CS5551 Lab Assignment-2

Name: Siva Sameer Krishna Yarlagadda

Class ID: 57

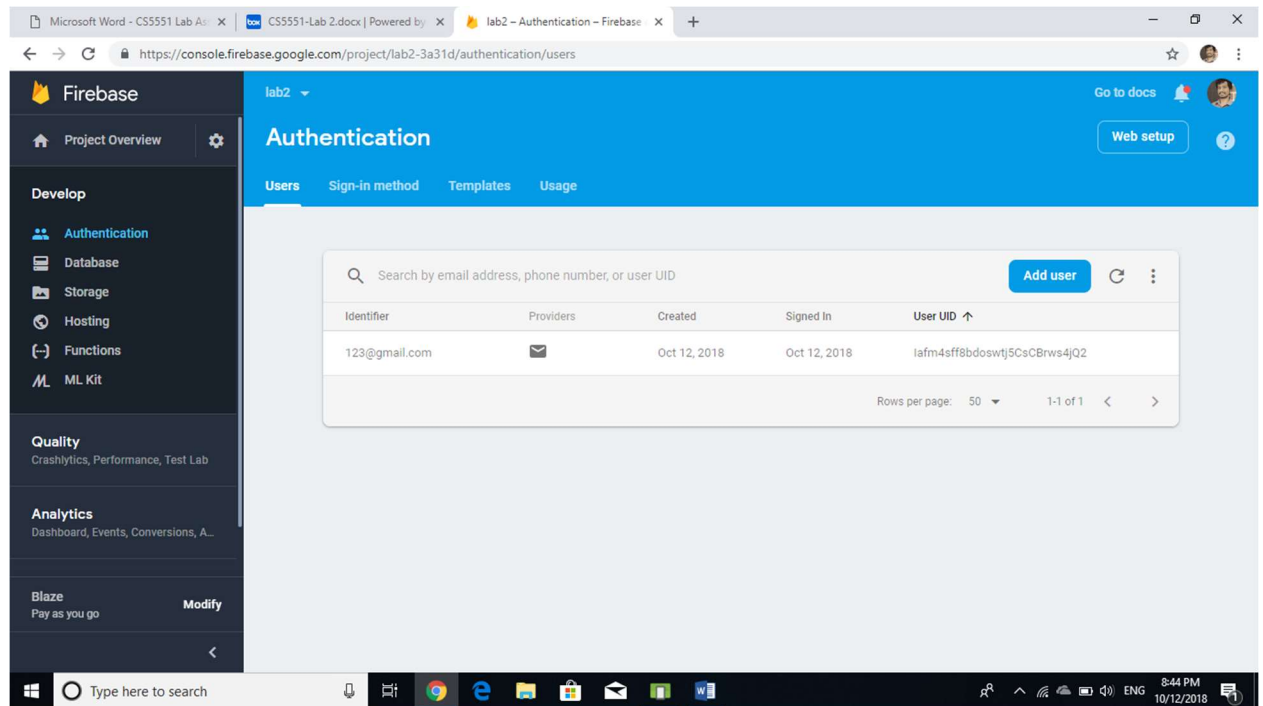**Team