Dr. Vishwanath Karad
MIT WORLD PEACE UNIVERSITY | PUNE
TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

Faculty of Science

School of Computer Science

2020-2021

A

PROJECT REPORT

ON

BY

Shubham Modi	**Roll no: MCA 37**
Shohrat Ali	**Roll no: MCA 03**
Himanshu Rajpurohit	**Roll No: MCA 45**
Pushkar Parmar	**Roll No: MCA 41**
Shubham Vyas	**Roll No: MCA 54**

IN PARTIAL FULFILLMENT OF

MASTERS OF COMPUTER APPLICATIONS

Dr. Vishwanath Karad MIT- World Peace University

Faculty of Science

School of Computer Science

CERTIFICATE

This is to certify that **Mr. Shubham Modi,** Student of MCA (Commerce) Semester VI has successfully completed a mini project in partial fulfilment of MCA (Science) Degree under Dr. Vishwanath Karad MIT-World Peace University, for the academic year 2020-2021.

(Project Guide)
Mrs. Surbhi Thatte

Head of School
Dr. C. H. Patil

Associate Dean
Dr. Shubhalaxmi Joshi
Date:

Seal

Examiners:

1.

2.

Faculty of Science

School of Computer Science

CERTIFICATE

This is to certify that **Mr. Pushkar Parmar,** Student of MCA (Commerce) Semester VI has successfully completed a mini project in partial fulfilment of MCA (Science) Degree under Dr. Vishwanath Karad MIT-World Peace University, for the academic year 2020-2021.

(Project Guide)
Mrs. Surbhi Thatte

Head of School
Dr. C. H. Patil

Associate Dean
Dr. Shubhalaxmi Joshi
Date:

Seal

Examiners:

1.

2.

Faculty of Science

School of Computer Science

CERTIFICATE

This is to certify that **Mr. Shubham Vyas,** Student of MCA (Commerce) Semester VI has successfully completed a mini project in partial fulfilment of MCA (Science) Degree under Dr. Vishwanath Karad MIT-World Peace University, for the academic year 2020-2021.

(Project Guide)
Mrs. Surbhi Thatte

Head of School
Dr. C. H. Patil

Associate Dean
Dr. Shubhalaxmi Joshi
Date:

Seal

Examiners:

1.

2.

Faculty of Science

School of Computer Science

CERTIFICATE

This is to certify that **Mr. Shohrat Ali,** Student of MCA (Commerce) Semester VI has successfully completed a mini project in partial fulfilment of MCA (Science) Degree under Dr. Vishwanath Karad MIT-World Peace University, for the academic year 2020-2021.

(Project Guide)
Mrs. Surbhi Thatte

Head of School
Dr. C. H. Patil

Associate Dean
Dr. Shubhalaxmi Joshi
Date:

Seal

Examiners:

1.

2.

Faculty of Science

School of Computer Science

CERTIFICATE

This is to certify that **Mr. Himanshu Purohit,** Student of MCA (Commerce) Semester VI has successfully completed a mini project in partial fulfilment of MCA (Science) Degree under Dr. Vishwanath Karad MIT-World Peace University, for the academic year 2020-2021.

(Project Guide)
Mrs. Surbhi Thatte

Head of School
Dr. C. H. Patil

Associate Dean
Dr. Shubhalaxmi Joshi
Date:

Seal

Examiners:

1.

2.

DECLARATION

I, Mr. PUSHKAR PARMAR hereby declare that this project is the record of authentic work carried out by me during the academic year 2020 . This project is plagiarism free and has not been submitted to any other University or Institute towards the award of any degree.

Signature of the student

(PUSHKAR PARMAR)

ACKNOWLEDGEMENT

I wish to express my deep sense of gratitude and honour toward for giving chance to work in project with my college. Project becomes successful weather it is big or small is just because of wonderful people working together in the project with their helping nature to each other.

I also wish to thanks to all people in the organization who help me during project development time to time.

I also express my honour and gratitude to **Mrs. Surbhi Thatte** and constant encouragement for completing my project work successfully.

I wish to express my honour and gratitude to **Dr. C. H. Patil** for providing the necessary facilities and encouragement for completing my project work successfully.

I wish to express my deep sense of gratitude and honour towards my internal guide **Mrs. Surbhi Thatte**. I committed devotion, dedication and encouragement with full faith on me.

Student Signature

INDEX

1. INTRODUCTION

1.1 Existing System

Throughout the decades our nation has been grown definitely, presently we are in this express we have a great deal of very much reached streets, business fabricating and expanding number of vehicles. While stopping these vehicles in parking spot we utilize the manual methodology of leaving. Which the greater part of the cases is spontaneous and absence of control because of this, individuals can leave their vehicles anyplace they need to, which makes a wreck as individuals don't pursue the specific signal more often than not. Therefore, a colossal congested driving conditions happens in that spot. While leaving in and recovering vehicle due fumble autos can get gouge by knocking with one another as there is absence of adequate space. This prompts contentions, battles among individuals which in some cases makes tremendous congested driving conditions. This is additionally an efficient misfortune as we have to fix our harmed vehicle and furthermore autos expend additional fuel while leaving in or out. Automobile overload is an issue here as it kills our valuable time. Because of this disarray in stopping our important time gets squandered. It hurts the understudies, office going staffs and crisis patients as it were.

1.2 Need of New System

New system is needed as it causes economical loss to commercial places like shopping malls, amusement parks, as people are more likely not to visit these places due to this parking hazard. As we are advancing with time, the manual car parking system in commercial spaces is creating hurdle which is causing wastage of time and some economic losses as well. Therefore, we need a solution which can overcome these problems. Here we are introducing Smart Parking App as a solution of these problems as well as a replacement to the manual car parking systems at commercial spaces. This system not only saves time and money, it can also earn money by charging for parking spaces.

2. PROPOSED SYSTEM

2.1 Proposed System: -

- **PARK SENSE** app is an IOT based technology which uses NodeMCU that sends the real time data from the sensors to efficiently find out and displays which parking slots are occupied or available.

- Infrared sensor will detect the vehicle at the entry and exit gate simultaneously keeps the count for incoming and outgoing vehicles resulting in opening and closing of respective gates automatically with the help of Arduino and Servo motor.

- User can navigate to the parking area using the functionality in the app which uses the Google Maps Android API for navigation.

2.2 Objectives of System: -

- No trouble in finding empty spots which is time consuming, thus won't result in towing and save our precious time.

- Reducing stress of being late to their lectures, or running across the campus to make it to their exams.

- Less Amount of traffic if there's a proper management of parking.

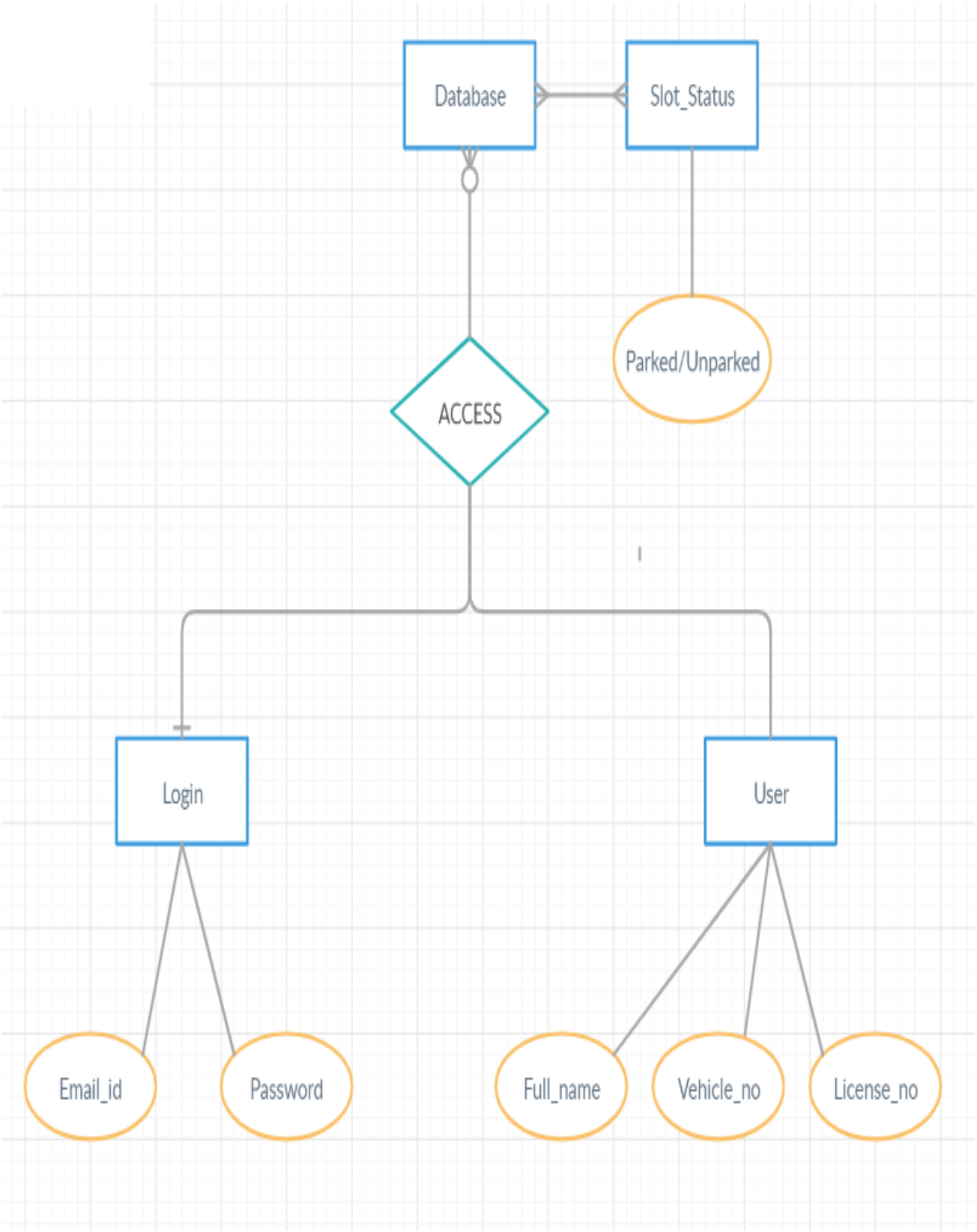
- It excludes the need of human efforts for managing parking spaces.

