TempoWala

A Project Report

Submitted in partial fulfillment of the Requirements for the award of the Degree of

BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)

By

Mayuresh Dinesh Rawal

3102974

Under the esteemed guidance of

Prof. R. Arogya Selvi

DEPARTMENT OF INFORMATION TECHNOLOGY

S.I.W.S

N.R. SWAMY COLLEGE OF COMMERCE AND ECONOMICS AND SMT.THIRUMALAI COLLEGE OF SCIENCE

Affiliated to University of Mumbai

MUMBAI , 400031

MAHARASHTRA

2020-2021

PROFORMA FOR THE APPROVAL PROJECT PROPOSAL

PRN No.: 2018016400862882 Roll no: 19036

Name of Student: Mayuresh Dinesh Rawal

Title of the Project: TempoWala

Name of the Guide: Prof. R. Arogya Selvi

Teaching Experience of the Guide:

Is this your first submission? Yes ● No □

Rawal

Signature of Student

Signature of Guide

Date: ……………..

Signature of the Co-ordinator

Date: ……………..

S.I.W.S

N.R. SWAMY COLLEGE OF COMMERCE AND ECONOMICS AND SMT.THIRUMALAI COLLEGE OF SCIENCE

Affiliated to University of Mumbai

MUMBAI-MAHARASHTRA-400031

DEPARTMENT OF INFORMATION TECHNOLOGY

CERTIFICATE

This is to certify that the project entitled, "**TempoWala**", is bonafied work of **Mayuresh Dinesh Rawal** bearing Seat. No: (**3102974**) submitted in partial fulfillment of the requirements for the award of degree of BACHELOR OF SCIENCE in INFORMATION TECHNOLOGY from University of Mumbai.

Internal Guide **Coordinator**

External Examiner

Date: **College Seal**

ABSTRACT

TempoWala is an Android app designed to help my friend's Tempo Service and also helping people to book without any hassles also allowing new TempoWalas to join in.

The system uses address based approach, thus it ensures that the pickup and destination locations are dealt with properly. It also ensures well documentation of the address. The user interface is designed for simplicity and its minimalistic design makes user experience smoother.

When the user opens the app there's a home screen showing 4 buttons. They are as follows: 1.Book Now, 2.Booking History, 3.Join us, 4.FAQ.
1. Book Now – Leads User to the booking page. The user can fill in details and book a tempo.
2. Booking History – Once booked the user can check the booking history from where the user can also EDIT/CANCEL/DELETE the booking.
3. Join us – This page opens a form to which new TempoWalas can enroll to join in. Once submitted it is reviewed by the admin.
4. FAQ – This is simple FAQ page, but wait there's more to it like Fare Calculator, Video Tutorials on how to BOOK/EDIT-CANCEL-DELETE/JOINUS, Email Complaint ,etc.

More side features include **offline compatibility**. Yes the user can use the app to **book Tempos even with no internet connection** this is done via SMS. Also the app comes with inbuilt notification and in app messaging listeners. Which means the app will receive notifications and in app messages by the admin.

The majority of these functionalities depend upon Google's own Firebase Database service. This is much more than a database and far superior than any other database systems in use. Using Firebase as database opened so many options for me to develop this app and extend its features. Overall the app is all complete functionally and can be used in real market.

ACKNOWLEDGEMENT

The completion of this project made me realize, that even I can code and develop apps. In this journey of developing this app I learnt a lot of new things and made so many mistakes which helped me to be a better developer.

Whenever I got stuck there was always a video, a forum, or just Stack Overflow for that matter. Every problem had a solution and which I overcame slowly and steadily and developed this app. Though the app is not that professional but it's functional and that's what matters most to me. There's a lot to learn and explore and the journey has just begun.

I would like to thank my teacher Prof. R. Arogya Selvi Mam for helping whenever I asked for help, without her kind gesture of helping this project wouldn't be possible. She also suggested some ideas (e.g.: Fare calculator) which I never thought of and implemented those ideas in my project.

DECLARATION

I hereby declare that the project entitled, "**TempoWala**" done at S.I.W.S College, MUMBAI, has not been in any case duplicated to submit to any other university for the award of any degree. To the best of my knowledge other than me, no one has submitted to any other university.

The project is done in partial fulfillment of the requirements for the award of degree of **BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)** to be submitted as final semester project as part of our curriculum.

Rawal

Mayuresh Dinesh Rawal

CONTENTS

Chapter 1 - Introduction

1.1 Background:

The background of this project comes from a friend looking to expand his business of Tempo Service. Porter an existing app, working in the same field has been providing tempo services from around 2014. This means the app has seen a lot of development changes and updates through the time and also has become advanced. By adding new features and becoming advanced Porter became more and more complex.

The complex it got, the harder the user experience became. While inspecting the top critic's review on playstore, people stated that the app was harder to use, to a point that people entered wrong pickup places and were charged for that.

That's where TempoWala steps up to provide such a simple interface that any layman could use the app easily.

1.2 Objectives:

Providing the most simple user interface that a person without any technical knowledge should be able to use the app without any hassles.The app has to be minimalistic and has to avoid and hide terms which are not required. The user should be able to use the app like - Come>Book>Leave This will give confidence to the user while booking a tempo service next time and would also save his/her time.

1.3 Purpose, Scope and Applicability

1.3.1 Purpose:

The main purpose is to create a strong end user interface, that a user feels confident about using our app instead of other similar services. This would add value to the system for users which are looking to book & leave.

This App will provide:

1. Easy To Understand and Use User Interface
2. Help To Book a Tempo easily.
3. Help To Create Employment By Letting other TempoWalas join in
4. Book & Leave System.
5. Provide end user with hassle free booking service.

1.3.2 Applicability and Scope:

1. Saves Time - Book & Leave Service

2. User Friendly – User can easily navigate through the app

3. Speeds Up Working Process - Helps to get more bookings as less time is spent on assisting each customer

4. Reduces Paper Work – Online database helps get rid of paper work

1.4 Achievements

The App will help layman, the minimalistic and simplistic design will avoid any complexity. People will feel confident using the app and navigating through it would be a smooth experience. As, they say first experience is the last experience, The app will meet the same.

Chapter 2: SURVEY OF TECHNOLOGIES

1. The IDE:

The IDE which I chose was Android Studio, because it’s far more superior to any other ide’s for developing android apps.

I can test the code which I write & execute directly on my phone or use an emulator built in android studio.

This helps to cut the time to develop and test an app; it also helps to check if everything is going right on the user’s end.

2. The Database:

I am using Google’s own database service known as firebase.

The reason I choose this service is because of its integration with android studio IDE, Android OS and Google play services which come by default in an android phone.

Firebase also stores data in real-time also it helps to reduce time to go through databases and search for a particular record.

3. The Android OS:

The main goal of this app was to book tempos at the ease of a click.

Mobile phones are portable and 85 % percent of smart phones globally run android OS.

This promises the app a wide range of user base to explore and prove beneficial to.

CHAPTER 3: REQUIREMENTS AND ANALYSIS

3.1 Problem Definition

These days the apps which are built are focused solely on putting in their technologies prior. Forgetting the end user's needs and end users understanding of the technology.

Often times less relevant information is displayed to the users which is not required. For example: I want to book a tempo, the app starts recommending useless offers and ads or pop ups which ruin the entire purpose of user experience. Thus leaving a bad remark on the app. This means the user would probably not use the app again

Bringing a clean, simple and minimalistic user interface is the key to success for an app which is a simple but less thought about fact.

This app tends to solve the complexity problem and will prove to give a better fresh user experience and make a user confident and comfortable about using this app.

3.2 Requirement Specifications

Front end (USER)– In the front end we will be using Java code combined with xml for user interface and navigation

.This will contain the following:

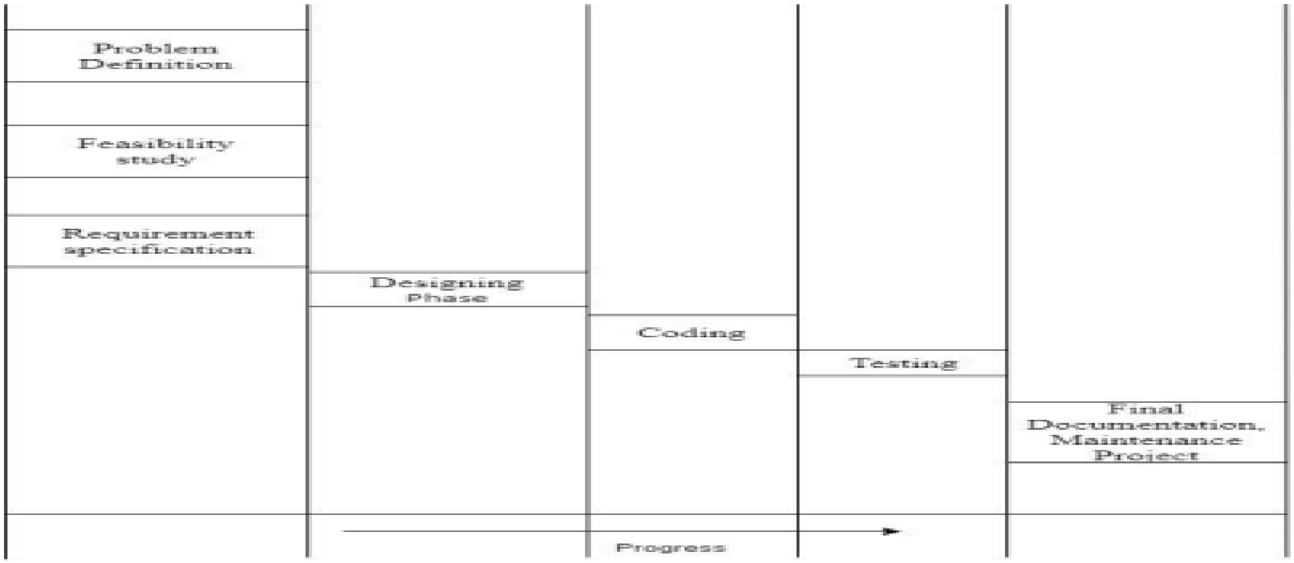
- Register Page
- Login Page
- Verify Email
- Home Page
- Booking Button

- Pickup & Destination Page
- Booking History Page
  - Edit, Cancel, Delete.
- Join Us
- FAQ
- **Back end (ADMIN) –** Monitoring the data, adding, updating, deleting data.(only if needed)
  We can modify data from the firebase console if we need to.

3.3 Planning and Scheduling

Planning can be determining all the small tasks that must be carried out in order to accomplish the goal. Planning also takes into account, rules which control when certain tasks can happen or not.Scheduling can be, determining whether adequate resources are available to carry out the plan. We have to schedule what task needs to be completed first and what to hold on for later completion. Scheduling should help us in achieving the goal at pre determined time and it should not pull us away from the deadline.

GNATT CHART:



3.4 Software and Hardware Requirements

- Software Requirements:
    - OS: Windows, Android OS.
    - Google Chrome: Firebase Console.
    - Android Studio.
        - Languages: java, xml.
- **Hardware Requirements:**
    - Smartphone: 50 mb storage, 500mb Ram, 1 GHz processor.

3.5 Preliminary Product Description

The TempoWala app is for users to book tempos without any hassles and technical knowledge.

It would contain a user Registration, Login, Verify Email page for authentication. Home page, Booking page, Booking History page, Join us page, FAQ page.

Once the booking is done, the data will be fetched in real-time and sent to the database which will lead to dispatching of the TempoWala at the desired address.

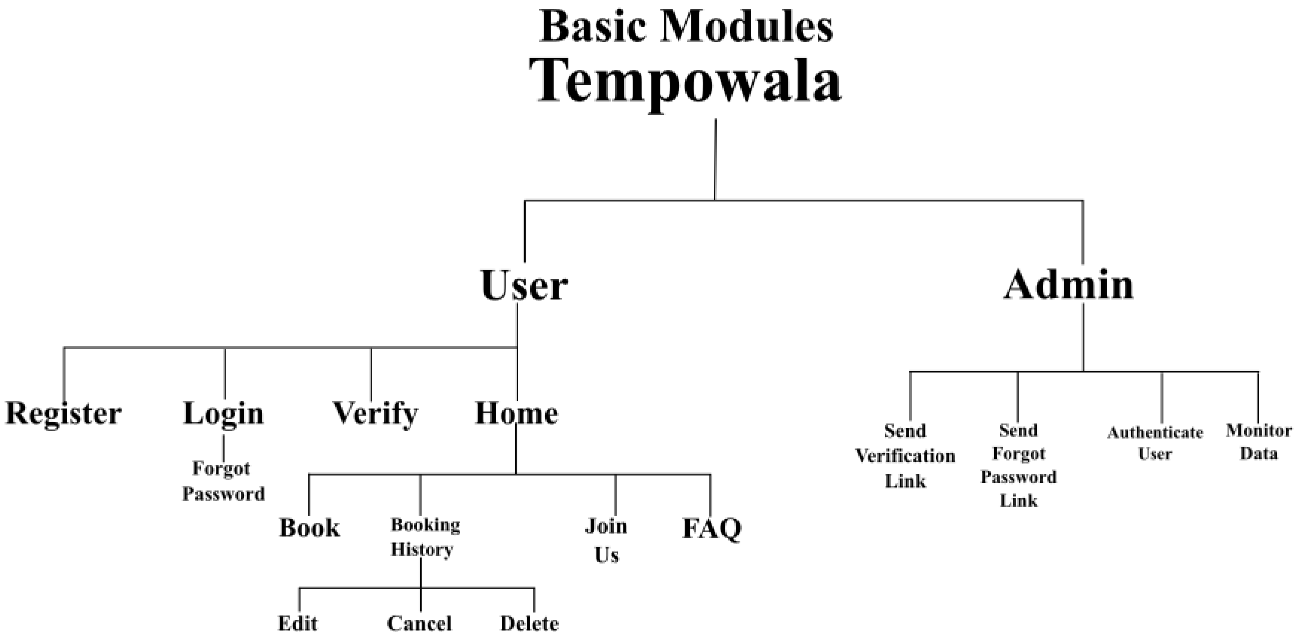
3.6 Conceptual models

This app reduces the large number of manpower to handle bookings. Also the database is much more consistent and works real time and also contains backup mechanism. The database is flexible to perform read, write & delete functions. This adds to being an efficient booking service.

CHAPTER 4: SYSTEM DESIGN

4.1 Basic Modules

The following diagram shows the basic modules in the app:



4.2 Data Design

Register Page:

Field Name	Data type	Description
Fullname	String	Enter Full Name
Email	String	Enter Email
Password	String	Enter password
Phone No	Int(10)	Enter Phone

Login Page:

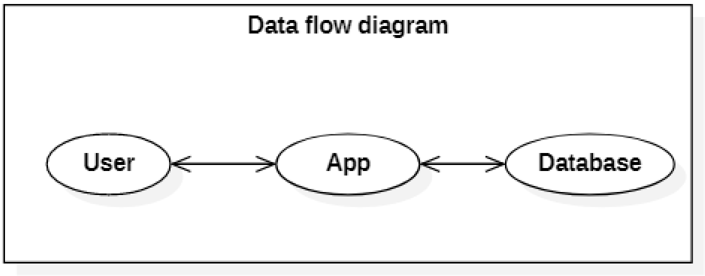
Email	String	Enter email
Password	String	Enter password

Booking /EditBooking Page:

Fullname	String	First name/Last name
Phone no.	Int(10)	10 digits Phone no.
Pincode	Int(6)	6 digits Pin code.
Flatno	String	Flat no , Bldg, Company
Area	String	Area,colony,street,sector.
Landmark	String	Landmark(eg:near post)
Town	String	Town ,City, Village
State	String	State Name

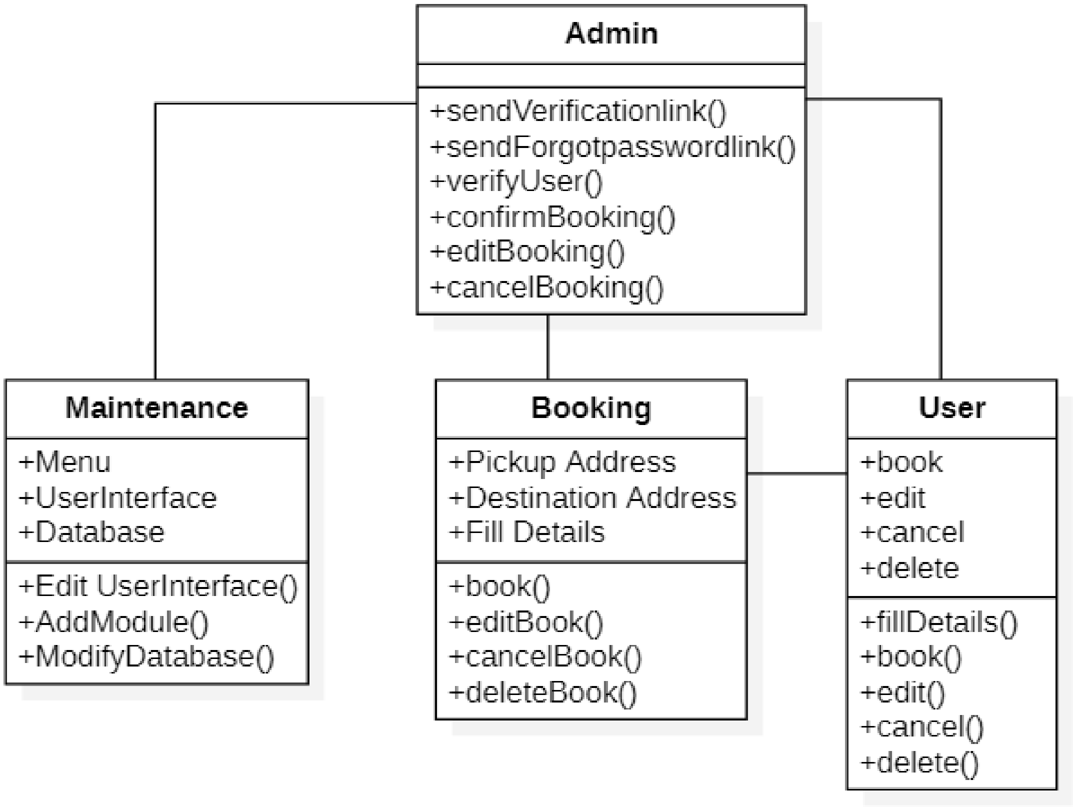
4.3 Procedural Design

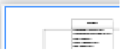
- Procedural design is a systematic way for developing algorithms or procedurals.
- Data flow diagram:



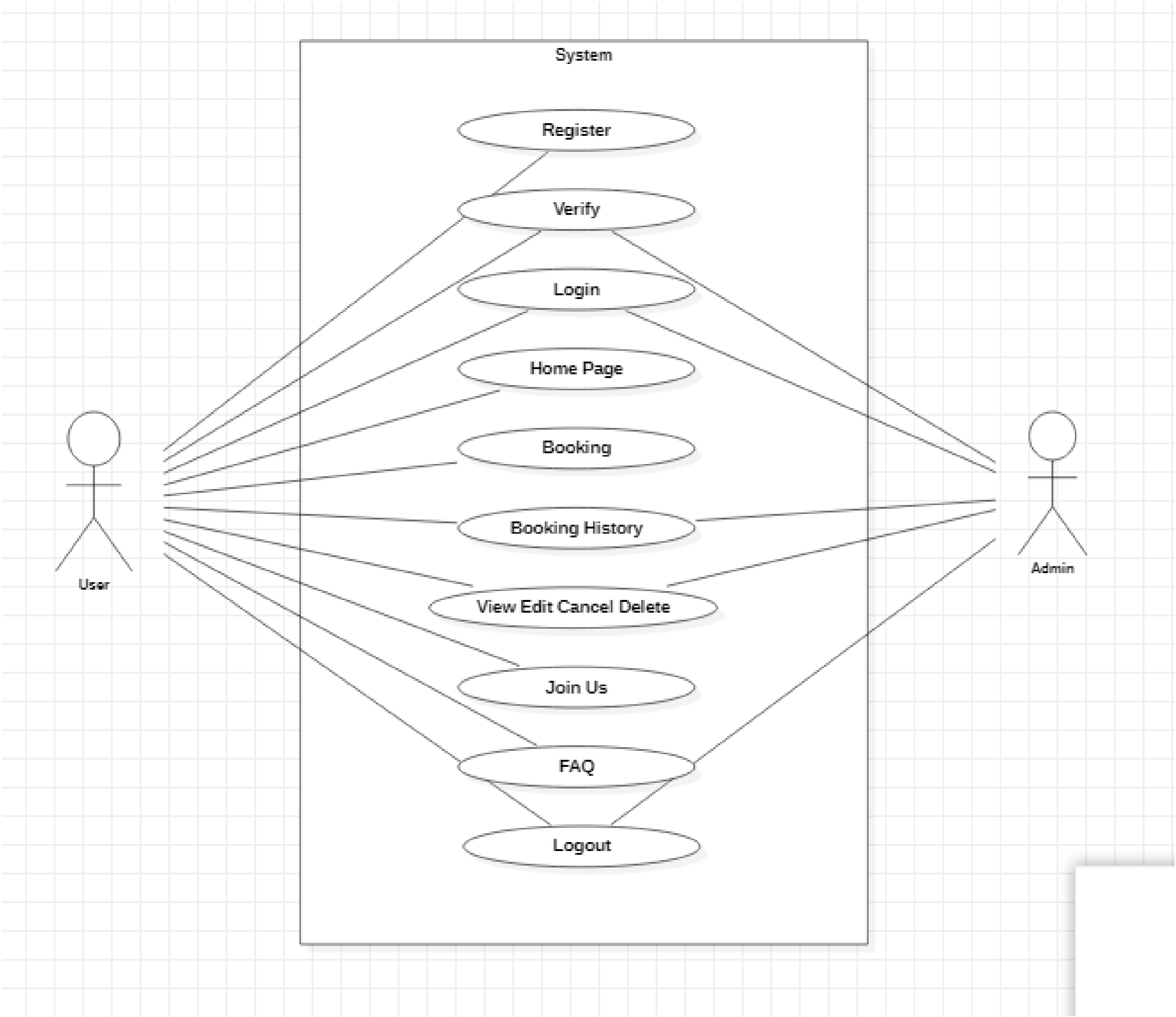
4.3.1 Logic Diagram

- **Class Diagram**



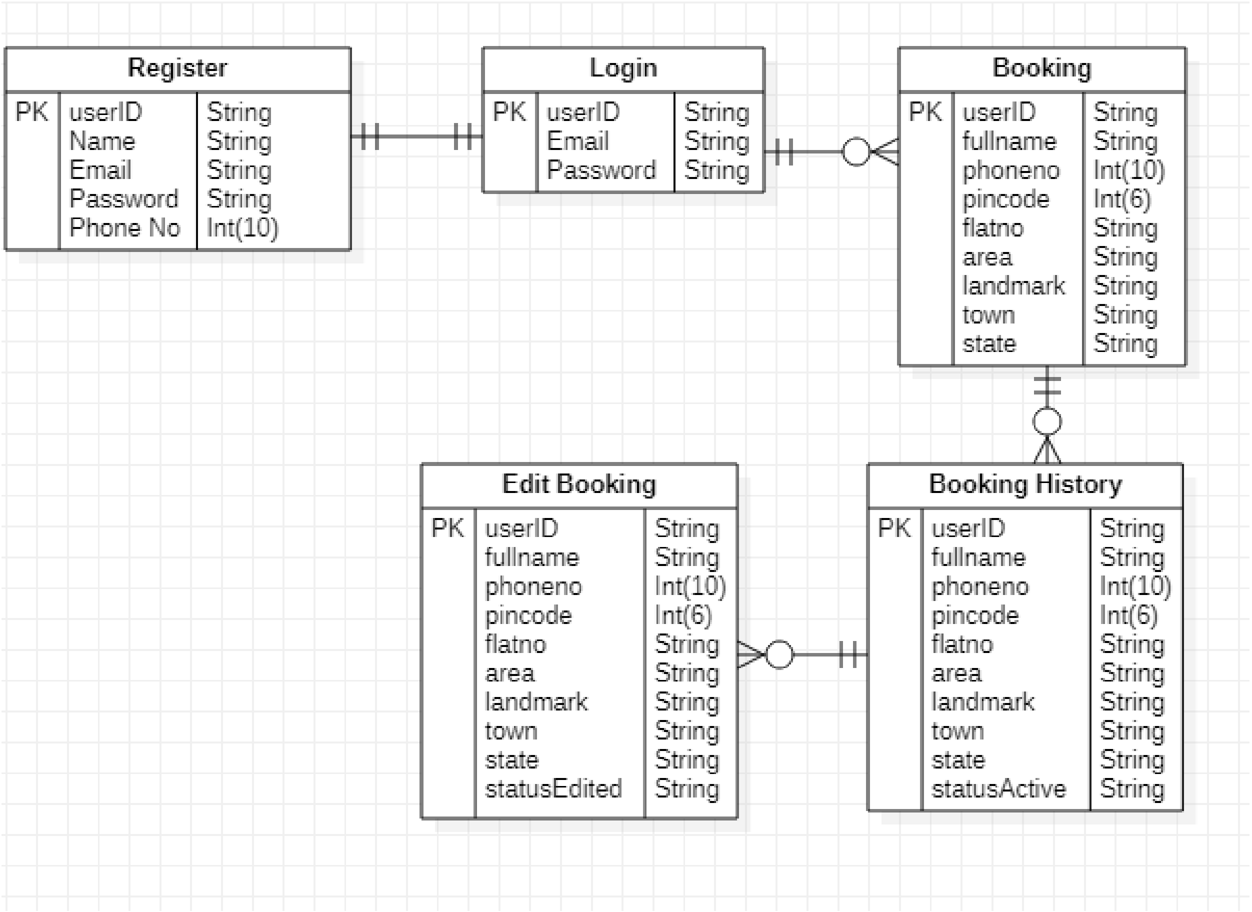


- **Use Case Diagram**



The User will Register, Verify, Login and book the tempo and fill the details, while the admin will Verify User, confirm the booking and maintain the record.

- **ER Diagram:**



Register		
PK	userID Name Email Password Phone No	String String String String Int(10)

Login		
PK	userID Email Password	String String String

Booking		
PK	userID fullname phoneno pincode flatno area landmark town state	String String Int(10) Int(6) String String String String String

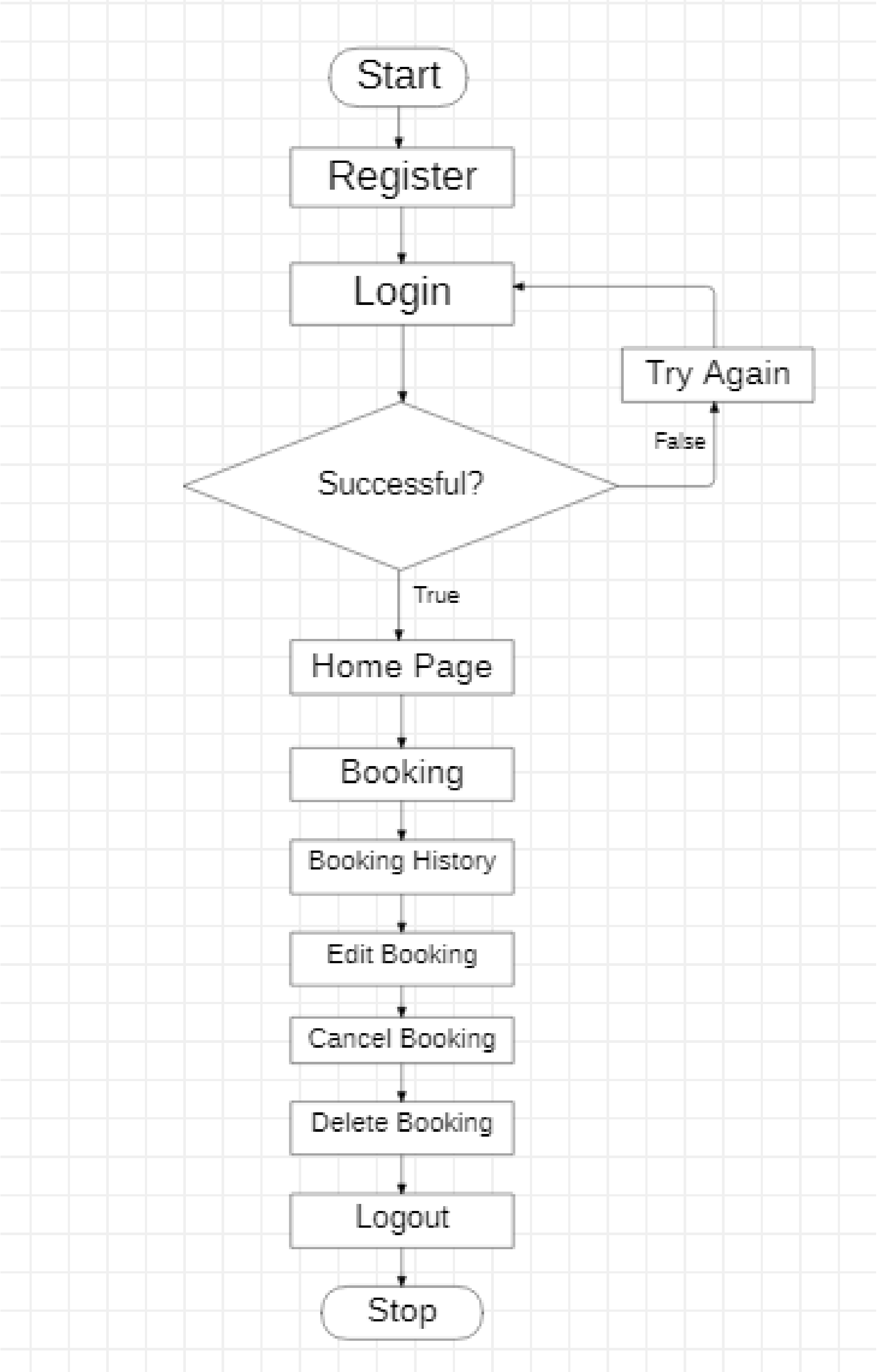
Edit Booking		
PK	userID fullname phoneno pincode flatno area landmark town state statusEdited	String String Int(10) Int(6) String String String String String String

Booking History		
PK	userID fullname phoneno pincode flatno area landmark town state statusActive	String String Int(10) Int(6) String String String String String String

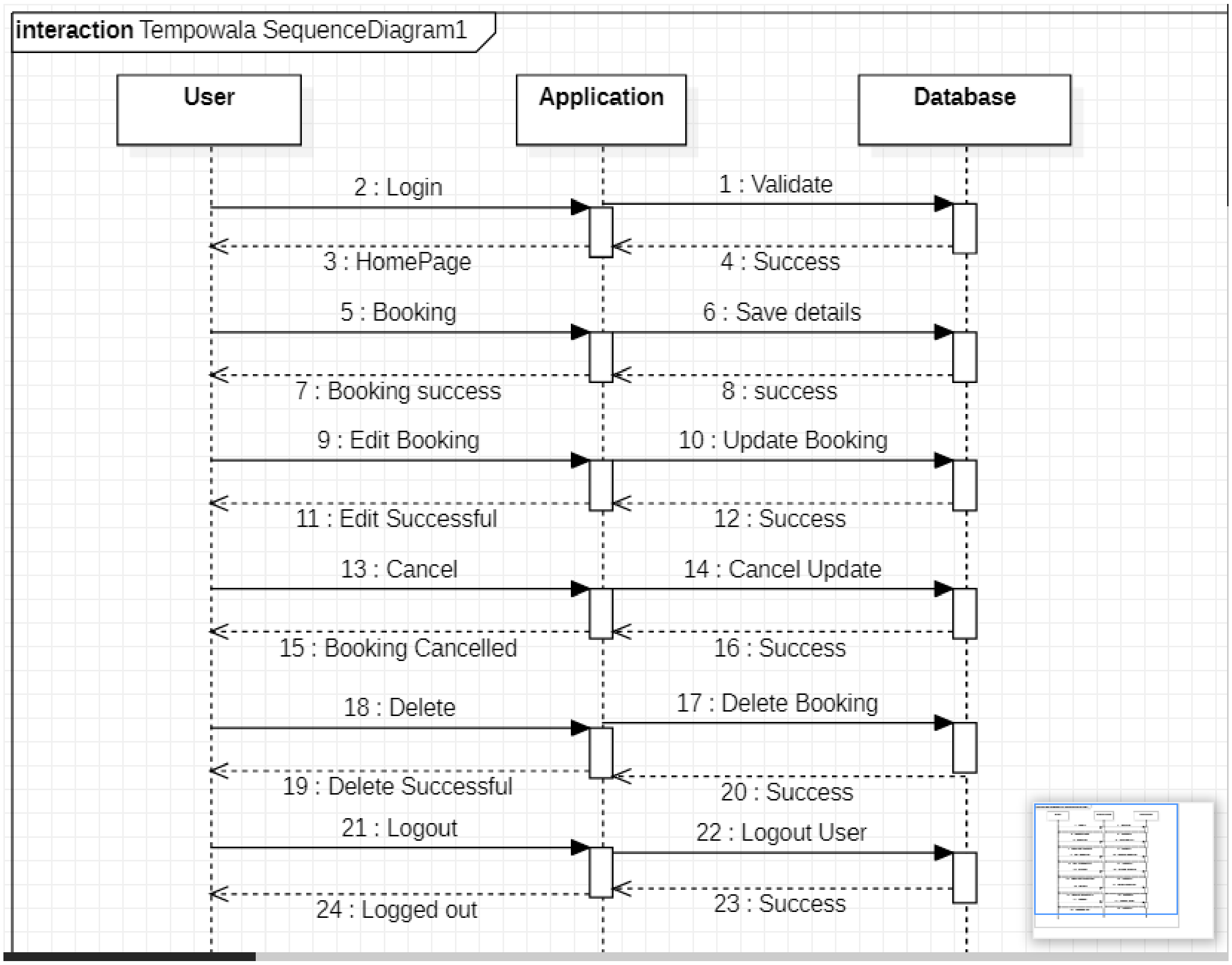
An entity relationship diagram (ERD) shows the relationship of entity sets stored in a database.

The above ER Diagram shows 5 attribute Register, Login, Booking, Booking History, Edit Booking.

