A MINI PROJECT REPORT

**On**

**Design and implementation of Android application**

**for easy driving**

**Submitted by**

**Ayaz Khan Honey Vishwakarma**

**Roll No: 161500154 Roll No: 161500243**

**Shivangi Tripathi Zaved Alam**

**Roll No: 161500278 Roll No: 161500646**

**To**

**Mr. Neeraj Khanna**

Department of Computer Engineering & Applications

**Institute of Engineering & Technology**



**GLA University**

**Mathura- 281406, INDIA**

**April, 2019**

**Department of Computer Engineering & Applications**



**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha, Mathura – 281406**

***Declaration***

*We hereby declare that the work which is being presented in the Mini Project “****Design and implementation of Android application for easy driving”,*** *in partial fulfillment of the requirements for Mini-Project LAB, is an authentic record of our own work carried under the supervision of* ***Mr. Neeraj Khanna, Technical Trainer, GLA University, Mathura****.*

**Name of Students with signature**

**1. Ayaz Khan**

**2. Honey Vishwakarma**

**3. Shivangi Tripathi**

**4. Zaved Alam**



**Department of Computer Engineering & Applications**

**GLA University, Mathura**

**17 km. Stone NH#2, Mathura-Delhi Road, P.O. – Chaumuha,**

**Mathura – 281406**

**CERTIFICATE**

*This is to certify that the project entitled* ***“Design and implementation of Android application for easy driving”*** *carried out in Mini Project Lab is a bon-a-fide work done by* ***Ayaz Khan (161500154), Honey Vishwakarma (161500243), Shivangi Tripathi (161500278) and Zaved Alam (161500646)*** *and is submitted in partial fulfillment of the requirements for the award of the degree Bachelor of Technology (Computer Science & Engineering).*

**Signature of Supervisor:**

**Name of Supervisor:**

**Date:**

**ACKNOWLEDGEMENT**

*It gives us a great sense of pleasure to present the report of the B. Tech Mini Project undertaken during B. Tech. Third Year. This project in itself is an acknowledgement to the inspiration, drive and technical assistance contributed to it by many individuals. This project would never have seen the light of the day without the help and guidance that we have received.*

*Our heartiest thanks to* ***Dr. (Prof). Anand Singh Jalal,*** *Head of Dept., Department of CEA for providing us with an encouraging platform to develop this project, which thus helped us in shaping our abilities towards a constructive goal.*

*We owe special debt of gratitude to* ***Mr. Neeraj Khanna,*** *Assistant Professor Department of CEA, for his constant support and guidance throughout the course of our work. His sincerity, thoroughness and perseverance have been a constant source of inspiration for us. He has showered us with all his extensively experienced ideas and insightful comments at virtually all stages of the project & has also taught us about the latest industry-oriented technologies.*

*We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind guidance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.*

Zaved Alam

Honey Vishwakarma

Shivangi Tripathi

Ayaz Khan

**Abstract**

Driving a vehicle is a very careful task. It requires a level of expertise and license along with the documents. As a vehicle owner, you are responsible for keeping your vehicle in roadworthy condition as well as presenting it for roadworthiness testing. It requires documents/ certificate for proof.

Carrying all the documents and certificates daily is challenging. Sometimes people lose or forget them to carry. Suppose the vehicle is stolen, we lose our documents too as we tend to keep the documents inside the vehicle.

We will use agile methodology. Agile mobile app development methodology is one of the most effective approaches to all the software development businesses, it ensures a proper channel of communication, which helps both the clients and App Developers execute the desired mobile application or in fact any software. The characteristics of Agile Methodology makes an easy job for mobile app development so that the mobile application outcome is adaptable after its release.

We need to develop two interfaces. An android application for the users and a web portal for RTO. The android application will require its users to sign up with the registered mobile number. An OTP will be generated and the person will be able to register with the number and generate a password. Portal of the RTO will be used to upload and maintain the required details.

**Table of Contents**

|  |  |
| --- | --- |
| Declaration | 2 |
| Certificate | 3 |
| Acknowledgments | 4 |
| Abstract | 5 |
| Table of Contents | 6 |
| 1. **Introduction** (This chapter must describe introduction about your project) | **7** |
| 1.1 Motivation and Overview ………………………………........ | 7 |
| 1.3 Objective .………………………………………………….... | 7 |
| 1. **Software Requirement Analysis** | **8** |
| 2.1 Define the problem ………………………………………………….. | 8 |
| 2.2 Define the modules and their functionalities (SRS) ……………..………………………………… | 8 |
| 1. **Software Design** | **10** |
| 3.1 UML Diagram | 10 |
| * 1. Database Diagram | 14 |
| 1. **Implementation and User Interface** | **16** |
| 4.1 Interface Screen | 16 |
| 4.2 Database Screen | 21 |
| **5.References/Bibliography** | **22** |
| 1. **Appendices**   6.1 Coding/Code Templates | **23** |

**1. Introduction**

**Motivation**

Obtaining a lost document requires a lot of time and money. People also face registering complaints of a lost vehicle as they would have also lost their documents which proves their point. We have come up with the idea of developing a solution to this problem. To address the discussed issues, we propose a solution by which all the documents would be kept in a mobile phone application.

We won’t need to worry about carrying and thus losing our documents. We could also save our time and money to obtain a new document.

**Objective**

To develop an android application for the necessary documents requires. By the help of application we won’t need to worry about carrying and thus losing our documents. We could also save our time and money to obtain a new document.

**2. SOFTWARE REQUIREMENT ANALYSIS**

**2.1 Define the problem**

Driving a vehicle is a very careful task. It requires a level of expertise and license along with the documents. As a vehicle owner, you are responsible for keeping your vehicle in roadworthy condition as well as presenting it for roadworthiness testing. It requires documents/ certificate for proof.

Carrying all the documents and certificates daily is challenging. Sometimes people lose or forget them to carry. Suppose the vehicle is stolen, we lose our documents too as we tend to keep the documents inside the vehicle.

In India around 2 million people lose their driving license or documents every year. However, the figures also show that your age and where you live could decide where you keep your driving license. Younger drivers are far more likely to choose to carry their license with them. The older a driver gets the more likely they are to keep their license at home. Approximately 87% young drivers do not carry their License.

