ABSTRACT

Tracking college buses has been a mainstream issue for city college students. We are never aware of the right time of bus arrival in a bus stop due to common issues like traffic, bus break down. Waiting time for transport in such crowded cities leads to less productivity on a whole. Students also pay bus fees early in the year and most of the time do not use bus services on a regular basis. This burdens them financially and they tend to use public transport for cheaper transport. College Administration can also not change bus route timings at their discretion since students are not given live bus tracking and bus updates if driver not available. Hence, we came up with an application which will allow students to get live status of the bus running in all routes, details of the driver and also will help the student to know how much time the bus will take to reach the current stop location of the student. The application will also allow booking and reserving seats in any bus on a daily basis irrespective of whether you belong to the bus route or not.

TABLE OF CONTENTS

1.INTRODUCTION

In the busy metropolitan cities, students don't have time to invest in waiting for college buses. Waiting time for transport in such crowded cities leads to less productivity on a whole. Students face this problem in their daily life where they have no idea about the current status of the college bus. So, the proposed solution is an android based application that will help the student to check out the current location of the college bus and also will help the student to know how much time the bus will take to reach the current location of the student. Basic android application will be interfacing with the updated database to provide the real-time data to the student, hence enhancing their experience.

YELLOW BUS is an application that tracks a bus and gathers the distance to station along its route. The application involves the installation of an electronic device in the bus, with an installed Android App on any smart phone to enable the students to track the vehicle's location. Buses carry GPS devices to track their positions. By this, positions to server are periodically updated. The application displays map showing the position of bus. It shows where bus is on the google map and provide students the updated information at different time interval. The bus-driver/student will monitor location and will store its data in the database. It is a real-time system as this method automatically sends the information on the GPS system to system/SMART phone. Since this is an android application, we use firebase server database for the backend. The students can get flexibility of planning travel to college using the app, to decide on which bus to take or when to catch the bus. Simple mode of communication is the key feature of the YELLOW BUS application.

This application can be easily extended for central tracking system to keep track of all the public vehicles. The different queries and efficient route management can be easily done through central server system.

2.PROPOSED SYSTEM

2.1. EXISTING SYSTEM

- *Live Tracking of Buses which have inbuilt GPS Modules fixed in them.*
- *Applications only provide live location of the driver*
- *Existing applications do not provide details of the current driver driving the bus as they return only GPS location of bus (and not of the driver's)*
- *Does not allow students to travel in other bus routes as seats in the bus are reserved for the whole year.*
- *Existing applications does not show Estimated time of arrival considering factors like traffic.*

2.2. PROPOSED SYSTEM

- *Live Location by accessing GPS coordinates using application installed in driver's mobile.*
- *Student view shows all the bus routes available along with the driver details.*
- *Real time location storing in Firebase Real time database.*
- *Firebase Authentication for all users of the application.*
- *Bus seat booking on a daily basis to allow students to travel in the bus they prefer.*
- *Real time ETA provided as directions are shown in Google Maps.*
- *Student will be redirected to payment page for payment*

3.PROJECT REQUIREMENTS

3.1. HARWARE REQUIREMENTS

Developer

- PC With 8GB RAM Minimum
- 5GB HardDisk minimum
- Processor – Preferred anything powerful enough as intel i5/i7

3.2. SOFTWARE REQUIREMENTS –

Developer

- Operating System – Windows/Ubuntu
- Code – Java/XML/JSON Objects (Firebase Database)
- Tools -Android Studio/Firebase

User

- Operating System – Android OS
- Version – Android 4.2 Jelly Bean and above
- Space – Not more than 20MB
- Internet Connection – Required with min speed of 128Kbps