member:** Vivek Maliye

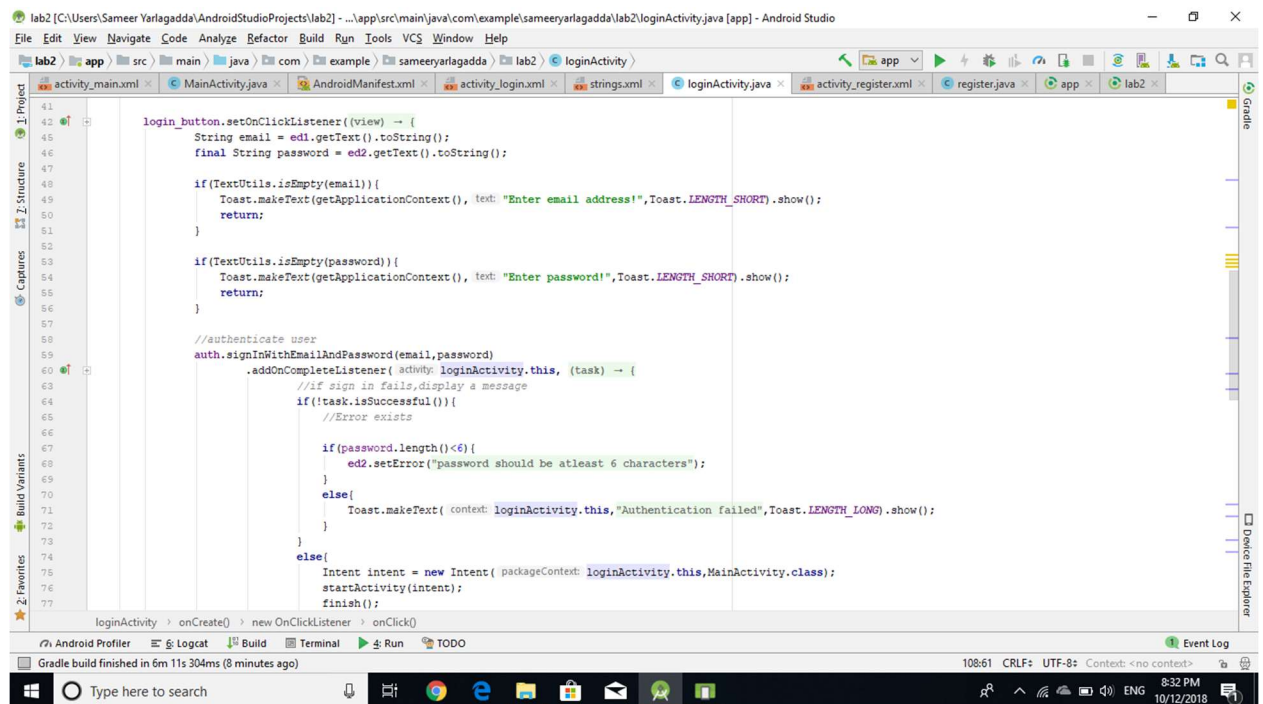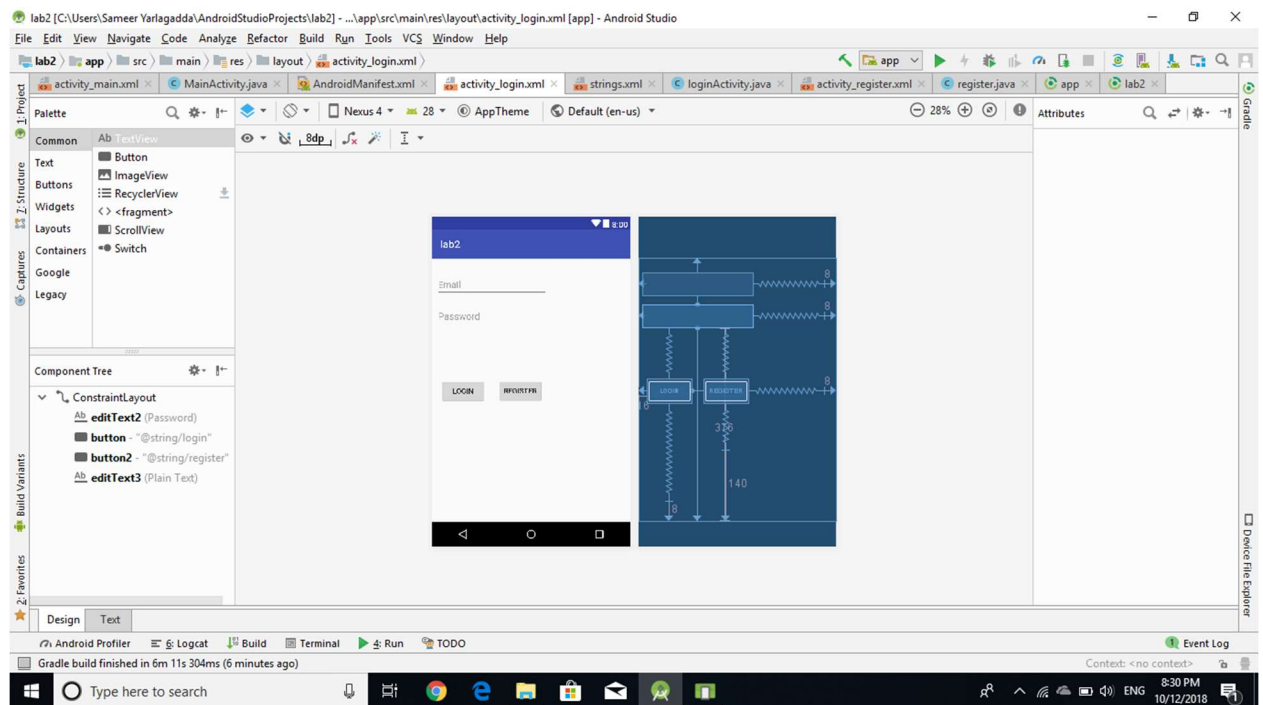**Partners wiki link:**