This is helpful for people as well as RTO and Police stations investigating a lost vehicle. The roadworthiness certificates are easy to check as it would be readily available.

**2.2 Define the modules and their functionalities (SRS)**

**Layout:**

It contains all the xml files for each activity in the android application.

* activity\_main.xml
* activity\_ownerlogin.xml
* activity\_owpage1.xml
* activity\_rtohome.xml
* activity\_rtologin.xml
* activity\_rtosearch.xml
* content\_main.xml

**Java files:**

These are the backend files for the development of the application.

* HomeActivity.java
* ImageAdapter.java
* ImagesActivity.java
* MainActivity.java
* Upload.java
* ownerlogin.java
* owpage1.java
* owpage11.java
* rtohome.java
* rtologin.java
* rtosearch.java
* rtouploadmain.java

**3. SOFTWARE DESIGN**

**3.1 UML Diagrams**

**3.1.1 Sequence Diagram**

RTO Login

System

Database

Enter ID and Password

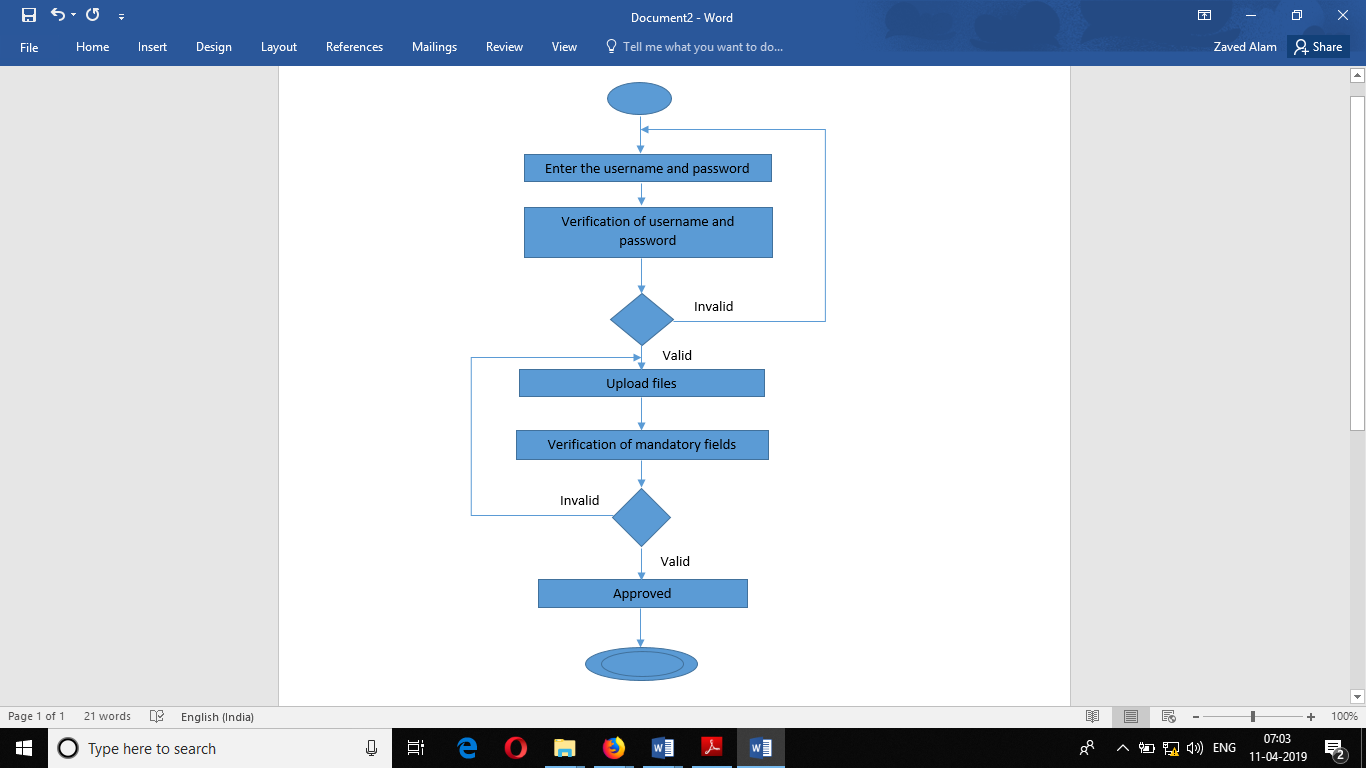
Verify ID and Password

Display Admin Page

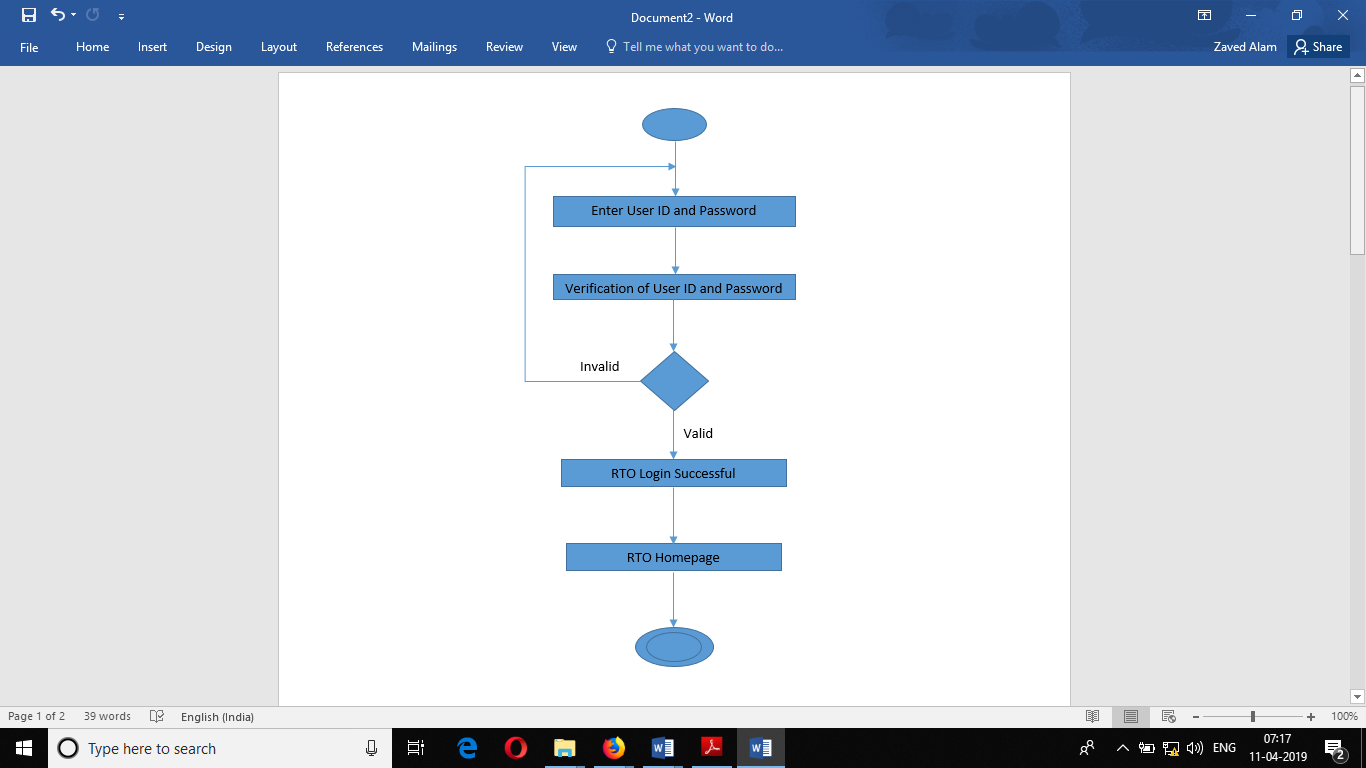
Message 1

Fig 3.4: Sequence Diagram For RTO Login

**3.1.2 Object Diagram**

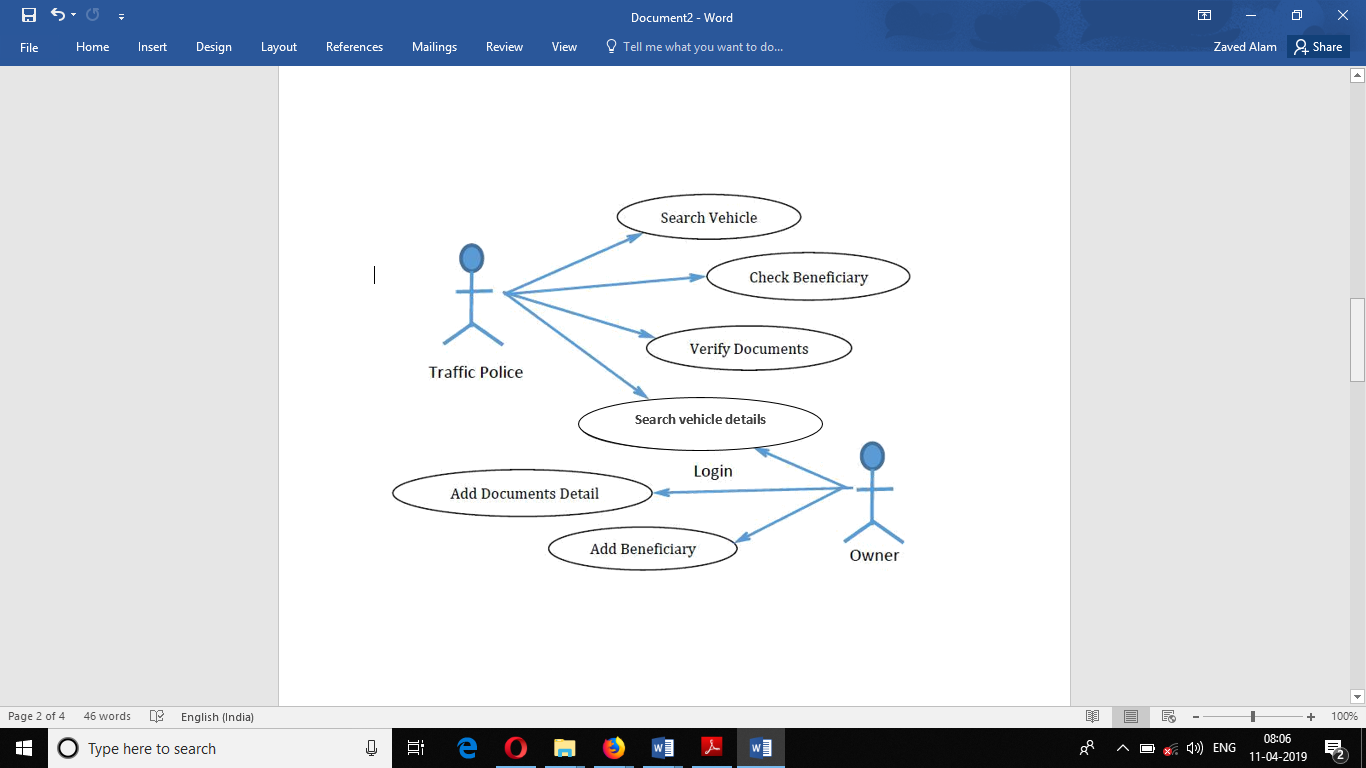