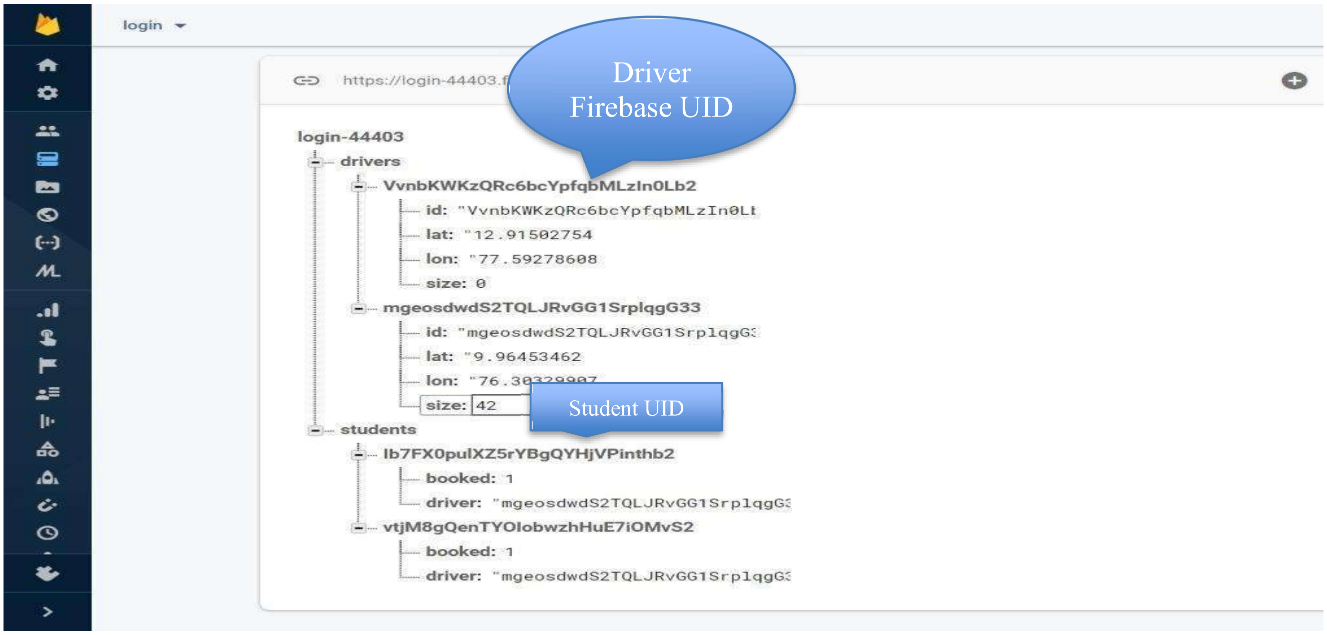
4.MODULE DESCRIPTION AND DATABASE ARCHITECTURE

4.1. Module Description

Register	Used to Register First time users into the App using Firebase Authentication.
Login	Used to Login existing users into the App using Firebase Authentication
Location	Used to Start Ride for Driver, Access Live current location of driver, store the driver details in Firebase Real-time database and End Ride on reaching destination.
Students Page	Used to Choose and reserve a seat in a convenient bus on a daily basis and view ETA of driver while student waits in the stop.
Seats	Used to book preferred seat on a seat matrix
GPay	Used to pay for Bus booking on a daily basis using GPay

TABLE 4.1. MODULE DESCRIPTION

4.2. Database Architecture



```
login-44403
  drivers
    VvnbKWKzQRc6bcYpfqbMLzIn0Lb2
      id: "VvnbKWKzQRc6bcYpfqbMLzIn0L
      lat: "12.91502754
      lon: "77.59278608
      size: 0
    mgeosdwdS2TQLJRvGG1SrplqgG33
      id: "mgeosdwdS2TQLJRvGG1SrplqgG
      lat: "9.96453462
      lon: "76.30329907
      size: 42
  students
    Ib7FX0pulXZ5rYBgQYHjVPinthb2
      booked: 1
      driver: "mgeosdwdS2TQLJRvGG1SrplqgG
    vtjM8gQenTYOlobwzhHuE7iOMvS2
      booked: 1
      driver: "mgeosdwdS2TQLJRvGG1SrplqgG
```

Fig 4.2. FIREBASE REALTIME DATABASE

The Firebase Realtime Database is a cloud-hosted database. Data is stored as JSON and synchronized in real-time to every connected client. Every time a driver starts the ride, his details get stored in the driver section of the database. Whenever a student books a seat in the bus, his details along the driver UID gets stored in the students' structure of the database as well as the number of seats gets reduced in the respective driver'

5.SYSTEM DESIGN

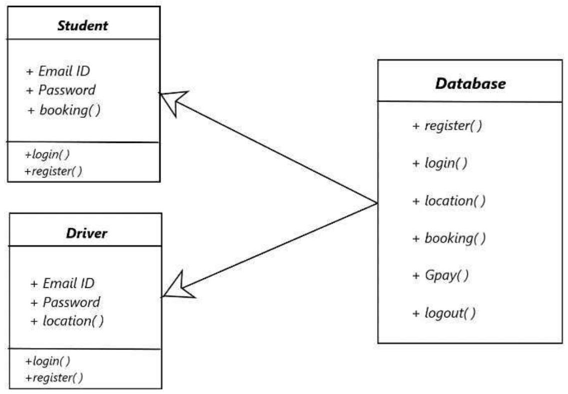
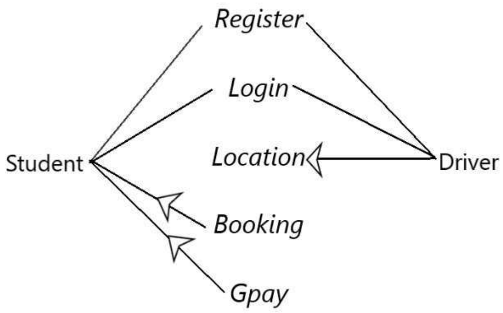


Fig 5.1.
Class Diagram of the Application (Conceptual Structure of the Application)

Fig 5.2.
Use Case Diagram of the Application (To Determine Linkage between different components of the system)



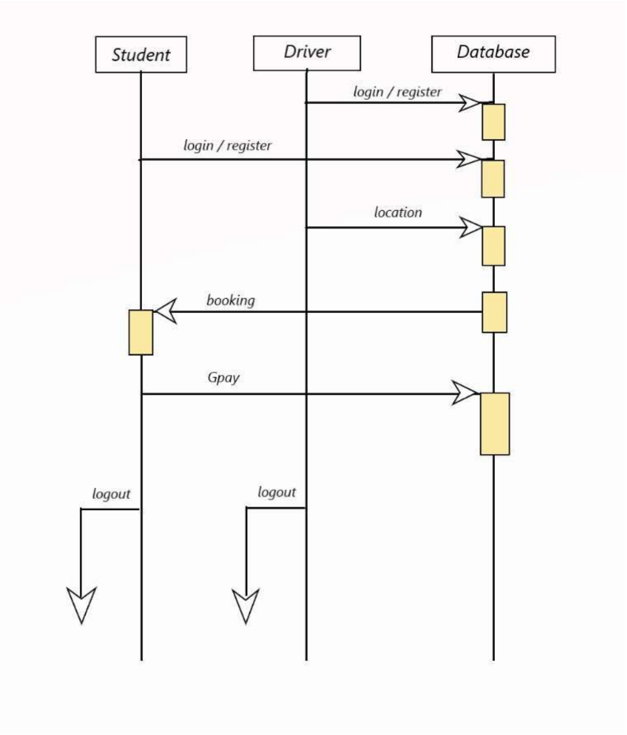


Fig 5.3.
Sequence Diagram of the Application (Flow of communication between various objects of the system)