Class ID: 33

Part-1

Objective: To develop an Android application with the following features
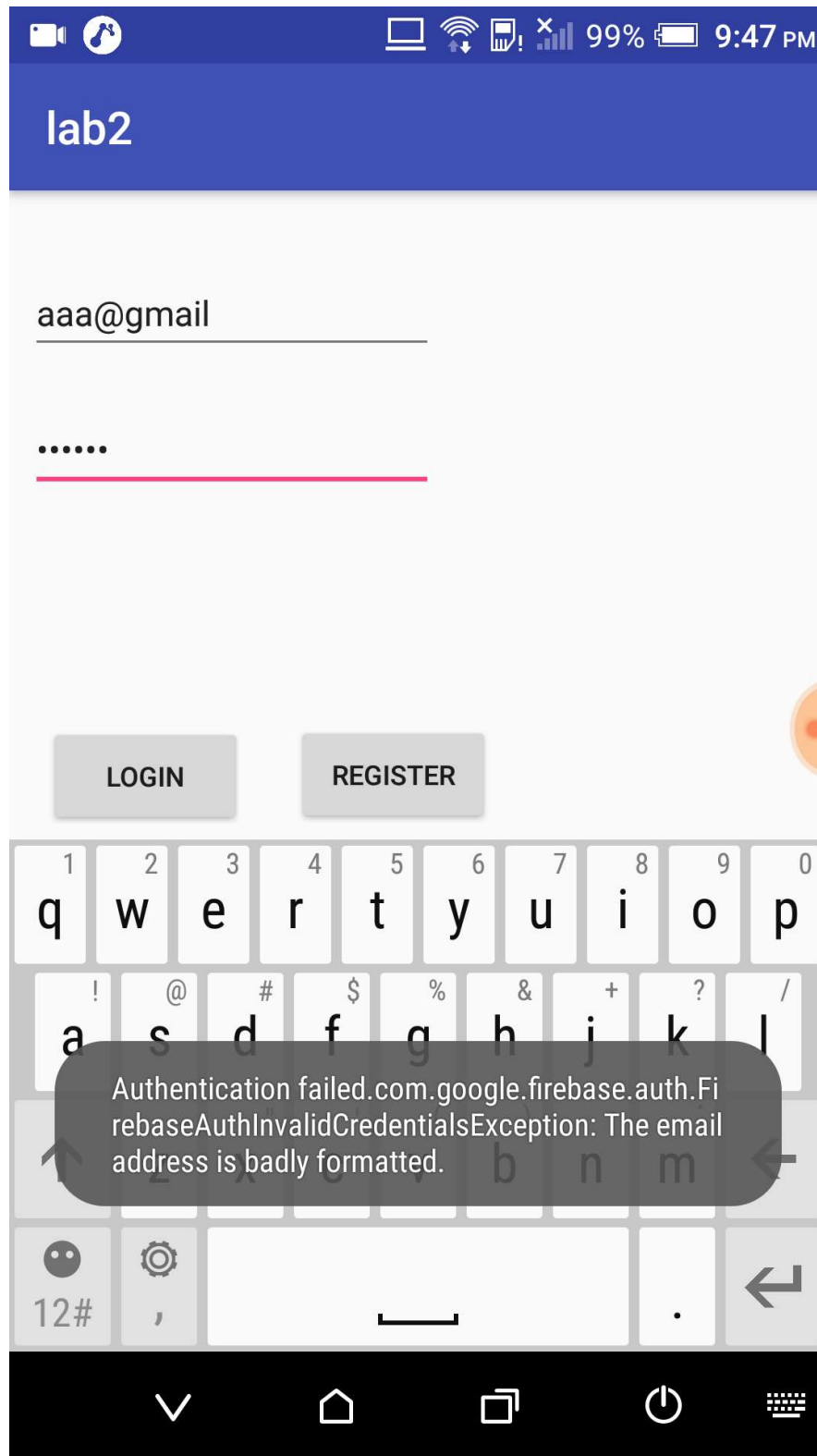
1. **FireBase Login and Registration:**
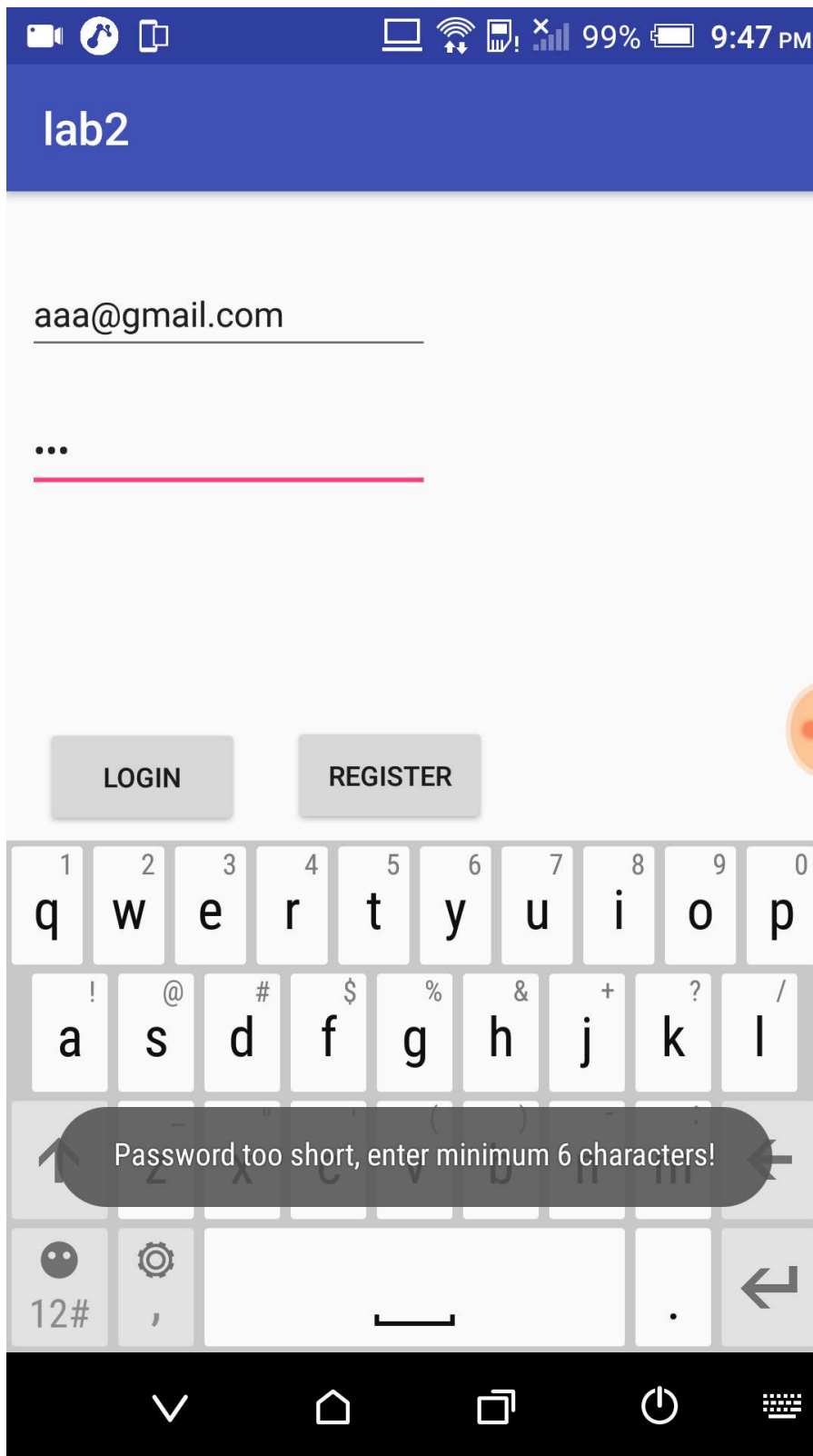   FireBase authentication is used for logging into the application.

