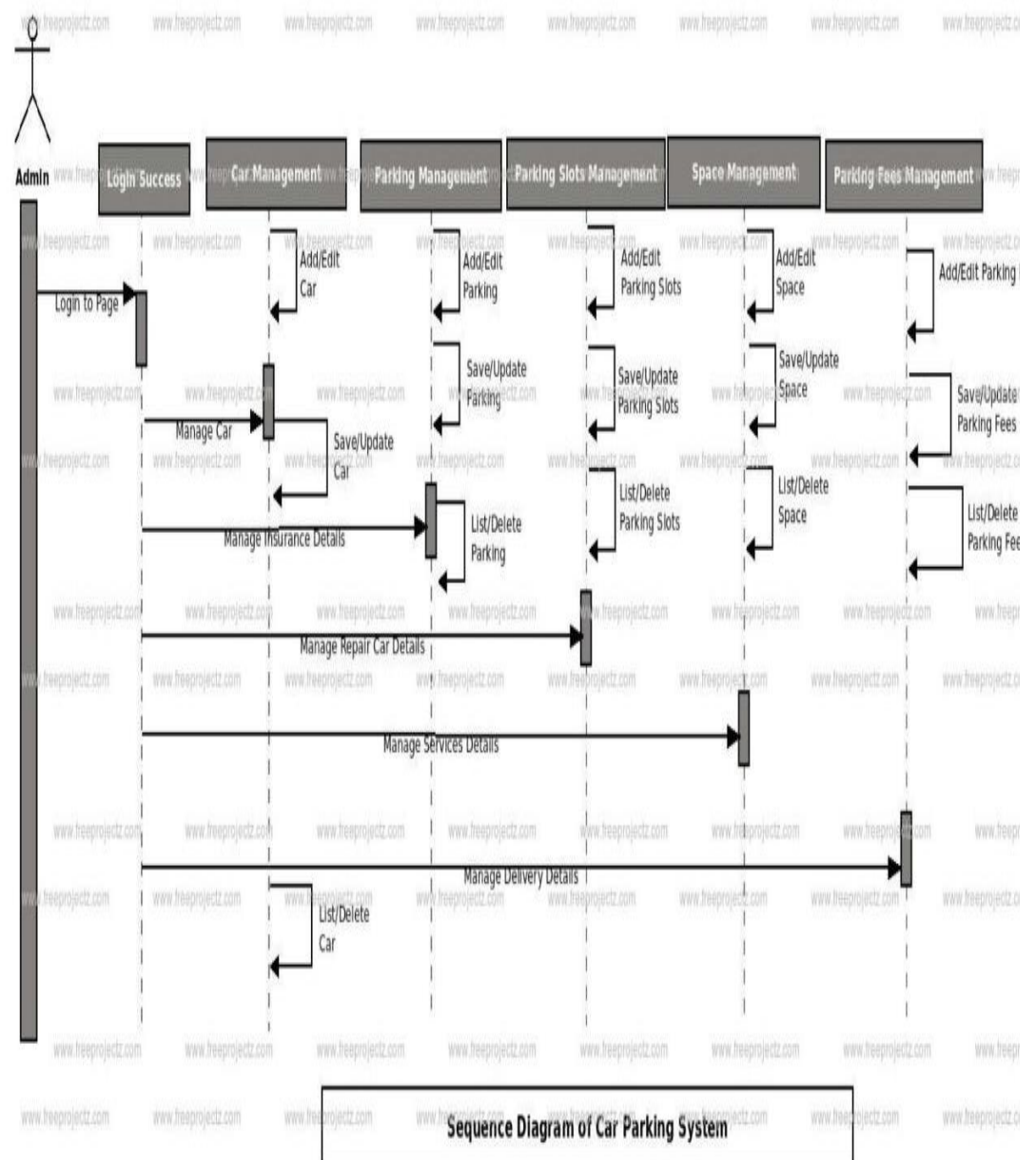
Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.