6.PROJECT IMPLEMENTATION

6.1. MODULES

6.1.1. LOGIN AND REGISTRATION

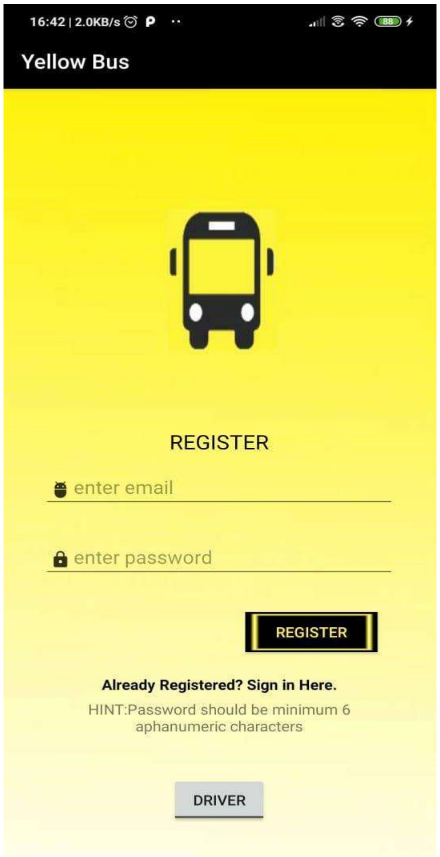
The Student and driver can register when they use the app for the first time using their email id and a password of length 6 which contain alphanumeric characters.

And all the users will be saved in the firebase authentication, the admin of the firebase can add users manually, remove users, send a mail to them to reset password in case they forgot it.

There will be a toggle button to choose whether they want to sign in as a driver or a student.

Once the registration is done it directly take the student and driver to their respective page depending on what they choose in the toggle button .Student to the page where the list of all drivers is available and can choose one , the driver will be taken to the location page where he/she should access their location and submit it for the first time.

On opening the app again it will directly take you to the respective page i.e student to the student page and driver to the location page without login again if they don’t want this to happen they can click on the logout button so their sign in details will be removed and the user will have to login again.

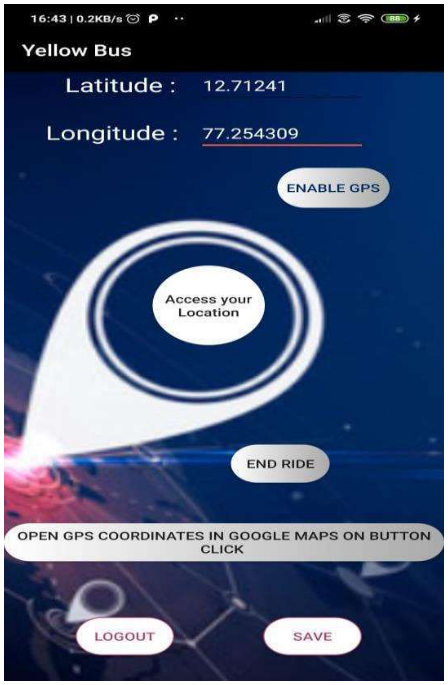


Already Registered? Sign in Here.

HINT:Password should be minimum 6 aphanumeric characters

Fig 6.1.1. Register Screen

6.1.2. DRIVER MODULE


16:43 | 0.2KB/s

Yellow Bus

Latitude : 12.71241

Longitude : 77.254309

Fig 6.1.2. Driver Screen

When the user login in successfully or registers successfully as a driver he/she will be redirected to location page where the driver will receive a toast message if their GPS is switched off or on , if off he/she will have to click on the enable button which will redirect to the GPS setting page of their phone to enable it.

After the GPS is enabled the driver will have to click “Access your location” which will fetch the users current location and put it in the textfield, if it’s the first time or a start of a new trip the user will get a toast telling to click on the save button which will save the location and reset the number of seats available in the firebase database after this the location gets auto updated after every 10 seconds and gets uploaded to the database since it is a real-time database it gets updated within 1second.

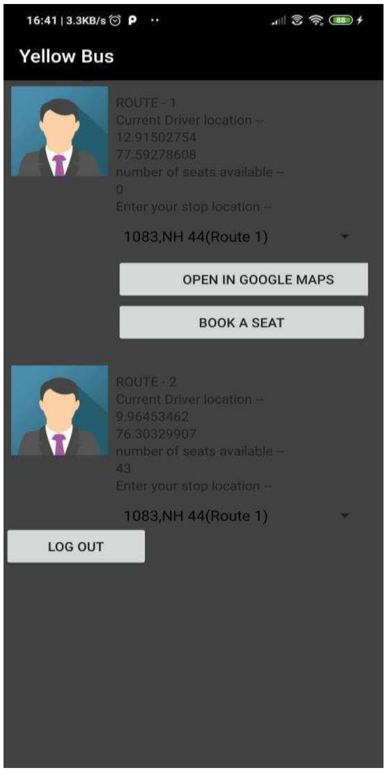
Once the trip is over the driver can click on end trip which will delete the driver details like location seats available etc. and all the students who booked under that driver and the driver can logout of the app and sign in again when his trip is about to start and repeat the process again.

6.1.3. STUDENT MODULE

When the user login in successfully or registers successfully as a student he/she will be redirected to student page where there is a recycler view of all the drivers who have started a trip.