Design view of Login page

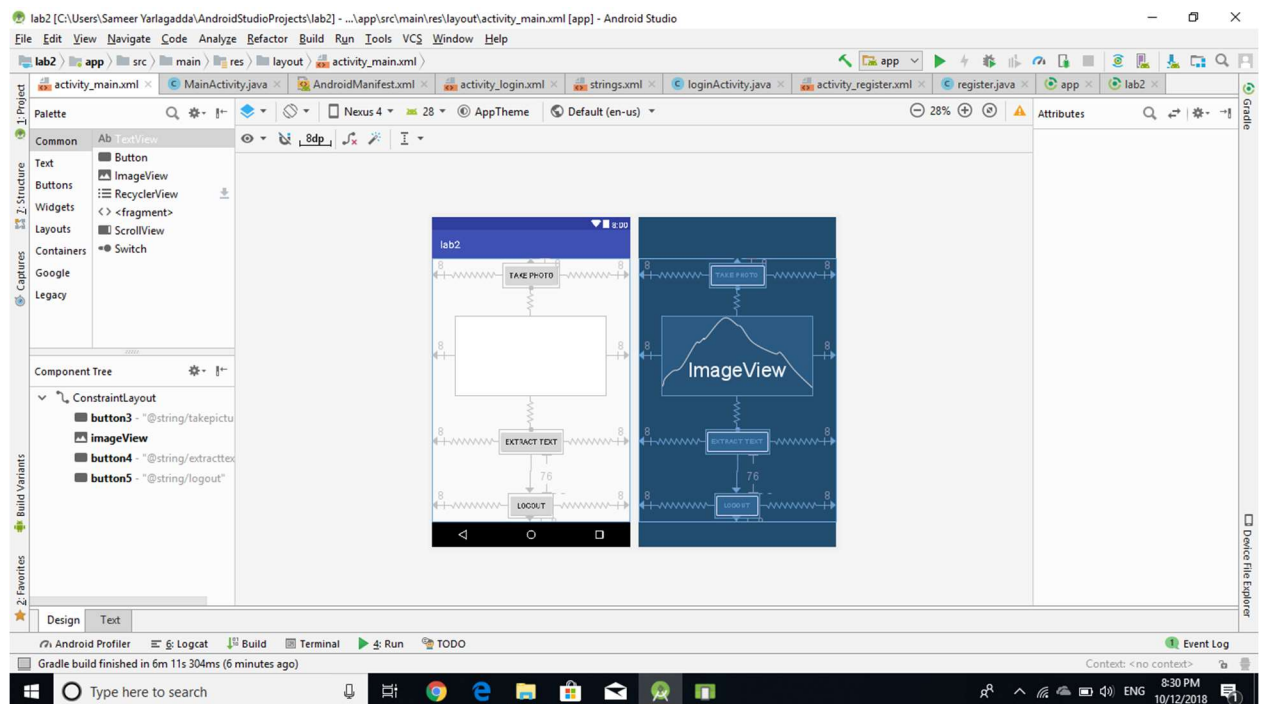During Registration and login, Validation is performed.