2.3 User Requirements: -

- **Software Requirements**
  - Android Studio - Building the Application.
  - Arduino IDE - IDE for Arduino Components.
  - Firebase - Realtime Database.
- **Hardware Requirements**
  - Arduino Uno - Controller for the incoming & outgoing vehicles.
  - NodeMCU - Controller for the parking slot.
  - Infrared Sensors - Sensor for detecting vehicles.
  - Jumper Wires - Overall connection of the components.
  - Servo Motors - Entry & Exit Gate.
  - Breadboard - Micro Controller Unit (MCU) to be placed.
  - LCD - Displaying the total count of slots available.
  - LED - Indicating whether the slots are available or not.

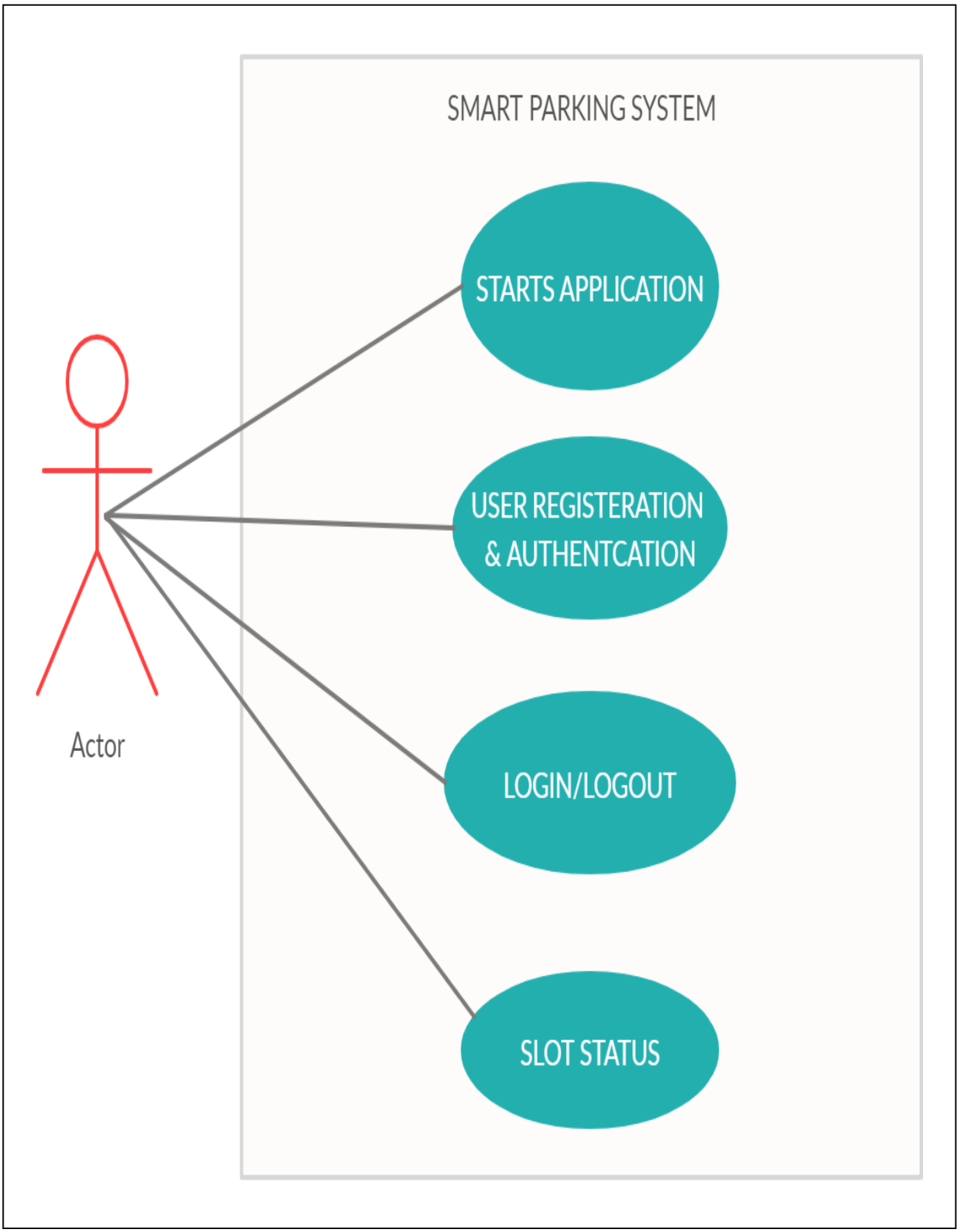
3. ANALYSIS & DESIGN