It shows details like the current location of the driver, how many seats available in that bus, the stops of that driver, button which opens google maps to view ETA from driver’s location to the users stop and a book button. The user can select the stop which he wants to get in in the drop down present next to that particular driver, all drivers may or may not have the same stops so the user can decide which driver to choose, after selecting the stop the user can click on open in google maps will open google maps and show the ETA from the drivers location to the users stop.

When the user clicks on book a seat it will reserve a seat in the bus by updating it in the database it reduces one seat from the seats available of that driver and adds the students details in the database and redirects them to the seats booking page.The user can book only one seat.


16:41 | 3.3KB/s

Yellow Bus

ROUTE - 1
Current Driver location --
12.91502754
77.59278608
number of seats available --
0
Enter your stop location --
1083,NH 44(Route 1)

ROUTE - 2
Current Driver location --
9.96453462
76.30329907
number of seats available --
43
Enter your stop location --
1083,NH 44(Route 1)

Fig 6.1.3. Student Screen

6.1.4. SEAT BOOKING AND PAYMENT MODULE

The application also allows students to book their preferred seat in the route they wish to travel in. From the student view, if the student is eligible to book a seat(i.e. if he hasn't booked a seat in any other route or the seats of the preferred routes isn't full),the student can click on the Book a seat which reserve a seat for the student and takes him to the screen which displays the seat matrix.

The student chooses his preferred seat based on availability and clicks on the Book option which stores the details in the database and redirects the student to the payment page.

The payment layout page is the screen where the student gives his name in a Textview and the Pay Now Button takes him to Google Pay which is the only mode of payment for the application. The student is expected to have a Google Pay Application and a UPI ID for the same for this process to be performed. OnClick the Pay Now button just starts and intent for the GPay.

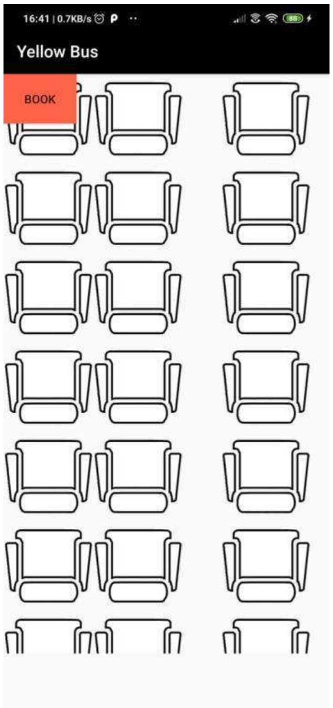
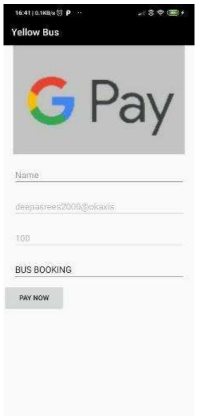


Fig 6.1.4. Seat Book Screen



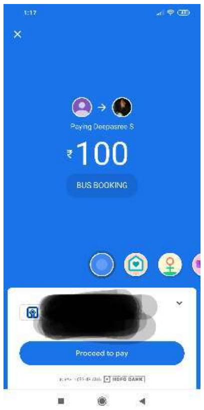
Name

deepasrees2000@okaxis

100

BUS BOOKING

Fig 6.1.5. Payment Screen



Paying Deepasree S

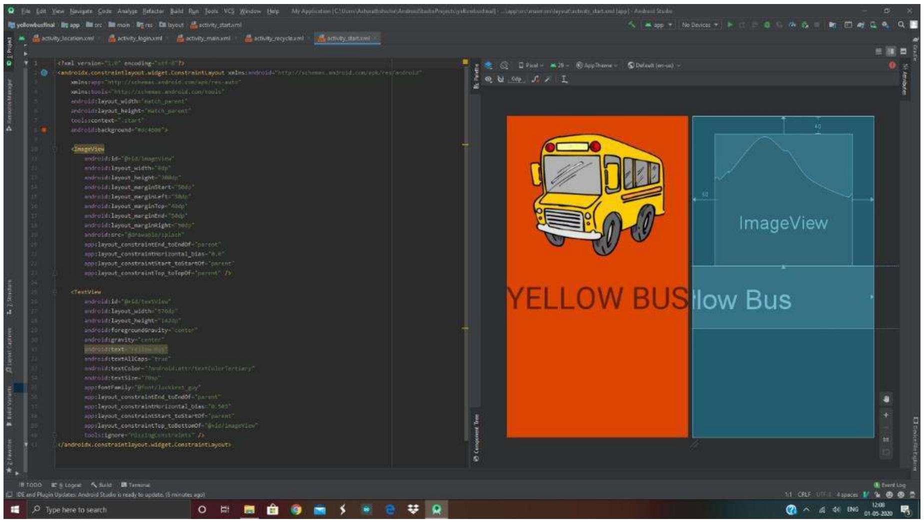
₹100

BUS BOOKING

[redacted]