After successful registration and login app redirects the user to home page.

2. Home page with Google Cloud Vision API:

Here Google Cloud Vision API is used to extract the textual content from the image.
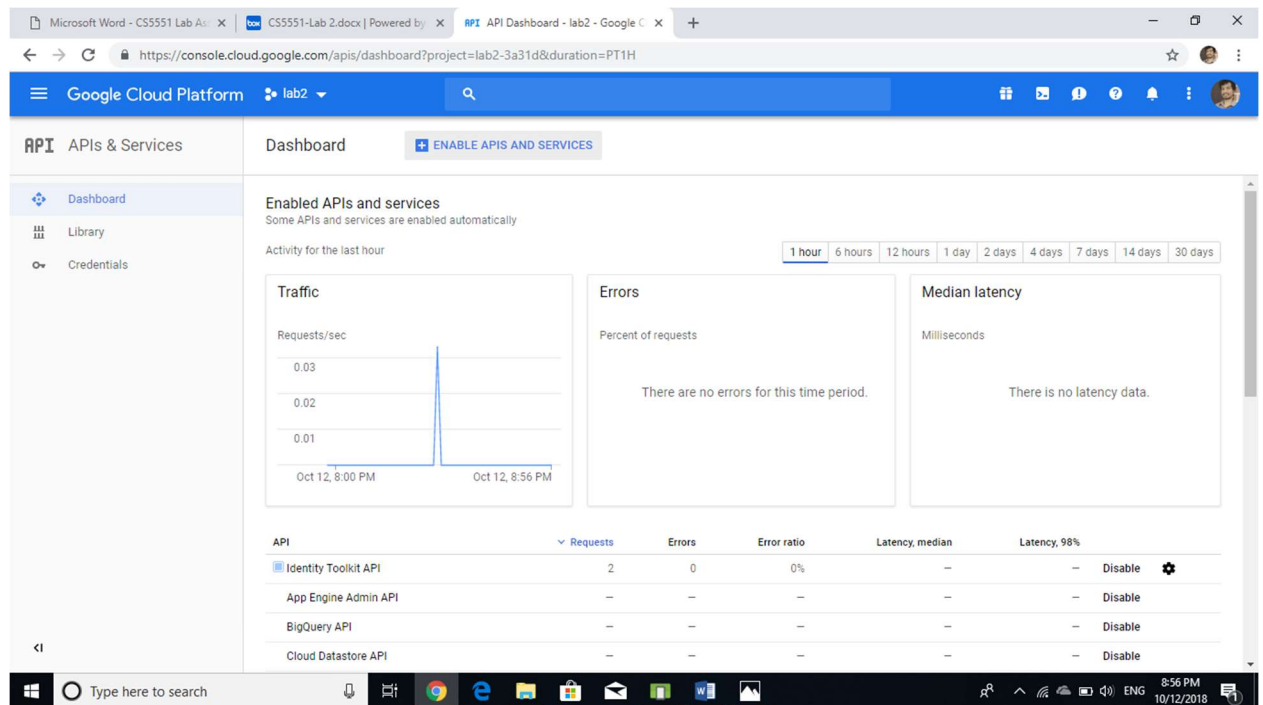
3. Camera (hardware feature):