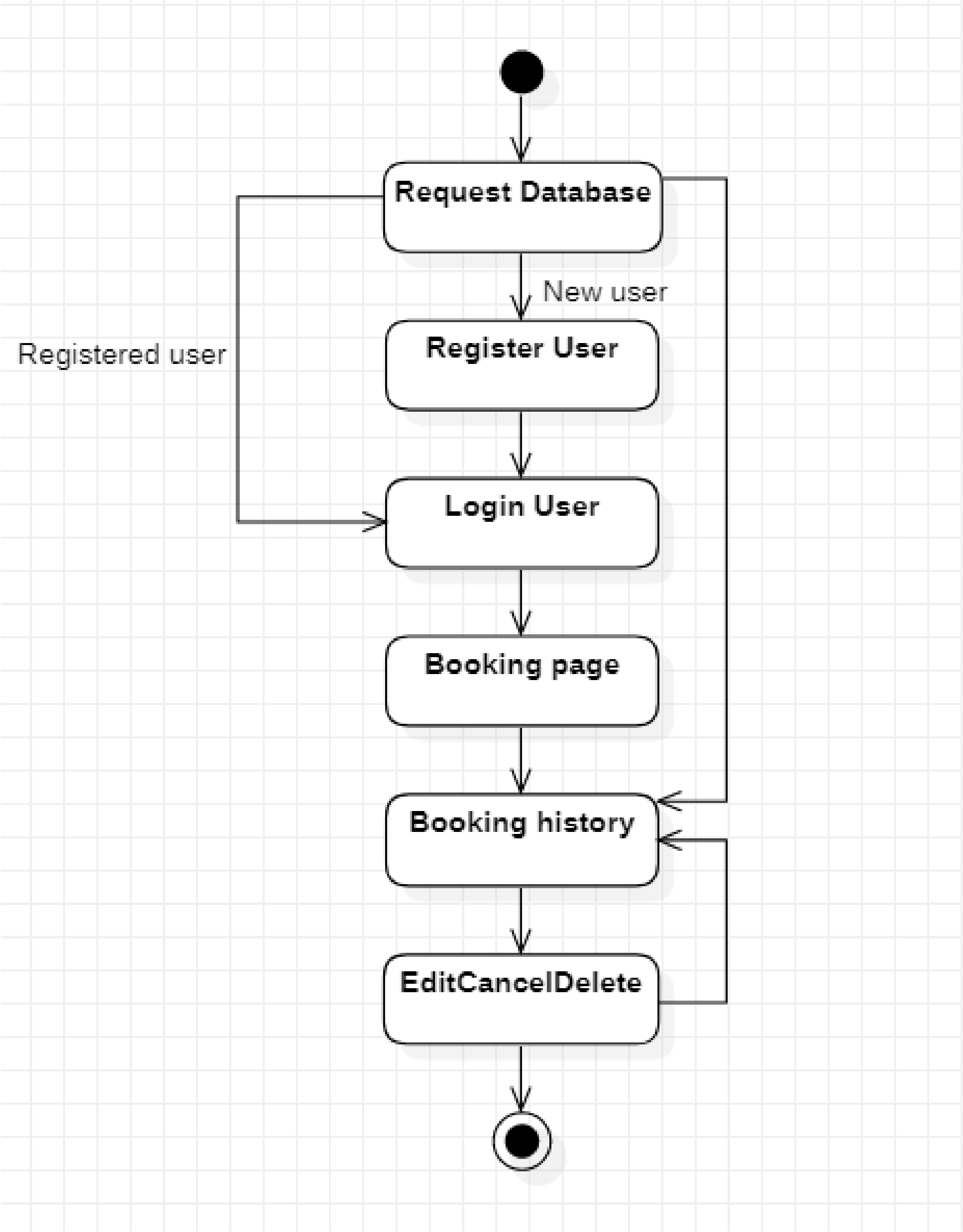
- **Flow Chart:**


Start
Register
Login
Try Again
False
Successful?
True
Home Page
Booking
Booking History
Edit Booking
Cancel Booking
Delete Booking
Logout
Stop

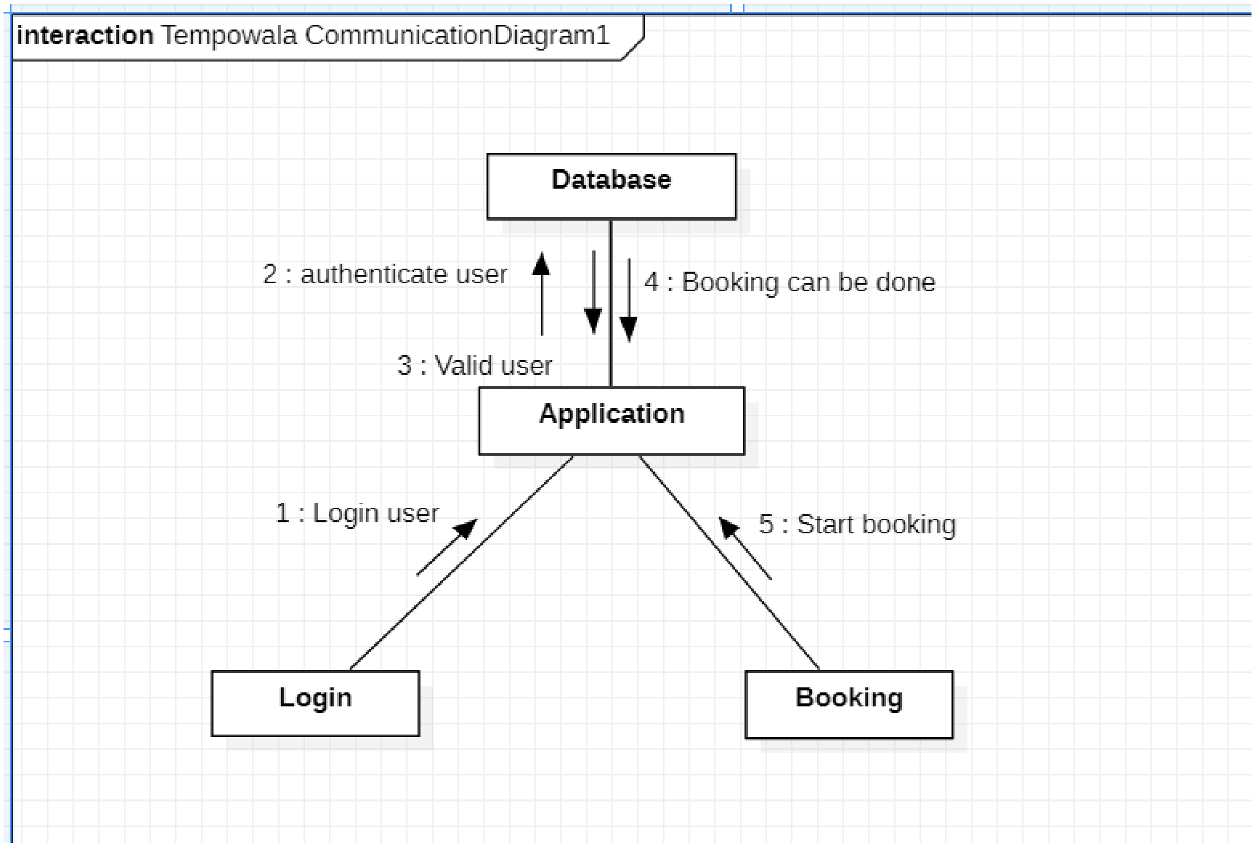
- **Sequence Diagram:**



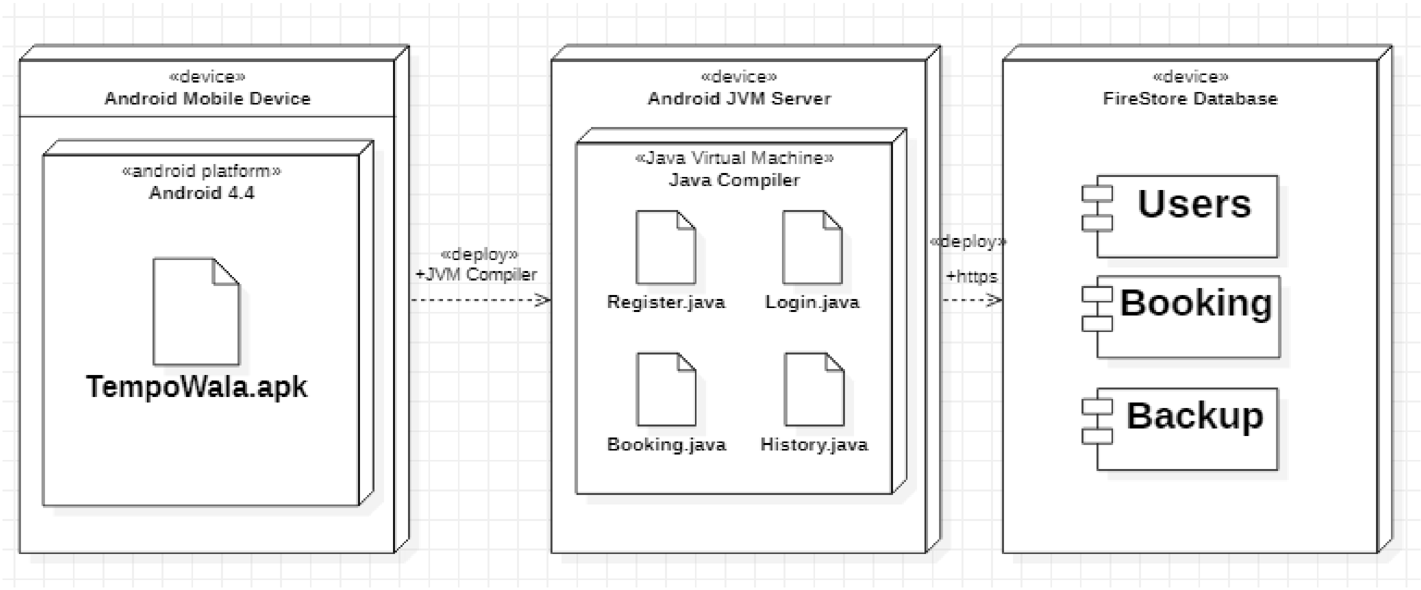
- **State Transition Diagram:**



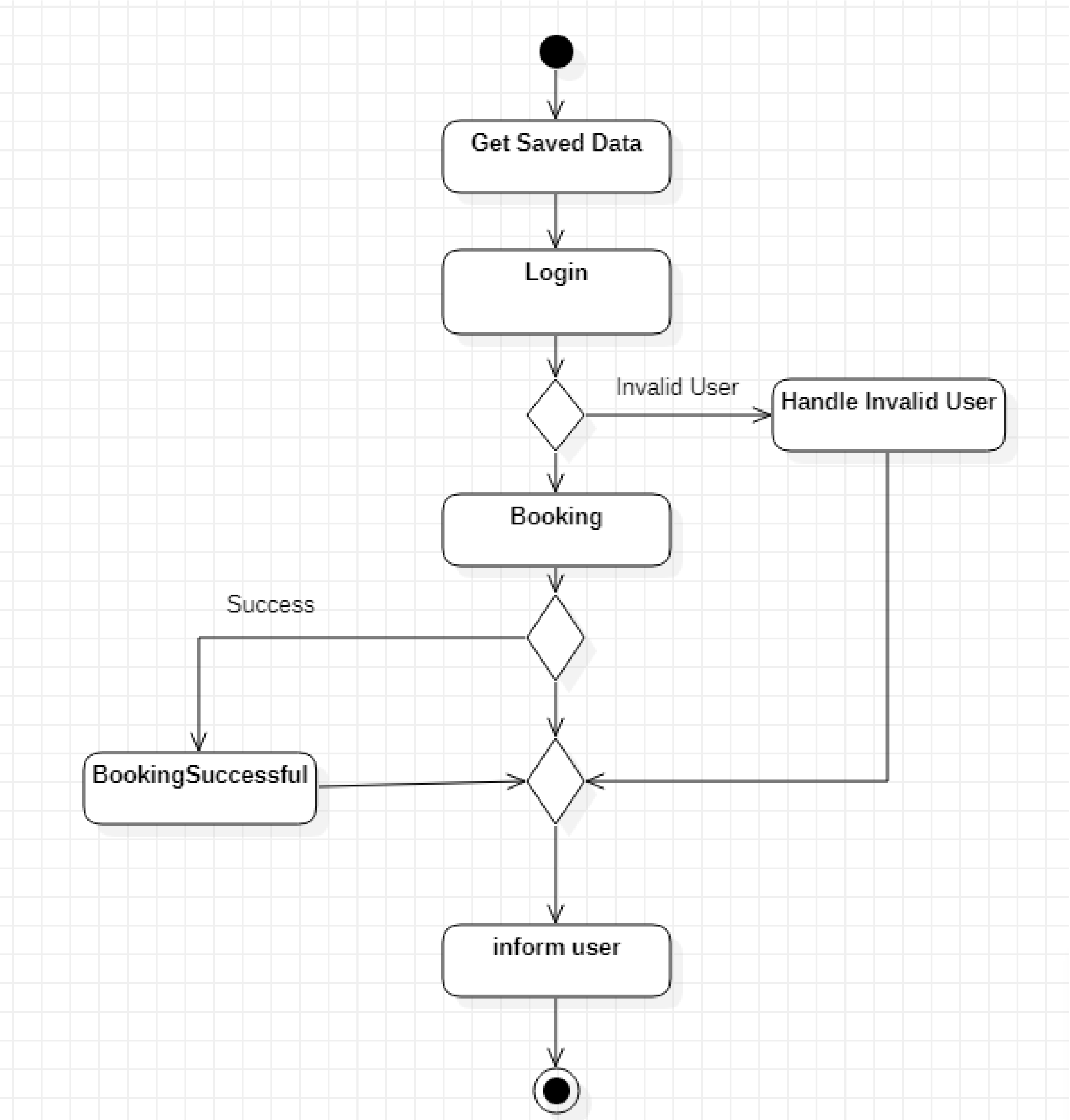
- **Collaboration/Communication Diagram:**



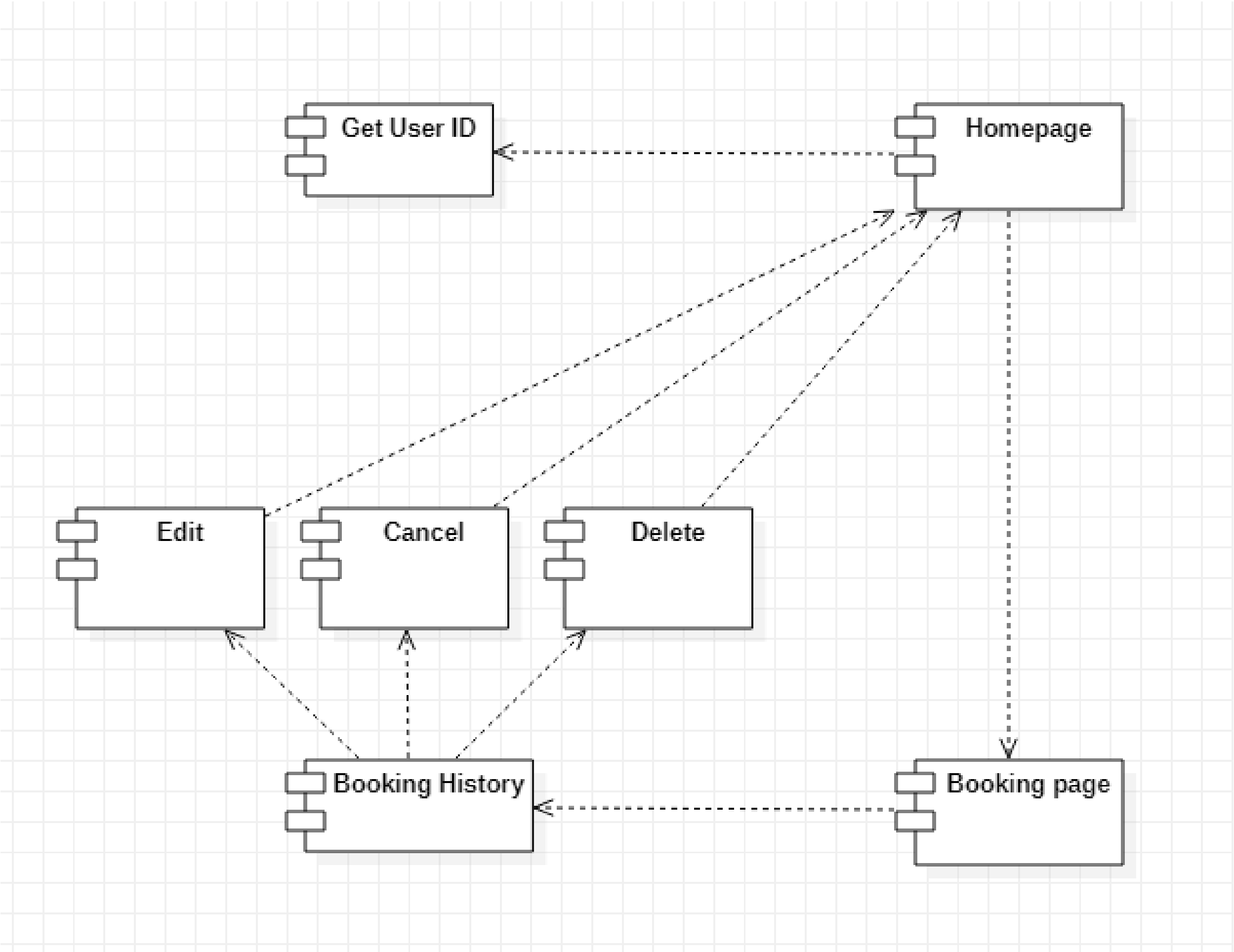
- **Deployment Diagram:**



- **Activity Diagram:**



- **Component Diagram:**



4.5 Security Issues

When the user opens the app first time, the user is asked to register and verify their email. So no other person can access the user's id. This also helps to verify the authenticity of the user, further this email is required to change the password if the user forgets it.

While booking users data is sent securely to the database which cannot be accessed by anyone except me. I can allow the permissions to view access modify data to someone if required.

The database Google Cloud FireStore is among the most secure databases in the world and bigger companies rely on it. Its been around for quite a long time so it's been tested many times for breaches and whenever they found it they fixed it.