Fig 6.1.6. GPay Payment Screen

6.2 SCREENSHOTS AND XML FILES

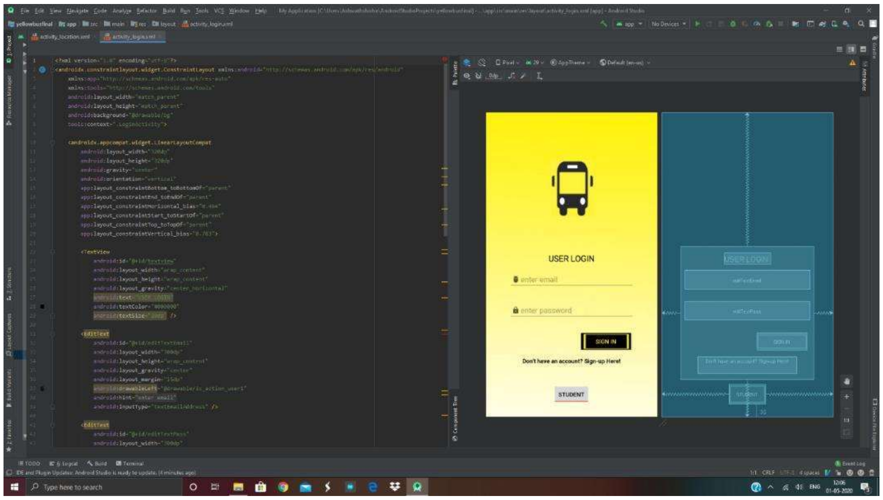

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
yellowbusfinal app src main res layout activity_start.xml
activity_location.xml activity_login.xml activity_main.xml activity_recycle.xml activity_start.xml
YELLOW BUS
ImageView
llow Bus
TODO 6: Logcat Build Terminal
Type here to search

```
<?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=".start"
    android:background="#dc4000">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="0dp"
        android:layout_height="300dp"
        android:layout_marginStart="50dp"
        android:layout_marginLeft="50dp"
        android:layout_marginTop="40dp"
        android:layout_marginEnd="50dp"
        android:layout_marginRight="50dp"
        android:src="@drawable/splash"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.0"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <TextView
        android:id="@+id/textView"
        android:layout_width="570dp"
        android:layout_height="142dp"
        android:foregroundGravity="center"
        android:gravity="center"
        android:text="Yellow Bus"
        android:textAllCaps="true"
        android:textColor="?android:attr/textColorTertiary"
        android:textSize="70sp"
        app:fontFamily="@font/luckiest_guy"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.503"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/imageView"
        tools:ignore="MissingConstraints" />
</androidx.constraintlayout.widget.ConstraintLayout>
```

IDE and Plugin Updates: Android Studio is ready to update. (6 minutes ago)

Fig 6.2.1.
Opening Screen Views –

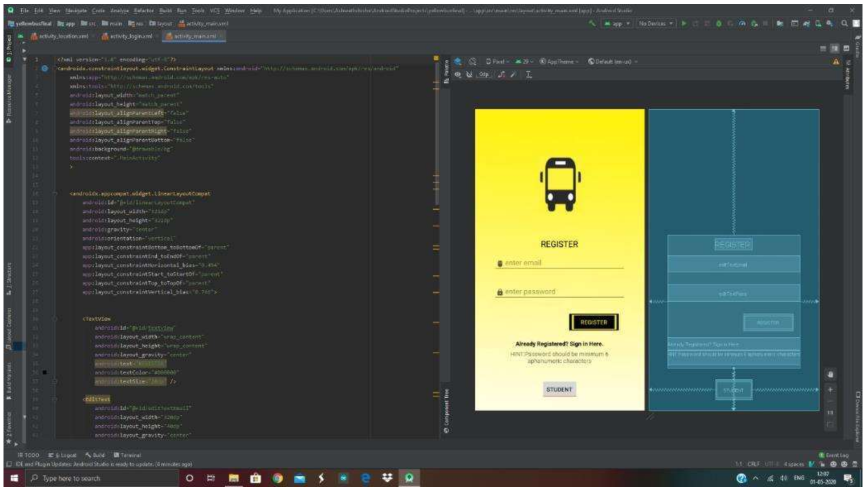
1. Image View
2. Text View


File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
yellowbusfinal app src main res layout activity_login.xml
activity_location.xml activity_login.xml
USER LOGIN
enter email
enter password
SIGN IN
Don't have an account? Sign-up Here!
STUDENT
TODO 6: Logcat Build Terminal
Type here to search

```
<?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/bg"
    tools:context=".LoginActivity">

    <androidx.appcompat.widget.LinearLayoutCompat
        android:layout_width="320dp"
        android:layout_height="320dp"
        android:gravity="center"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.494"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.783">

        <TextView
            android:id="@+id/textview"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center_horizontal"
            android:text="USER LOGIN"
            android:textColor="#000000"
            android:textSize="20sp" />

        <EditText
            android:id="@+id/editTextEmail"
            android:layout_width="300dp"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:layout_margin="15dp"
            android:drawableLeft="@drawable/ic_action_user1"
            android:hint="enter email"
            android:inputType="textEmailAddress" />

        <EditText
            android:id="@+id/editTextPass"
            android:layout_width="300dp"
```

IDE and Plugin Updates: Android Studio is ready to update. (4 minutes ago)

Fig 6.2.2.
Login Screen Views –
1.Text View
2.Image View
3.Edit Text
4. Buttons
5.Toggle Button