3.1 Entity Relationship Diagram: -

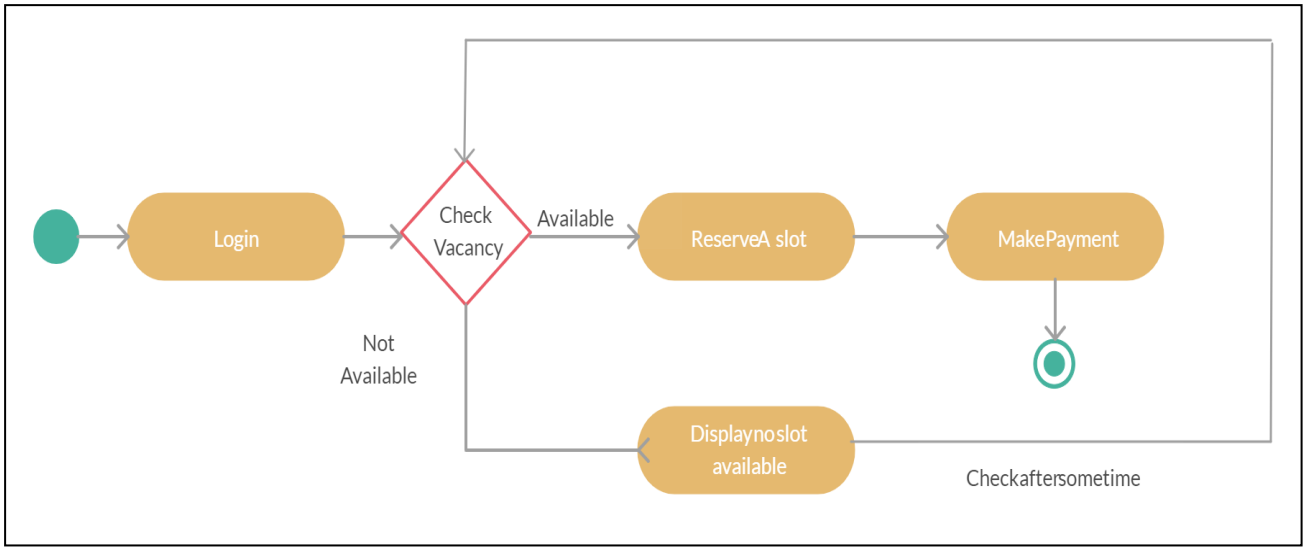


3.2 UML DIAGRAMS: -

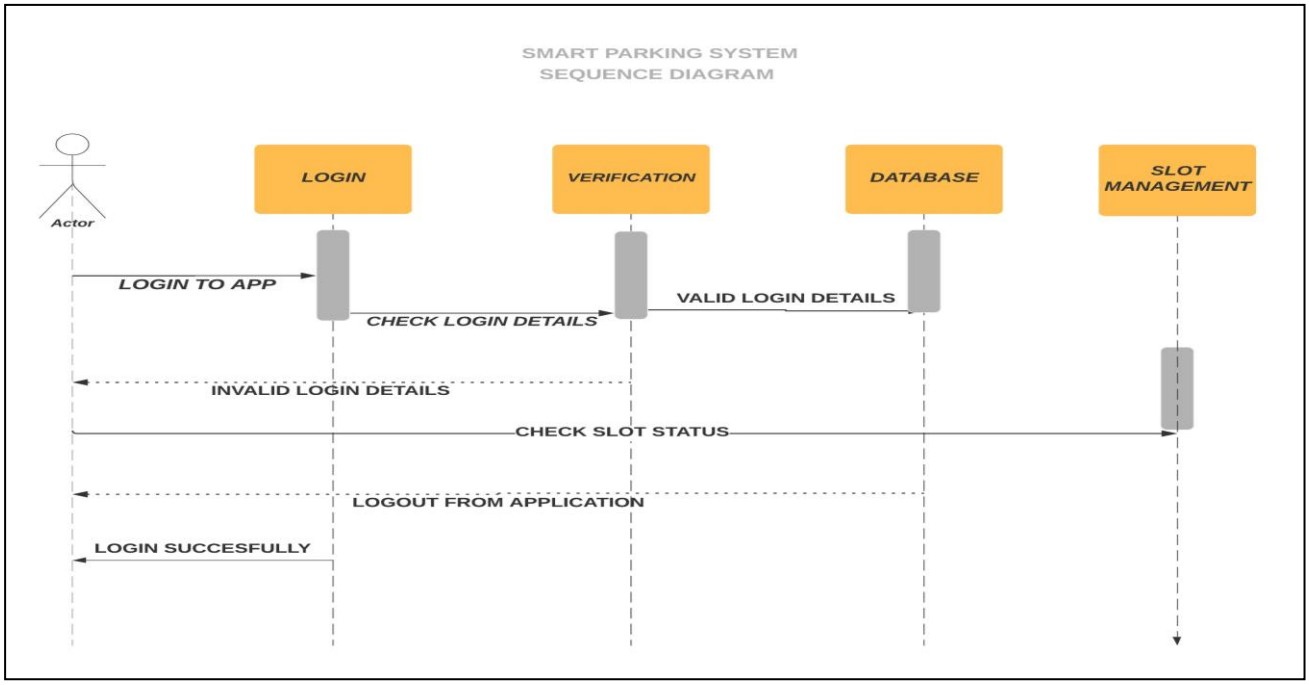
3.2.1 Use Case Diagram: -



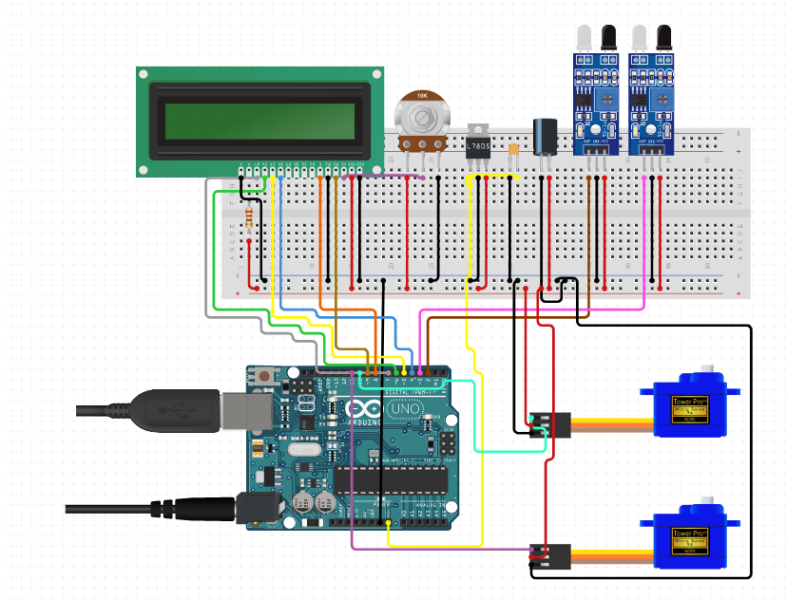
3.2.2 Activity Diagram: -

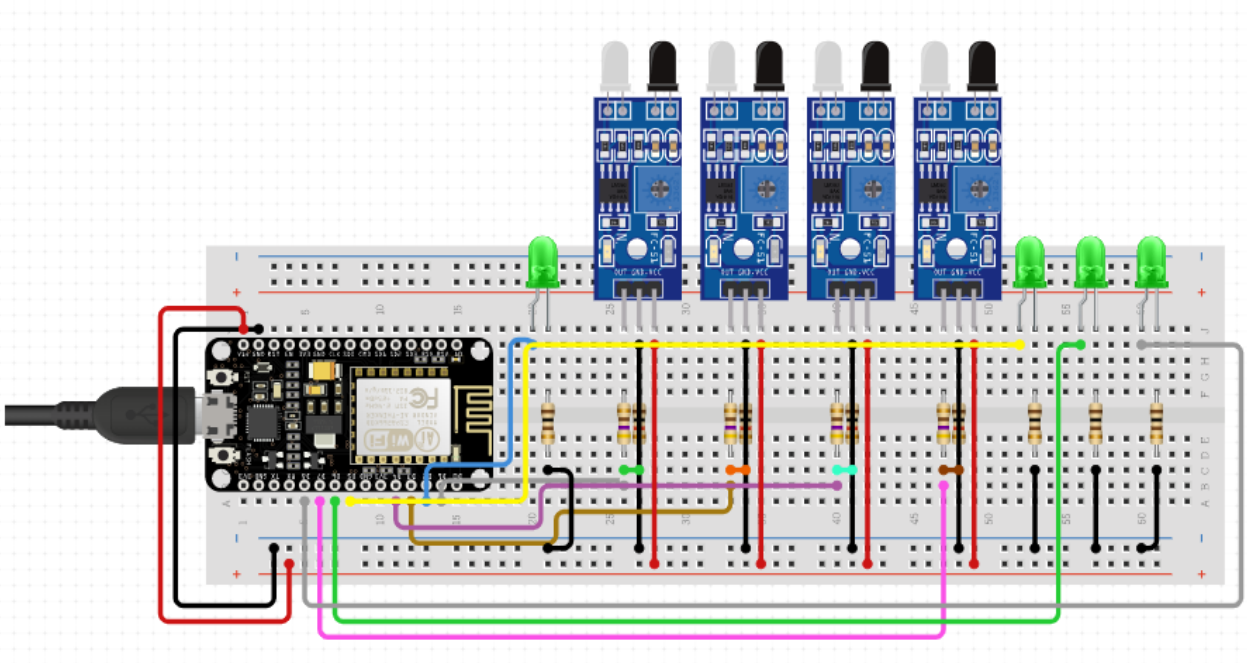


3.2.3 Sequence Diagram: -

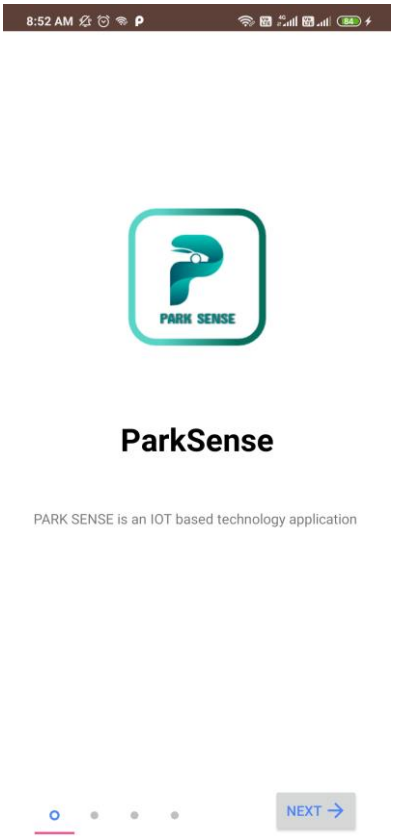


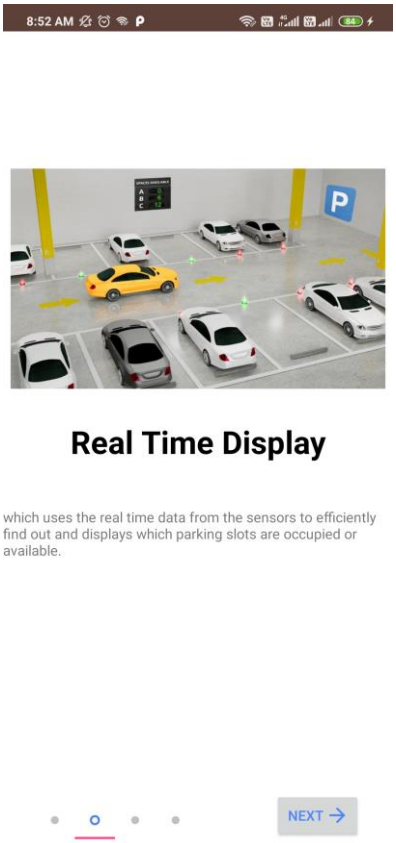
3.3 CIRCUIT DIAGRAM: -

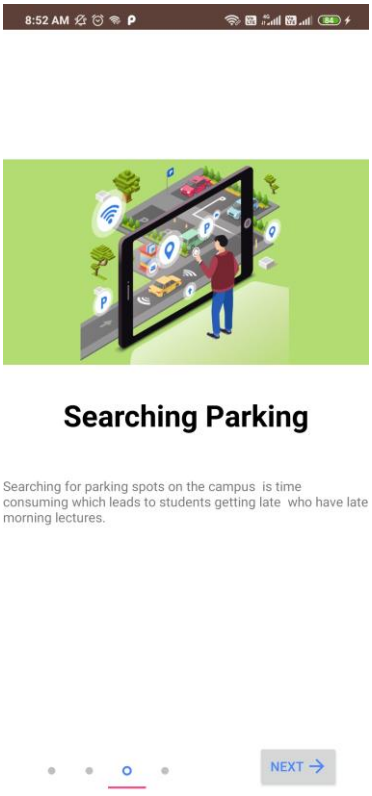


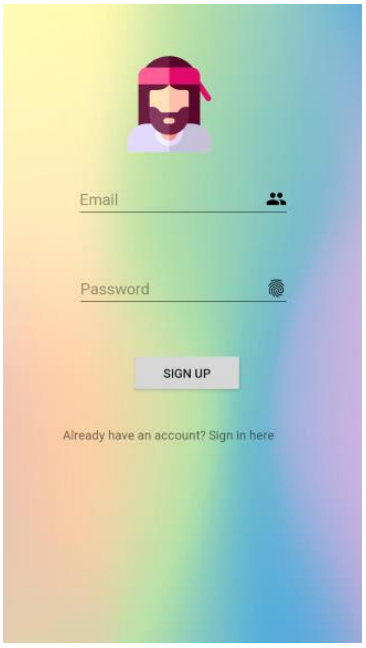


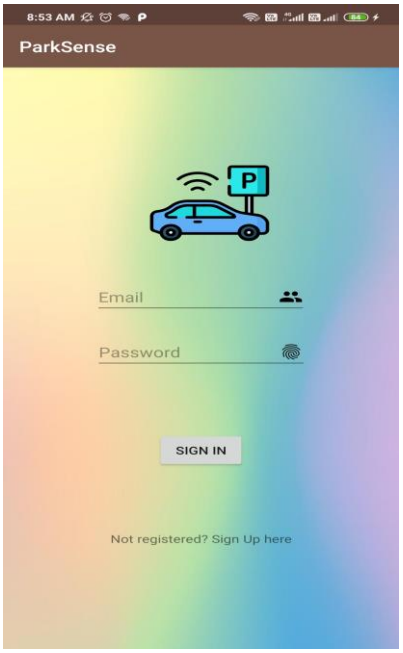
3.3 SCREEN SHOTS: -

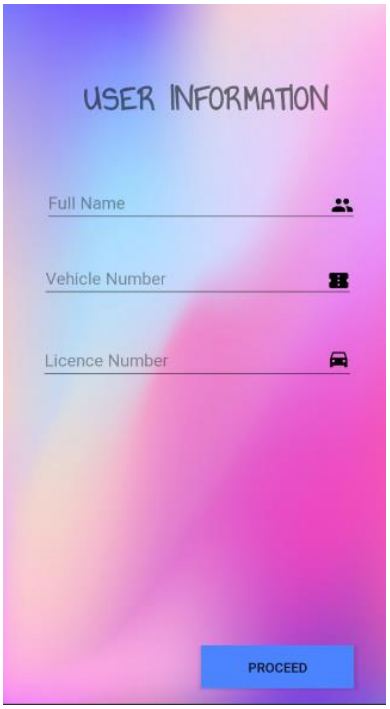


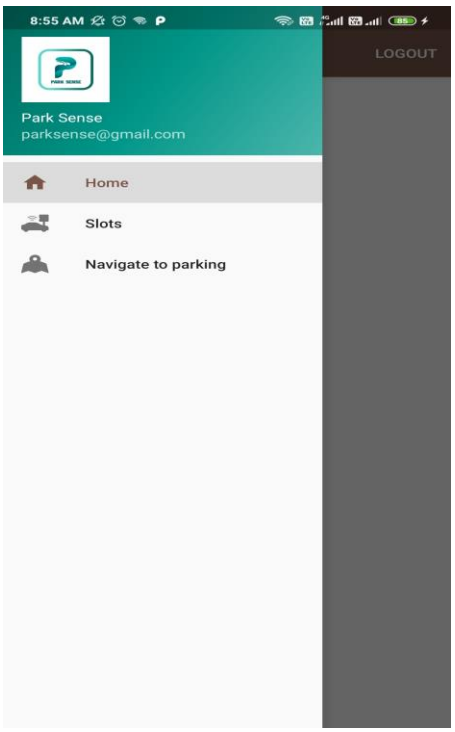


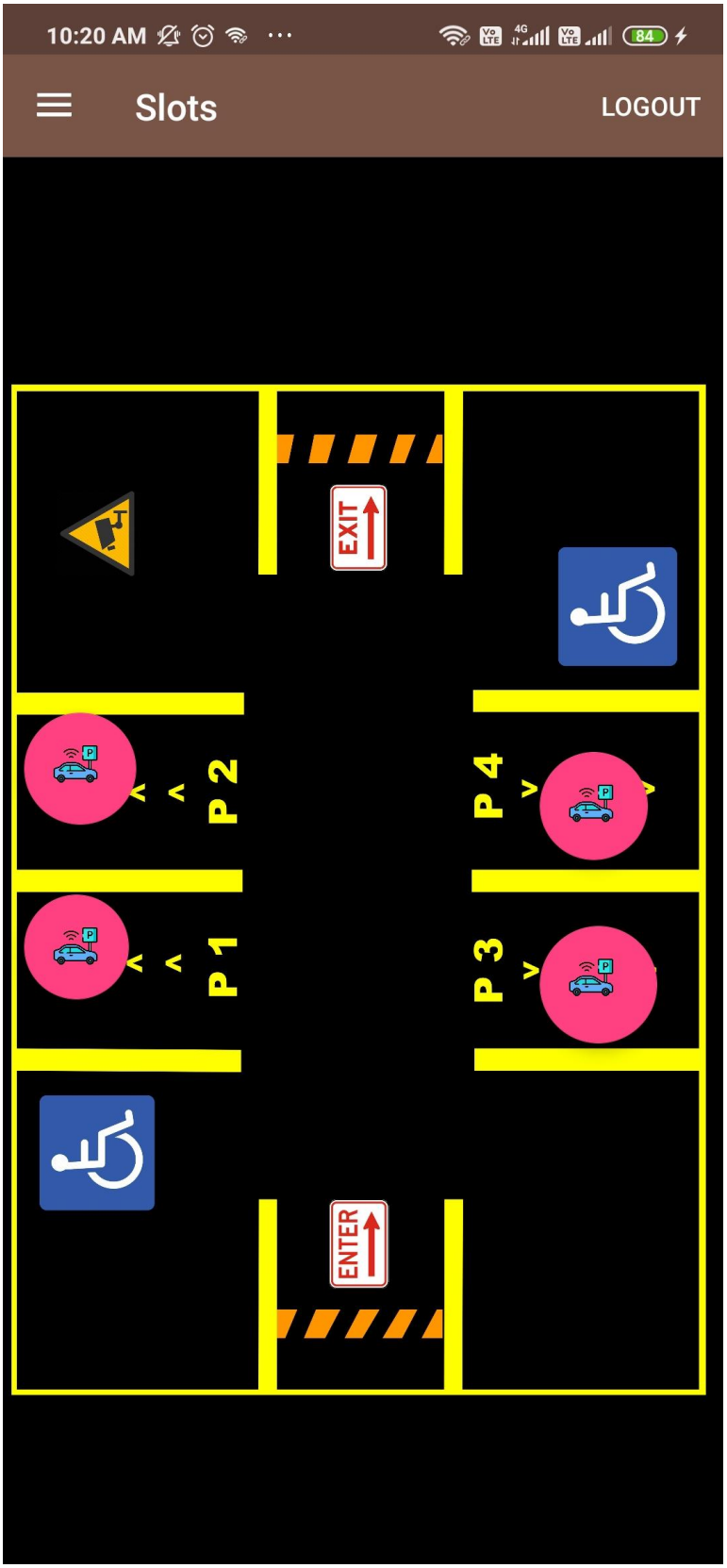



Email
Password
SIGN UP
Already have an account? Sign in here


8:53 AM
ParkSense
Email
Password
SIGN IN
Not registered? Sign Up here


USER INFORMATION
Full Name
Vehicle Number
Licence Number
PROCEED


8:55 AM
LOGOUT
Park Sense
parksense@gmail.com
Home
Slots
Navigate to parking


10:20 AM
Slots
LOGOUT
EXIT
P2
P4
P1
P3
ENTER

4. TESTING

4.1 TESTING & CHARACTERISTICS OF TESTING: -

Software testing determines the correctness, completeness and quality of software being developed. IEEE defines testing as 'the process of exercising or evaluating a system or system component by manual or automated means to verify that it satisfies specified requirements or to identify differences between expected and actual results.'