4.6 Test cases

SR. NO	ACTION	INPUTS	EXPECTED OUTPUTS	ACTUAL OUTPUT	TEST APP	TEST RESULT
1.	Launch App	Pre installed app	Login Page	Login Page	TempoWala	Pass
2.	Login with unverified Email id	Em:xys3doodma@13.com Pass:13232333	Login Failed	Login Failed this email and password is not verified.	TempoWala	Pass
3.	Login with verified Email id	Em:test123@gmail.com Pass:test1234	Login Success Redirected to Homepage	Login success Redirected to Homepage	TempoWala	Pass

CHAPTER 5: IMPLEMENTATION & TESTING

5.1 Implementation Approaches:

The app will be implemented via Playstore and will be directly connected to our TempoWala service. Where employees will constantly monitor the incoming request and dispatch tempos accordingly.

On booking the user sends the data to our Mumbai database which updates in real time. After booking we wait 5 minutes if the user needs to edit or cancel the booking. Then our employee calls the user about the booking and asks for more details and discuss about the fare. Once completed the tempo is then dispatched at the required destination.

Since the online payment system is still a bit new in India and many elderly people don't know how to do it, I decided it would be better if it's done cash on delivery approach. This also gives customer some trust to book and need not hesitate about paying online.

5.2 Coding details & Code Efficiency:

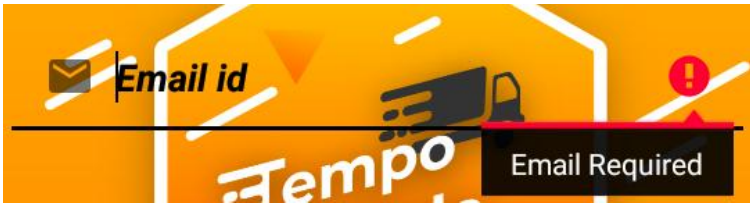
1. Register Page:

**Since the register page is the first page after user installs this app we need to show it.
and if already logged in then send to homepage
so this gets the current user from the database.**

```
if(fAuth.getCurrentUser() != null){
   final FirebaseUser fUser=fAuth.getCurrentUser();

//if the email is verified which means user has already completed registration redirect to homepage.
   if(fUser.isEmailVerified()){
       startActivity(new Intent(getApplicationContext(),MainActivity2.class));
       finish();
   }
   else {
//redirect to verify page.
       startActivity(new Intent(getApplicationContext(),MainActivity.class));
       finish();
```

When clicked on signup first validate:

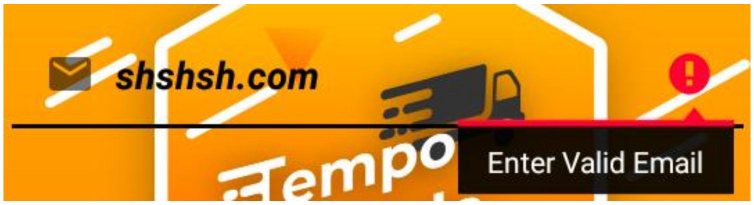
```
//setting on click listener
mRegisterBtn.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
//getting the text from user
        final String fullName = mFullName.getText().toString().trim();
        final String email = mEmail.getText().toString().trim();
        final String password = mPassword.getText().toString().trim();
        final String phone= mPhone.getText().toString().trim();

        String character = "@"; //defining @ for email validation.

        if(TextUtils.isEmpty(email)){  // if email is empty then set error
            mEmail.setError("Email Required");
            return;
        }
```



Email id

Email Required

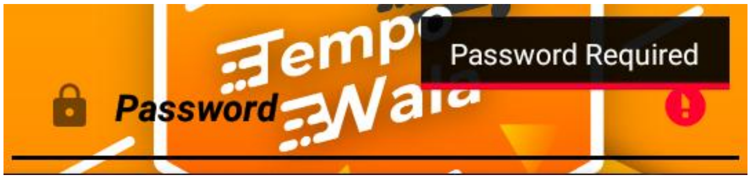
```
        if(!email.contains(character)) //if email doesn’t contain @ symbol
        {
            mEmail.setError("Enter Valid Email");
            return;
        }
```



shshsh.com

Enter Valid Email

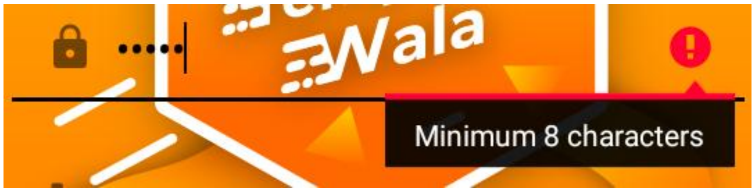
```
        if(TextUtils.isEmpty(password)){ // if password is empty
            mPassword.setError("Password Required");
            return;
```



Password Required

Password

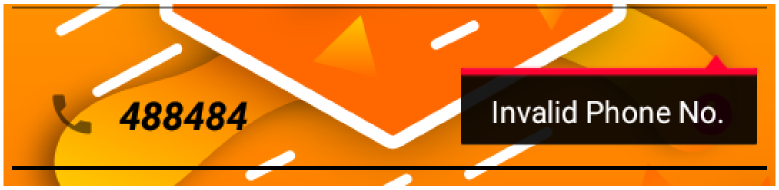
```
if(password.length()<8){  //if password is less than 8 characters
        mPassword.setError("Minimum 8 characters");
        return;
     }
```



•••••

Minimum 8 characters

```
     if(TextUtils.isEmpty(phone)){  //if phone no is empty
        mPhone.setError("Phone No. Required");
        return;
     }


     if(phone.length()<10){  // if greater than 10
        mPhone.setError("Invalid Phone No.");
        return;
     }
```



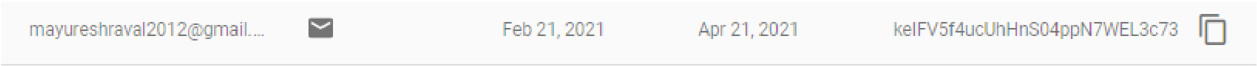
488484

Invalid Phone No.

//**Register the user in database**

```
fAuth.createUserWithEmailAndPassword(email,password).addOnCompleteListener(new

               //saving data to cloud firestore
               userID=fAuth.getCurrentUser().getUid(); //getting the current users user id


               //creating a document reference creating a collection users and making a new doc using useid
               DocumentReference documentReference = fstore.collection("users").document(userID);


               //creating a hashmap to send data
               Map<String,Object> user = new HashMap<>();
               user.put("fName",fullName);
               user.put("email",email);
               user.put("password",password);
```

```
                user.put("phone",phone);

                //using the document reference to set user document
                documentReference.set(user).addOnSuccessListener(new OnSuccessListener<Void>() {


                    @Override
                    public void onSuccess(Void aVoid) {
                        Log.d(TAG,"onSuccess: user profile created for "+ userID);
                    }
                });
//if successful start verify page
                startActivity(new Intent(getApplicationContext(),MainActivity.class));
                finish();

            }
            else{

//show the error
                Toast.makeText(Register.this, "Error!" + task.getException().getMessage(),
Toast.LENGTH_LONG).show();
                progressBar.setVisibility(View.GONE);
            }
        }
```

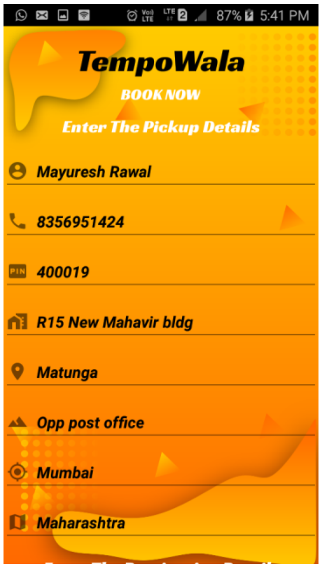


mayureshraval2012@gmail.... ✉ Feb 21, 2021 Apr 21, 2021 keIFV5f4ucUhHnS04ppN7WEL3c73

Booking page:

The booking page has a lot of things I will just show how the data is put to the database.

```
//getting the current users unique id
userID=fAuth.getCurrentUser().getUid();
//creating a document reference creating a collection booking and making a new doc using user id
DocumentReference documentReference = fstore.collection("Booking").document(userID);
//creating a hashmap to send data
Map<String,Object> book = new HashMap<>();
//setting status
book.put("Status -","Active");
//pickup
book.put("a1 - Fullname",firstname);
book.put("a2 - PhoneNo",phoneno);
book.put("a3 - Pincode",pincode);
book.put("a4 - Flatno",flatno);
book.put("a5 - Area",area);
book.put("a6 - Landmark",landmark);
book.put("a7 - Town",town);
book.put("a8 - State",state);
//destination
book.put("d1 - Fullname2",firstname2);
book.put("d2 - PhoneNo2",phoneno2);
book.put("d3 - Pincode2",pincode2);
book.put("d4 - Flatno2",flatno2);
book.put("d5 - Area2",area2);
book.put("d6 - Landmark2",landmark2);
book.put("d7 - Town2",town2);
book.put("d8 - State2",state2);
```



Mayuresh Rawal

8356951424

400019

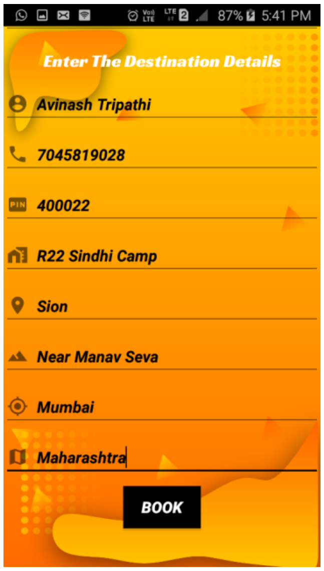
R15 New Mahavir bldg

Matunga

Opp post office

Mumbai

Maharashtra



Avinash Tripathi

7045819028

400022

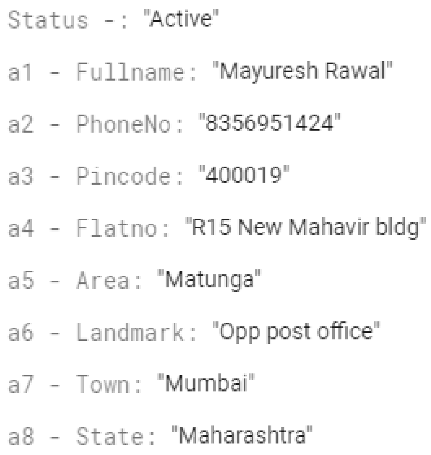
R22 Sindhi Camp

Sion

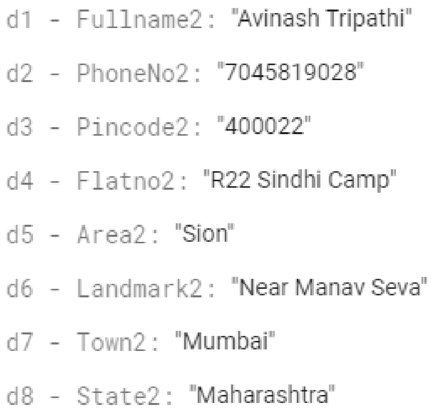
Near Manav Seva

Mumbai

Maharashtra



```
Status -: "Active"
a1 - Fullname: "Mayuresh Rawal"
a2 - PhoneNo: "8356951424"
a3 - Pincode: "400019"
a4 - Flatno: "R15 New Mahavir bldg"
a5 - Area: "Matunga"
a6 - Landmark: "Opp post office"
a7 - Town: "Mumbai"
a8 - State: "Maharashtra"
```



```
d1 - Fullname2: "Avinash Tripathi"
d2 - PhoneNo2: "7045819028"
d3 - Pincode2: "400022"
d4 - Flatno2: "R22 Sindhi Camp"
d5 - Area2: "Sion"
d6 - Landmark2: "Near Manav Seva"
d7 - Town2: "Mumbai"
d8 - State2: "Maharashtra"
```

Booking history:

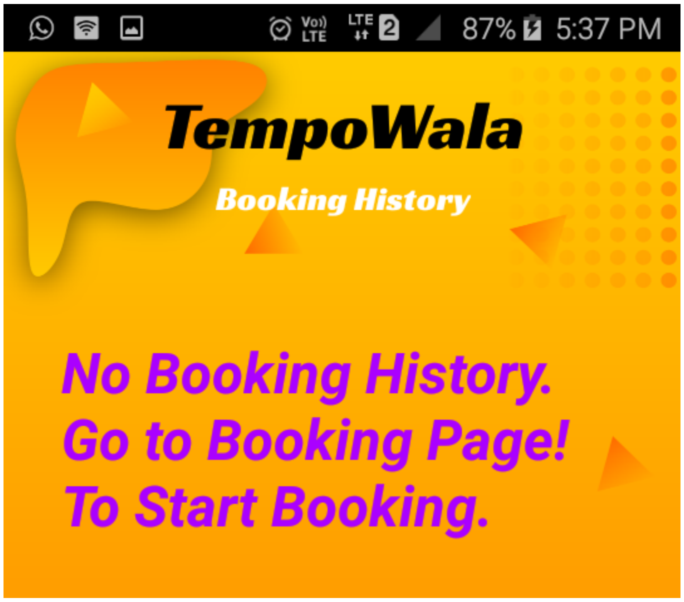
The actual code is lot bigger I have trimmed the code

This one checks booking exists or not

```
fstore.collection("Booking").document(userID).get().addOnCompleteListener(new
OnCompleteListener<DocumentSnapshot>() {
    @Override
    public void onComplete(@NonNull Task<DocumentSnapshot> task) {
```

if theres no document for userid

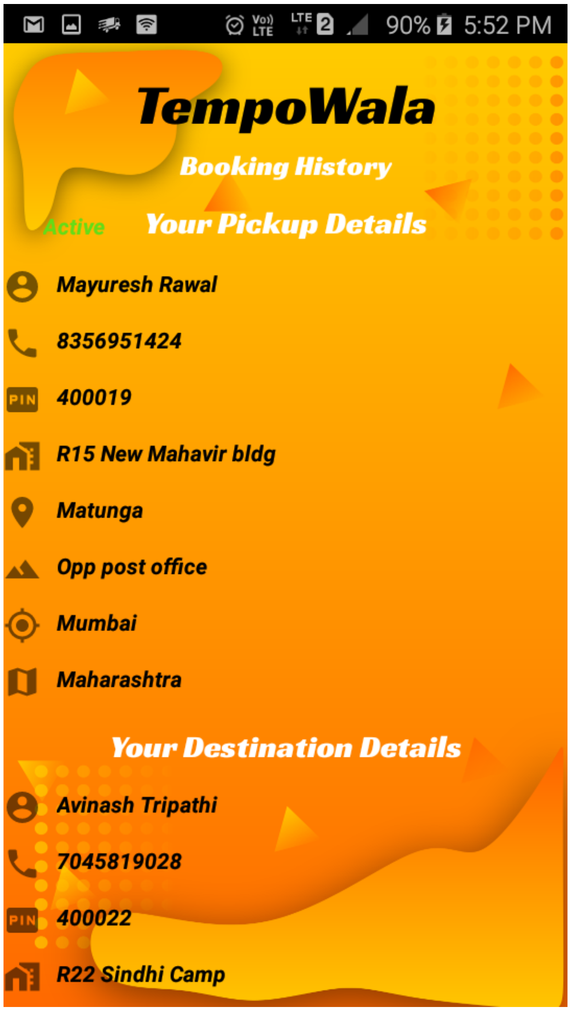
```
 if (!task.getResult().exists()) {
//then no booking exists.
}
```



No Booking History.
Go to Booking Page!
To Start Booking.

if theres a document for userid then show the booking

```
// this one is used to set the text
    final DocumentReference documentReference = fstore.collection("Booking").document(userID);
    documentReference.addSnapshotListener(this, new EventListener<DocumentSnapshot>() {
            @Override
            public void onEvent(@Nullable DocumentSnapshot documentSnapshot, @Nullable
FirebaseFirestoreException e) {

//displaying the booking from the db to the ui.

 mStatus.setText(documentSnapshot.getString("Status -"));
mFirstnamelastname.setText(documentSnapshot.getString("a1 - Fullname"));

mMobnum.setText(documentSnapshot.getString("a2 - PhoneNo"));

mPincode.setText(documentSnapshot.getString("a3 - Pincode"));

}
```



Your Pickup Details

Mayuresh Rawal

8356951424

PIN 400019

R15 New Mahavir bldg

Matunga

Opp post office

Mumbai

Maharashtra

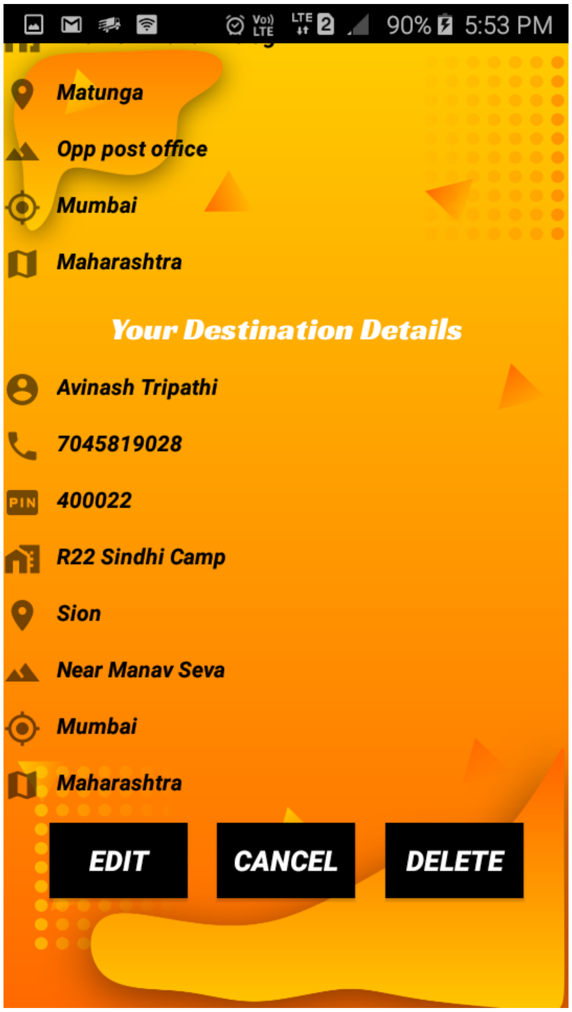
Your Destination Details

Avinash Tripathi

7045819028

PIN 400022

R22 Sindhi Camp



Matunga

Opp post office

Mumbai

Maharashtra

Your Destination Details

Avinash Tripathi

7045819028

PIN 400022

R22 Sindhi Camp

Sion

Near Manav Seva

Mumbai

Maharashtra

FAQ PAGE:

This is the most interesting page it has a lot of feature like farecalculaor, youtube video tutorials,email complaint opener etc.