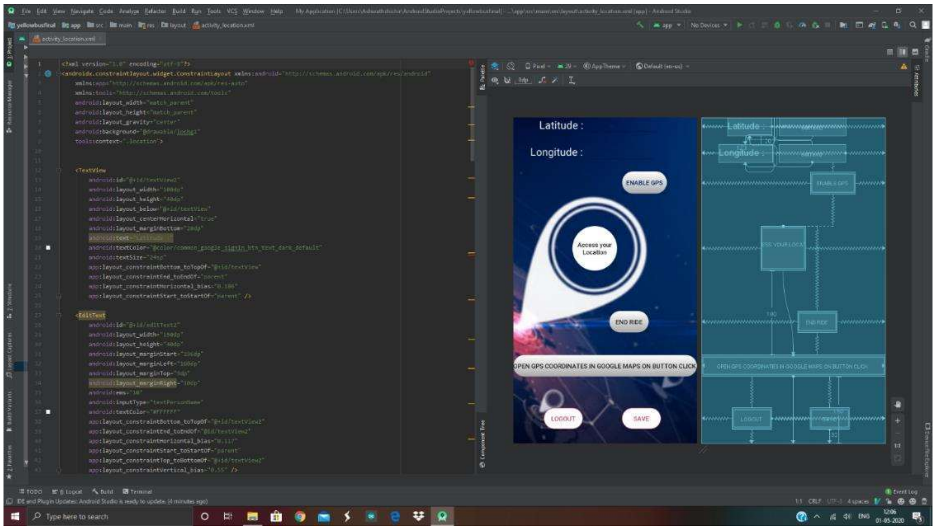

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
yellowbusfinal app src main res layout activity_main.xml
activity_location.xml activity_login.xml activity_main.xml
REGISTER
enter email
enter password
REGISTER
Already Registered? Sign in Here.
HINT:Password should be minimum 8 alphanumeric characters
STUDENT
TODO 6: Logcat Build Terminal
Type here to search

```
<?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:layout_alignParentLeft="false"
    android:layout_alignParentTop="false"
    android:layout_alignParentRight="false"
    android:layout_alignParentBottom="false"
    android:background="@drawable/bg"
    tools:context=".MainActivity"
    >


    <androidx.appcompat.widget.LinearLayoutCompat
        android:id="@+id/linearLayoutCompat"
        android:layout_width="321dp"
        android:layout_height="321dp"
        android:gravity="center"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.494"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.765">


        <TextView
            android:id="@+id/textView"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="REGISTER"
            android:textColor="#000000"
            android:textSize="20sp" />

        <EditText
            android:id="@+id/edittextemail"
            android:layout_width="320dp"
            android:layout_height="40dp"
            android:layout_gravity="center"
```

IDE and Plugin Updates: Android Studio is ready to update. (4 minutes ago)

Fig 6.2.3.
Register Screen Views –

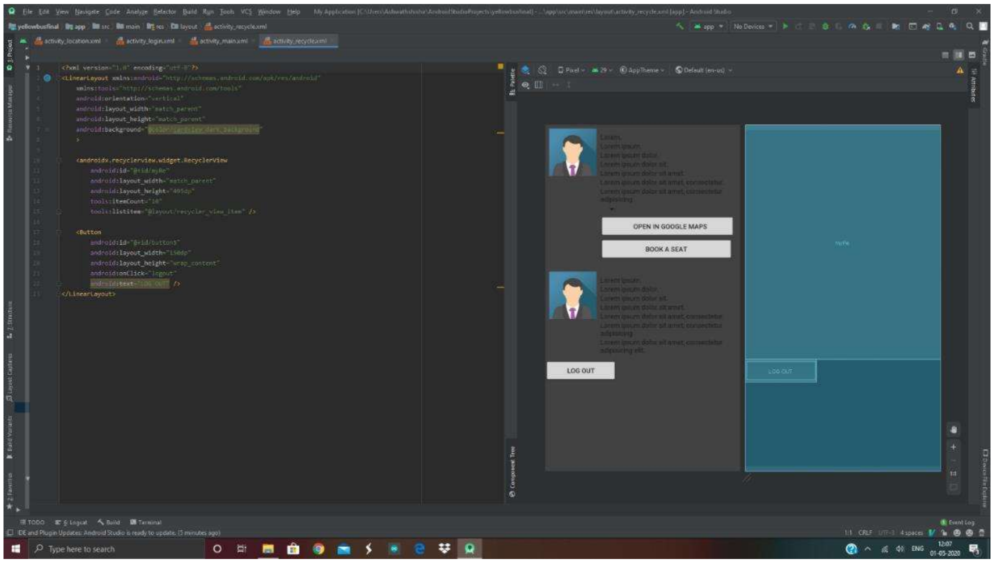
1. Text View
2. Edit Text
3. Buttons
4. Toggle Button



```
<?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:layout_gravity="center"
    android:background="@drawable/locbg1"
    tools:context=".location">


    <TextView
        android:id="@+id/textView2"
        android:layout_width="100dp"
        android:layout_height="40dp"
        android:layout_below="@+id/textView"
        android:layout_centerHorizontal="true"
        android:layout_marginBottom="20dp"
        android:text="Latitude :"
        android:textColor="@color/common_google_signin_btn_text_dark_default"
        android:textSize="24sp"
        app:layout_constraintBottom_toTopOf="@+id/textView"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintHorizontal_bias="0.186"
        app:layout_constraintStart_toStartOf="parent" />

    <EditText
        android:id="@+id/editText2"
        android:layout_width="150dp"
        android:layout_height="40dp"
        android:layout_marginStart="150dp"
        android:layout_marginLeft="100dp"
        android:layout_marginTop="5dp"
        android:layout_marginRight="10dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:textColor="#FFFFFF"
        app:layout_constraintBottom_toTopOf="@+id/textView2"
        app:layout_constraintEnd_toEndOf="@id/textView2"
        app:layout_constraintHorizontal_bias="0.117"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toBottomOf="@+id/textView2"
        app:layout_constraintVertical_bias="0.55" />
```

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
yellowbusfinal app src main res layout activity_location.xml
activity_location.xml
Latitude :
Longitude :
ENABLE GPS
Access your Location
END RIDE
OPEN GPS COORDINATES IN GOOGLE MAPS ON BUTTON CLICK
LOGOUT
SAVE
TODO
6: Logcat
Build
Terminal
IDE and Plugin Updates: Android Studio is ready to update. (4 minutes ago)
Type here to search