****

**Fig. Object diagram of RTO upload**

****

**Fig. Object diagram of RTO login**

**3.1.3 Use case diagram**



**Fig. Use Case diagram**

**3.2 Database design**

**3.2.1 Tables:**

report

Owner\_detail

RTO\_detail

report

comments

vehicle\_no

vehicle\_doc

login\_id

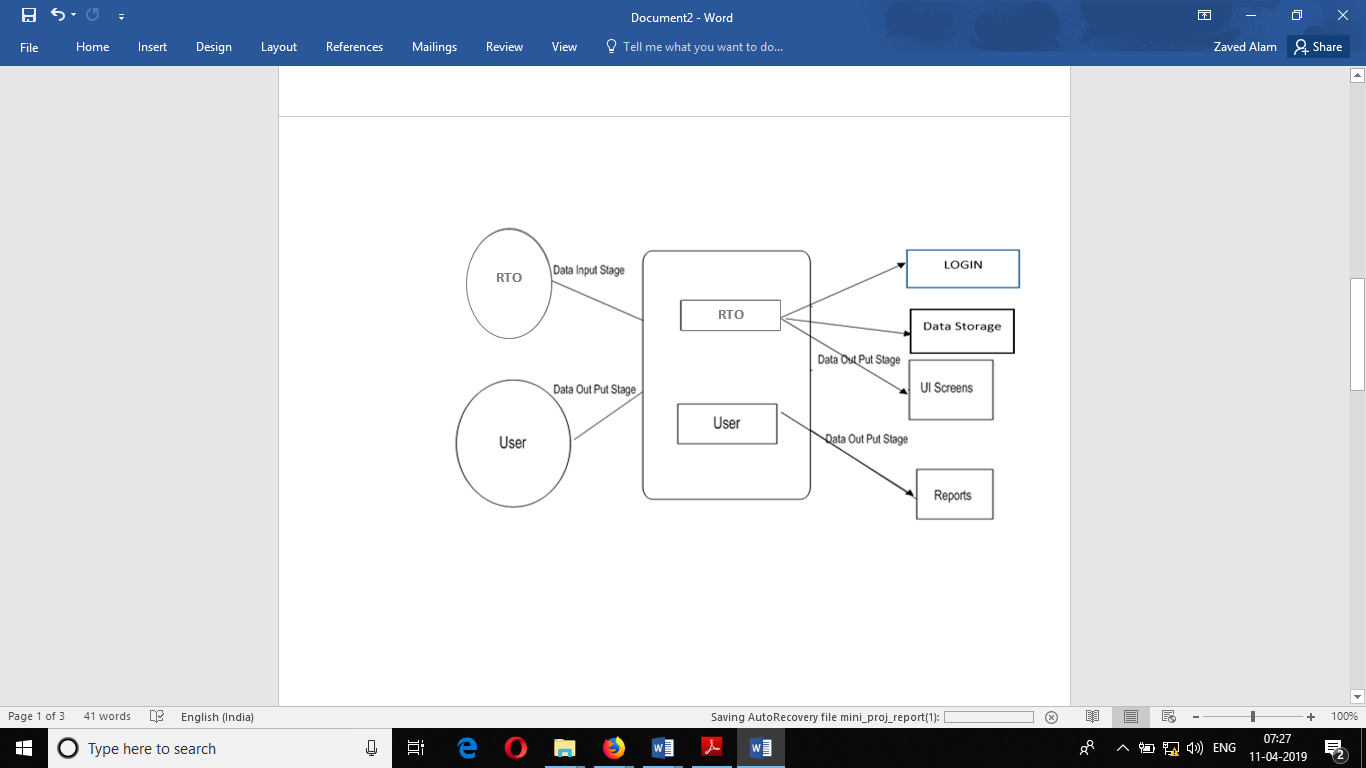
password

login\_id