When user clicks the button to "Take photo", it initially requests the permission for camera to access the application.
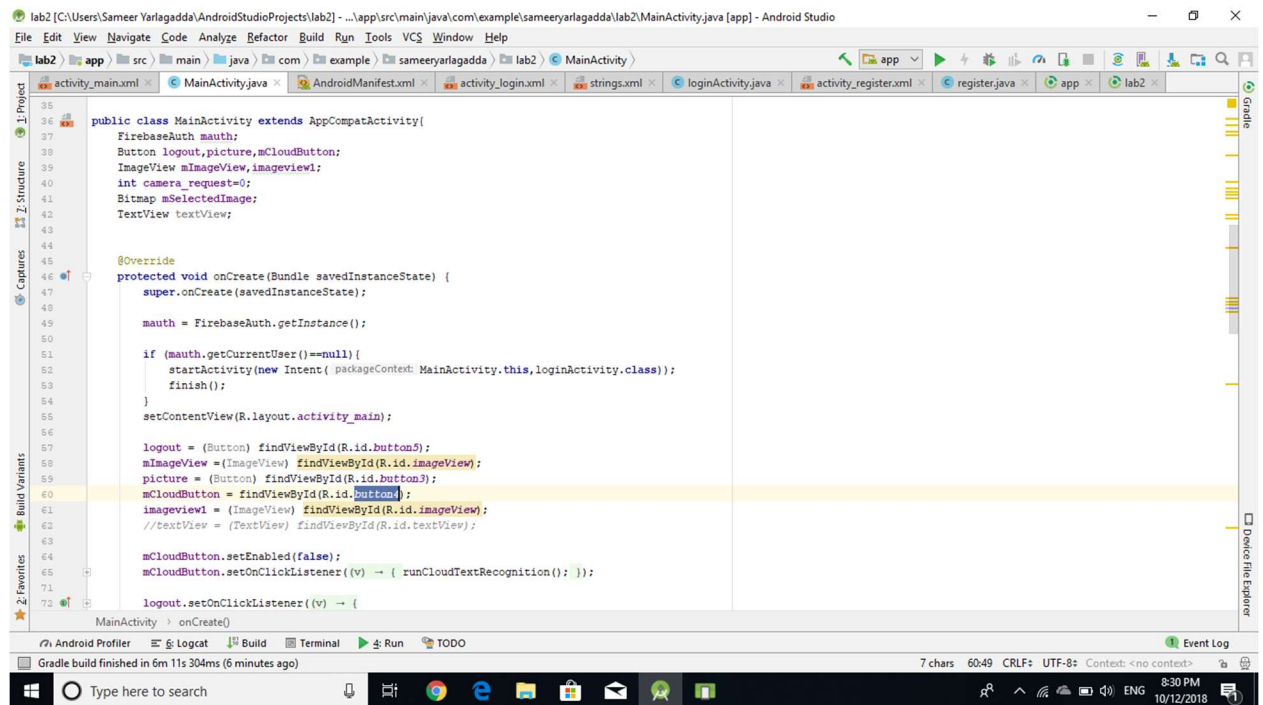
It then take a snap and after clicking the "Extract text" button, API extracts the text in the picture.

"runCloudTextRecognition()" is used for extraction of text from the image.

File Edit View Navigate Code Analyze Refactor Build Run Tools VCS Window Help

lab2 ⟩ app ⟩ src ⟩ main ⟩ java ⟩ com ⟩ example ⟩ sameeryarlagadda ⟩ lab2 ⟩ MainActivity

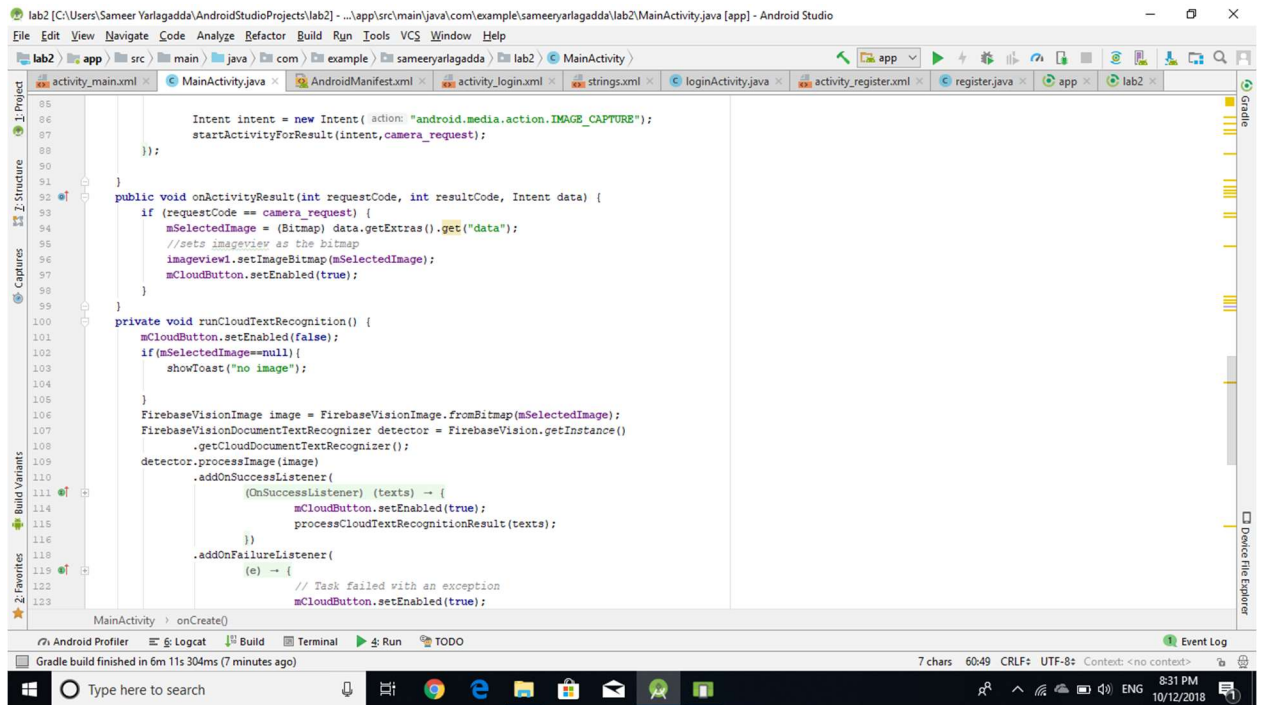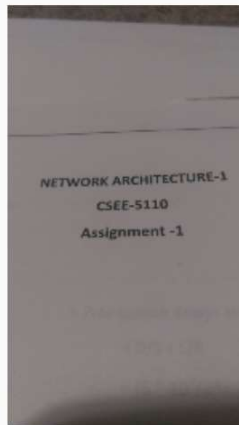activity_main.xml   MainActivity.java   AndroidManifest.xml   activity_login.xml   strings.xml   loginActivity.java   activity_register.xml   register.java   app   lab2

```java
85
86                    Intent intent = new Intent( action: "android.media.action.IMAGE_CAPTURE");
87                    startActivityForResult(intent,camera_request);
88            });
89
90
91        }
92    public void onActivityResult(int requestCode, int resultCode, Intent data) {
93        if (requestCode == camera_request) {
94            mSelectedImage = (Bitmap) data.getExtras().get("data");
95            //sets imageview as the bitmap
96            imageview1.setImageBitmap(mSelectedImage);
97            mCloudButton.setEnabled(true);
98        }
99    }
100    private void runCloudTextRecognition() {
101        mCloudButton.setEnabled(false);
102        if(mSelectedImage==null){
103            showToast("no image");
104
105        }
106        FirebaseVisionImage image = FirebaseVisionImage.fromBitmap(mSelectedImage);
107        FirebaseVisionDocumentTextRecognizer detector = FirebaseVision.getInstance()
108                .getCloudDocumentTextRecognizer();
109        detector.processImage(image)
110                .addOnSuccessListener(
111                        (OnSuccessListener) (texts) -> {
114                            mCloudButton.setEnabled(true);
115                            processCloudTextRecognitionResult(texts);
116                        })
118                .addOnFailureListener(
119                        (e) -> {
122                            // Task failed with an exception
123                            mCloudButton.setEnabled(true);
```