## 4.8 SEQUENCE DIAGRAM

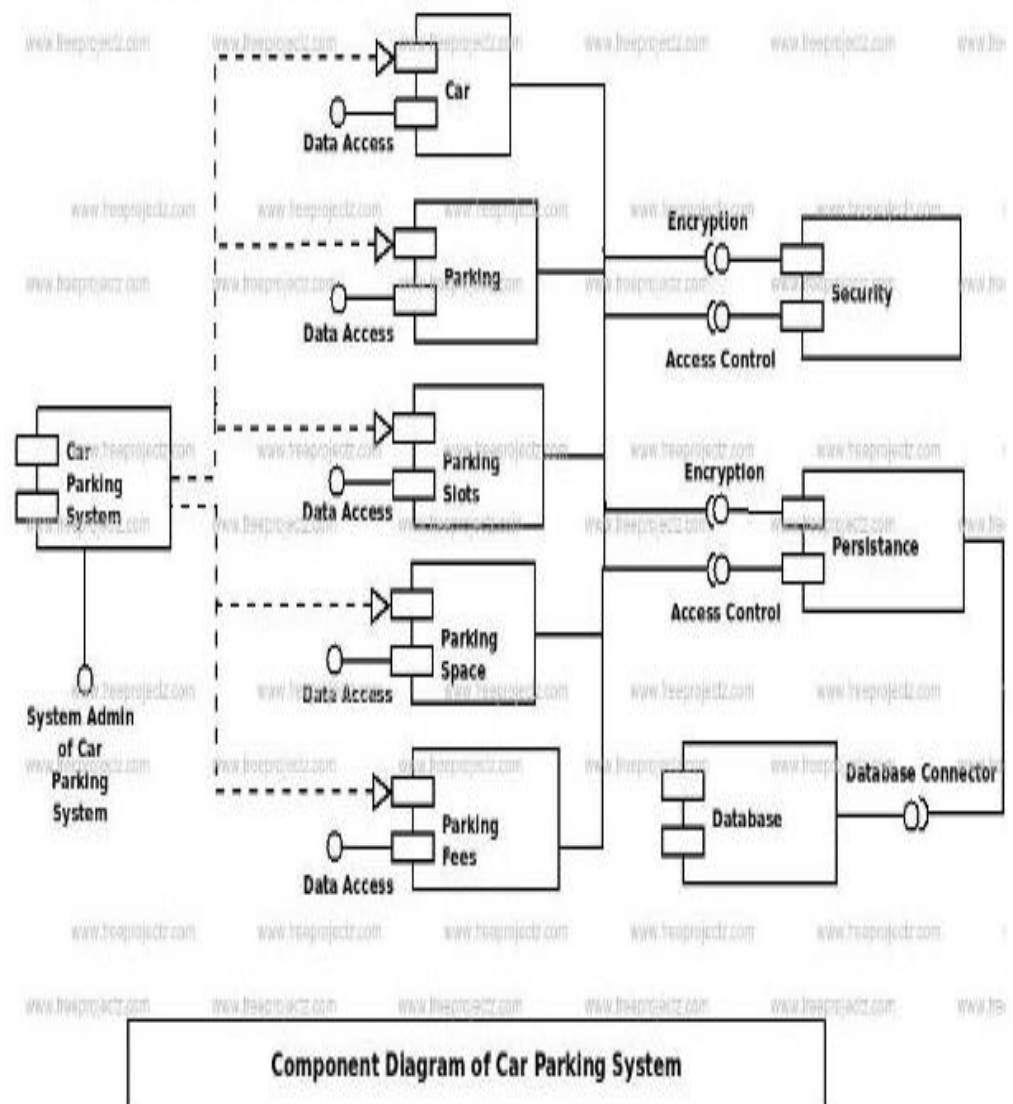
A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.



## 4.9 COMPONENT DIAGRAM

A component is something required to execute a stereotype function. Examples of stereotypes in components include executables, documents, database tables, files, and library files. Components are wired together by using an assembly connector to connect the required interface of one component with the provided interface of another component. This illustrates the service consumer - service provider relationship between the two components.

Component Diagram:



## CHAPTER 5 IMPLEMENTATION

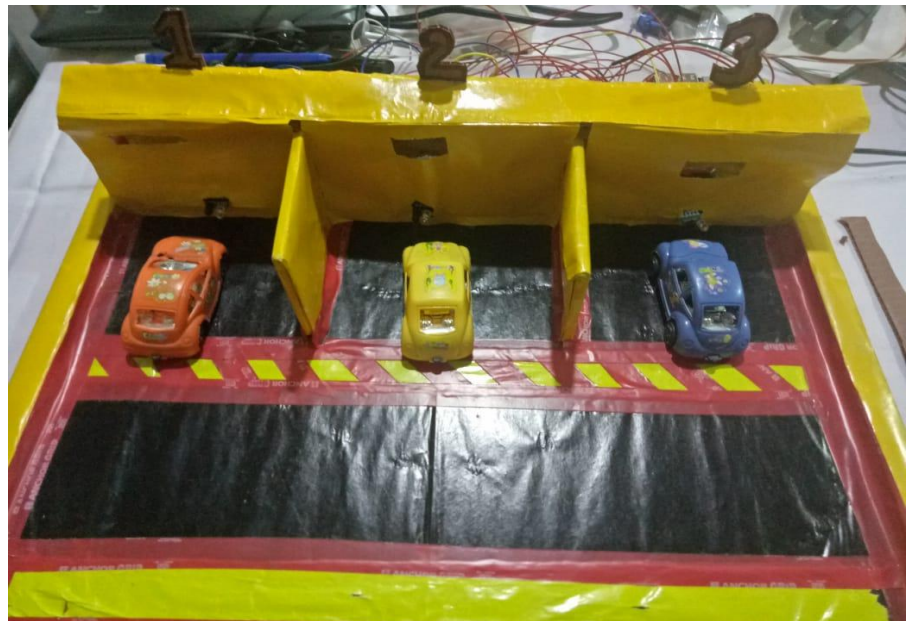
The Smart Parking System is designed by making use of some IOT supportable hardware's such as raspberry pi, RFID and infrared sensors, here we focussing on less power consumption and more performance device so raspberry pi is the suitable microcontroller for our implementation. And rasbian OS is loaded into the storage device that is a memory card because this operating systems basically consumes less power.

Our project focuses on delivering a problem solving idea for parking, basically it aims at saving time that is - time management .Not only it focuses on saving time but also it provides the idea of proper utilization of space, proper information of vehicles, safety and avoid unnecessary parking. We have implemented this with the help of raspberrypi 3, RFID and Infrared sensors.

- Firstly the raspberrypi 3 hardware is installed with rasbian operating system which is the official operating system of raspberry Pi. After the setup of raspberryPi, through GPIO pins we have accessed infrared sensors.
- Every parking slot consist of an IR sensor which is used to sense the presence of object. The output of these sensors are displayed through led's. Every slot consist of two led's each. Out of the two one is red and the other is green.
- We have used red and green colours to show the availability of the lot.
- The connections of led and infrared sensors are done with the help of wires and jumpers, through bread board to the GPIO pins of the raspberry pi.
- After all the physical setup, then the appropriate logic behind the glowing of the led's through IR sensors is coded in python using rasbian OS.
- The logic behind smart parking that we have applied is ,when the vehicle will approach to the parking lot then if the slot is empty this result will be displayed by green led ,this shows that the particular slot is empty or available ,next if red led glows that means the slot is already filled .
- We have used red and green colours to show the availability of the lot.
- We have implemented this approach by making a hardware model .Also a software approach has been made which shows the availability through a website.
- Php my admin is installed in raspberry pi with the help of that connectivity is established between the website and the RaspberryPi.

- The user is required to log-in in the application ,fill all the details in the form to log-in and check the parking status of a particular place.
- The users will be provided with RFID which can be used to store their personal information provided by them.
- This information can be useful for security purposes also.
- If a person lost track of his/her destination or he/she is in some kind of danger then all their information will be sent to the nearest police station and his/her registered emergency number through GPS.
- And because the user have already logged-in in the app we will be having all their information including the number of the vehicle so that the police would be able to find them easily.
- The output of the system can be seen on laptop or mobile phone through a app/webpage.The output is shown such that , the parking lots that are booked are coloured red whereas it displays green colour if the parking lots are empty