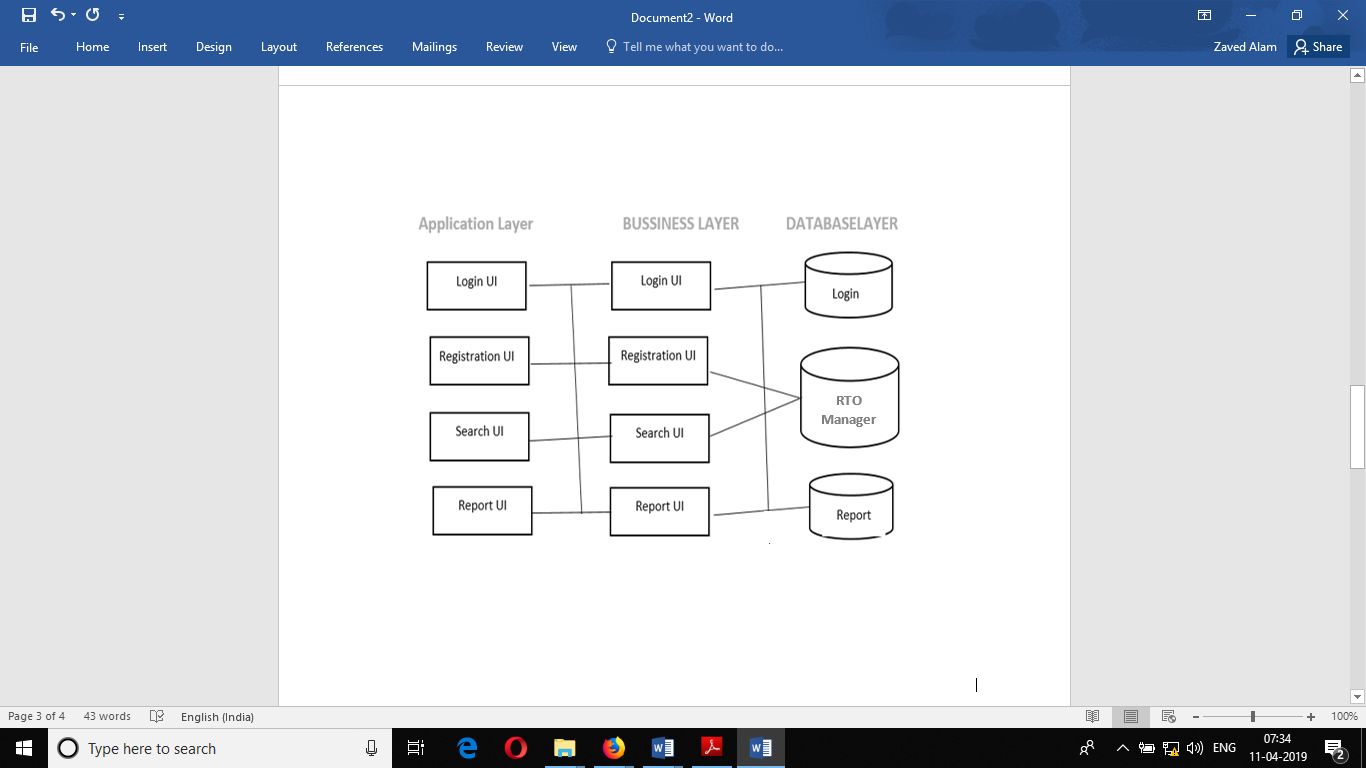
password

comments

**3.2.2 ER Diagram:**

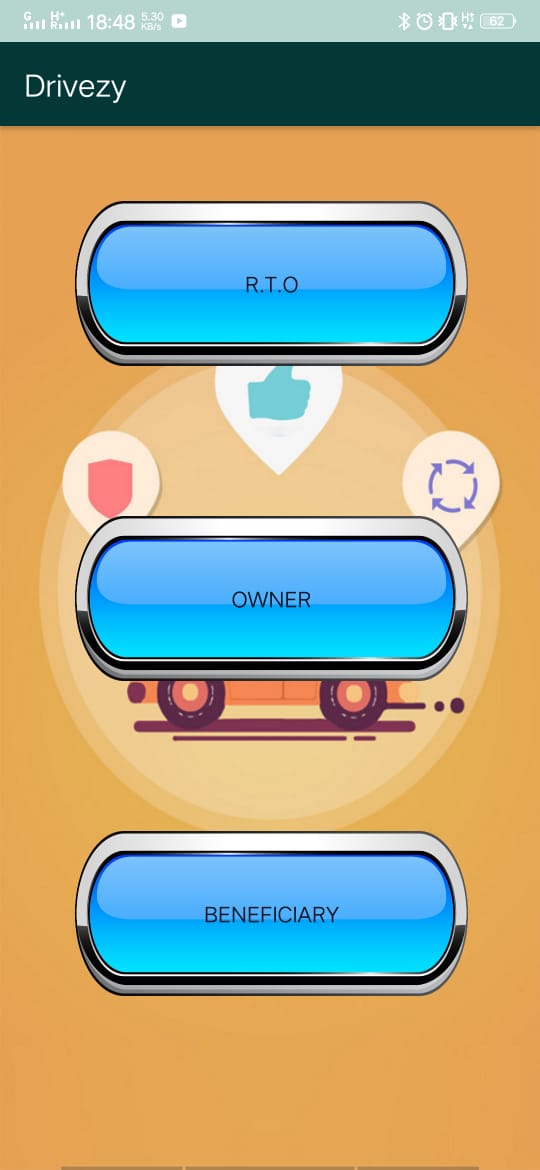
****

* + 1. **Stored Procedure:**

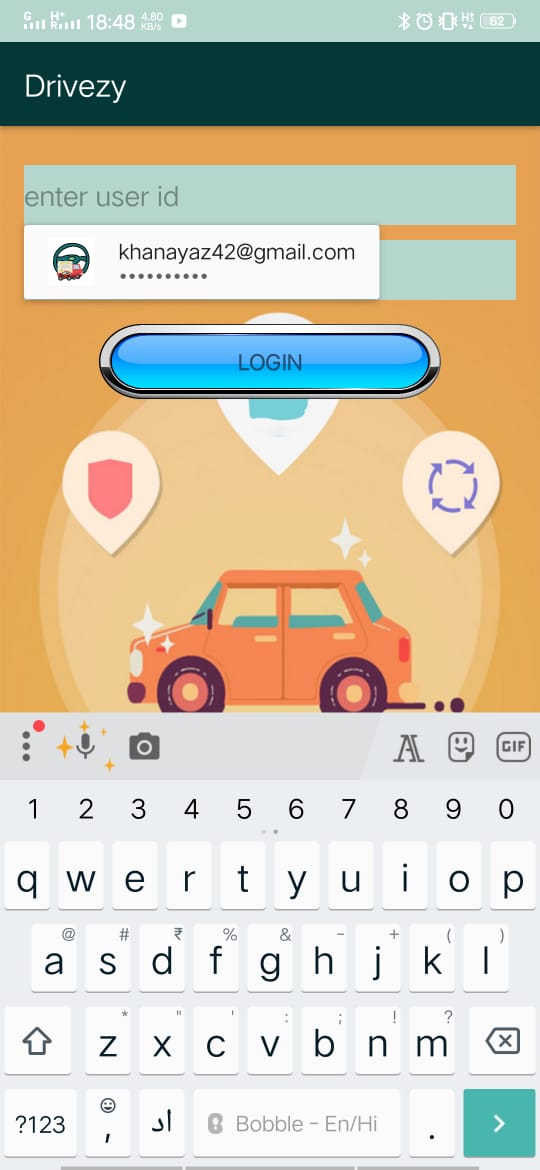


**4. Implementation and User Interface**

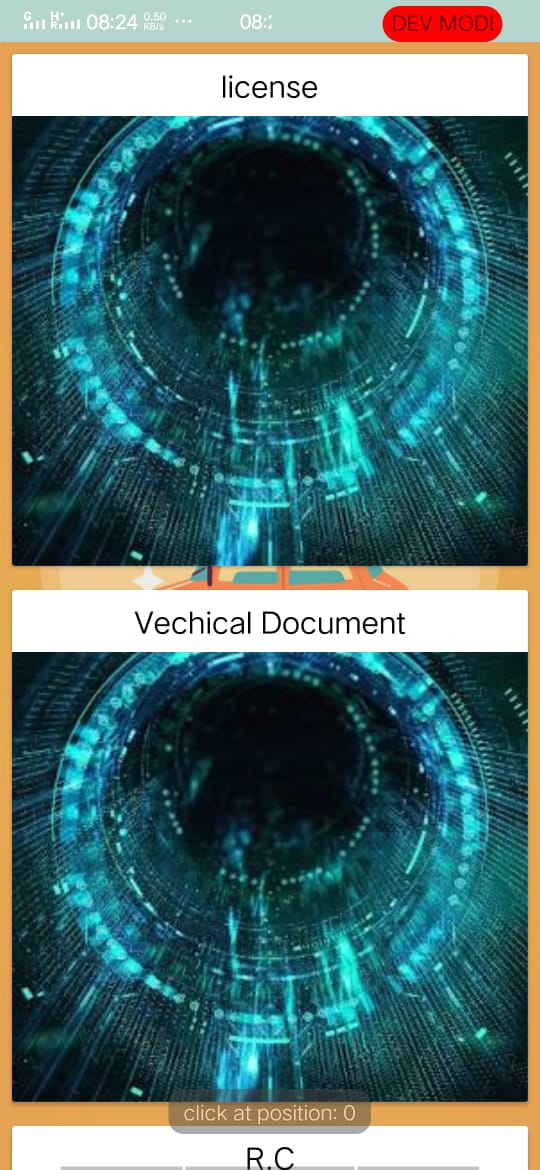
**4.1 Interface Screen**

****

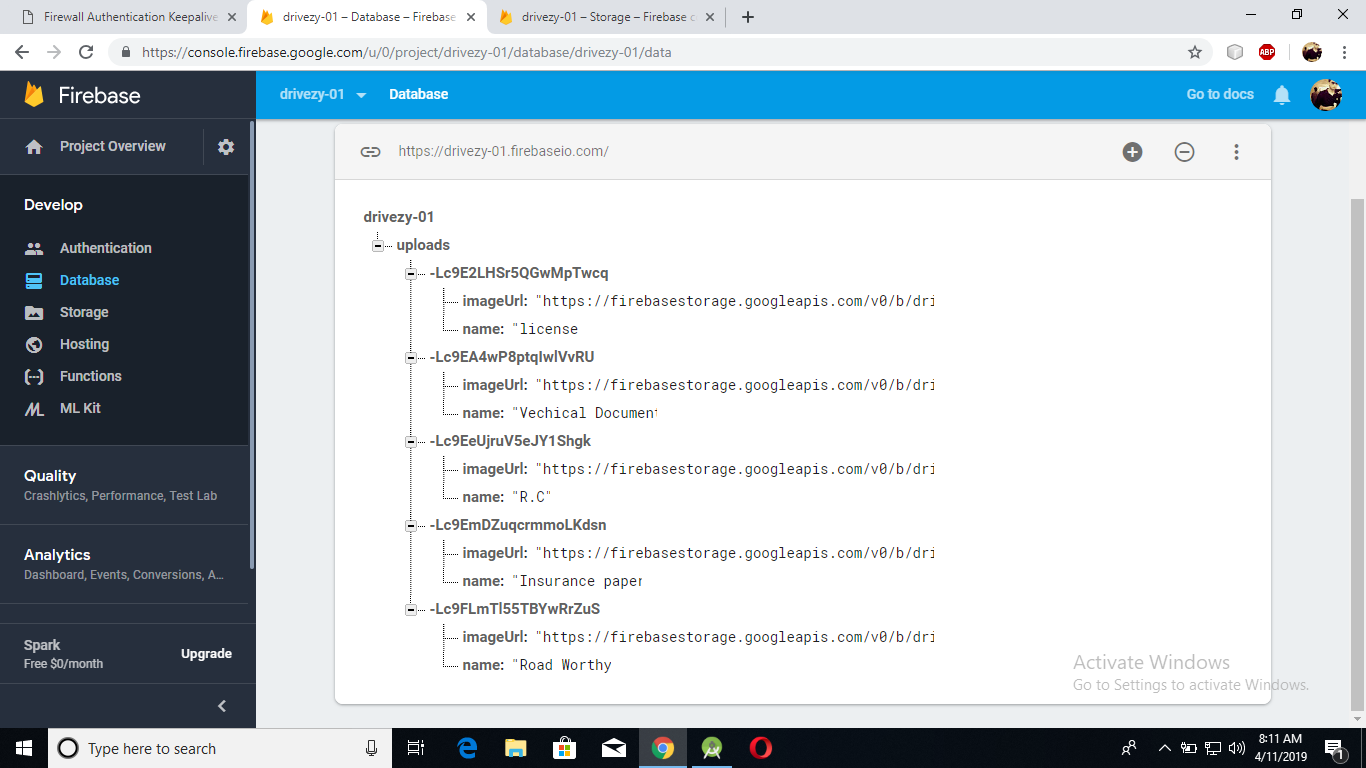
