

EXAM APP WITH IMAGE PROCESSING USING ML

Submitted in partial fulfillment of the requirements for
the award of Master of Engineering degree in
Computer Science and Engineering

by

B.SELVAAMBIGAI(40310003)



**DEPARTMENT OF COMPUTER SCIENCE AND
ENGINEERING**

SCHOOL OF COMPUTING

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI - 600 119

MAY - 2022

DECLARATION

I , **B.SELVAAMBIGAI (40310003)** hereby declare that the Project Report entitled “**EXAM APP WITH IMAGE PROCESSING USING ML**”done by me under the guidance of **Dr.S.VIGNESHWARI M.E.,Ph.D.** at **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** is submitted in partial fulfillment of the requirements for the award of **Master ofEngineering degree in Department of Computer Science and Engineering.**

DATE:

PLACE:

SIGNATURE OF THE CANDIDATE



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **B.Selvaambigai(4031003)** who carried out the project entitled **“EXAM APP WITH IMAGE PROCESSING USING ML”** under my supervision from November 2021 to May 2022

Internal guide

Dr.S.VIGNESHWARI M.E.,PhD

Head of department

Dr.S.VIGNESHWARI M.E.,PhD

Submitted for Viva voce Examination held on_____

Internal Examiner

External Examiner

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T.Sasikala M.E., Ph.D, Dean, School of computing and Dr.S.Vigneshwari M.E., Ph.D and Dr.L.Lakshmanan M.E.,Ph.D.,-** Heads of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.S.Vigneshwari M.E., Ph.D** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

TABLE OF CONTENTS

SNO	TOPIC	PAGE
1	INTRODUCTION	1
2	LITERATURE SURVEY	2
3	AIM	7
4	WHAT IS MACHINE LEARNING	17
5	RESULTS	25
6	CONCLUSION	32
	REFERENCES	33
	APPENDIX	34

CHAPTER – 1

1. INTRODUCTION

Image processing is a way to convert an image to a digital aspect and perform certain functions on it, in order to get an enhanced image or extract other useful information from it. It is a type of signal time when the input is an image, such as a video frame or image and output can be an image or features associated with that image. Usually, the Image Processing system includes treating images as two equal symbols while using the set methods used. It is one of the fastest growing technologies today, with its use in various business sectors. Graphic Design forms the core of the research space within the engineering and computer science industry as well.

CHAPTER – 2

2. LITERATURE SURVEY

[1] China has the world's largest Android population with 270 million active users. However, Google Play is only accessible by about 30% of them, and third-party app stores are thus used by 70% of them for daily Android apps (applications) discovery. The trustworthiness of Android app stores in China is still an open question. In this paper, we present a comprehensive study on the trustworthy level of top popular Android app stores in China, by discovering the identicalness and content differences between the APK files hosted in the app stores and the corresponding official APK files.

First, we have selected 25 top apps that have the highest installations in China and have the corresponding official ones downloaded from their official websites as oracle, and have collected total 506 APK files across 21 top popular app stores (20 top third party stores as well as Google Play). Afterwards, APK identical checking and APK difference analysis are conducted against the corresponding official versions. Next, assessment is applied to rank the severity of APK files. All the apps are classified into 3 severity levels, ranging from safe (identical and higher level), warning (lower version or modifications on resource related files) to critical (modifications on permission file and/or application codes).

Finally, the severity levels contribute to the final trustworthy ranking score of the 21 stores. The study indicates that about only 26.09% of level APK files are safe, 37.74% of them are at warning level, and 36.17% of them are surprisingly at critical level. We have also found out that 10 (about 2%) APK files are modified and resigned by unknown third-parties. In addition, the average trustworthy ranking score (47.37 over 100) has also highlighted that the trustworthy level of the Android app stores in China is relatively low.

[2] the improvement of the trustworthiness of android app stores in china

the absence of Google Play has created a booming area for Android app distribution through third-party app stores in China. Since the study showed that the trustworthy level of app stores was fairly low in 2014, much attention should be paid on the changes of the trustworthiness of Android app stores.

we present a method to analyze the changes of trustworthiness of the top popular Android app stores in China. In this method, we evaluate the target app stores by analyzing the sampled apps hosted in them. Further more, we have used this method to track the changes of trustworthy level of Android app stores in China about two years.

The results indicate that the trustworthy level of the top popular Android app stores in China has been improved 24% on average. It can be seen that the positive changes may be related to the development of the China's mobile market, the improvement of Android system, and the introduced policies. Although the trustworthy level of top popular Android app stores in China is still low, it is predicted to be improving in the future.

[3] a forensics method for android devices based on the technique of temporary root

Nowadays, Android devices become more and more popular, which attracts the attention of the forensics workers. As Android devices have a perfect security mechanism, the root privileges of Android devices are required to obtain electronic evidence effectively.

The existing methods of obtaining the root privileges in Android devices mainly based on the third-party tool. Although the methods are widely used, the drawback of this root method based on the third-party tools is uncontrollable. In this article, a temporary root method is presented. It is based on the vulnerabilities of the Android system.

Moreover, this root method is applied to the acquisition of the physical images for Android devices. This root method utilizes an appropriate strategy and the vulnerabilities of the Android system to acquire the root privileges effectively. In the course of practice, this root method has the advantages of completely controllable and convenient.

[4] the improvement of the trustworthiness of android app stores in china

A promising L4-approach known as multipath TCP (MPTCP) has recently been standardized which allows a single transport connection to use multiple paths simultaneously over multiple interfaces. The use of MPTCP in offloading (i.e., multipath offloading) provides the potential to both maintain service connectivity and maximize the usage of the cellular resource.

However, current Long Term Evolution (LTE) Policy and Charging Control (PCC) system lacks the ability to realize the coexistence of a cellular path (i.e., primary path) and an alternative path (i.e., secondary path) belonging to the same application session. As a result, it always determines the guaranteed bit-rate to the user under the assumption that the user has only a single LTE connection, and hence the usage efficiency of the cellular resources is seriously impaired.

This proposes an enhanced PCC system with the consideration of using multipath offloading in cellular/WiFi networks. We propose a transport-based multipath offloading approach in which the LTE PCC system is rendered aware of alternative connection paths and allocates the available network resources accordingly.

[4] an inter-store transaction mechanism to distribute mobile application

Android is one of the popular OSs and application platforms for mobile devices. There are numerous appstores in the world, including Google Play and Amazon Appstore for Android. These appstores with DRM-protection and In-App Purchasing APIs provide various business models to their app developers. However, the APIs and implementations are very different among the appstores. Therefore, accommodating the mobile apps to other appstores may require extra maintenance and porting overheads.

This proposes a mechanism that supports inter-store transactions to distribute appstore-dependent apps by slightly modifying the appstore client and service logic. Consequently, the app developers can publish their DRM-protected apps and purchasable items in the apps at once to wide markets without rewriting the programs.

CHAPTER – 3

3. AIM

In this project we are going to implement the concept of image processing for validating the certificate

Objective

The project we are going to implement the concept of image processing for validating the certificate Machine learning techniques are increasingly popular in research fields like bio- and chemo-informatics, text and web mining, as well as many other areas of research and industry. In this project we present Java-ML: a cross-platform, open source machine learning library written in Java Image processing is a way to convert an image to a digital aspect and perform certain functions on it, in order to get an enhanced image or extract other useful information from it. It is a type of signal time when the input is an image, such as a video frame or image and output can be an image or features associated with that image. Usually, the Image Processing system includes treating images as two equal symbols while using the set methods used.