Software testing is closely related to the terms verification and validation. Verification refers to the process of ensuring that the software is developed according to its specifications. For verification, techniques like reviews, analysis, inspections and walkthroughs are commonly used. While validation refers to the process of checking that the developed software meets the requirements specified by the user. Verification and validation can be summarized thus as given here.

Verification: Is the software being developed in the right way?

Validation: Is the right software being developed?

Software testing is performed either manually or by using automated tools to make sure that the software is functioning in accordance with the user requirements. Various advantages associated with testing are listed below.

- It removes errors, which prevent software from producing outputs according to user requirements.
- It removes errors that lead to software failure.
- It ensures that the software conforms to business as well as user's needs.
- It ensures that the software is developed according to user requirements.
- It improves the quality of the software by removing maximum possible errors from it.

4.2 TEST CASES: -

DECESSION TABLE TEST CASES

Conditions	Test Case 1	Test Case 2	Test Case 3	Test Case 4
Sensor 1	T	F	F	T
Sensor 2	T	F	F	T
Sensor 3	T	T	F	F
Sensor 4	T	T	F	F
Empty Slot		X	X	X
Occupide Slot	X	X		X

S.no	Sensor 1	Sensor 2	Sensor 3	Sensor 4	Expected Output	Actual Output
1	Blocked	Unblocked	Unblocked	Unblocked	Empty/Occupied	Empty/Occupied
2	Blocked	Blocked	Unblocked	Unblocked	Empty/Occupied	Empty/Occupied
3	Unblocked	Unblocked	Unblocked	Unblocked	Empty	Empty
4	Blocked	Blocked	Blocked	Blocked	Occupied	Slot Full / Occupied

5. CODING

5.1 ANDROID CODE: -

Activity_intro.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".IntroActivity"
    android:background="#ffffff">

    <androidx.viewpager.widget.ViewPager
        android:id="@+id/screen_pager"
        android:layout_width="wrap_content"
        android:layout_height="0dp"
        android:layout_marginBottom="8dp"
        app:layout_constraintBottom_toTopOf="@+id/tabLayout"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0" >

    </androidx.viewpager.widget.ViewPager>

    <Button
        android:id="@+id/btn_next"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginEnd="32dp"
        android:layout_marginBottom="8dp"
        android:textColor="@color/gradient_start"
        android:drawableTint="@color/gradient_start"
        android:drawableRight="@drawable/ic_arrow_forward_black_24dp"
        android:text="Next"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent" />

    <com.google.android.material.tabs.TabLayout
        android:id="@+id/tabLayout"
        android:layout_width="162dp"
        android:layout_height="38dp"
        android:layout_marginStart="32dp"
        android:layout_marginBottom="8dp"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toStartOf="@+id/button"
        app:layout_constraintHorizontal_bias="0.533"
        app:layout_constraintStart_toStartOf="parent"
        app:tabBackground="@drawable/indicator_selector">


    </com.google.android.material.tabs.TabLayout>

    <Button
        android:id="@+id/btn_getStart"
        android:layout_width="134dp"
```

```
        android:layout_height="44dp"
        android:layout_marginBottom="64dp"
        android:background="@drawable/btn_gradient_style"
        android:text="Get Start"
        android:visibility="invisible"
        android:textColor="#fff"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />
```

IntroActivity.java

```
package com.example.parksense;

import androidx.appcompat.app.AppCompatActivity;
import androidx.viewpager.widget.ViewPager;

import android.content.Intent;
import android.content.SharedPreferences;
import android.os.Bundle;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.Button;

import com.google.android.material.tabs.TabLayout;

import java.util.ArrayList;
import java.util.List;

public class IntroActivity extends AppCompatActivity {

    private ViewPager screenPager;
    IntroViewPageAdapter introViewPageAdapter;
    TabLayout tabindicator;
    Button btnNext;
    int position = 0;
    Button btngetstarted;
    Animation btnAnim;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_intro);

        // full screen
      //  requestWindowFeature(Window.FEATURE_NO_TITLE);
       // getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
            //   WindowManager.LayoutParams.FLAG_FULLSCREEN);

        if (restorePrefData()){
            Intent mainActivity = new
Intent(getApplicationContext(),MainActivity.class);
            startActivity(mainActivity);
            finish();
        }
```

```
       // setContentView(R.layout.activity_intro);

        //hide action bar

        getSupportActionBar().hide();

        tabindicator = findViewById(R.id.tabLayout);
        btnNext = findViewById(R.id.btn_next);
        btngetstarted = findViewById(R.id.btn_getStart);
        btnAnim =
AnimationUtils.loadAnimation(getApplicationContext(),R.anim.button_animation);

        final List<Screenitem> mList = new ArrayList<>();
        mList.add(new Screenitem("ParkSense","PARK SENSE is an IOT based technology
application ",R.drawable.logo));
        mList.add(new Screenitem("Real Time Display","which uses the real time data from
the sensors to efficiently find out and displays which parking slots are occupied or
available.",R.drawable.img2));
        mList.add(new Screenitem("Searching Parking","Searching for parking spots on the
campus  is time consuming which leads to students getting late  who have late morning
lectures.",R.drawable.img3));
        mList.add(new Screenitem("Smart Parking","Park Sense provides simple solution
for campus as well as commercial parking space around the city",R.drawable.img4));

        screenPager = findViewById(R.id.screen_pager);
        introViewPageAdapter = new IntroViewPageAdapter(this,mList);
        screenPager.setAdapter(introViewPageAdapter);

        tabindicator.setupWithViewPager(screenPager);

        btnNext.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                position = screenPager.getCurrentItem();
                if (position < mList.size()){

                    position++;
                    screenPager.setCurrentItem(position);
                }

                if (position == mList.size()-1){

                    loadLastScreen();
                }

            }
        });

        tabindicator.addOnTabSelectedListener(new TabLayout.BaseOnTabSelectedListener()
{
            @Override
            public void onTabSelected(TabLayout.Tab tab) {

                if (tab.getPosition() == mList.size()-1) {
                    loadLastScreen();
                }
            }
```

```
            @Override
            public void onTabUnselected(TabLayout.Tab tab) {

            }

            @Override
            public void onTabReselected(TabLayout.Tab tab) {

            }
        });

        btngetstarted.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {

                Intent mainActivity = new
Intent(getApplicationContext(),MainActivity.class);
                startActivity(mainActivity);

                savePrefsData();
                finish();
            }
        });

    }

    private boolean restorePrefData() {

        SharedPreferences pref =
getApplicationContext().getSharedPreferences("mypref",MODE_PRIVATE);
        Boolean isIntroActivityOpenBefore = pref.getBoolean("isIntroOpened",false);
        return  isIntroActivityOpenBefore;


    }

    private void savePrefsData() {

        SharedPreferences pref =
getApplicationContext().getSharedPreferences("mypref",MODE_PRIVATE);
        SharedPreferences.Editor editor = pref.edit();
        editor.putBoolean("isIntroOpened",true);
        editor.commit();
    }

    private void  loadLastScreen() {

        btnNext.setVisibility(View.INVISIBLE);
        btngetstarted.setVisibility(View.VISIBLE);
        tabindicator.setVisibility(View.INVISIBLE);

        btngetstarted.setAnimation(btnAnim);

    }
}
```

IntroViewPageAdapter.java

```java
package com.example.parksense;

import android.content.Context;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.ImageView;
import android.widget.TextView;

import androidx.annotation.NonNull;
import androidx.viewpager.widget.PagerAdapter;

import java.util.List;

public class IntroViewPageAdapter extends PagerAdapter {

    Context mContext;
    List<Screenitem> mListScreen;

    public IntroViewPageAdapter(Context mContext, List<Screenitem> mListScreen) {
        this.mContext = mContext;
        this.mListScreen = mListScreen;
    }


    @NonNull
    @Override
    public Object instantiateItem(@NonNull ViewGroup container, int position) {
        LayoutInflater inflater = (LayoutInflater)
mContext.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        View layoutScreen = inflater.inflate(R.layout.layout_screen,null);