****

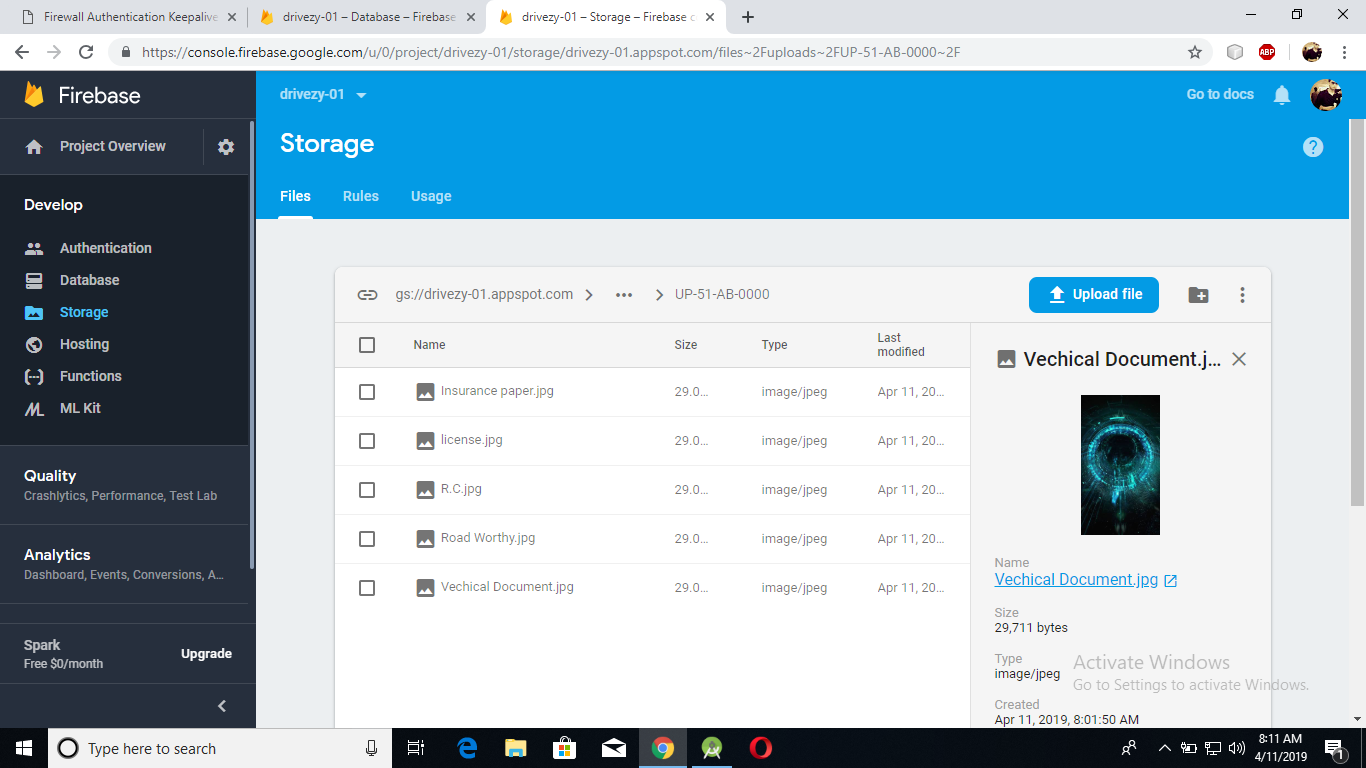
****

****

****

**4.2 Database Screen**

****

****

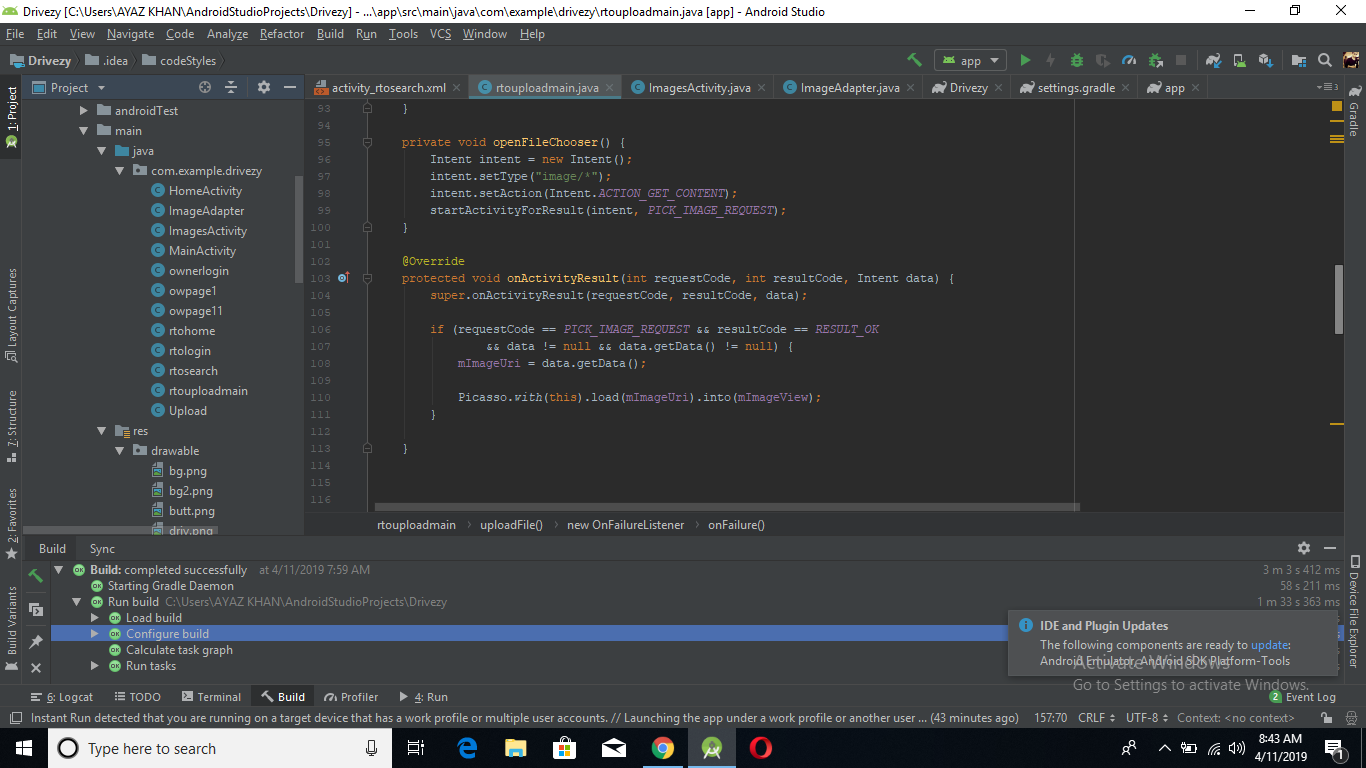
**5. References**

* Android System Programming –Roger Ye
* ANDROID: [www.W3school.com](http://www.w3school.com)
* ANDROID: [www.Javatpoint.com](http://www.javatpoint.com)
* Udemy Android Course
* Wikipedia: [www.wikipedia.com](http://www.wikipedia.com)