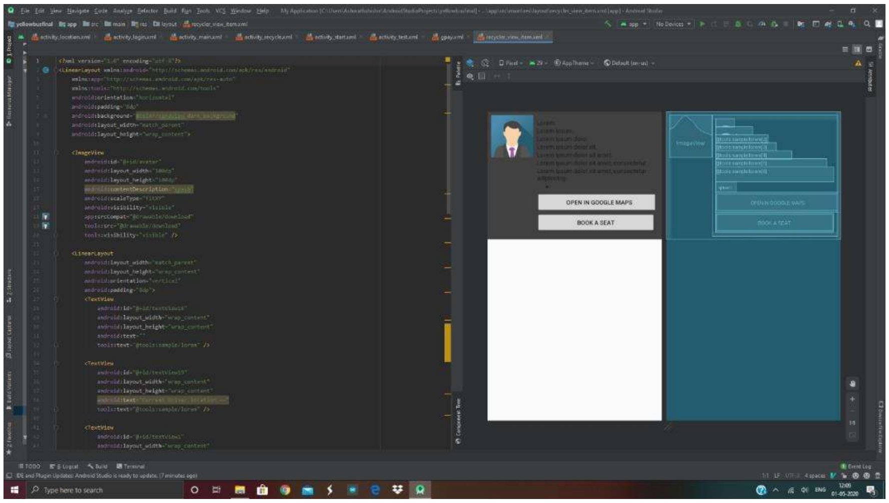
Fig 6.2.4.
Driver Screen Views-
1.Text View
2. Edit Text
3.Buttons



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/cardview_dark_background"
    >

    <androidx.recyclerview.widget.RecyclerView
        android:id="@+id/myRe"
        android:layout_width="match_parent"
        android:layout_height="405dp"
        tools:itemCount="10"
        tools:listitem="@layout/recycler_view_item" />

    <Button
        android:id="@+id/button5"
        android:layout_width="150dp"
        android:layout_height="wrap_content"
        android:onClick="logout"
        android:text="LOG OUT" />
</LinearLayout>
```

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
yellowbusfinal app src main res layout activity_recycle.xml
activity_location.xml
activity_login.xml
activity_main.xml
activity_recycle.xml
OPEN IN GOOGLE MAPS
BOOK A SEAT
LOG OUT
TODO
6: Logcat
Build
Terminal
IDE and Plugin Updates: Android Studio is ready to update. (3 minutes ago)
Type here to search

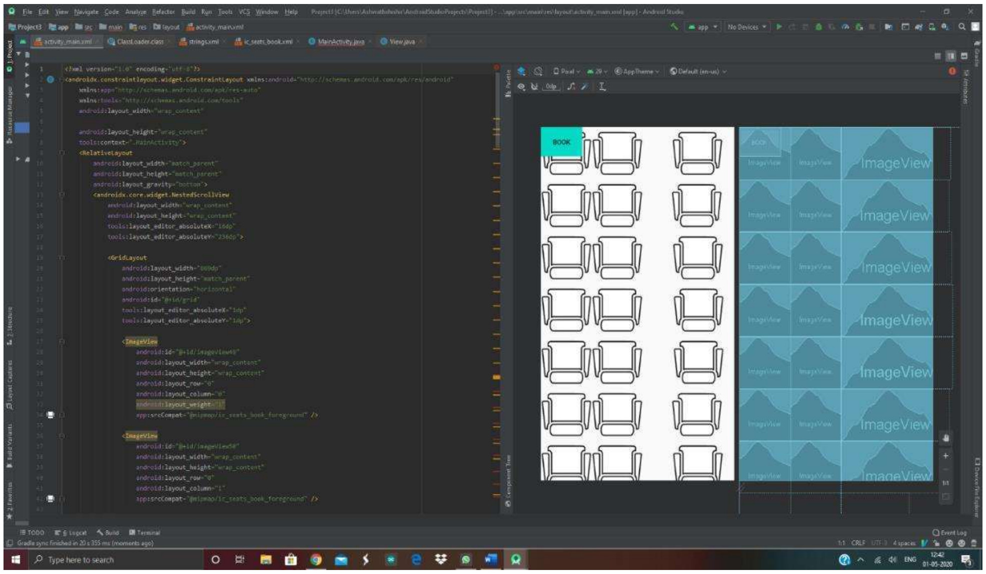
Fig 6.2.5.
Student Screen View –
1. Recycler View
2. Button



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout 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:orientation="horizontal"
    android:padding="5dp"
    android:background="@color/cardview_dark_background"
    android:layout_width="match_parent"
    android:layout_height="wrap_content">

    <ImageView
        android:id="@+id/avatar"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:contentDescription="[illegible]"
        android:scaleType="fitXY"
        android:visibility="visible"
        app:srcCompat="@drawable/download"
        tools:src="@drawable/download"
        tools:visibility="visible" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="vertical"
        android:padding="5dp">
        <TextView
            android:id="@+id/textView18"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text=""
            tools:text="@tools:sample/lorem" />

        <TextView
            android:id="@+id/textView19"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Current Driver location :-"
            tools:text="@tools:sample/lorem" />

        <TextView
            android:id="@+id/textView1"
            android:layout_width="wrap_content"
```

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help
yellowbusfinal app src main res layout recycler_view_item.xml
activity_location.xml
activity_login.xml
activity_main.xml
activity_recycle.xml
activity_start.xml
OPEN IN GOOGLE MAPS
BOOK A SEAT
TODO
6: Logcat
Build
Terminal
IDE and Plugin Updates: Android Studio is ready to update. (7 minutes ago)
Type here to search