        ImageView imageView = layoutScreen.findViewById(R.id.intro_img);
        TextView title = layoutScreen.findViewById(R.id.intro);
        TextView desc = layoutScreen.findViewById(R.id.intro_desc);

        title.setText(mListScreen.get(position).getTitle());
        desc.setText(mListScreen.get(position).getDesc());
        imageView.setImageResource(mListScreen.get(position).getScreenImg());

        container.addView(layoutScreen);

        return layoutScreen;

    }

    @Override
    public int getCount() {
        return mListScreen.size();
    }

    @Override
    public boolean isViewFromObject(@NonNull View view, @NonNull Object o) {
        return view == o;
    }

    @Override
    public void destroyItem(@NonNull ViewGroup container, int position, @NonNull Object
object) {
```

```
        container.removeView((View)object);
    }
}
```

ScreenItem.java

```
package com.example.parksense;

public class Screenitem {

    String Title,Desc;
    int ScreenImg;

    public Screenitem(String title, String desc, int screenImg) {
        Title = title;
        Desc = desc;
        ScreenImg = screenImg;
    }


    public void setTitle(String title) {
        Title = title;
    }

    public void setDesc(String desc) {
        Desc = desc;
    }

    public void setScreenImg(int screenImg) {
        ScreenImg = screenImg;
    }

    public String getTitle() {
        return Title;
    }

    public String getDesc() {
        return Desc;
    }

    public int getScreenImg() {
        return ScreenImg;
    }
}
```

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/dsa"
    tools:context=".MainActivity">

    <EditText
        android:id="@+id/editText"
```

```
        android:layout_width="326dp"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="84dp"
        android:layout_marginLeft="84dp"
        android:layout_marginTop="203dp"
        android:layout_marginEnd="85dp"
        android:layout_marginRight="85dp"
        android:layout_marginBottom="483dp"
        android:ems="10"
        android:hint="Email"
        android:drawableRight="@drawable/ic_people_black_24dp"
        android:inputType="textEmailAddress"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editText2"
        android:layout_width="278dp"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="85dp"
        android:layout_marginTop="304dp"
        android:layout_marginEnd="85dp"
        android:layout_marginBottom="382dp"
        android:ems="10"
        android:hint="Password"
        android:drawableRight="@drawable/ic_fingerprint_black_24dp"
        android:inputType="textPassword"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText"
        app:layout_constraintVertical_bias="0.052" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="242dp"
        android:layout_height="35dp"
        android:layout_alignParentStart="true"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="69dp"
        android:layout_marginEnd="100dp"
        android:layout_marginBottom="215dp"
        android:text="Already have an account? Sign in here"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />

    <Button
        android:id="@+id/button2"
        android:layout_width="125dp"
        android:layout_height="wrap_content"
        android:layout_alignParentStart="true"
```

```
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginStart="147dp"
        android:layout_marginTop="8dp"
        android:layout_marginEnd="139dp"
        android:layout_marginBottom="286dp"
        android:text="Sign Up"
        app:layout_constraintBottom_toTopOf="@+id/textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText2" />

    <ImageView
        android:id="@+id/imageView3"
        android:layout_width="108dp"
        android:layout_height="129dp"
        android:layout_alignParentEnd="true"
        android:layout_alignParentBottom="true"
        android:layout_marginEnd="168dp"
        android:layout_marginBottom="549dp"
        app:srcCompat="@drawable/ic_user" />
</RelativeLayout>
```

MainActivity.java

```
package com.example.parksense;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
public class MainActivity extends AppCompatActivity {
    EditText emailId, password;
    Button btnSignUp;
    TextView tvSignIn;
    FirebaseAuth mFirebaseAuth;


    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        mFirebaseAuth = FirebaseAuth.getInstance();
        emailId = findViewById(R.id.editText);
        password = findViewById(R.id.editText2);
        btnSignUp = findViewById(R.id.button2);
        tvSignIn = findViewById(R.id.textView);
        btnSignUp.setOnClickListener(new View.OnClickListener() {
            @Override
```

```
            public void onClick(View v) {
                String email = emailId.getText().toString();
                String pwd = password.getText().toString();
                if(email.isEmpty()){
                    emailId.setError("Please enter email id");
                    emailId.requestFocus();
                }
                else  if(pwd.isEmpty()){
                    password.setError("Please enter your password");
                    password.requestFocus();
                }
                else  if(email.isEmpty() && pwd.isEmpty()){
                    Toast.makeText(MainActivity.this,"Fields Are
Empty!",Toast.LENGTH_SHORT).show();
                }
                else  if(!(email.isEmpty() && pwd.isEmpty())){
                    mFirebaseAuth.createUserWithEmailAndPassword(email,
pwd).addOnCompleteListener(MainActivity.this, new OnCompleteListener<AuthResult>() {
                        @Override
                        public void onComplete( Task<AuthResult> task) {
                            if(!task.isSuccessful()){
                                Toast.makeText(MainActivity.this,"SignUp Unsuccessful,
Please Try Again",Toast.LENGTH_SHORT).show();
                            }
                            else {
                                startActivity(new
Intent(MainActivity.this,HomeActivity.class));
                            }
                        }
                    });
                }
                else{
                    Toast.makeText(MainActivity.this,"Error
Occurred!",Toast.LENGTH_SHORT).show();

                }
            }
        });

        tvSignIn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent i = new Intent(MainActivity.this,LoginActivity.class);
                startActivity(i);
            }
        });


    }
}
```

activity_login.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
```

```
    android:layout_height="match_parent"
    android:background="@drawable/dsa"
    tools:context=".LoginActivity">

    <EditText
        android:id="@+id/editText"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="313dp"
        android:layout_marginEnd="85dp"
        android:layout_marginRight="85dp"
        android:layout_marginStart="84dp"
        android:layout_marginLeft="84dp"
        android:layout_marginTop="152dp"
        android:ems="10"
        android:hint="Email"
        android:drawableRight="@drawable/ic_people_black_24dp"
        android:inputType="textEmailAddress"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:id="@+id/editText2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginEnd="85dp"
        android:layout_marginStart="85dp"
        android:layout_marginTop="8dp"
        android:ems="10"
        android:hint="Password"
        android:drawableRight="@drawable/ic_fingerprint_black_24dp"
        android:inputType="textPassword"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText"
        app:layout_constraintVertical_bias="0.052" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="136dp"
        android:layout_marginEnd="163dp"
        android:layout_marginStart="163dp"
        android:text="Not registered? Sign Up here"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent" />

    <Button
        android:id="@+id/button2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginEnd="8dp"
```

```
        android:layout_marginStart="8dp"
        android:layout_marginTop="8dp"
        android:text="Sign In"
        app:layout_constraintBottom_toTopOf="@+id/textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/editText2" />

    <ImageView
        android:id="@+id/imageView4"
        android:layout_width="251dp"
        android:layout_height="118dp"
        android:layout_marginTop="120dp"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.562"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:srcCompat="@drawable/ic_car" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

LoginActivity.java

```
package com.example.parksense;

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;

public class LoginActivity extends AppCompatActivity {

    EditText emailId, password;
    Button btnSignIn;
    TextView tvSignUp;
    FirebaseAuth mFirebaseAuth;

    private FirebaseAuth.AuthStateListener mAuthStateListener;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);

        mFirebaseAuth = FirebaseAuth.getInstance();
        emailId = findViewById(R.id.editText);
        password = findViewById(R.id.editText2);
```

```
        btnSignIn = findViewById(R.id.button2);
        tvSignUp = findViewById(R.id.textView);

        mAuthStateListener = new FirebaseAuth.AuthStateListener() {
            @Override
            public void onAuthStateChanged( FirebaseAuth firebaseAuth) {
                FirebaseUser mFirebaseUser = mFirebaseAuth.getCurrentUser();
                if( mFirebaseUser != null ){
                    Toast.makeText(LoginActivity.this,"You are logged
in",Toast.LENGTH_SHORT).show();
                    Intent i = new Intent(LoginActivity.this, HomeActivity.class);
                    startActivity(i);
                }
                else{
                    Toast.makeText(LoginActivity.this,"Please
Login",Toast.LENGTH_SHORT).show();
                }
            }
        };

        btnSignIn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String email = emailId.getText().toString();
                String pwd = password.getText().toString();
                if(email.isEmpty()){
                    emailId.setError("Please enter email id");
                    emailId.requestFocus();
                }
                else  if(pwd.isEmpty()){
                    password.setError("Please enter your password");
                    password.requestFocus();
                }
                else  if(email.isEmpty() && pwd.isEmpty()){
                    Toast.makeText(LoginActivity.this,"Fields Are
Empty!",Toast.LENGTH_SHORT).show();
                }
                else  if(!(email.isEmpty() && pwd.isEmpty())){
                    mFirebaseAuth.signInWithEmailAndPassword(email,
pwd).addOnCompleteListener(LoginActivity.this, new OnCompleteListener<AuthResult>() {
                        @Override
                        public void onComplete( Task<AuthResult> task) {
                            if(!task.isSuccessful()){
                                Toast.makeText(LoginActivity.this,"Login Error, Please
Login Again",Toast.LENGTH_SHORT).show();
                            }
                            else{
                                Intent intToHome = new
Intent(LoginActivity.this,DashboardActivity.class);
                                startActivity(intToHome);
                            }
                        }
                    });
                }
                else{
                    Toast.makeText(LoginActivity.this,"Error
Occurred!",Toast.LENGTH_SHORT).show();

                }
```

```
            }
        });

        tvSignUp.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intSignUp = new Intent(LoginActivity.this, MainActivity.class);
                startActivity(intSignUp);
            }
        });
    }

    @Override
    protected void onStart() {
        super.onStart();
        mFirebaseAuth.addAuthStateListener(mAuthStateListener);
    }
}
```

activity_home.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@drawable/grad"
    tools:context=".HomeActivity">