Scope of the present investigation

As a response, erasure coding as an alternative to backup has emerged as a method of protecting against drive failure .Raid just does not cut it in the age of high-capacity HDDs. The larger a disk's capacity, the greater the chance of bit error and when a disk fails, the Raid rebuild process begins, during which there is no protection against a second (or third) mechanism failure. So not only has the risk of failure during normal operation grown with capacity, it is much higher during Raid rebuild, too. Also, rebuild times were once measured in minutes or hours, but disk transfer rates have not kept pace with the rate of disk capacity expansion, so large Raid rebuilds can now take days or even longer.

EXISTING SYSTEM:

- ✓ In existing system, they are also implement the application. But their no ml techniques for validation.
- ✓ App contains about classes and exams schedule.

Disadvantages:

- ✓ It contains only exam schedules, we can't able to make exam in that app.
- ✓ Less reliability and security

PROPOSED SYSTEM:

- ✓ In our proposed system we are introducing the concept of image processing for certificate verification.
- ✓ In our app connect with MySQL database for stores the details which has collect from users. Also we can able retrieve that information from MySQL.
- ✓ Exam options are there in app those who have eligibility they can able to attend the exams in our app. After finishing the exams score will be generated automatically.

Advantages:

- ✓ Improves reliability in our app.
- ✓ Introduce the image processing concept for validation.
- ✓ User can able to write over the exam in this application.

SYSTEM REQUIREMENTS:

HARDWARE REQUIREMENTS:

- System : Dual Core
- Hard Disk : 160 GB.
- Monitor : 15 VGA Colour.
- Mouse : Logitech.
- Ram : 4 GB

SOFTWARE REQUIREMENTS:

- Operating system : Windows XP/7/8/10
- Coding Language : JAVA
- Front End : NetBeans Software
- Backend : My SQLyog Community 32

METHODOLOGY

SYSTEM MODULES:

- Web service
- Service
- Database
- Image processing

Modules:**Web Service**

Web service is like connecting android application and server. Server should run 24 hours and it has to give all the details to database which data's we are getting from users. Then using SOAP protocol we can connect android application to server. If we are using SOAP protocol, it will collect all the details from android application and it will send to server.

Server:

Server will be running continuously because then only we can access the DB from Android application. Server will receive the request from SOAP protocol and receive the request with help of WSDL (Web Service Description Language). WSDL is help response the client request which is communicate with DB. The server Communicate with DB with the help of JDBC (Java Database Connectivity Driver) connection.

Data Base:

In this module we have implement the MYSQL queries for storing and retrieving the data from DB. DB it will store all the details about app such user, admin, product, location details. All the details will store depends on the table which is allocated

Image processing:

In this module we are implement the ml with the help of OCR technique. With the help of image processing we detect the certificate register number. When we select the register number in that certificate automatically it compares with DB weather they are valid or not valid.

System Architecture:

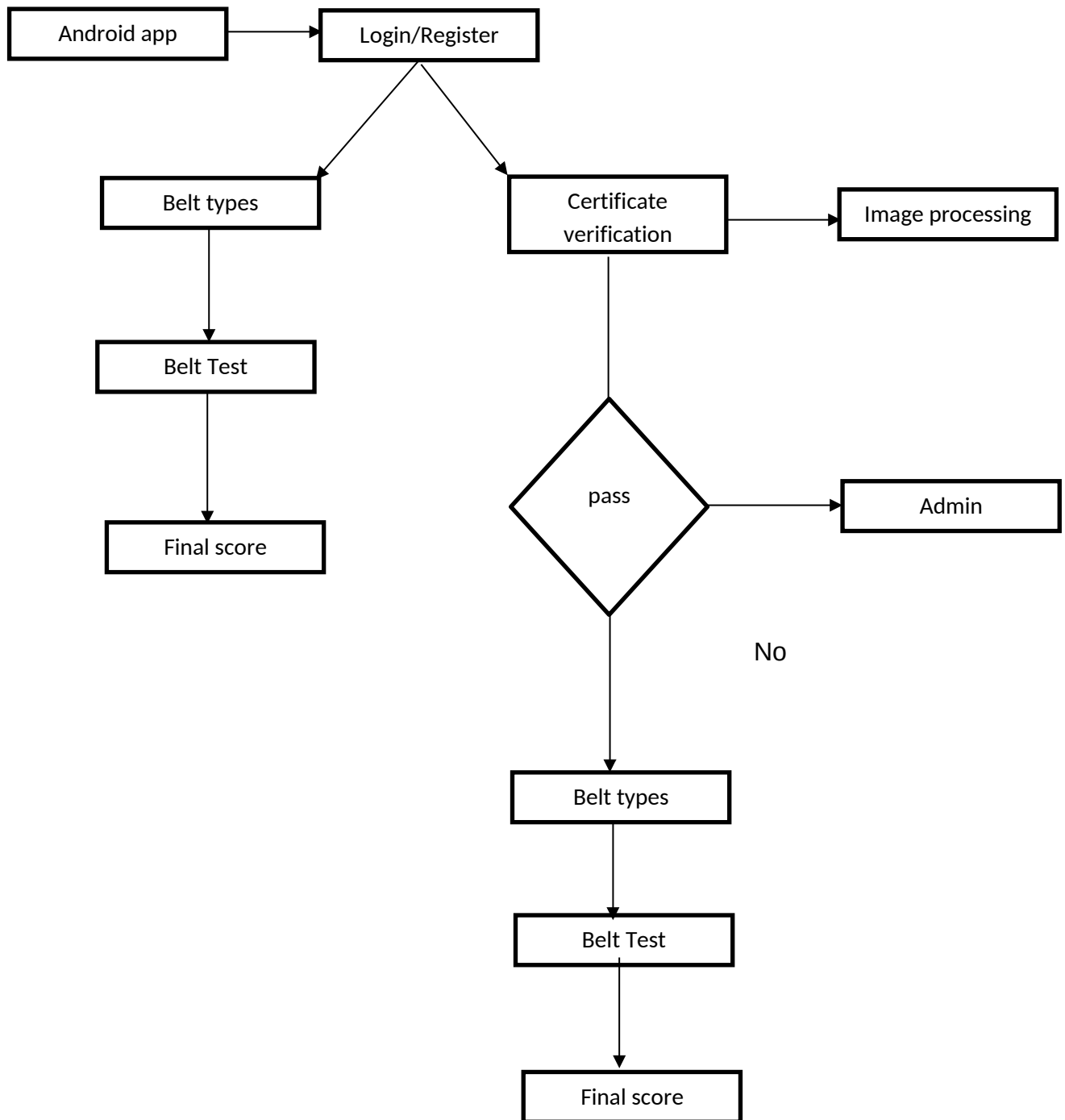
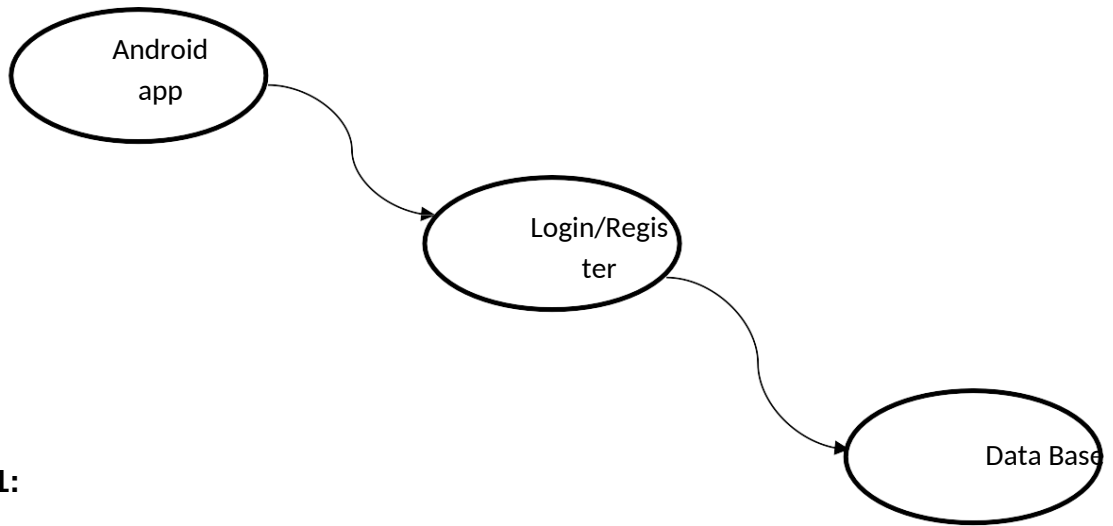