The standard way of creating faq page in android is using expandable list views, what I have done is more simpler and so unique that I didn’t find anyone using this approach.

So basically what I do is

1. When a question is clicked

```
mfaq.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
```

2. I take the visbility of the ans and check if its already visible first.

```
            if(mAns.getVisibility()== View.VISIBLE){
```

3. If it is visible then make it invisible.

```
            mAns.setVisibility(View.GONE);
            }
            else {
```

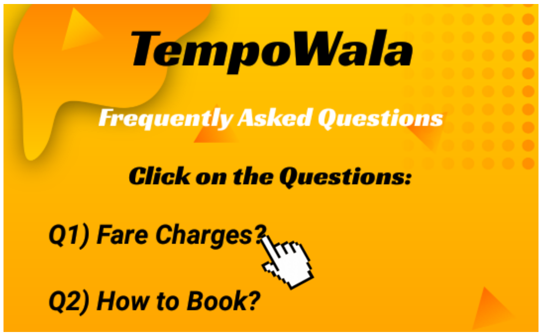
4. Else make it visible

```
                mAns.setVisibility(View.VISIBLE);
            }
        }
    });
```

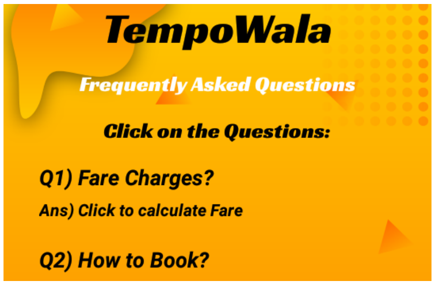
Still confused?

Let’s go to through the life cycle

1. User clicks a question 1st time



2. Then we check if the ans is already visible or not
3. Which in our case is not
4. Then we show the ans by setting it visible



TempoWala

Frequently Asked Questions

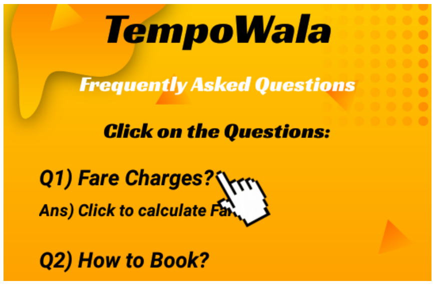
Click on the Questions:

Q1) Fare Charges?

Ans) Click to calculate Fare

Q2) How to Book?

5. User clicks on the same question 2nd time



TempoWala

Frequently Asked Questions

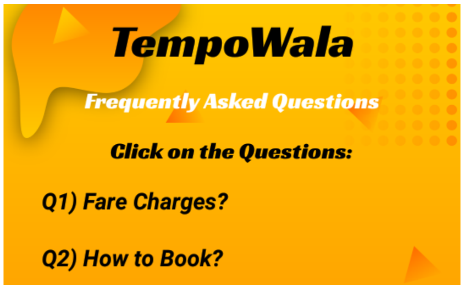
Click on the Questions:

Q1) Fare Charges?

Ans) Click to calculate Fa

Q2) How to Book?

6. We again check if the ans is already visible or not
7. Which for now is visible
8. So we hide it by setting it invisible.



TempoWala

Frequently Asked Questions

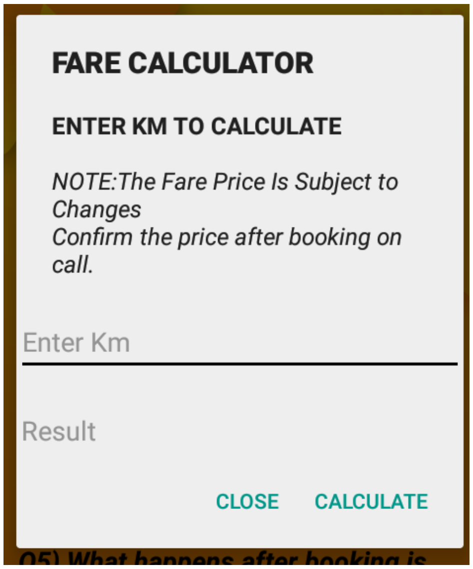
Click on the Questions:

Q1) Fare Charges?

Q2) How to Book?

The Fare calculator

When a user clicks on click to calculate fare ans. I create an alert builder dialog for user to calculate fare.



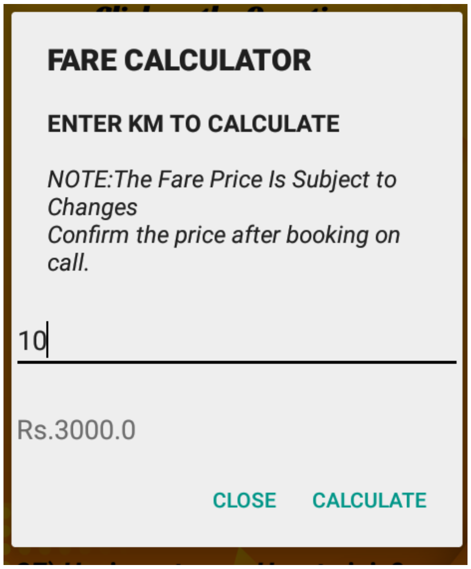
FARE CALCULATOR

ENTER KM TO CALCULATE

*NOTE:The Fare Price Is Subject to Changes
Confirm the price after booking on call.*

Enter Km

Result



FARE CALCULATOR

ENTER KM TO CALCULATE

*NOTE:The Fare Price Is Subject to Changes
Confirm the price after booking on call.*

10

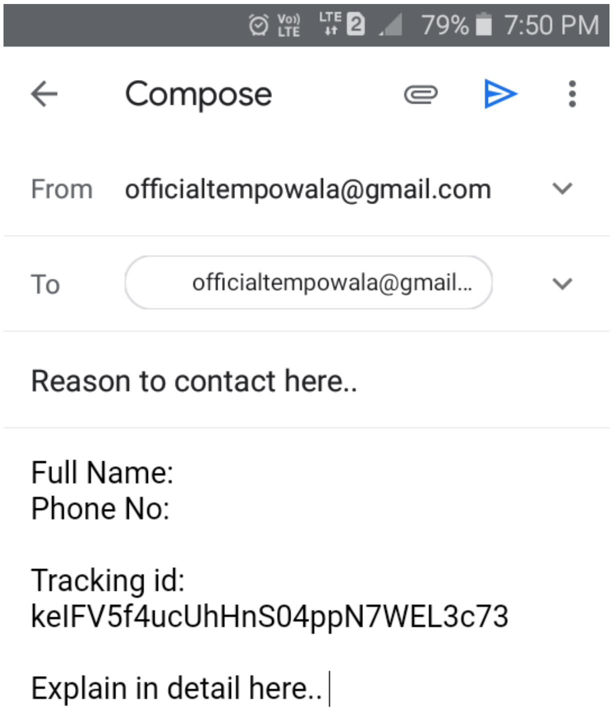
Rs.3000.0

The code is below:

```
mAns.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        AlertDialog.Builder farecalculator = new AlertDialog.Builder(view.getContext());
                farecalculator.setTitle(R.string.Title) // setting title


                .setMessage(R.string.Message); //showing message

        // Set an EditText view to get user input
        final EditText KM = new EditText(view.getContext()); // for user input
        KM.setInputType(InputType.TYPE_CLASS_NUMBER); //taking number type
        KM.setRawInputType(Configuration.KEYBOARD_12KEY);//no special char
        KM.setHint(KM_HINT);//showing hint
        final TextView FARE = new TextView(view.getContext());
        FARE.setHint(FARE_HINT);
        FARE.setPadding(5,5,5,5);//setting 5 padding for top bottom left right
        FARE.setTextSize(18);
        LinearLayout layout = new LinearLayout(getApplicationContext()); //creating a linear layout
        layout.setOrientation(LinearLayout.VERTICAL);//setting it to vertical
        layout.addView(KM); //adding the above created Edittext view to enter km
        layout.addView(FARE);// adding the above created Text view to display result
        farecalculator.setView(layout); //showing layout
        //to overide the closing of the dialog when calculate is pressed
        farecalculator.setPositiveButton("Calculate",null)
});



                .setNegativeButton("Close",null);
        final AlertDialog dialog=farecalculator.create();
        dialog.show();
        dialog.getButton(AlertDialog.BUTTON_POSITIVE).setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                /*
                so what happened here was that the double value is a primitive data type which cannot be null
thus cannot be compared to null.
                so the app was crashing when clicked on alert builder positive button when the data was null
                so i took the value into a string first checked it if it was empty then if it wasn't empty i parsed it
to double value to execute further
                took me 10 hours to figure out this one. started at morning 10 , 9:34 i solved it.
                */




                String nullCheck=KM.getText().toString();
//checking if null
                if(TextUtils.isEmpty(nullCheck)){
                    Toast.makeText(faq.this, "Enter KM or Press Close!", Toast.LENGTH_SHORT).show();
                }
                else{
//converting string to null
                    Double n1,Result;
                    n1=Double.parseDouble(nullCheck);
//displaying fare result
                    if(n1==0){
                        FARE.setText("\nRs. 0");
                    }
                    else if(n1==1){
                        FARE.setText("\nRs.500");
                    }
                    else if(n1==2){
                        FARE.setText("\nRs.700");
                    }
                    else{
                        Result=n1*300;
                        FARE.setText("\nRs."+ Result);
                    }
                }
            }
        });
    }
});
```

The Email complaint:

```
mAns9.setOnClickListener(new View.OnClickListener() {
                @Override
                public void onClick(View view) {
                   String[] ourMail = {"officialtempowala@gmail.com"}; //our mail addresss

                   Intent intent =new Intent(Intent.ACTION_SENDTO);
                   intent.setData(Uri.parse("mailto:"));
                   intent.putExtra(Intent.EXTRA_EMAIL, ourMail); //this sets our email In gmail
                   intent.putExtra(Intent.EXTRA_SUBJECT,"Reason to contact here.."); //sets some text
                   intent.putExtra(Intent.EXTRA_TEXT,"Full Name:\nPhone No:\n\nTracking
id:\n"+userId+"\n\nExplain in detail here.."); //sets some text
    startActivity(Intent.createChooser(intent,"Choose Application To Send Email"));
//creates chooser if users phone has different apps for mailing.

                }
             });
```



From officialtempowala@gmail.com

To officialtempowala@gmail...

Reason to contact here..

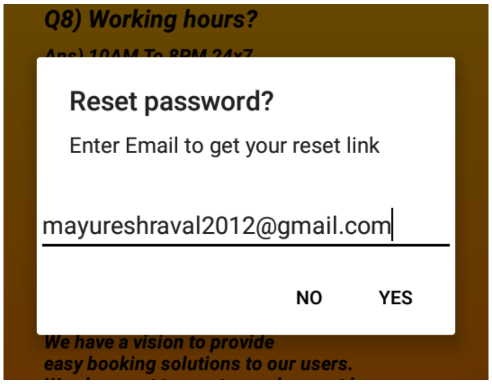
Full Name:
Phone No:

Tracking id:
keIFV5f4ucUhHnS04ppN7WEL3c73

Explain in detail here..

The Reset password dialog:

```
final EditText resetMail = new EditText(view.getContext());
androidx.appcompat.app.AlertDialog.Builder passwordResetDialog= new
androidx.appcompat.app.AlertDialog.Builder(view.getContext());
passwordResetDialog.setTitle("Reset password?");
passwordResetDialog.setMessage("Enter Email to get your reset link");
passwordResetDialog.setView(resetMail);
passwordResetDialog.setPositiveButton("Yes", new DialogInterface.OnClickListener() {
    @Override
    public void onClick(DialogInterface dialogInterface, int i) {
        //setting new logic to prevent crash somehow this fixes crash
        String mail=resetMail.getText().toString();
          if(TextUtils.isEmpty(mail)){
            resetMail.setError("Email Required!");
        }
        else {
            fAuth.sendPasswordResetEmail(mail).addOnSuccessListener(new OnSuccessListener<Void>() {
                @Override
                public void onSuccess(Void aVoid) {
                    Toast.makeText(faq.this, "Reset Link Sent!", Toast.LENGTH_SHORT).show();
                }
            }).addOnFailureListener(new OnFailureListener() {
                @Override
                public void onFailure(@NonNull Exception e) {
                    Toast.makeText(faq.this, "Reset Link Not Sent" + e.getMessage(),
Toast.LENGTH_SHORT).show();

                }
            });
```



Q8) Working hours?

Ans) 10AM To 8PM 24x7

Reset password?

Enter Email to get your reset link

mayureshraval2012@gmail.com

We have a vision to provide easy booking solutions to our users.

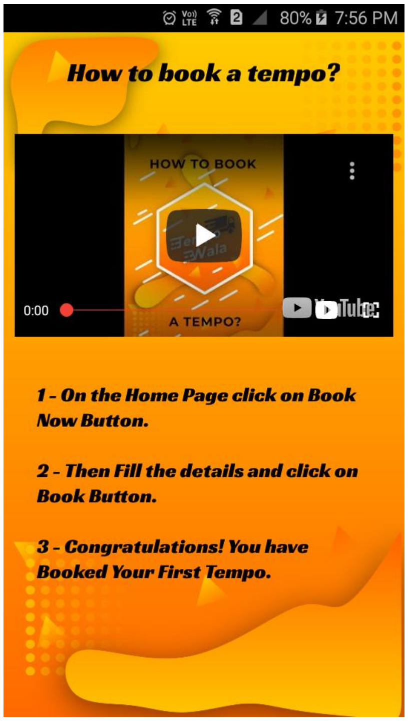
The YouTube player:

//Java code to declare the youtube player.

```
YouTubePlayerView youTubePlayerView = findViewById(R.id.youtube_player_view);
//        youTubePlayerView.enterFullScreen();
      getLifecycle().addObserver(youTubePlayerView);
```

//XML code in which the actual magic happens

```
<com.pierfrancescosoffritti.androidyoutubeplayer.core.player.views.YouTubePlayerView
    android:id="@+id/youtube_player_view"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"

    android:layout_gravity="center_horizontal"
    android:layout_marginTop="10dp"
    android:padding="10dp"
    app:autoPlay="false"
    app:showFullScreenButton="true"
    app:useWebUi="false"
    app:videoId="Hu8HXktvGYo">  // my video id.

</com.pierfrancescosoffritti.androidyoutubeplayer.core.player.views.YouTubePlayerView>
```



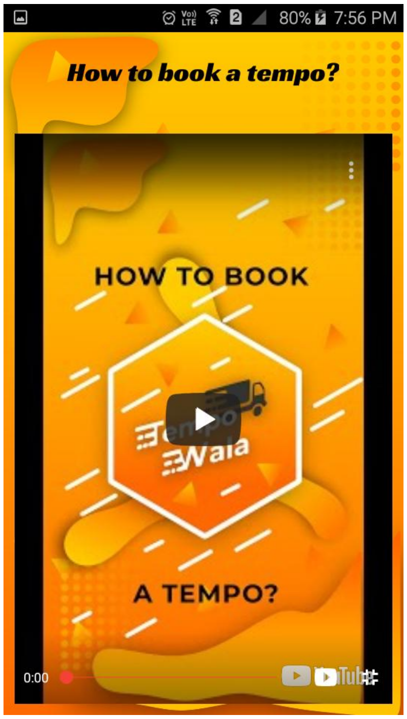
80% 7:56 PM

How to book a tempo?

1 - On the Home Page click on Book Now Button.

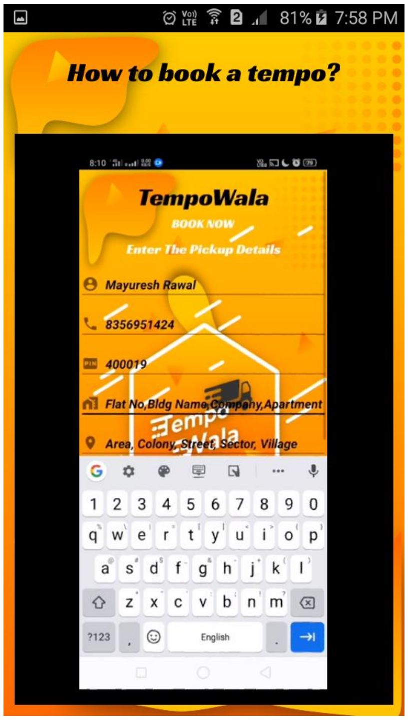
2 - Then Fill the details and click on Book Button.

3 - Congratulations! You have Booked Your First Tempo.



80% 7:56 PM

How to book a tempo?



81% 7:58 PM

How to book a tempo?