    <EditText
        android:id="@+id/username"
        android:layout_width="328dp"
        android:layout_height="39dp"
        android:layout_marginTop="132dp"
        android:layout_marginEnd="36dp"
        android:layout_marginBottom="500dp"
        android:drawableRight="@drawable/ic_people_black_24dp"
        android:ems="10"
        android:hint="Full Name"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="1.0" />

    <EditText
        android:id="@+id/vehiclenumber"
        android:layout_width="327dp"
        android:layout_height="40dp"
        android:layout_marginTop="39dp"
        android:layout_marginEnd="40dp"
        android:layout_marginBottom="200dp"
        android:drawableRight="@drawable/ic_confirmation_number_black_24dp"
        android:ems="10"
        android:hint="Vehicle Number"
```

```
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/username"
        app:layout_constraintVertical_bias="0.0" />

    <EditText
        android:id="@+id/licence"
        android:layout_width="328dp"
        android:layout_height="45dp"
        android:layout_marginTop="39dp"
        android:layout_marginEnd="40dp"
        android:layout_marginBottom="300dp"
        android:drawableRight="@drawable/ic_drive_eta_black_24dp"
        android:ems="10"
        android:hint="Licence Number"
        android:inputType="textPersonName"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/vehiclenumber"
        app:layout_constraintVertical_bias="0.043" />

    <Button
        android:id="@+id/save"
        android:layout_width="160dp"
        android:layout_height="45dp"
        android:layout_marginTop="325dp"
        android:layout_marginEnd="40dp"
        android:layout_marginBottom="16dp"
        android:text="Proceed"
        android:background="#4c84ff"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/licence"
        app:layout_constraintVertical_bias="1.0" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="275dp"
        android:layout_height="55dp"
        android:layout_marginStart="88dp"
        android:fontFamily="@font/annie_use_your_telescope"
        android:text="USER INFORMATION"
        android:textSize="36sp"
        android:textStyle="bold"
        app:layout_constraintBottom_toTopOf="@+id/username"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.503" />

</androidx.constraintlayout.widget.ConstraintLayout>
```

HomeActivity.java

```
package com.example.parksense;

import androidx.appcompat.app.AppCompatActivity;
```

```
import android.content.Intent;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;

import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;

public class HomeActivity extends AppCompatActivity {

    private EditText username,vehiclenumber,licence;
    private Button save,next;

    DatabaseReference databaseReference;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_home);

        databaseReference = FirebaseDatabase.getInstance().getReference("user");
        username = (EditText)findViewById(R.id.username);
        vehiclenumber = (EditText)findViewById(R.id.vehiclenumber);
        licence = (EditText)findViewById(R.id.licence);

        save = (Button)findViewById(R.id.save);


        save.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                adduser();
            }
        });
    }

    public void  adduser(){

        String name = username.getText().toString();
        String vehicle = vehiclenumber.getText().toString();
        String licenceno = licence.getText().toString();

        if (!TextUtils.isEmpty(name) && !TextUtils.isEmpty(vehicle) &&
!TextUtils.isEmpty(licenceno)){

            String id = databaseReference.push().getKey();
             UserInfo userInfo = new UserInfo(id,name,vehicle,licenceno);
             databaseReference.child(id).setValue(userInfo);
             username.setText("");
             vehiclenumber.setText("");
             licence.setText("");

            Intent dashboardActivity  = new
Intent(getApplicationContext(),DashboardActivity.class);
            startActivity(dashboardActivity);
        }
```

```
        else {
            Toast.makeText(HomeActivity.this,"Enter the
Details",Toast.LENGTH_LONG).show();
        }
    }
}
```

DashboardActivity.java

```
package com.example.parksense;

import android.content.Intent;
import android.os.Bundle;
import android.view.MenuItem;
import android.view.View;
import android.view.Menu;

import com.google.android.material.floatingactionbutton.FloatingActionButton;
import com.google.android.material.snackbar.Snackbar;
import com.google.android.material.navigation.NavigationView;

import androidx.navigation.NavController;
import androidx.navigation.Navigation;
import androidx.navigation.ui.AppBarConfiguration;
import androidx.navigation.ui.NavigationUI;
import androidx.drawerlayout.widget.DrawerLayout;
import androidx.appcompat.app.AppCompatActivity;
import androidx.appcompat.widget.Toolbar;

public class DashboardActivity extends AppCompatActivity {

    private AppBarConfiguration mAppBarConfiguration;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_dashboard);
        Toolbar toolbar = findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);

        DrawerLayout drawer = findViewById(R.id.drawer_layout);
        NavigationView navigationView = findViewById(R.id.nav_view);
        // Passing each menu ID as a set of Ids because each
        // menu should be considered as top level destinations.
        mAppBarConfiguration = new AppBarConfiguration.Builder(
                R.id.nav_home, R.id.nav_gallery, R.id.nav_slideshow)
                .setDrawerLayout(drawer)
                .build();
        NavController navController = Navigation.findNavController(this,
R.id.nav_host_fragment);
        NavigationUI.setupActionBarWithNavController(this, navController,
mAppBarConfiguration);
        NavigationUI.setupWithNavController(navigationView, navController);
    }

    @Override
    public boolean onCreateOptionsMenu(Menu menu) {
        // Inflate the menu; this adds items to the action bar if it is present.
```

```
        getMenuInflater().inflate(R.menu.dashboard, menu);
        return true;
    }

    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        switch (item.getItemId()) {
            case R.id.action_settings:
                Intent bck = new Intent(DashboardActivity.this,MainActivity.class);
                startActivity(bck);
                return true;



            default:
                // If we got here, the user's action was not recognized.
                // Invoke the superclass to handle it.
                return super.onOptionsItemSelected(item);

        }
    }


    @Override
    public boolean onSupportNavigateUp() {
        NavController navController = Navigation.findNavController(this,
R.id.nav_host_fragment);
        return NavigationUI.navigateUp(navController, mAppBarConfiguration)
                || super.onSupportNavigateUp();
    }
}
```

fragment_gallary.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#000000"
    tools:context=".ui.gallery.GalleryFragment">

    <ImageView
        android:id="@+id/imageView2"
        android:layout_width="438dp"
        android:layout_height="702dp"
        android:src="@drawable/vs"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.57"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.73" />

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/slot2"
```

```
        android:layout_width="61dp"
        android:layout_height="66dp"
        android:layout_marginEnd="320dp"
        android:clickable="true"
        android:src="@drawable/ic_car"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.429" />

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/slot1"
        android:layout_width="57dp"
        android:layout_height="84dp"
        android:layout_marginTop="35dp"
        android:layout_marginEnd="324dp"
        android:clickable="true"
        android:src="@drawable/ic_car"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/slot2"
        app:layout_constraintVertical_bias="0.01" />

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/slot4"
        android:layout_width="86dp"
        android:layout_height="59dp"
        android:layout_marginStart="220dp"
        android:layout_marginBottom="384dp"
        android:clickable="true"
        android:src="@drawable/ic_car"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toEndOf="@+id/slot2" />

    <com.google.android.material.floatingactionbutton.FloatingActionButton
        android:id="@+id/slot3"
        android:layout_width="64dp"
        android:layout_height="72dp"
        android:layout_marginStart="224dp"
        android:layout_marginBottom="284dp"
        android:clickable="true"
        android:src="@drawable/ic_car"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintStart_toEndOf="@+id/slot1" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

Google_maps_api.xml

```
<resources>
    <!--
    TODO: Before you run your application, you need a Google Maps API key.