Fig:3.1 System Architecture

Data Flow Diagram:

Level 0:



Level 1:

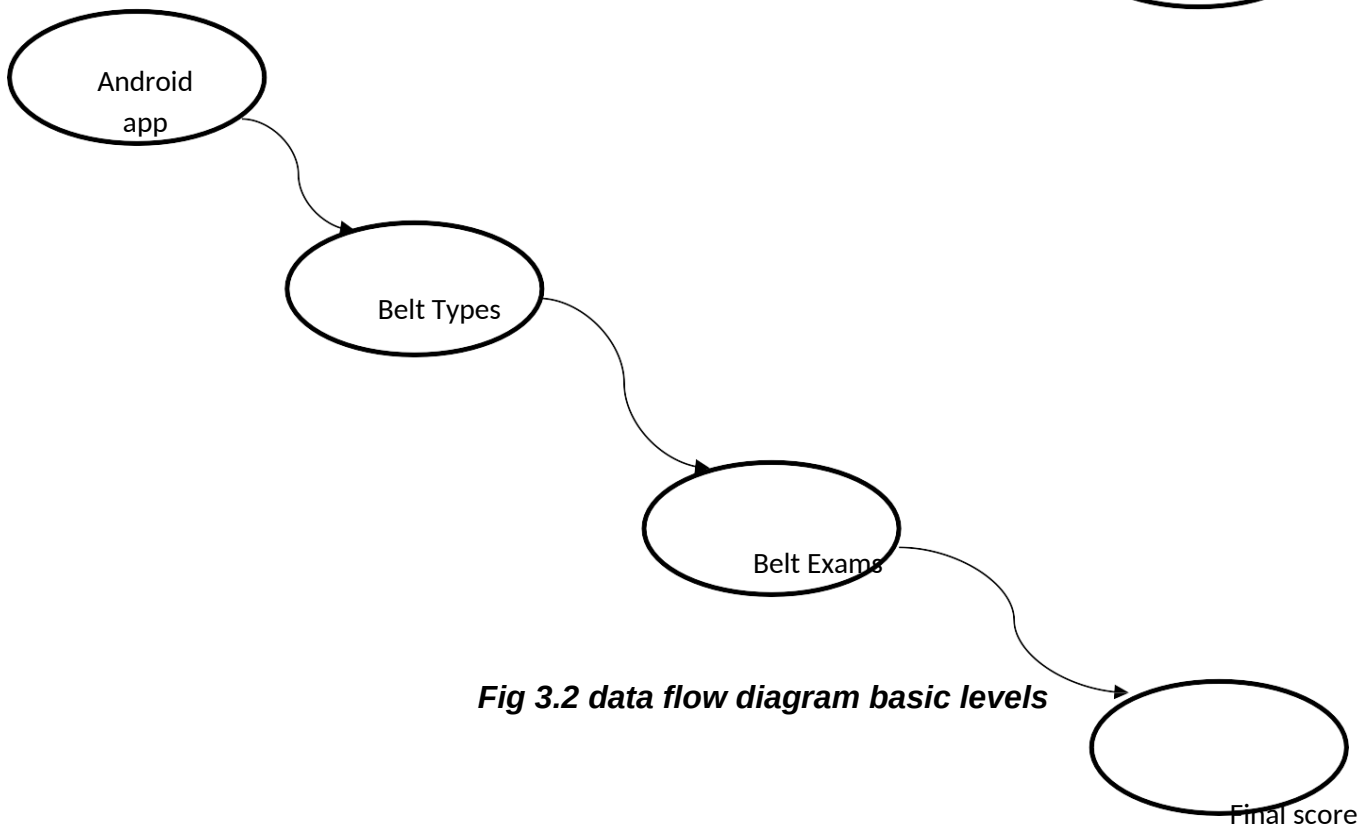
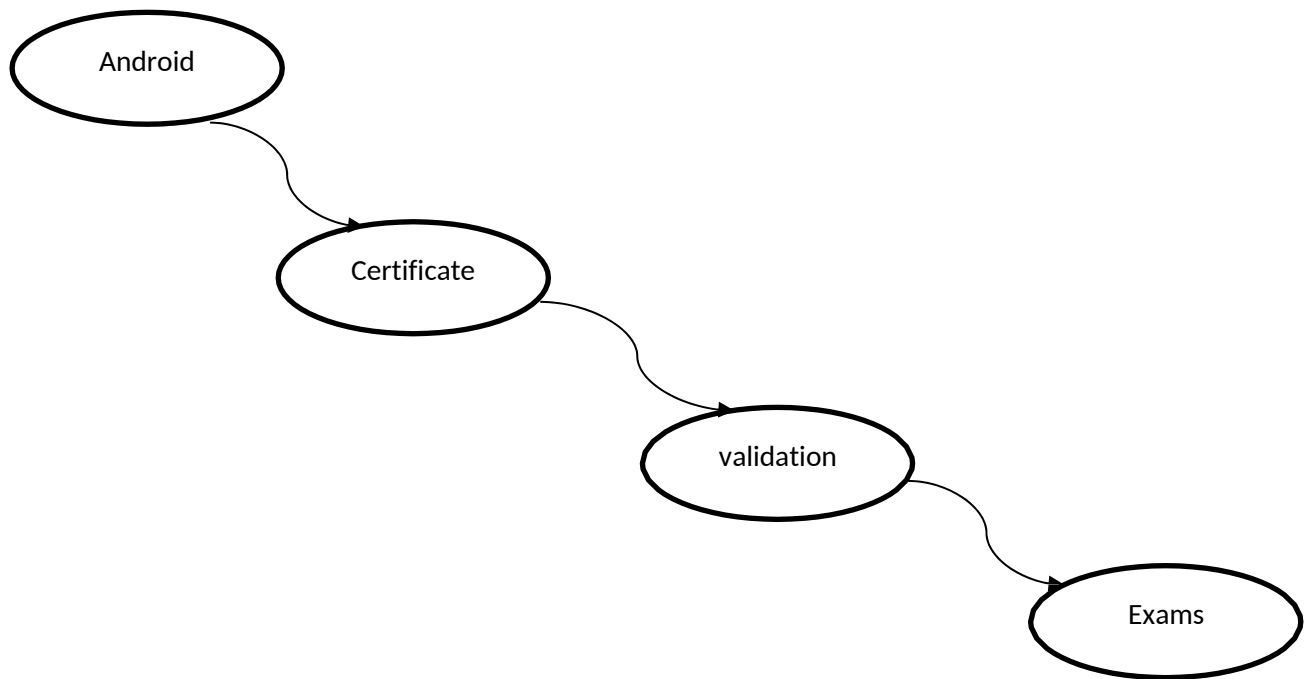