TempoWala

BOOK NOW

Enter The Pickup Details

Mayuresh Rawal

8356951424

400019

Flat No,Bldg Name,Company,Apartment

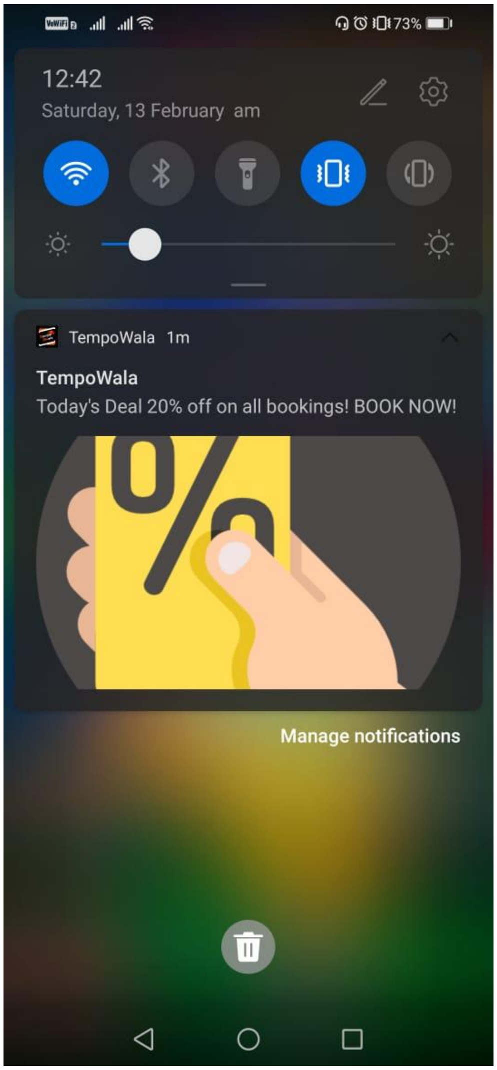
Area, Colony, Street, Sector, Village

The Notification Listeners and in app messaging:

I can send notifications to the users whenever required from my database console. The code for the service file which listens for notification is:

```
public class MyService extends FirebaseMessagingService {
    private static final String TAG ="Tag";
    FirebaseAuth fAuth;
    public MyService() {
    }
    @Override
    public void onMessageReceived(RemoteMessage remoteMessage) {
        fAuth=FirebaseAuth.getInstance();
        if(fAuth.getCurrentUser() != null){
            Intent intent = new Intent(this, MainActivity2.class);
            intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
            PendingIntent pi = PendingIntent.
                    getActivity(this, 0, intent, PendingIntent.FLAG_ONE_SHOT);
            String Channel_Id = "Default";
            NotificationCompat.Builder builder = new NotificationCompat.Builder(this, Channel_Id);
            builder.setSmallIcon(R.drawable.ic_notify)
                    .setContentTitle(remoteMessage.getNotification().getTitle())
                    .setContentText(remoteMessage.getNotification().getBody())
                    .setAutoCancel(true)
                    .setContentIntent(pi);
            NotificationManager manager = (NotificationManager) getSystemService(NOTIFICATION_SERVICE);

            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
                NotificationChannel channel = new NotificationChannel(Channel_Id, "Default channel",
NotificationManager.IMPORTANCE_DEFAULT);
                manager.createNotificationChannel(channel);
            }
            manager.notify(0, builder.build());
        }
        else {
            Intent intent = new Intent(this, Login.class);
            intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
            PendingIntent pi = PendingIntent.
                    getActivity(this, 0, intent, PendingIntent.FLAG_ONE_SHOT);
            String Channel_Id = "Default";
            NotificationCompat.Builder builder = new NotificationCompat.Builder(this, Channel_Id);
            builder.setSmallIcon(R.drawable.ic_notify)
                    .setContentTitle(remoteMessage.getNotification().getTitle())
                    .setContentText(remoteMessage.getNotification().getBody())
                    .setAutoCancel(true)
                    .setContentIntent(pi);
            NotificationManager manager = (NotificationManager) getSystemService(NOTIFICATION_SERVICE);

            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.O) {
```

```
            NotificationChannel channel = new NotificationChannel(Channel_Id, "Default channel",
NotificationManager.IMPORTANCE_DEFAULT);
            manager.createNotificationChannel(channel);
         }
         manager.notify(0, builder.build());
      }
   }
}
```



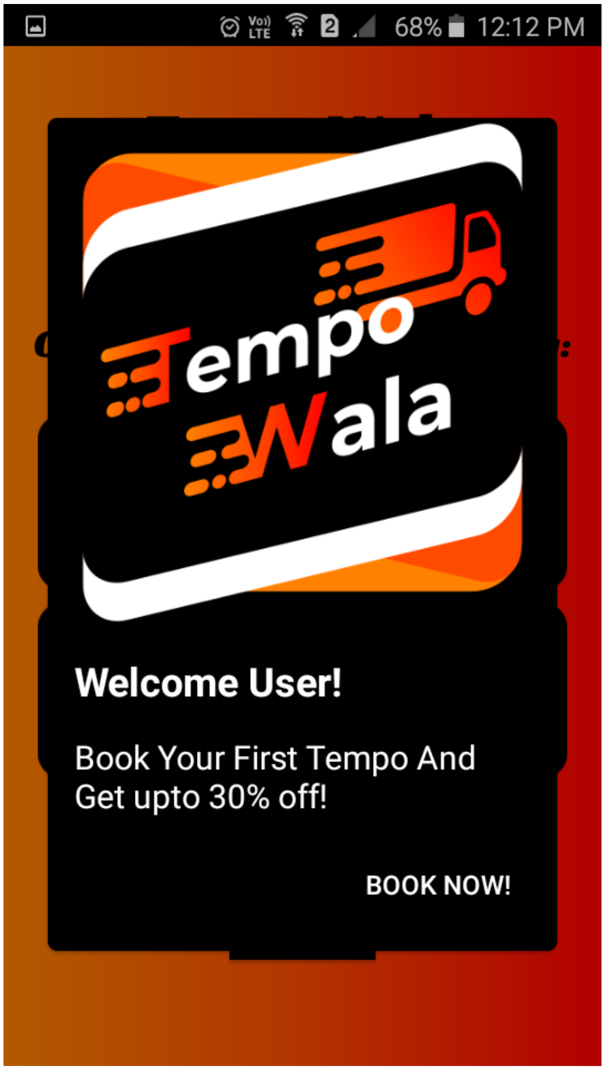
12:42

Saturday, 13 February am

TempoWala 1m

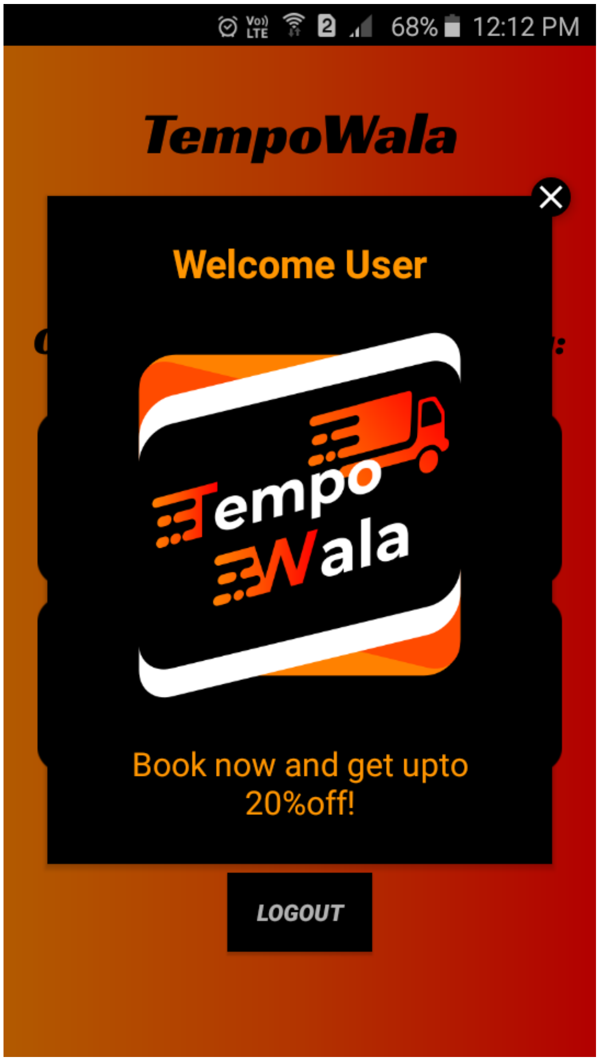
TempoWala

Today's Deal 20% off on all bookings! BOOK NOW!



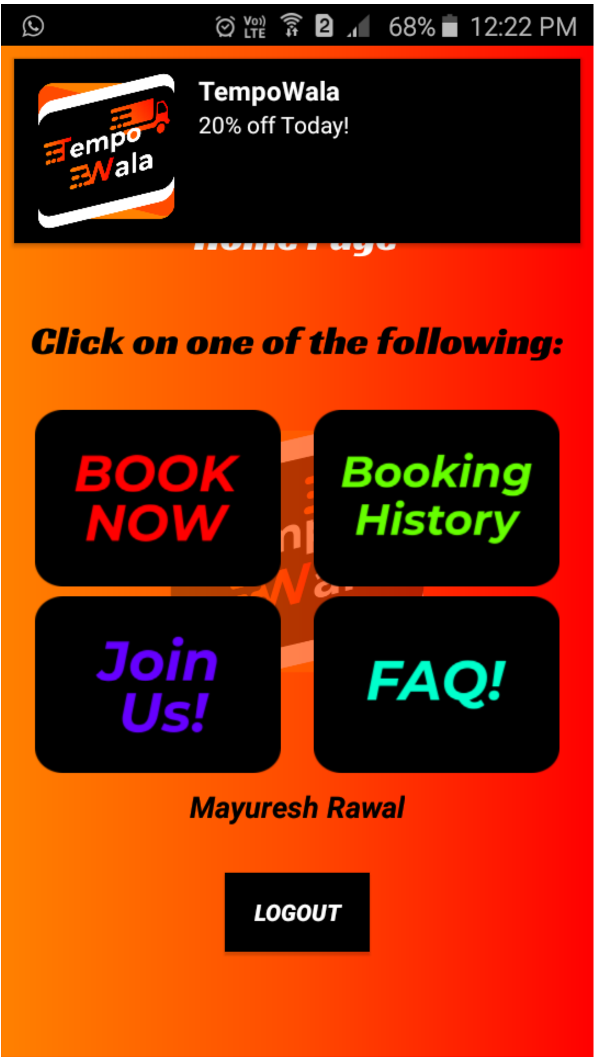
Welcome User!

Book Your First Tempo And Get upto 30% off!



Welcome User

Book now and get upto 20%off!



TempoWala
20% off Today!

Click on one of the following:

5.3 Testing:

The testing of the app has been done both via functional testing and user-acceptance testing.

The unit testing was done successfully during developing each unit.

The bugs were fixed during that phase and each unit as of now works properly.

The integration testing was also successful and the app didn’t break during integration.

Modifications and improvements were made whenever I found a bug big or small.

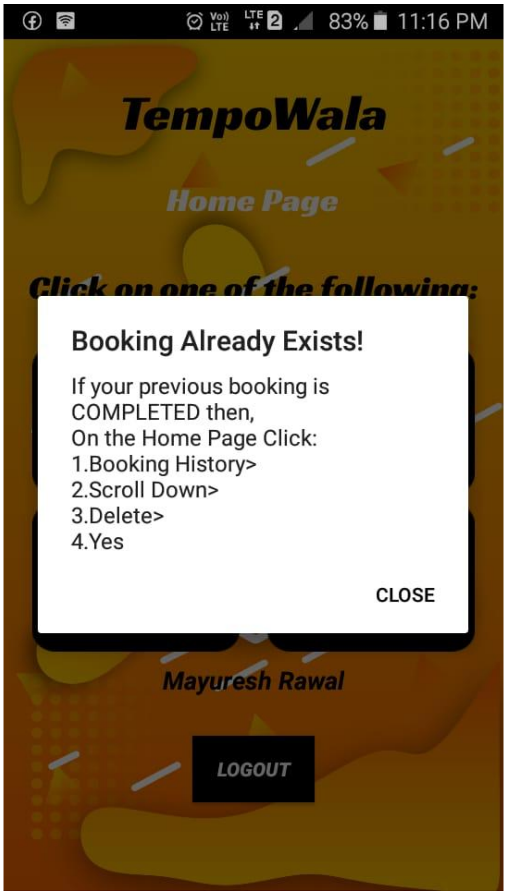
CHAPTER 6: RESULTS & DISCUSSION

6.1 Test Reports:

During Testing I saw that one of the more exploitable feature of this app was bombarding my database with multiple bookings which can hold only 10,000 bookings which is huge for testing purposes, but not good when it launches in the market.

So I locked the bookings to only at a time. This ensures proper flow of bookings and also my database can't be bombarded with multiple requests from the same user taking up space for other users.

So if the user wants to book again they have to wait for old booking to be finished and once finished then they can delete that booking and book a new one.



Booking Already Exists!

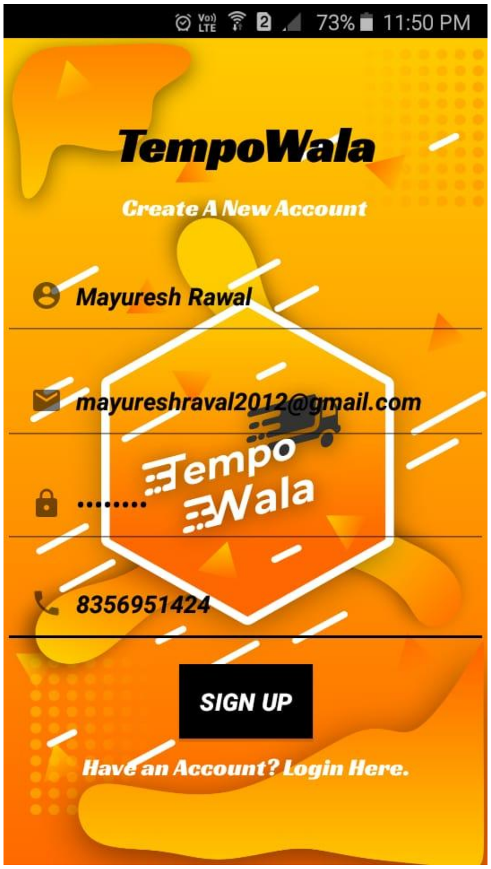
If your previous booking is
COMPLETED then,
On the Home Page Click:
1.Booking History>
2.Scroll Down>
3.Delete>
4.Yes

6.1 User Documentation:

Hello User welcome to the Documentation of TempoWala app

Let’s book our first tempo, but first we have to register ourselves.

Fill in the details and click on sign up button.

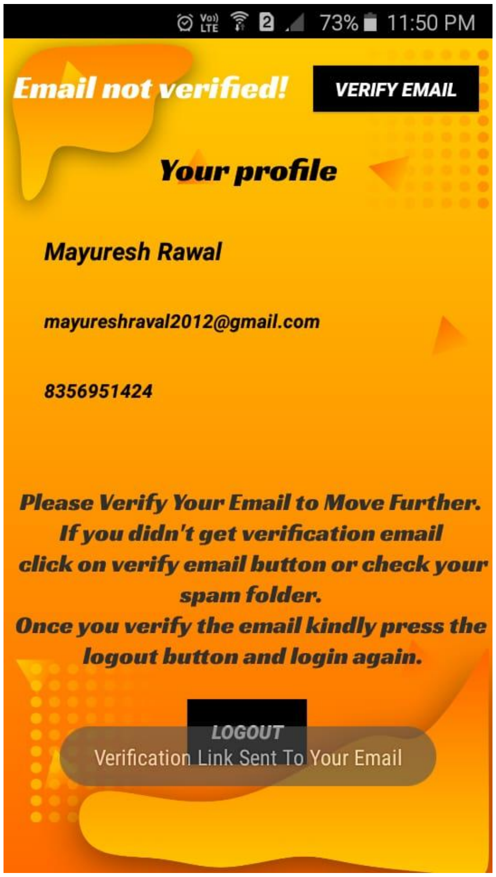


Mayuresh Rawal

mayureshraval2012@gmail.com

....

8356951424



Mayuresh Rawal

mayureshraval2012@gmail.com

8356951424

Please Verify Your Email to Move Further.
If you didn't get verification email
click on verify email button or check your
spam folder.
Once you verify the email kindly press the
logout button and login again.

Verification Link Sent To Your Email

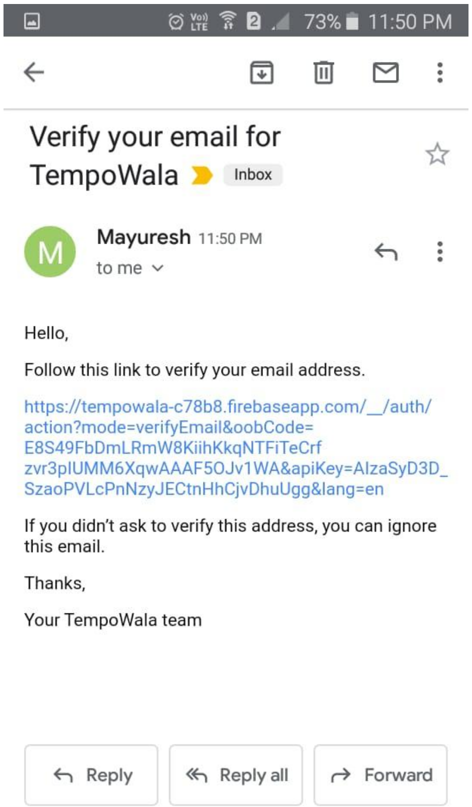
After that you would see the email verification page, to move further

You first have to verify your email.

To do this open your email app, Search for verify email for TempoWala click on the mail

And you will see a mail with link, click on that link to verify your email.

Once verified successfully you can logout and login again.