MainActivity ⟩ onCreate()

Android Profiler   6: Logcat   Build   Terminal   4: Run   TODO

Gradle build finished in 6m 11s 304ms (7 minutes ago)   7 chars   60:49   CRLF   UTF-8   Context: <no context>

TAKE PHOTO

EXTRACT TEXT

NETWORK ARCHITECTURE - 1 CSEE - 5110
Assignment - 1

LOGOUT

Part-2

Objective: To develop an Ionic Application with following features.

1. **Mashup application with different web services:**
   After user logs in mashup page is loaded. Two API's Clarifai and Kairos are used for Face and Visual detection in this application.

Lab2

Enter email address

Enter password

Login

Register

```javascript
var loginapp=angular.module("login", ["ionic","ngCordova","firebase"]);

loginapp.controller("login", function($scope, $state, $firebaseAuth,$window) {

    var fbAuth = $firebaseAuth();

    $scope.login = function(username, password) {

        fbAuth.$signInWithEmailAndPassword(username,password).then(function(authData) {

            $window.location.href = "Home.html";
            return false;
        }).catch(function(error) {
            console.error("ERROR: " + error);
        });

    }

    $scope.register = function(username, password,$window) {
        fbAuth.$createUserWithEmailAndPassword(username,password).then(function(userData) {
            return fbAuth.$signInWithEmailAndPassword(username,password);
        }).then(function(authData) {

            $window.location.href ="Home.html";
        }).catch(function(error) {
            console.error("ERROR: " + error);
        });
    }

});

    var exampleApp = angular.module('starter', ['ionic','ngCordova'])

    .run(function($ionicPlatform) {
        $ionicPlatform.ready(function() {
            // Hide the accessory bar by default (remove this to show the accessory bar above the keyboard
```

Window 1 — home.js

```javascript
$scope.upload = function(file) {
    var fname = file.base64;
    var itemList = "";

    const app = new Clarifai.App({apiKey: 'b09f0d3528d445ee8799fd370a631398'});
    app.models.predict(Clarifai.GENERAL_MODEL, {base64: fname}).then(

    function(response)  {
        console.log(response);
        console.log(response.outputs[0].data.concepts[0].name);
        console.log(response.outputs[0].data.concepts.length);

        for (var i = 0; i < 10;i++) {
            var temp = response.outputs[0].data.concepts[i].name;
            itemList = itemList+ temp +"<br/>";
        }

        document.getElementById("notes").innerHTML =itemList;
    }

    );
}]


$scope.faceDetails = function(file1) {

    var face64 = file1.base64;
    var appurl = "https://api.kairos.com/detect?";
    var appid ="1a2e035b";
    var appkey = "f8a08902f1888f67c412fb92f7d21479";
    $http({
        url: appurl,
```