Fig 3.2 data flow diagram basic levels

Level 2:



***Fig:3.3 Data Flow
Diagram:***

UML Diagrams;

Use case Diagram:

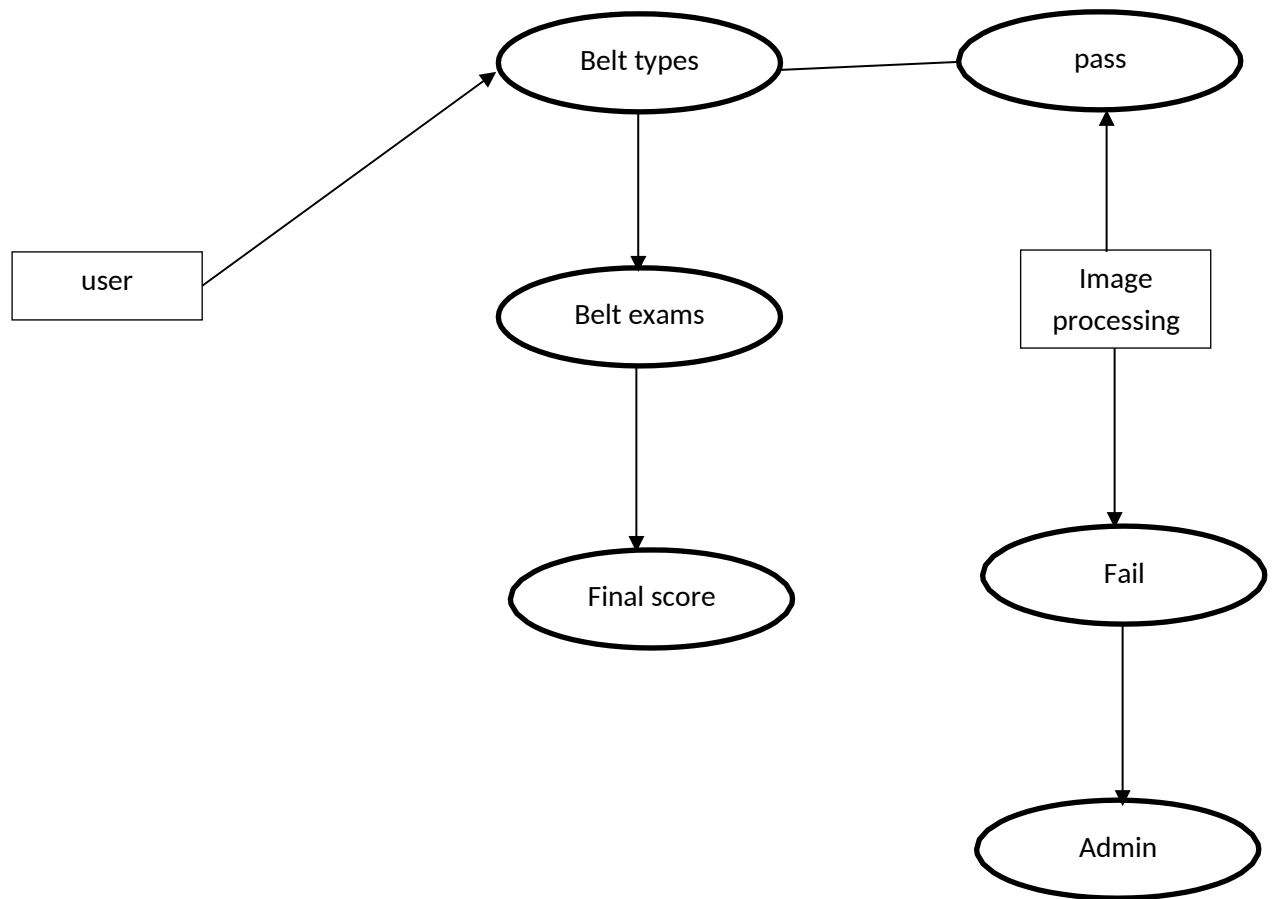


Fig:3.4 Use case Diagram

Class Diagram:

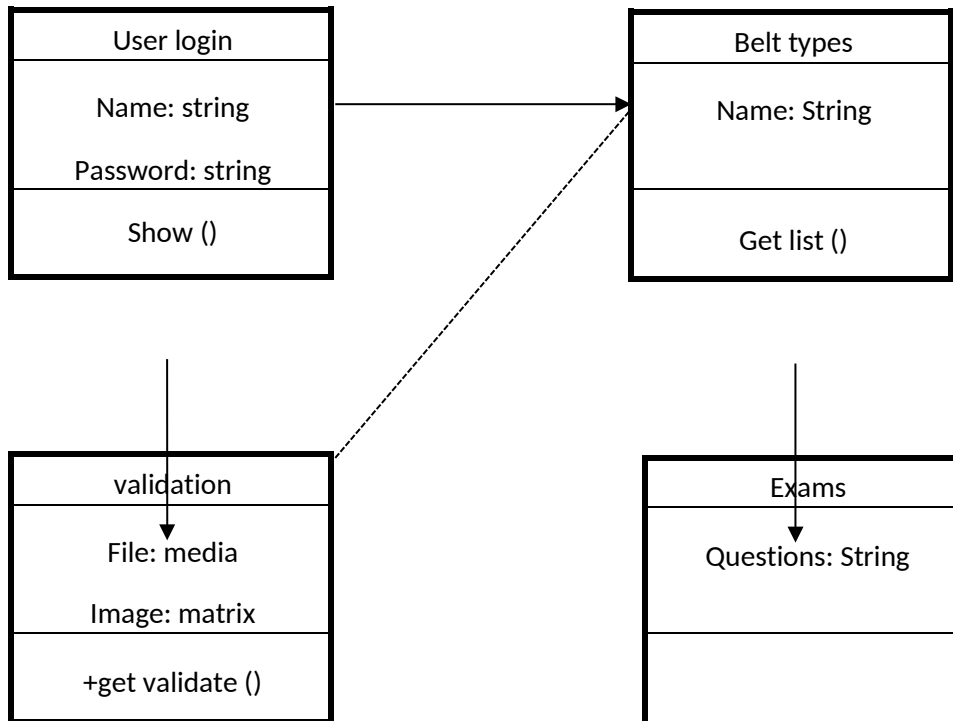


Fig: 3.5 Class Diagram

Activity Diagram:

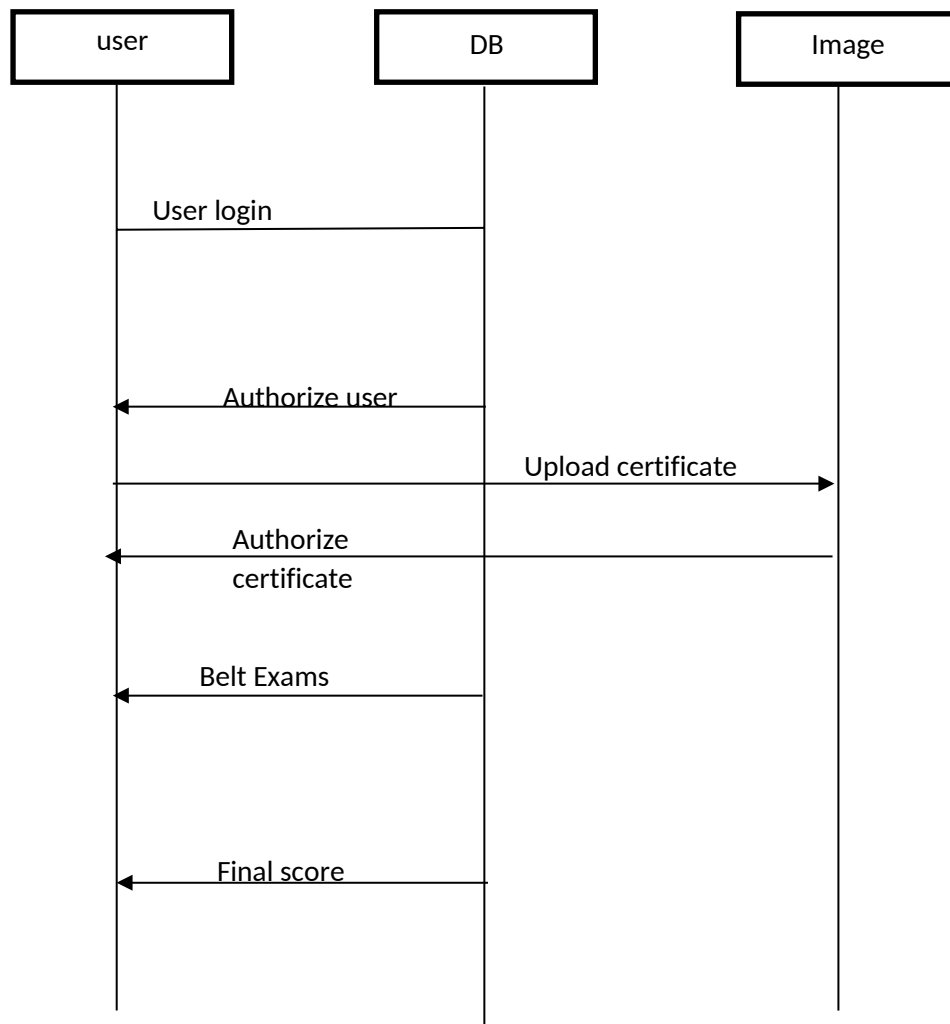


Fig:3.6 Activity Diagram

CHAPTER-4

WHAT IS MACHINE LEARNING?

Machine learning is a branch of artificial intelligence (AI) and computer science which focuses on the use of data and algorithms to imitate the way that humans learn, gradually improving its accuracy.

Machine learning is an important component of the growing field of data science. Through the use of statistical methods, algorithms are trained to make classifications or predictions, uncovering key insights within data mining projects. These insights subsequently drive decision making within applications and businesses, ideally impacting key growth metrics. As big data continues to expand and grow, the market demand for data scientists will increase, requiring them to assist in the identification of the most relevant business questions and subsequently the data to answer them.

How machine learning works?

learning system of a machine learning algorithm into three main parts.

1. **A Decision Process:** In general, machine learning algorithms are used to make a prediction or classification. Based on some input data, which can be labelled or unlabeled, your algorithm will produce an estimate about a pattern in the data.
2. **An Error Function:** An error function serves to evaluate the prediction of the model. If there are known examples, an error function can make a comparison to assess the accuracy of the model.
3. **An Model Optimization Process:** If the model can fit better to the data points in the training set, then weights are adjusted to reduce the discrepancy between the known example and the model estimate. The algorithm will repeat this evaluate and optimize process, updating weights autonomously until a threshold of accuracy has been met.

Supervised machine learning

Supervised learning, also known as supervised machine learning, is defined by its use of labeled datasets to train algorithms that to classify data or predict outcomes accurately. As input data is fed into the model, it adjusts its weights until the model ensure that the model avoids overfitting or underfitting. Supervised learning helps organizations solve for a variety of real-world problems at scale, such as classifying spam in a separate folder from your inbox. Some methods used in supervised learning include neural networks, naïve bayes, linear regression, logistic regression, random forest, support vector machine (SVM), and more.

Unsupervised machine learning

Unsupervised learning, also known as unsupervised machine learning, uses machine learning algorithms to analyze and cluster unlabeled datasets. These algorithms discover hidden patterns or data groupings without the need for human intervention. Its ability to discover similarities and differences in information make it the ideal solution for exploratory data analysis, cross-selling strategies, customer segmentation, image and pattern recognition. It's also used to reduce the number of features in a model through the process of dimensionality reduction; principal component analysis (PCA) and singular value decomposition (SVD) are two common approaches for this. Other algorithms used in unsupervised learning include neural networks, k-means clustering, probabilistic clustering methods, and more.

Semi-supervised learning

Semi-supervised learning offers a happy medium between supervised and unsupervised learning. During training, it uses a smaller labeled data set to guide classification and feature extraction from a larger, unlabeled data set. Semi-supervised learning can solve the problem of having not enough labeled data (or not being able to afford to label enough data) to train a supervised learning algorithm.

Introduction to image processing

Image processing is a method to perform some operations on an image, in order to get an enhanced image or to extract some useful information from it. It is a type of signal processing in which input is an image and output may be image or characteristics/features associated with that image. Nowadays, image processing is among rapidly growing technologies. It forms core research area within engineering and computer science disciplines too.

Image processing basically includes the following three steps:

Importing the image via image acquisition tools;

Analysing and manipulating the image;

Output in which result can be altered image or report that is based on image analysis.

There are two types of methods used for image processing namely, analogue and digital image processing. Analogue image processing can be used for the hard copies like printouts and photographs. Image analysts use various fundamentals of interpretation while using these visual techniques. Digital image processing techniques help in manipulation of the digital images by using computers. The three general phases that all types of data have to undergo while using digital technique are pre-processing, enhancement, and display, information extraction.

In this lecture we will talk about a few fundamental definitions such as image, digital image, and digital image processing. Different sources of digital images will be discussed and examples for each source will be provided. The continuum from image processing to computer vision will be covered in this lecture. Finally we will talk about image acquisition and different types of image sensors.

WHAT IS IMAGE PROCESSING?

Image processing is the process of transforming an image into a digital form and performing certain operations to get some useful information from it. The image processing system usually treats all images as 2D signals when applying certain predetermined signal processing methods.

There are five main types of image processing:

Visualization - Find objects that are not visible in the image

Recognition - Distinguish or detect objects in the image

Sharpening and restoration - Create an enhanced image from the original image

Pattern recognition - Measure the various patterns around the objects in the image

Retrieval - Browse and search images from a large database of digital images that are similar to the original image

Fundamental Image Processing Steps

Image Acquisition

Image acquisition is the first step in image processing. This step is also known as preprocessing in image processing. It involves retrieving the image from a source, usually a hardware-based source.