Verify your email for TempoWala

Mayuresh 11:50 PM
to me

Hello,

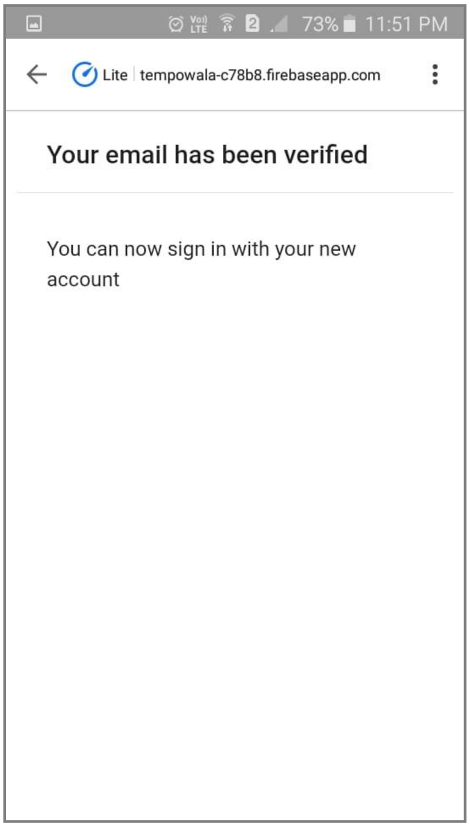
Follow this link to verify your email address.

https://tempowala-c78b8.firebaseapp.com/__/auth/action?mode=verifyEmail&oobCode=E8S49FbDmLRmW8KiihKkqNTFiTeCrfzvr3pIUMM6XqwAAAF5OJv1WA&apiKey=AIzaSyD3D_SzaoPVLcPnNzyJECtnHhCjvDhuUgg&lang=en

If you didn't ask to verify this address, you can ignore this email.

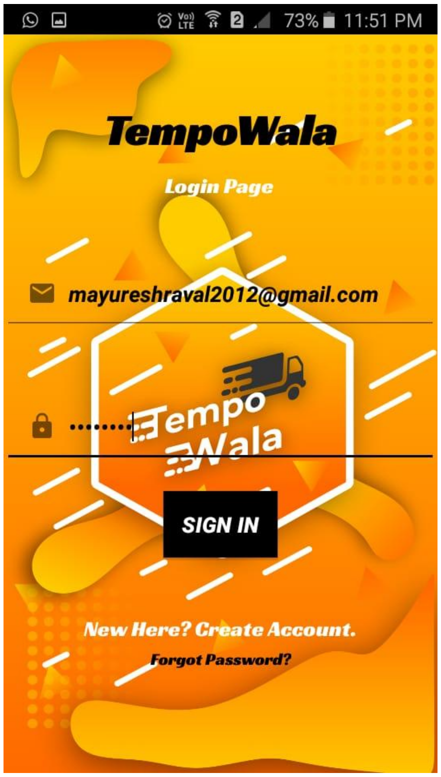
Thanks,

Your TempoWala team

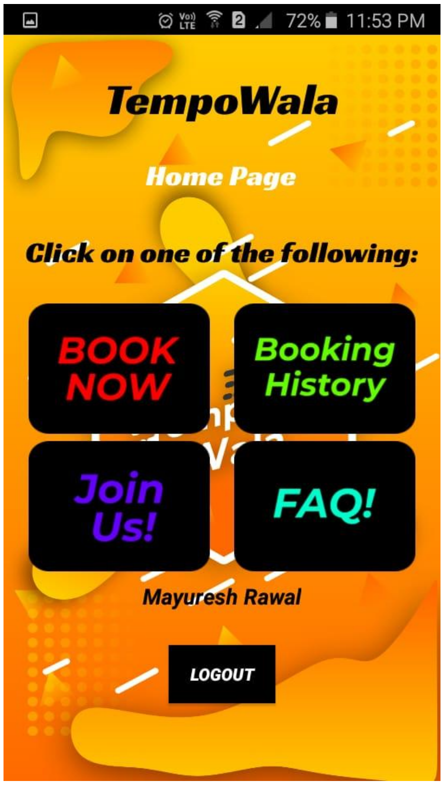


Your email has been verified

You can now sign in with your new account



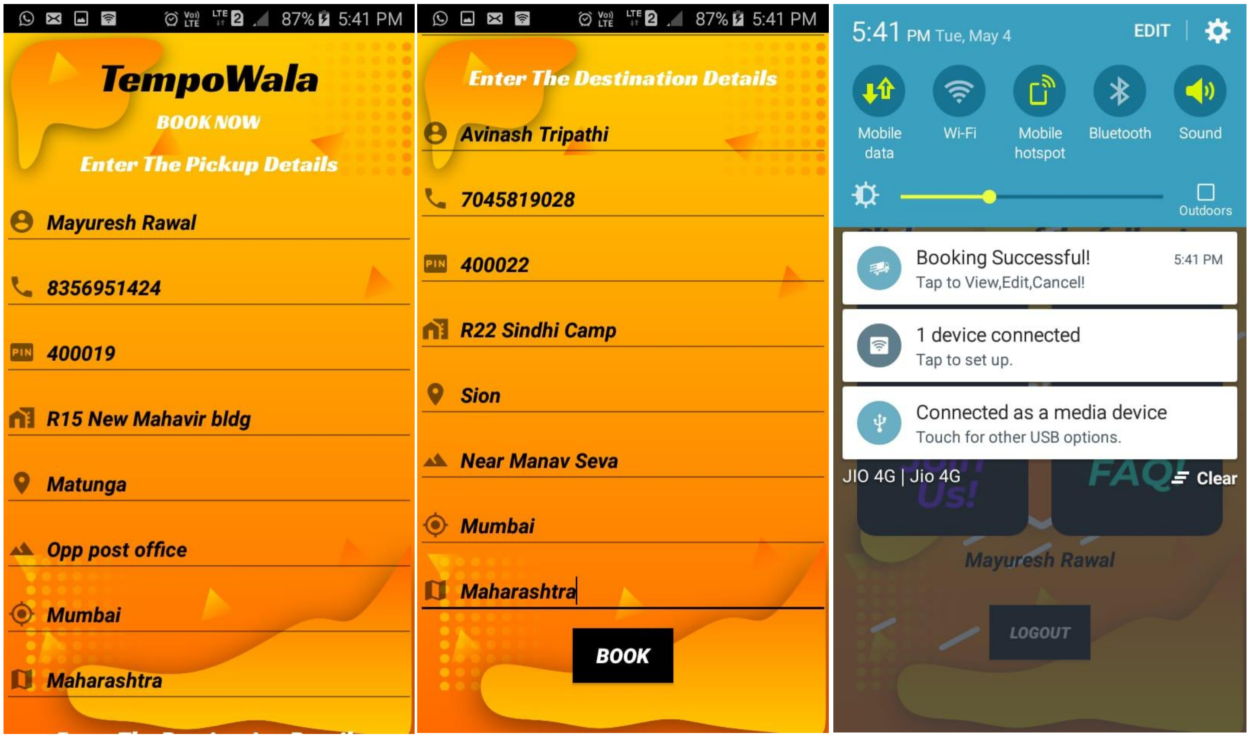
mayureshraval2012@gmail.com



Now once you Login Successfully you are all set to book tempos.

To book tempos click on the BOOK NOW button on the homepage.

Now fill in the details as required, Click on book, if successful you will see a notification above.



87% 5:41 PM

TempoWala

BOOK NOW

Enter The Pickup Details

Mayuresh Rawal

8356951424

400019

R15 New Mahavir bldg

Matunga

Opp post office

Mumbai

Maharashtra

87% 5:41 PM

Enter The Destination Details

Avinash Tripathi

7045819028

400022

R22 Sindhi Camp

Sion

Near Manav Seva

Mumbai

Maharashtra

BOOK

5:41 PM Tue, May 4

EDIT

Booking Successful! 5:41 PM
Tap to View,Edit,Cancel!

1 device connected
Tap to set up.

Connected as a media device
Touch for other USB options.

JIO 4G | Jio 4G

Clear

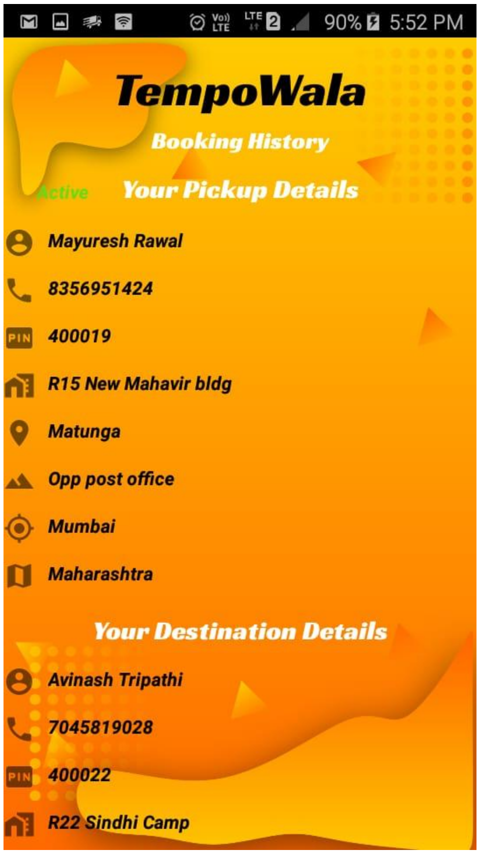
Mayuresh Rawal

LOGOUT

Congrats You Have Booked Your First Tempo!

__NOTE: VIDEO TUTORIALS FOR BOOKING ON FAQ PAGE.__

To view the booking history click on the BOOKING HISTORY button on home page , if there is no booking then it would show no booking , for now it will show the booking which we just made.



TempoWala

Booking History

Active

Your Pickup Details

Mayuresh Rawal

8356951424

400019

R15 New Mahavir bldg

Matunga

Opp post office

Mumbai

Maharashtra

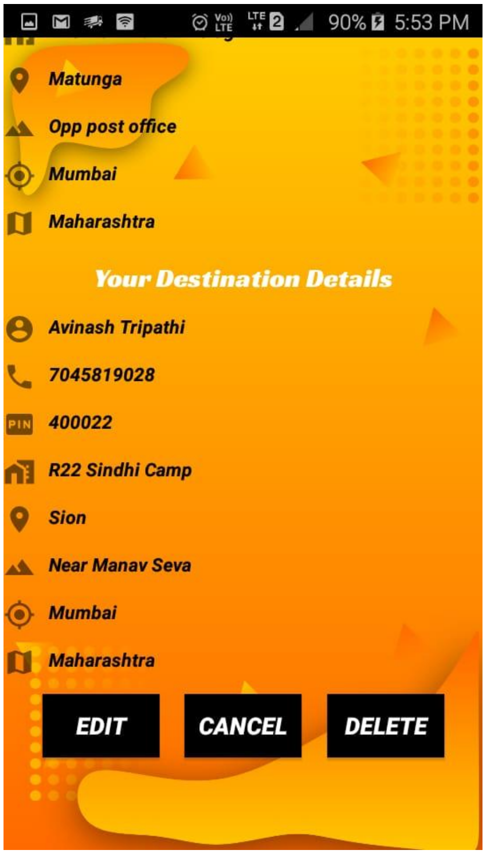
Your Destination Details

Avinash Tripathi

7045819028

400022

R22 Sindhi Camp



Matunga

Opp post office

Mumbai

Maharashtra

Your Destination Details

Avinash Tripathi

7045819028

400022

R22 Sindhi Camp

Sion

Near Manav Seva

Mumbai

Maharashtra

Booking history fully scrolled down

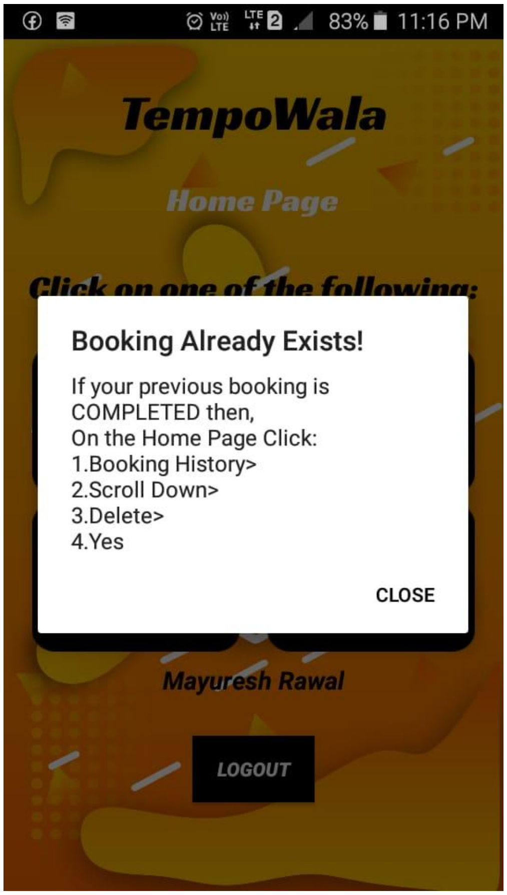
NOTE: VIDEO TUTORIALS FOR EDIT/CANCEL/DELETE ON FAQ PAGE.

To edit a booking simply click on the edit button edit whatever you like and once done click on the book button you will see an edit successful notification. The status will turn to active + updated.

To cancel a booking you can click on the cancel button which will cancel the current booking. The status will turn to cancelled.

And now what happens if you try to book another tempo,

Uh oh! You will see this



Booking Already Exists!

If your previous booking is COMPLETED then,
On the Home Page Click:
1.Booking History>
2.Scroll Down>
3.Delete>
4.Yes

To be able to book new tempos you need to delete the old booking so click on the delete button once you think the booking is not needed.

You will see a booking deleted notification and on clicking on booking history you will see No booking available.

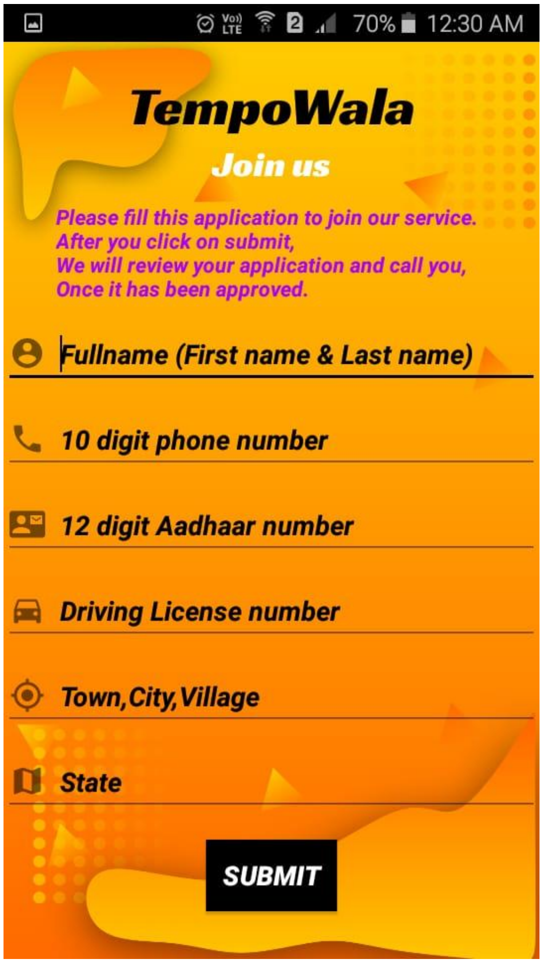
This will also remove your booking data from our database, your privacy in your hands.

Now you have become a master at using our app, more questions answered in the FAQ page which you can find by clicking on the FAQ page from the Home Page.

Let’s go to the join us page (Only for TempoWalas).

NOTE: VIDEO TUTORIALS FOR JOIN US ON FAQ PAGE.

Please fill in the details carefully and responsibly because you can only submit this form once. After clicking on submit your application will be reviewed by us, and once approved we will call you.



Please fill this application to join our service.
After you click on submit,
We will review your application and call you,
Once it has been approved.

Fullname (First name & Last name)

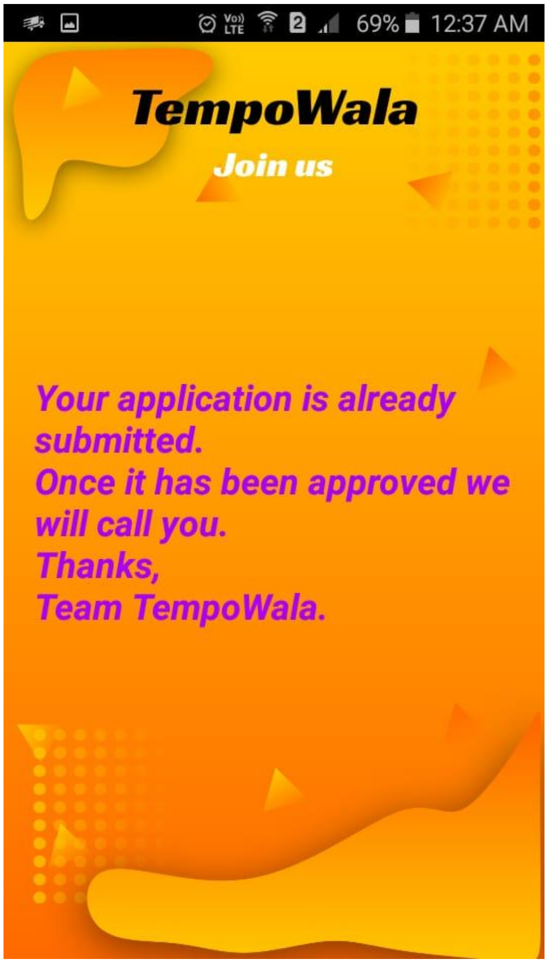
10 digit phone number

12 digit Aadhaar number

Driving License number

Town,City,Village

State



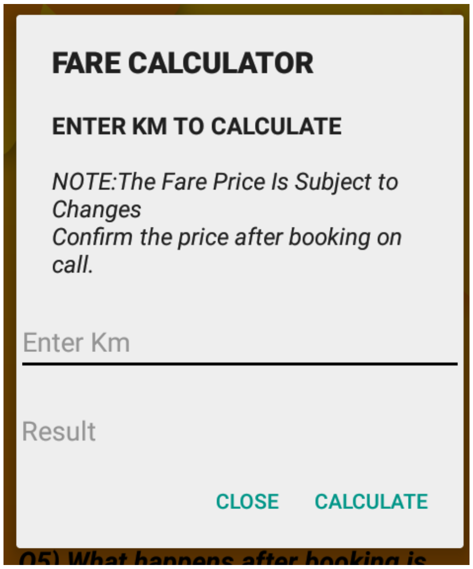
Your application is already submitted.
Once it has been approved we will call you.
Thanks,
Team TempoWala.