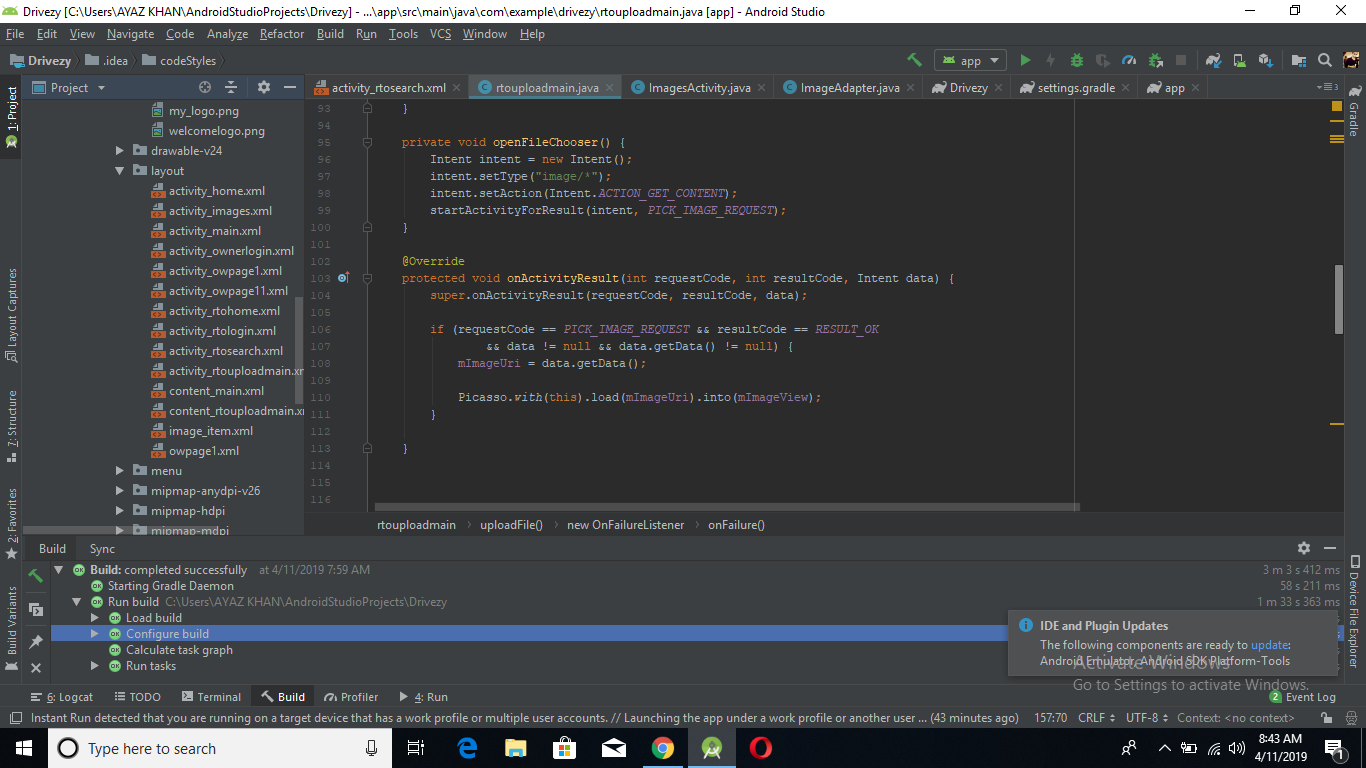
**6. Appendices**

**6.1 Coding/ Code Templates**

**Java activities:**

****

**XML activities:**

****

**Upload image:**

private void uploadFile() {  
 if (mImageUri != null) {  
 StorageReference fileReference = mStorageRef.child(mEditTextFileName1.getText().toString()+"/"+mEditTextFileName.getText().toString()  
 + "." + getFileExtension(mImageUri));  
 // StorageReference fileReference = mStorageRef.child(System.currentTimeMillis()  
 // + "." + getFileExtension(mImageUri));  
  
 mUploadTask = fileReference.putFile(mImageUri)  
 .addOnSuccessListener(new OnSuccessListener<UploadTask.TaskSnapshot>() {  
 @Override  
 public void onSuccess(UploadTask.TaskSnapshot taskSnapshot) {  
 Handler handler = new Handler();  
 handler.postDelayed(new Runnable() {  
 @Override  
 public void run() {  
 mProgressBar.setProgress(0);  
 }  
 }, 500);  
  
 Toast.*makeText*(rtouploadmain.this, "Upload successful", Toast.*LENGTH\_LONG*).show();  
  
 Upload upload = new Upload(mEditTextFileName.getText().toString().trim(),  
 taskSnapshot.getStorage().child(mEditTextFileName1.getText().toString()+"/"+mEditTextFileName.getText().toString()).getDownloadUrl().toString());  
  
 // mStorageRef.child(mEditTextFileName1.getText()+"/"+mEditTextFileName.getText() + "." + getFileExtension(mImageUri)).getDownloadUrl().toString());  
  
 // "<https://firebasestorage.googleapis.com/v0/b/drivezy-01.appspot.com/o/uploads%2Fayaz%20kam.jpg?alt=media&token=8ea5da5c-cdf0-415f-b9c7-466bc713ba58>"  
 String uploadId = mDatabaseRef.push().getKey();  
 //Toast.makeText(rtouploadmain.this, uploadId, Toast.LENGTH\_LONG).show();  
 mDatabaseRef.child(uploadId).setValue(upload);  
 }  
 })  
 .addOnFailureListener(new OnFailureListener() {  
 @Override  
 public void onFailure(@NonNull Exception e) {  
 Toast.*makeText*(rtouploadmain.this, e.getMessage(), Toast.*LENGTH\_SHORT*).show();  
 }  
 })  
 .addOnProgressListener(new OnProgressListener<UploadTask.TaskSnapshot>() {  
 @Override  
 public void onProgress(UploadTask.TaskSnapshot taskSnapshot) {  
 double progress = (100.0 \* taskSnapshot.getBytesTransferred() / taskSnapshot.getTotalByteCount());  
 mProgressBar.setProgress((int) progress);  
 }  
 });  
 } else {  
 Toast.*makeText*(this, "No file selected", Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
  
 private void openImagesActivity() {  
 Intent intent = new Intent(this, ImagesActivity.class);  
 startActivity(intent);  
 }  
}

**Image Retrieval:**

public ImageAdapter(Context context, List<Upload> uploads) {  
 mContext = context;  
 mUploads = uploads;  
}  
  
@Override  
public ImageViewHolder onCreateViewHolder(ViewGroup parent, int viewType) {  
 View v = LayoutInflater.*from*(mContext).inflate(R.layout.*image\_item*, parent, false);  
 return new ImageViewHolder(v);  
}  
  
@Override  
public void onBindViewHolder(ImageViewHolder holder, int position) {  
  
 Upload uploadCurrent = mUploads.get(position);  
 holder.textViewName.setText(uploadCurrent.getName());  
 Toast.*makeText*(mContext, "url:" + uploadCurrent.getImageUrl(), Toast.*LENGTH\_SHORT*).show();  
 Picasso.*with*(mContext)  
 .load(uploadCurrent.getImageUrl())  
 .placeholder(R.mipmap.*ic\_launcher*)  
 .fit()  
 .centerCrop()  
 .into(holder.imageView);  
 // Picasso.with(mContext).load(uploadCurrent.getImageUrl()).into(holder.imageView);  
}  
  
@Override  
public int getItemCount() {  
 return mUploads.size();  
}  
  
public class ImageViewHolder extends RecyclerView.ViewHolder implements View.OnClickListener,  
 View.OnCreateContextMenuListener, MenuItem.OnMenuItemClickListener {  
 public TextView textViewName;  
 public ImageView imageView;  
  
 public ImageViewHolder(View itemView) {  
 super(itemView);  
  
 textViewName = itemView.findViewById(R.id.*text\_view\_name*);  
 imageView = itemView.findViewById(R.id.*image\_view\_upload*);  
  
 itemView.setOnClickListener(this);  
 itemView.setOnCreateContextMenuListener(this);  
 }