Image Enhancement

Image enhancement is the process of bringing out and highlighting certain features of interest in an image that has been obscured. This can involve changing the brightness, contrast, etc.

Image restoration

Image restoration is the process of improving the appearance of an image. However, unlike image enhancement, image restoration is done using certain mathematical or probabilistic models.

Color Image Processing

Color image processing includes a number of color modeling techniques in a digital domain. This step has gained prominence due to the significant use of digital images over the internet

Wavelets and Multiresolution Processing

Wavelets are used to represent images in various degrees of resolution. The images are subdivided into wavelets or smaller regions for data compression and for pyramidal representation.

Compression

Compression is a process used to reduce the storage required to save an image or the bandwidth required to transmit it. This is done particularly when the image is for use on the Internet.

Morphological Processing

Morphological processing is a set of processing operations for morphing images based on their shapes.

Segmentation

Segmentation is one of the most difficult steps of image processing. It involves partitioning an image into its constituent parts or objects.

Representation and Description

After an image is segmented into regions in the segmentation process, each region is represented and described in a form suitable for further computer processing. Representation deals with the image's characteristics and regional properties. Description deals with extracting quantitative information that helps differentiate one class of objects from the other.

Recognition

Recognition assigns a label to an object based on its description.

Applications of Image Processing

Medical Image Retrieval

Image processing has been extensively used in medical research and has enabled more efficient and accurate treatment plans. For example, it can be used for the early detection of breast cancer using a sophisticated nodule detection algorithm in breast scans. Since medical usage calls for highly trained image processors, these applications require significant implementation and evaluation before they can be accepted for use.

Traffic Sensing Technologies

In the case of traffic sensors, we use a video image processing system or VIPS. This consists of a) an image capturing system b) a telecommunication system and c) an image processing system. When capturing video, a VIPS has several detection zones which output an “on” signal whenever a vehicle enters the zone, and then output an “off” signal whenever the vehicle exits the detection zone. These detection zones can be set up for multiple lanes and can be used to sense the traffic in a particular station

Besides this, it can auto record the license plate of the vehicle, distinguish the type of vehicle, monitor the speed of the driver on the highway and lots more.

Image Reconstruction

Image processing can be used to recover and fill in the missing or corrupt parts of an image. This involves using image processing systems that have been trained extensively with existing photo datasets to create newer versions of old and damaged photos.

Face Detection

One of the most common applications of image processing that we use today is face detection. It follows deep learning algorithms where the machine is first trained with the specific features of human faces, such as the shape of the face, the distance between the eyes, etc. After teaching the machine these human face features, it will start to accept all objects in an image that resemble a human face. Face detection is a vital tool used in security, biometrics and even filters available on most social media apps these days.

Benefits of Image Processing

The implementation of image processing techniques has had a massive impact on many tech organizations. Here are some of the most useful benefits of image processing, regardless of the field of operation:

The digital image can be made available in any desired format (improved image, X-Ray, photo negative, etc)

It helps to improve images for human interpretation

Information can be processed and extracted from images for machine interpretation