Window 2 — home.js

```javascript
    );
}]


$scope.faceDetails = function(file1) {

    var face64 = file1.base64;
    var appurl = "https://api.kairos.com/detect?";
    var appid ="1a2e035b";
    var appkey = "f8a08902f1888f67c412fb92f7d21479";
    $http({
        url: appurl,
        method: 'POST',
        headers: { app_id: appid,app_key: appkey},
        data: { image:face64 }

    }).success(function(data){
        var age = data.images[0].faces[0].attributes.age;
        var gender = data.images[0].faces[0].attributes.gender.type;
        document.getElementById("facenotes").innerHTML = "Age : " +age+"<br/>"+"Gender : "+gender;
        console.log(data);

    }).error(function(){
        alert("error");
    });

    }

});
```

Ionic Mashup

Facial Recognition

Choose file | No file chosen

Predict Face

Visual Recognition

Choose file | No file chosen

Predict Image

Scan QR | Logout

Ionic Mashup

Facial Recognition

| Choose file | IMG_20181011_021501.jpg |

Predict Face

Age : 26
Gender : M

Visual Recognition

| Choose file | No file chosen |

Predict Image

Scan QR   Logout

Visual Recognition

Choose file   IMG-20181012-WA0003.jpg
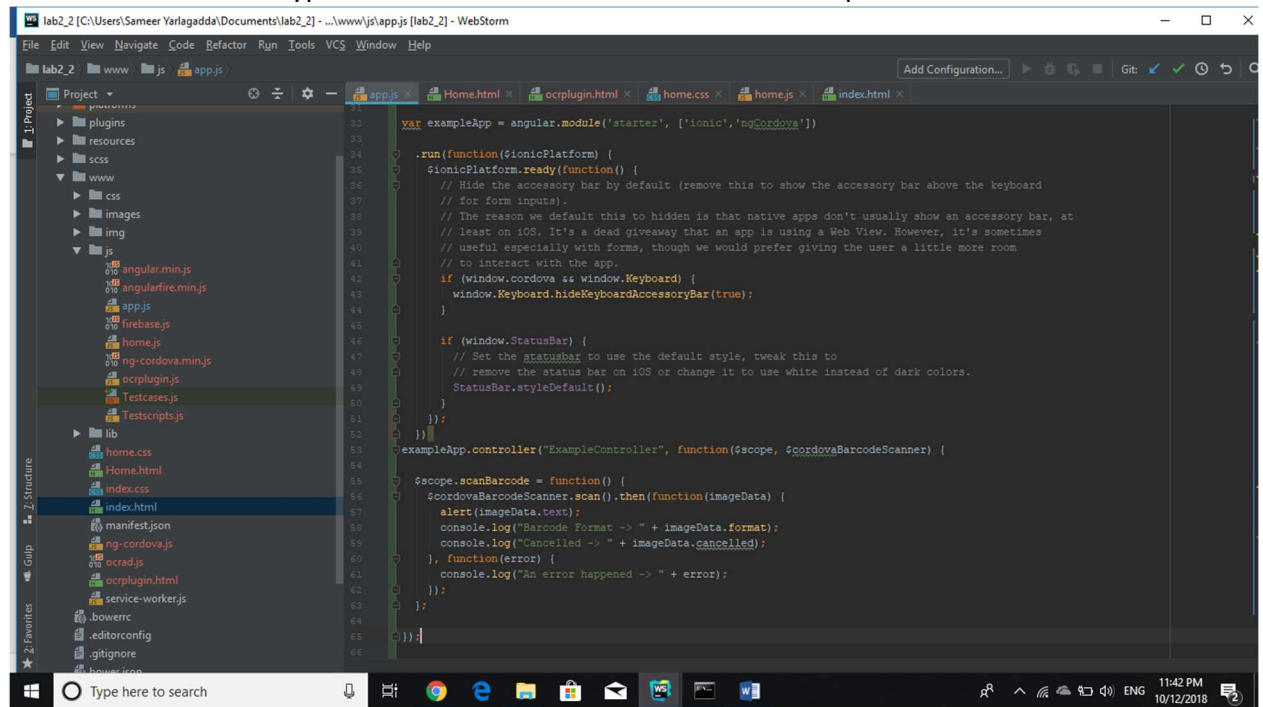
Predict Image

sculpture
art
statue
religion
ancient
old
travel
architecture
no person
Buddha

2. Cordova plugin: Bar code scanner is implemented.
   It scans for different types of codes and returns the information present in it.

QR Detection

scan

Bar code Scanner output:

QR code details:

2594373860856

References:

1. https://console.firebase.google.com
2. https://cloud.google.com/vision/
3. https://clarifai.com/developer/account/applications
4. https://www.kairos.com/docs/api/
5. https://developer.android.com/guide/
6. https://ionicframework.com/docs
7. https://www.thepolyglotdeveloper.com/2014/09/implement-barcode-scanner-using-ionic-framework/