Before submitting

After submitting

Now it’s time to explore some extra hidden features!

The fare calculator on FAQ page helps you to calculate fare price.



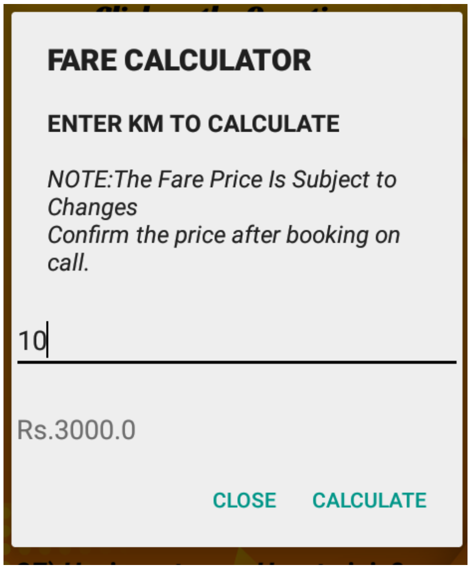
FARE CALCULATOR

ENTER KM TO CALCULATE

NOTE:The Fare Price Is Subject to Changes
Confirm the price after booking on call.

Enter Km

Result



FARE CALCULATOR

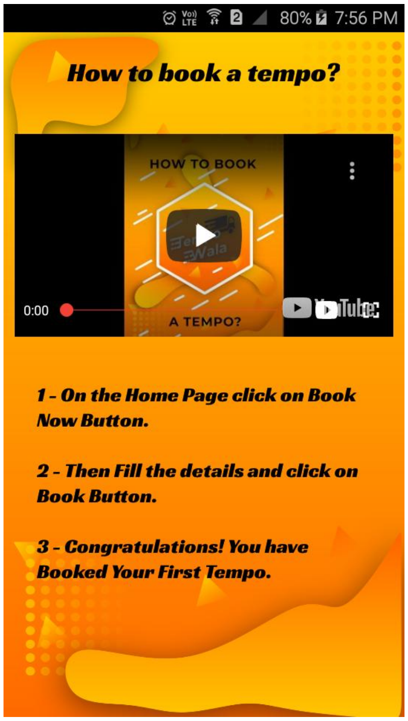
ENTER KM TO CALCULATE

NOTE:The Fare Price Is Subject to Changes
Confirm the price after booking on call.

10

Rs.3000.0

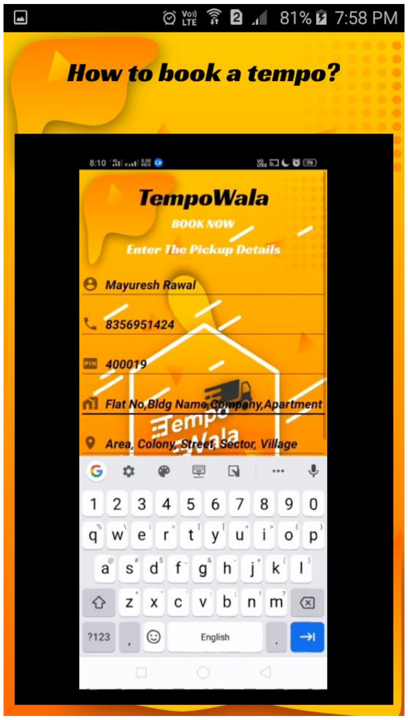
How to book? Click to watch video will show you video tutorial



1 - On the Home Page click on Book Now Button.

2 - Then Fill the details and click on Book Button.

3 - Congratulations! You have Booked Your First Tempo.



Mayuresh Rawal

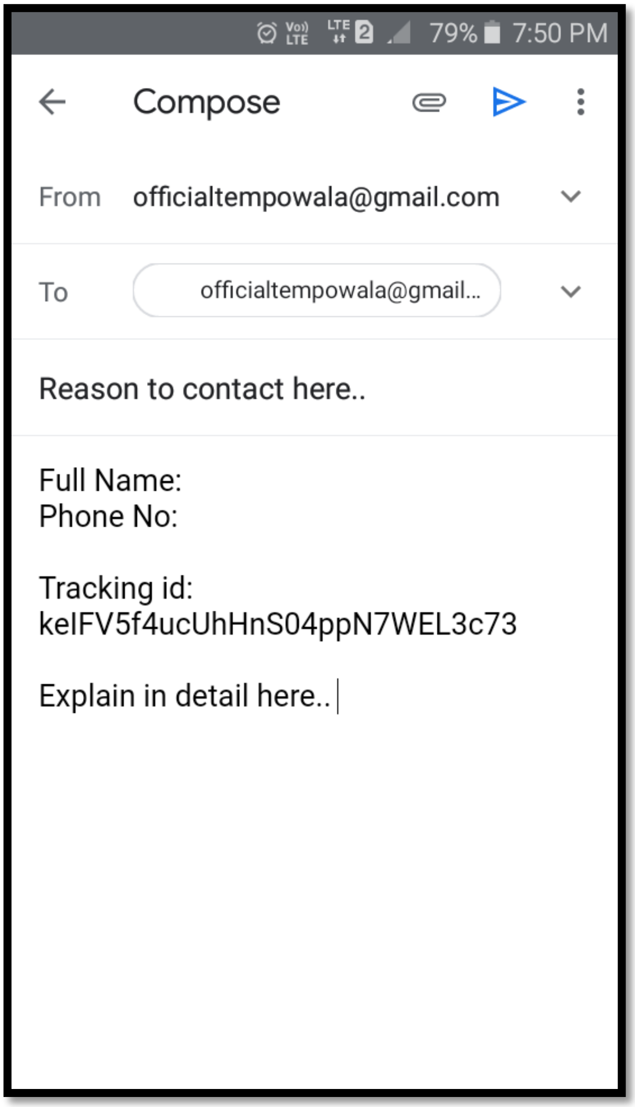
8356951424

400019

Flat No,Bldg Name,Company,Apartment

Area, Colony, Street, Sector, Village

Contact Us/Lodge Complaint opens up email for you .



From officialtempowala@gmail.com

To officialtempowala@gmail...

Reason to contact here..

Full Name:
Phone No:

Tracking id:
keIFV5f4ucUhHnS04ppN7WEL3c73

Explain in detail here..

In the subject enter the reason to contact in short

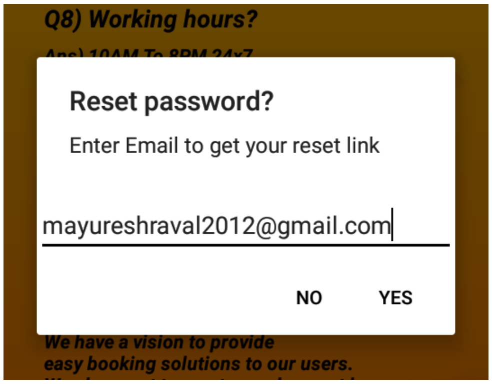
Enter your full name and phone no as mentioned

And don’t touch the tracking id it helps us to view your order,

And then explain in detail below.

Easy as that!

To reset password click on Forgot password/Reset Password on Faq page.



Q8) Working hours?

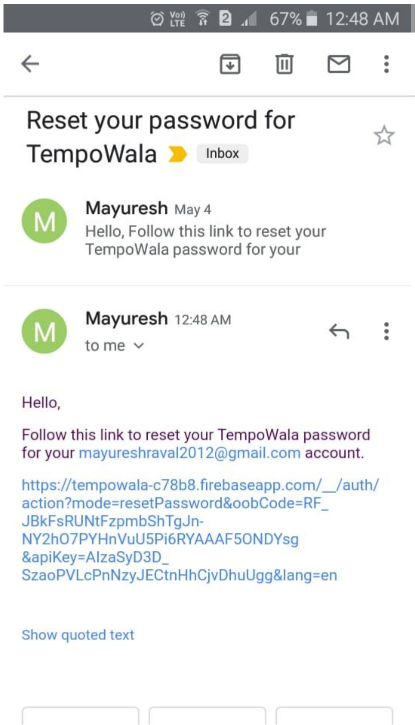
Ans) 10AM To 8PM 24x7

Reset password?

Enter Email to get your reset link

mayureshraval2012@gmail.com

We have a vision to provide easy booking solutions to our users.



Reset your password for TempoWala

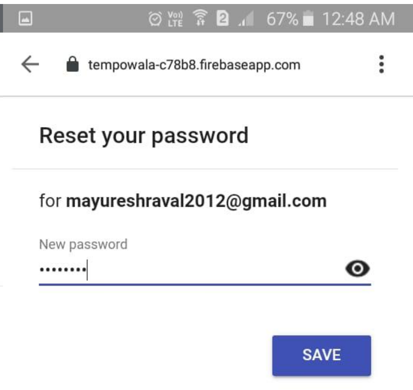
Mayuresh May 4
Hello, Follow this link to reset your TempoWala password for your

Mayuresh 12:48 AM
to me

Hello,

Follow this link to reset your TempoWala password for your mayureshraval2012@gmail.com account.

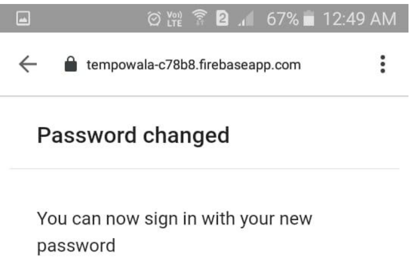
https://tempowala-c78b8.firebaseapp.com/__/auth/action?mode=resetPassword&oobCode=RF_JBkFsRUNtFzpmbShTgJn-NY2hO7PYHnVuU5Pi6RYAAAF5ONDYsg&apiKey=AIzaSyD3D_SzaoPVLcPnNzyJECtnHhCjvDhuUgg&lang=en



Reset your password

for mayureshraval2012@gmail.com

New password: ••••••••



Password changed

You can now sign in with your new password

Open email Enter new pass Success

Click the link

The last but not the least the top secret feature is that the app can book even while offline. Turn off internet to test it. The booking will be done via SMS. Mostly these days we get free SMS daily even if not it will cost less than 3 rupees.

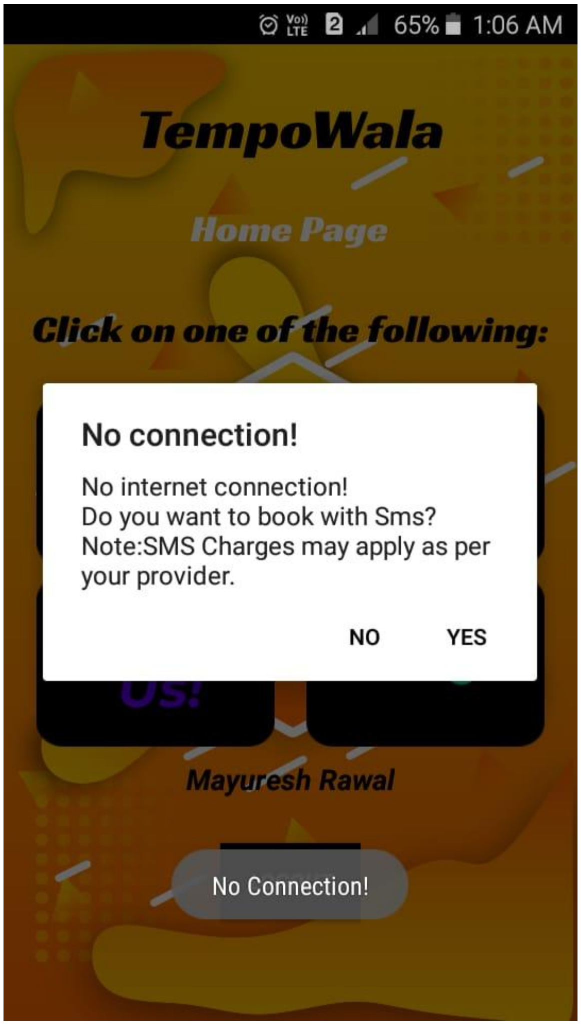
The limitations here

1. SMS support only 165 chars max, I need some extra characters for tracking id. So the address is broken into 2 parts each will have 130 chars max, and once you click on book the app will send 2 SMS. One for pickup and second for destination.
2. When you turn on internet you wouldn't be able to see this booking in the booking history, nor you would be able to edit it cancel it or delete it. To do such our employee will call you in few minutes Ask them to edit/cancel/delete it.

To book via offline mode turn off internet or when there's no connection click on BOOK now button it will show you a message.

Press yes to continue, fill in details don't worry about characters if your address is longer than 130 chars it will automatically prompt you on pressing book button. Once everything is perfect. Hit the book button and there you go you have booked via offline SMS mode.

THE SCREENSHOTS ARE ON THE NEXT PAGE

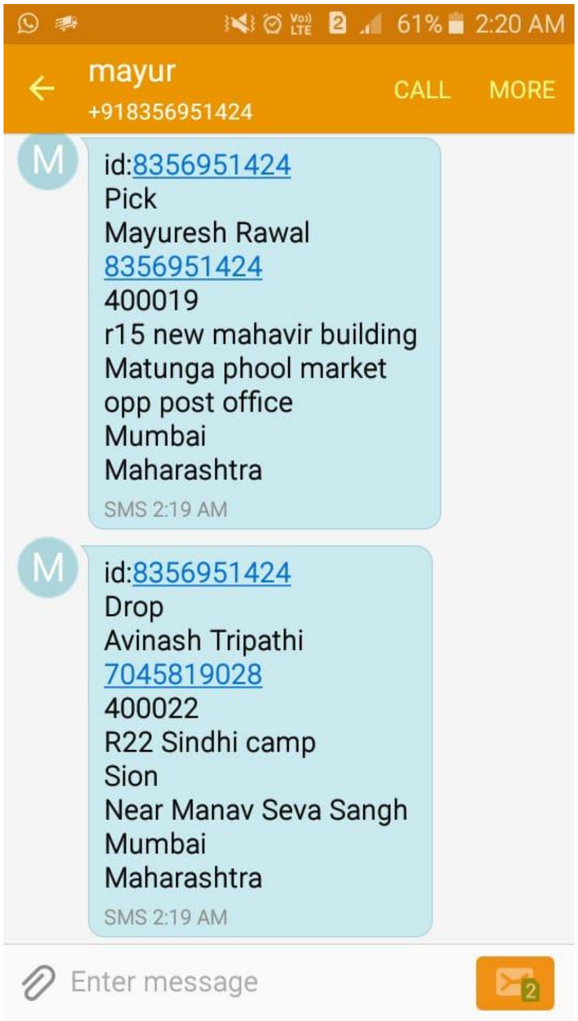


No connection!

No internet connection!
Do you want to book with Sms?
Note:SMS Charges may apply as per your provider.

No Connection!

No connection prompt



id:8356951424
Pick
Mayuresh Rawal
8356951424
400019
r15 new mahavir building
Matunga phool market
opp post office
Mumbai
Maharashtra
SMS 2:19 AM

id:8356951424
Drop
Avinash Tripathi
7045819028
400022
R22 Sindhi camp
Sion
Near Manav Seva Sangh
Mumbai
Maharashtra
SMS 2:19 AM

This is how I receive
Your bookings.

Chapter 7 – Conclusion

So we can conclude that the app is functional and can perform in different scenarios. It serves meaningful purpose in real world application and can be launched to public for use.

The app approaches simple address based approach which is easier to understand and use and anyone without too much technical knowledge can easily figure out how to book a tempo.

The user interface as promised is also simple and easy to use that there's no confusion about how things work.

The app also does unique things which other rival like porter can't do like offline SMS booking.

It also can be concluded that there are no bugs left in the app and is pretty stable and does not crash.

Let's talk about the limitations which cannot be fixed (for now)

1. Database can take only 10000 entries, so I have to limit each user to 1 booking per time.
2. SMS booking can only be limited to 165 chars 130 for user.
3. SMS booking can't be shown in booking history.

Future Scopes:

Improving UI, Scaling database, adding new features, crash analytics, dynamic links, advanced fare calculator, custom alert dialogs, remote config.

This concludes my project and so glad that it came out to be functional which matters the most to me, before making this I didn’t even know how to use android studio and how to code properly, after this project I am confident that I can make such apps now. Thank you for viewing this documentation, I really appreciate your time and effort to read through my documentation.

REFERENCES:

https://www.youtube.com/watch?v=tbh9YaWPKKs&t=186s

Small academy- How to create a login page using firebase?

https://stackoverflow.com/questions/66042279/how-to-visible-invisible-using-onclicklistener?noredirect=1#comment116769289_66042279

My question on how to set visible invisible, which I later used to make FAQ page.

https://stackoverflow.com/questions/65965848/how-to-check-if-a-firebase-document-exists-or-not

again my question on how to check if a document exists or not.

https://github.com/PierfrancescoSoffritti/android-youtube-player

YouTube viewer library by PIER FRANCESCO SOFFRITTI

Whole code:

https://github.com/mayureshraval/TempoWala