The pixels in the image can be manipulated to any desired density and contrast

Images can be stored and retrieved easily

It allows for easy electronic transmission of images to third-party providers

CHAPTER – 5

5.RESULTS AND DISCUSSION, PERFORMANCE ANALYSIS

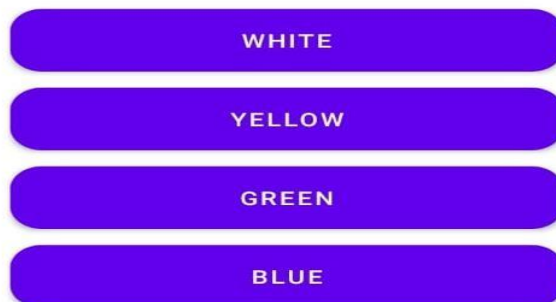


Fig 5.1 belt type display page

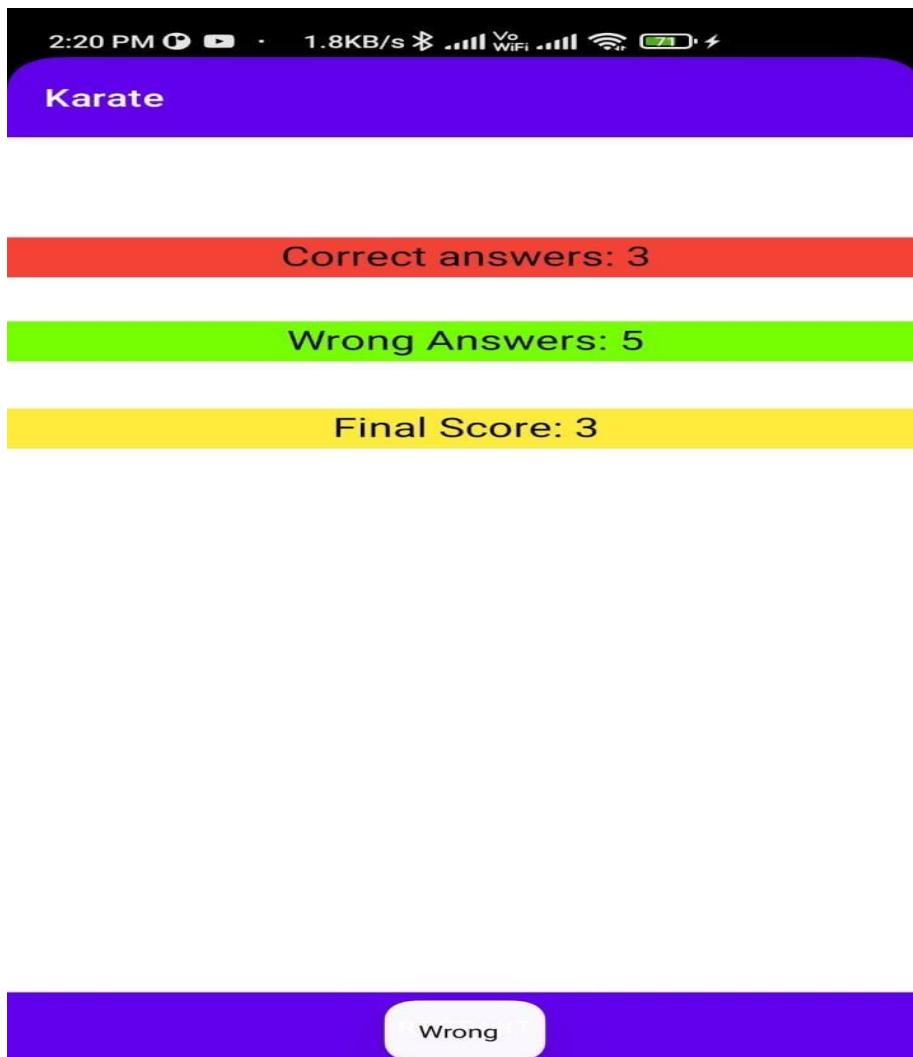


Fig 5.2 mark evaluation page

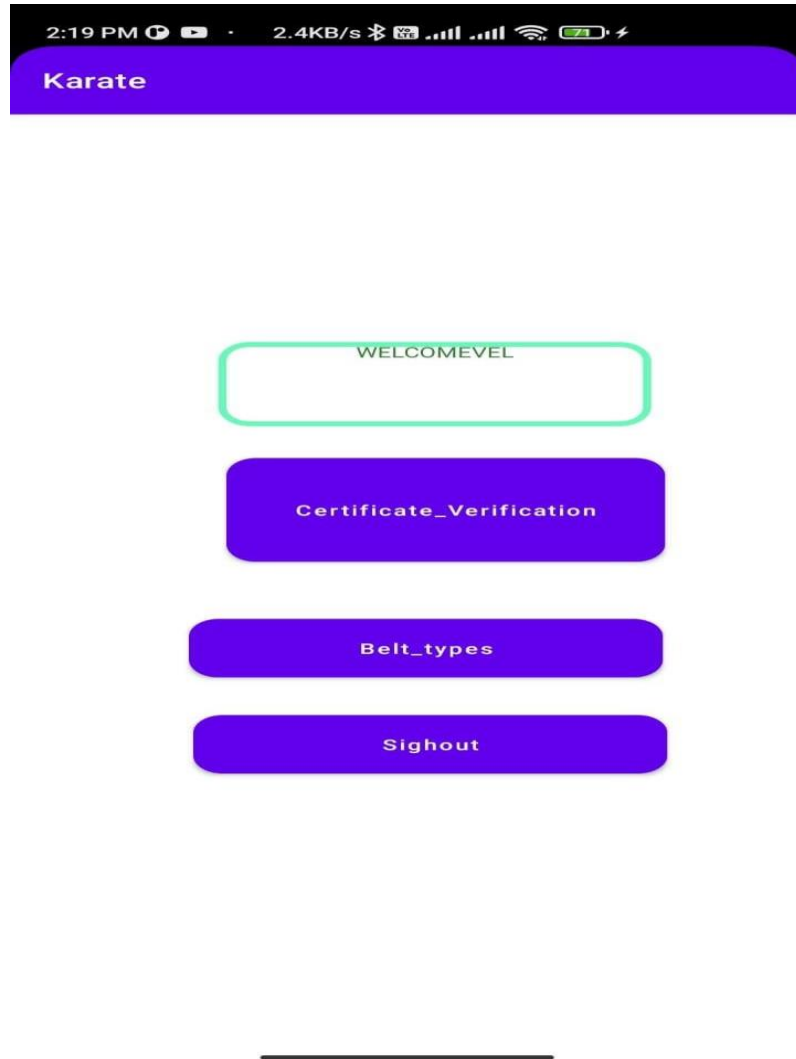


Fig 5.3 welcome page

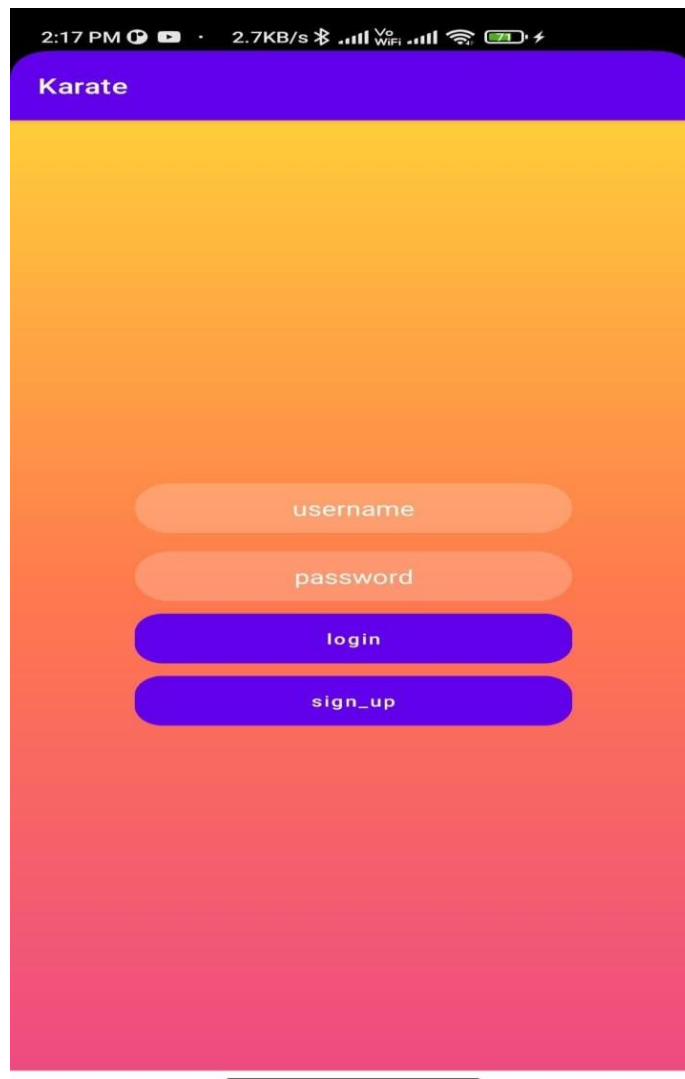


Fig 5.4 login page

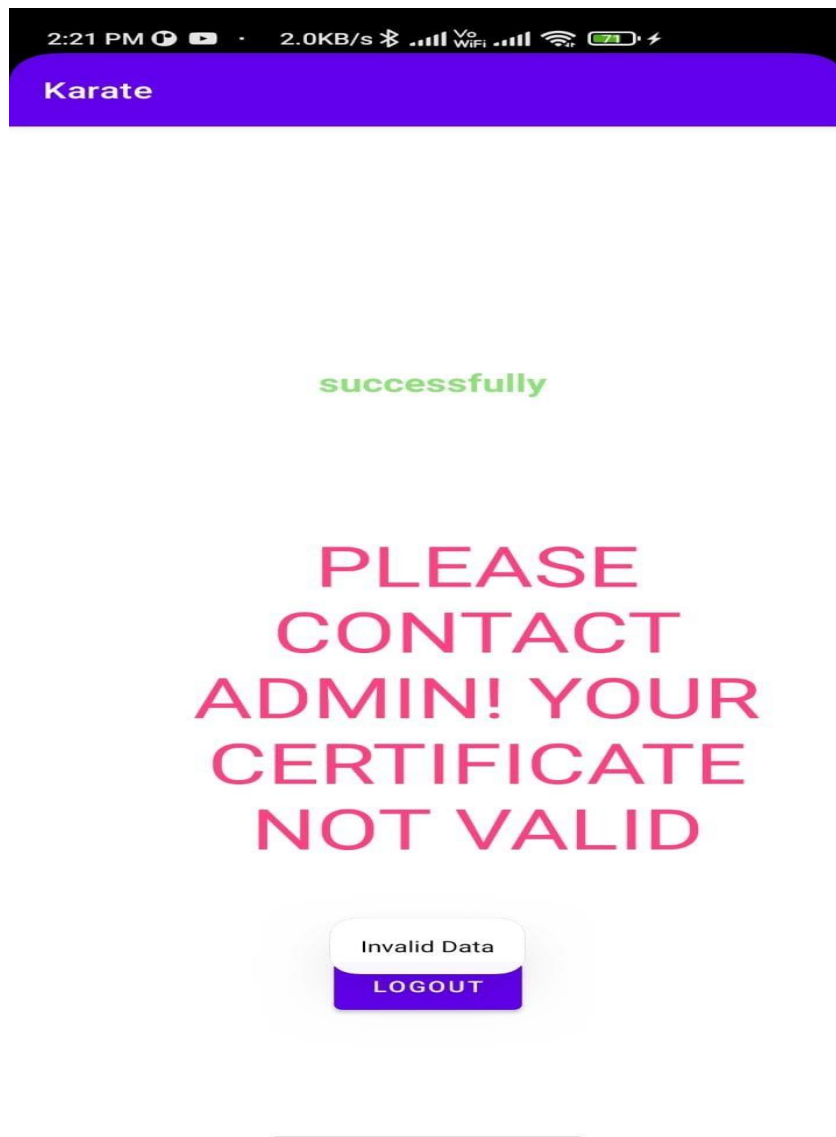


Fig 5.5 certificate verification result page



SELECT IMAGES

Fig 5.6 image uploading page

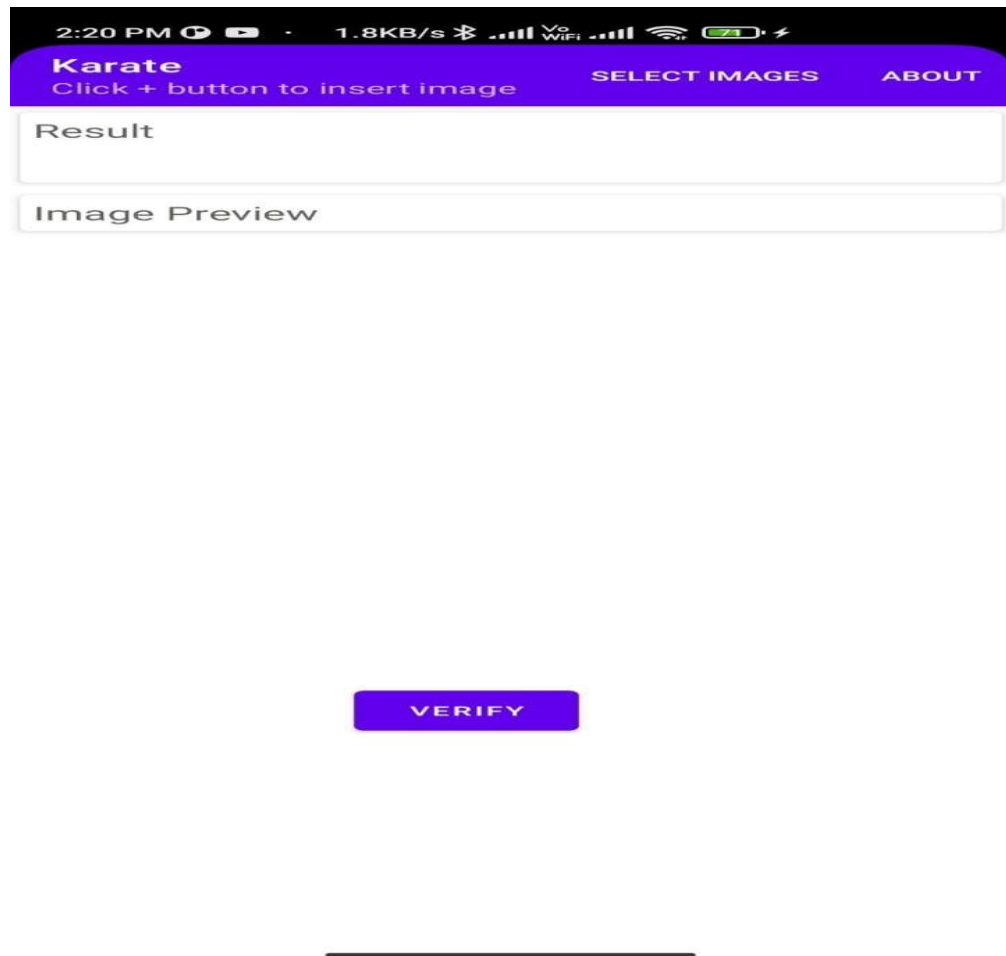


Fig 5.7 image preview page

CHAPTER – 6

CONCLUSIONS

In this project we have implement the concept of digital validation for certificate images. From this app we can able to see who are attending the exam and getting the belt. So that easy to check the user's details who are having exam, who are finishing the exams like that. Finally, we complete the process of image processing to detect the validate parameters

APPENDIX

A) REFERENCES:

- [1] Google, "Android," <http://www.android.com/>, 2014.
- [2] IDC, "Smartphone os market share, 2015 q2," <http://www.idc.com/prodserv/smartphone-os-market-share.jsp>, 2015.
- [3] Google, "Open distribution," <http://developer.android.com/distribute/open.html>, 2014.
- [4] TechInAsia, "China now has 386 million active android users," <https://www.techinasia.com/china-386-million-active-android-users-q2-2014>, 2014.
- [5] Appannie, "App annie mobile app forecast: China to surpass the us in 2016," <http://blog.appannie.com/mobile-app-forecast-china-to-surpass-us-in-2016/>, 2016.
- [6] Tech, "3545 malicious apps are reported to be pulled off shelves," <http://w.huanqiu.com/r/MV8wXzg1NDU1MzVfOTU4XzE0NTU1ODQ4MjA=>, 2016.
- [7] Y. Ng, H. Zhou, Z. Ji, H. Luo, and Y. Dong, "Which android app store can be trusted in china?" in In Computer Software and Applications Conference (COMPSAC), 2014 IEEE 38th Annual, 2014.