    To get one, follow this link, follow the directions and press "Create" at the end:


https://console.developers.google.com/flows/enableapi?apiid=maps_android_backend&keyType
=CLIENT_SIDE_ANDROID&r=3B:E7:9F:1C:9B:C7:76:EC:7C:BD:A1:0D:A4:FF:4B:C6:75:E6:73:92%3Bcom
.example.parksense.ui.slideshow
```

```
    You can also add your credentials to an existing key, using these values:

    Package name:
    com.example.parksense.ui.slideshow

    SHA-1 certificate fingerprint:
    3B:E7:9F:1C:9B:C7:76:EC:7C:BD:A1:0D:A4:FF:4B:C6:75:E6:73:92

    Alternatively, follow the directions here:
    https://developers.google.com/maps/documentation/android/start#get-key

    Once you have your key (it starts with "AIza"), replace the "google_maps_key"
    string in this file.
    -->
    <string name="google_maps_key" templateMergeStrategy="preserve"
translatable="false">AIzaSyCZKifoe7UpP2DBSPBVWH4wJZY4TSIbesU</string>
</resources>
```

MapActivity.java

```
package com.example.parksense;

import androidx.fragment.app.FragmentActivity;

import android.os.Bundle;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.SupportMapFragment;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

public class MapsActivity extends FragmentActivity implements OnMapReadyCallback {

    private GoogleMap mMap;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_maps);
        // Obtain the SupportMapFragment and get notified when the map is ready to be
used.
        SupportMapFragment mapFragment = (SupportMapFragment)
getSupportFragmentManager()
                .findFragmentById(R.id.map);
        mapFragment.getMapAsync(this);
    }


    /**
     * Manipulates the map once available.
     * This callback is triggered when the map is ready to be used.
     * This is where we can add markers or lines, add listeners or move the camera. In
this case,
     * we just add a marker near Sydney, Australia.
     * If Google Play services is not installed on the device, the user will be prompted
```

```
to install
     * it inside the SupportMapFragment. This method will only be triggered once the
user has
     * installed Google Play services and returned to the app.
     */
    @Override
    public void onMapReady(GoogleMap googleMap) {
        mMap = googleMap;

        // Add a marker in Sydney and move the camera
        LatLng pune = new LatLng(18.516268, 73.815122);
        mMap.addMarker(new MarkerOptions().position(pune).title("MIT World Peace
University,Pune"));
        mMap.moveCamera(CameraUpdateFactory.newLatLng(pune));
    }
}
```

5.2 ARDUNIO CODE

```
#include <ESP8266WiFi.h>

#include <SoftwareSerial.h>

#include <FirebaseArduino.h>

#include <ArduinoJson.h>

#include <ESP8266HTTPClient.h>


// Set these to run example.

#define FIREBASE_HOST "smart-parking-system-3546b.firebaseio.com"

#define FIREBASE_AUTH "yYuliNzC1IkBhcIfuGjm4gIzUbnokvZzJJTZsNGf"

#define WIFI_SSID "Redmi"

#define WIFI_PASSWORD "ufuckoff"


//String myString;

int s1 = D0;// variable resistor connected

int s2 = D1;

int s3 = D2;

int s4 = D3;
```

```
int sdata1 = 0;// The variable resistor value will be stored in sdata.
int sdata2 = 0;
int sdata3 = 0;
int sdata4 = 0;

void setup()
{
  // Debug console
  Serial.begin(9600);
  pinMode(s1 ,INPUT);
  pinMode(s2,INPUT);
   pinMode(s3,INPUT);
    pinMode(s4,INPUT);
  // connect to wifi.
  pinMode(A0,OUTPUT);
  WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
  Serial.print("connecting");
  while (WiFi.status() != WL_CONNECTED)
     {
    Serial.print(".");
    delay(500);
     }
  Serial.println();
  Serial.print("connected: ");
  Serial.println(WiFi.localIP());

  Firebase.begin(FIREBASE_HOST, FIREBASE_AUTH);
```

```
}

void loop()
{



sdata1 = digitalRead(s1);

if(sdata1 == 1)
{
  Serial.println("SLOT 1: UN-PARKED");
Firebase.setInt("SENSOR 1 value",sdata1);
}
else
{
    Serial.println("SLOT 1: PARKED");
Firebase.setInt("SENSOR 1 value",sdata1);
}
sdata2 = digitalRead(s2);

if(sdata2 == 1)
{
  Serial.println("SLOT 2: UN-PARKED");
Firebase.setInt("SENSOR 2 value",sdata2);
}
```

```
else
{
   Serial.println("SLOT 2: PARKED");
Firebase.setInt("SENSOR 2 value",sdata2);
}
sdata3 = digitalRead(s3);

if(sdata3 == 1)
{
  Serial.println("SLOT 3: UN-PARKED");
Firebase.setInt("SENSOR 3 value",sdata3);
}
else
{
   Serial.println("SLOT 3: PARKED");
Firebase.setInt("SENSOR 3 value",sdata3);
}
sdata4 = digitalRead(s4);

if(sdata4 == 1)
{
  Serial.println("SLOT 4: UN-PARKED");
Firebase.setInt("SENSOR 4 value",sdata4);
}
else
{
   Serial.println("SLOT 4: PARKED");
```

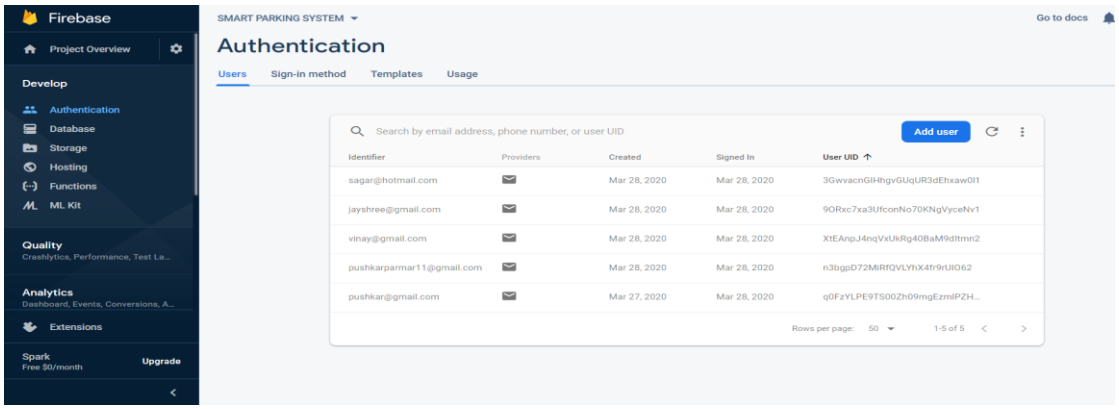
Firebase.setInt("SENSOR 4 value",sdata4);

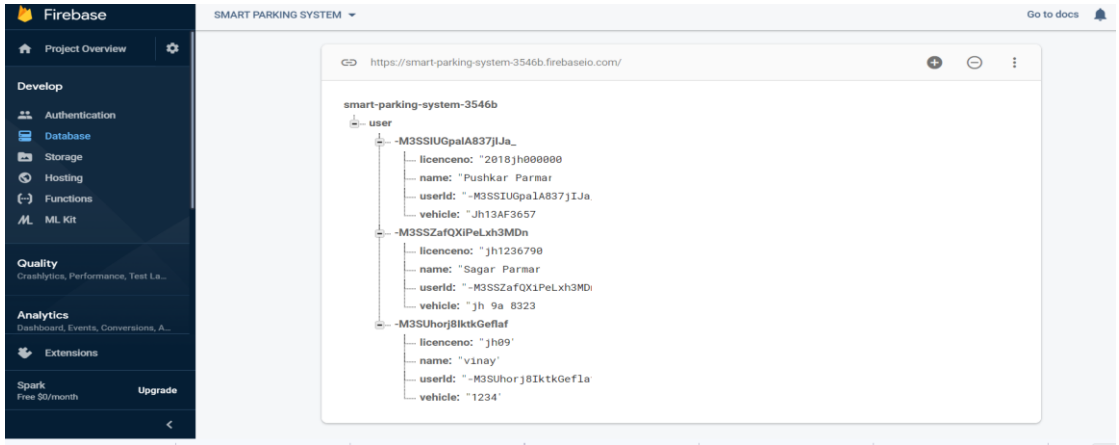
}

delay(150);

}

FIREBASE





6. CONCLUSION

6.1 Limitations and Drawbacks

1. Our Application doesn’t have booking slots.
2. Our system doesn’t keep record of the driver who parked the car.
3. Maps and navigation are not so perfect they need more advancement.

6.2 Future Enhancement

1. We will make app for facial recognition and Number Plate Recognition.
2. We will keep the record of driver.
3. There will Parking Slots for booking.
4. We charge some amounts for parking as compare to time they park.
5. Payment method will be included.

6.3 Conclusion

This project focuses on implementation of car parking place detection using Internet of Things and we make android application for this parking solution.

The system benefits of smart parking go well beyond avoiding time wasting.

Developing a smart parking solution with in a city solves the pollution problem.