Fig 6.2.6.
Student Screen Adapter View –
1. Image View
2. Text View
3. Button
4. Spinner

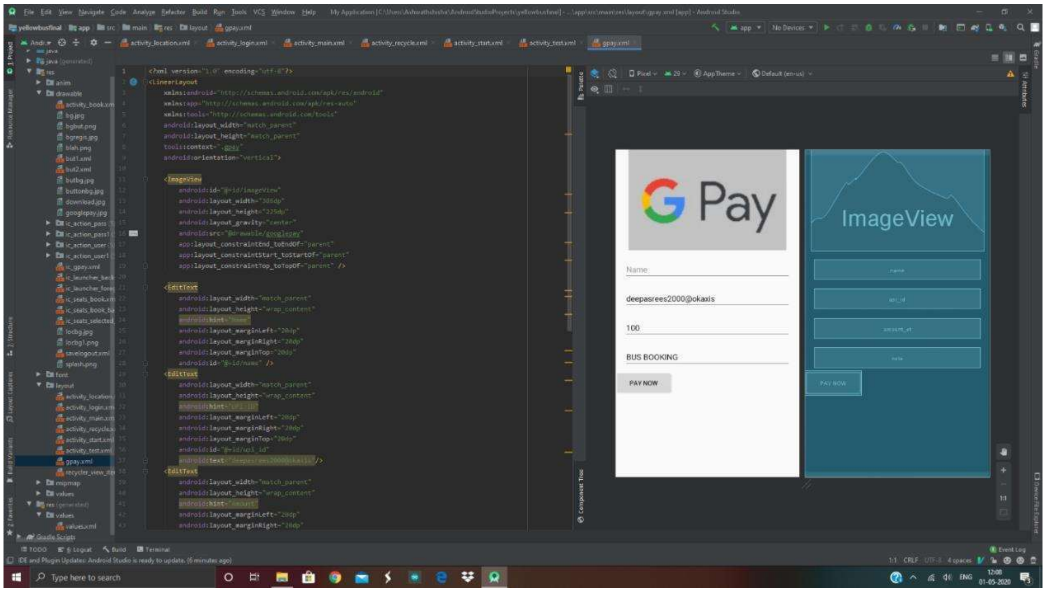


```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="wrap_content"

    android:layout_height="wrap_content"
    tools:context=".MainActivity">
    <RelativeLayout
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:layout_gravity="bottom">
        <androidx.core.widget.NestedScrollView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            tools:layout_editor_absoluteX="16dp"
            tools:layout_editor_absoluteY="236dp">

            <GridLayout
                android:layout_width="809dp"
                android:layout_height="match_parent"
                android:orientation="horizontal"
                android:id="@+id/grid"
                tools:layout_editor_absoluteX="1dp"
                tools:layout_editor_absoluteY="1dp">

                <ImageView
                    android:id="@+id/imageView49"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_row="0"
                    android:layout_column="0"
                    android:layout_weight="1"
                    app:srcCompat="@mipmap/ic_seats_book_foreground" />

                <ImageView
                    android:id="@+id/imageView50"
                    android:layout_width="wrap_content"
                    android:layout_height="wrap_content"
                    android:layout_row="0"
                    android:layout_column="1"
                    app:srcCompat="@mipmap/ic_seats_book_foreground" />
```

Gradle sync finished in 20 s 355 ms (moments ago)

Fig 6.2.7.
Seat Book Screen Views –

1. Scroll View
2. Image View
3. Button



```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    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=".gpay"
    android:orientation="vertical">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="300dp"
        android:layout_height="225dp"
        android:layout_gravity="center"
        android:src="@drawable/googlepay"
        app:layout_constraintEnd_toEndOf="parent"
        app:layout_constraintStart_toStartOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Name"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:layout_marginTop="20dp"
        android:id="@+id/name" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="upi id"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
        android:layout_marginTop="20dp"
        android:id="@+id/upi_id"
        android:text="deepasrees2000@okaxis"/>
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Amount"
        android:layout_marginLeft="20dp"
        android:layout_marginRight="20dp"
```

Name
deepasrees2000@okaxis
100
BUS BOOKING

IDE and Plugin Updates: Android Studio is ready to update. (6 minutes ago)

Fig 6.2.8.
Payment Screen Views-
1.Image View
2.Edit Text
3.Button

7.FUTURE ASPECTS AND CONCLUSION

7.1. FUTURE ASPECTS

- Scheduled Bus booking which opens at a particular time of the day.
- Monthly/Yearly payment of bills on a POSTPAID basis.
- Sending Notifications to users on a regular basis.
- Sort Driver list in Student Screen based on ETA.
- Store Emergency Contact Details of students for Emergency Purposes.

7.2. CONCLUSION

For College students, this is a very beneficial application as students need not pay bus fees early during the year and not use bus services on a regular basis. Instead whenever the student is in need of a transport service, he may use the bus service. This lowers the burden faced by students due to their financial expenses. The college department can also use the app to make route timing changes based on their discretion as every update is made real-time and students have access to live location of driver. Student also gets to see traffic conditions and ETA of driver.

Hence this proposed solution is a basic necessity for every college student to travel and the college bus department to work smoothly.

8.BIBLIOGRAPHY

- https://firebase.google.com/docs/database/
- https://developer.android.com/
- https://www.blueappsoftware.com/google-pay-integration-android/
- https://www.androidhive.info/2016/01/android-working-with-recycler-view/