### **Hardware :**



## CHAPTER 6 TESTING APPROACH

### 6.1 Test case for an android app

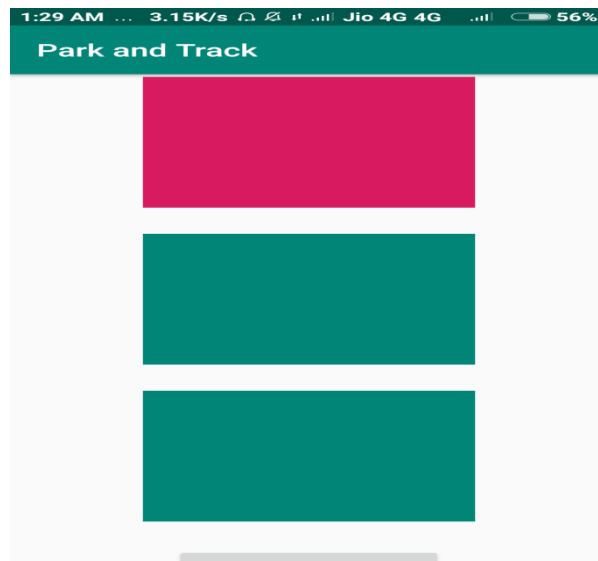
TEST CASE(S)	Steps	Expected Results
Test Case:	1	
1)	Enter a valid username and valid password. Click login button.	The application should display the home page.
2)	Log out. Enter a valid username and an <b>invalid</b> password. Click login button.	The application should display an error message and re-open the login page.
3)	Log out. Enter an <b>invalid</b> username and an <b>invalid</b> password. Click login button.	The application should display an error message and re-open the login page.
Test Case:	2	
1)	Verify the following username and password combinations. Note V means valid and I means invalid.	Note: E means error message and H means home page.
1.1)	blank, blank	E
1.2)	blank, I	E
1.3)	blank, V	E
1.4)	I, blank	E
1.5)	I, I	E
1.6)	I, V	E
1.7)	V, blank	E
1.8)	V, I	E
1.9)	V, V	H

### 6.2 Test case for IR sensors

Test Case		
2)	Intruder Detection. Note T means true and F means false	Note: R means red and G means green
2.1)	T	R
2.2)	F	G

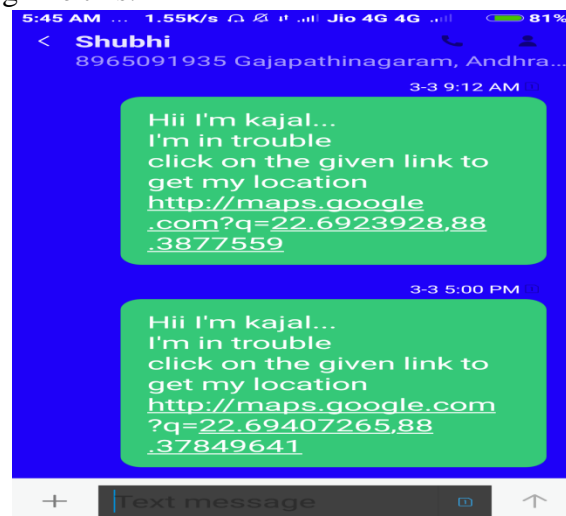
## CHAPTER 7 RESULT AND DISCUSSION

As a result we can see current status a parking slot from anywhere with the help of our app as shown . The slots which are filled or booked they will be shown in red colour while those which are empty that means will be shown in green colour



Status of parking slot

As soon as you press the panic button it sends the message to the three registered number. The message would look something like this.



Result of panic alert

## CHAPTER 8 CONCLUSION

It's inevitable for the people to update with the growing technology. And generally people are facing problems on parking vehicles in parking slots in a city. In this study we design a smart parking system which enables the user to find the nearest parking area and gives availability of parking slots

in that respective parking area. And it mainly focus on reducing the time in finding the parking lots in a parking area. Thus it reduces the fuel consumption which in turn reduces carbon footprints in an atmosphere. Our system is a raspberry pi based parking system and the implementation is through infrared sensors connected to the GPIO pins of the raspberry p to detect the empty parking spaces and sends this data to server, this stored data is accessed by users. This enhances the user to check the availability of parking spaces before setting their journey. Here the challenge is to use the existing resources in optimum level to reduce the searching time, traffic congestion in the city.

## **CHAPTER 9 REFERENCES**

- 1.Kyaw<sup>1</sup>, Ar. Kar. , Truong<sup>3</sup>, Hong. Phat. (2018). Low-Cost Computing Using Raspberry Pi 2 Model B.Journal of Computers, 13(3), 287- 296
2. <https://www.elprocus.com/infrared-ir-sensor-circuit-and-working/>
- 3.<https://circuitdigest.com/article/servo-motor-basics>
- 4.<https://www.epc-rfid.info/rfid>
- 5.<https://skyline-parking.com/automated-parking-systems/how-aps-work/>