- [8] A. Shabtai, Y. Fledel, and Y. Elovici, "Automated static code analysis for classifying android applications using machine learning," in Proceedings of the 2010 International Conference on Computational Intelligence and Security, ser. CIS '10. IEEE, 2010, pp. 329–333.
- [9] W. Zhou, Y. Zhou, X. Jiang, and P. Ning, "Detecting repackaged smartphone applications in third-party android marketplaces," in Proceedings of the Second ACM Conference on Data and Application Security and Privacy. ACM, 2012, pp. 317–326.

B) CODING

```
package newpackage;
import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.sql.Statement;
import javax.jws.WebService;
import javax.jws.WebMethod;
import javax.jws.WebParam;

/**
 *
 * @author White_devil
 */
@WebService(serviceName = "karatewebservice")
public class karatewebservice {
    /**
     * This is a sample web service operation
     * @param user_name
     * @param password
     * @return
     */

    @WebMethod(operationName = "login")
    public String login(@WebParam(name = "username") String user_name,
```

```

@WebParam(name = "password") String password) {

    System.out.println(user_name);
    System.out.println(password);
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/karate","root","password");
        Statement st=con.createStatement();
        ResultSet rs=st.executeQuery("Select * from user where username='"+user_name+"'
and password='"+password+"' ");
        if(rs.next())
        {
            return "success";
        }
        else
        {
            return "Invalid username or password";
        }
    }
    catch(ClassNotFoundException | SQLException e){
        System.out.println(e);
    }
    return "internal server error";

}

```

```

@WebMethod(operationName = "register")
public String register(@WebParam(name = "username") String username,
@WebParam(name = "password") String password, @WebParam(name = "email") String
email, @WebParam(name = "phone") String mobile, @WebParam(name="address")
String address) {

```

```

try{
    Class.forName("com.mysql.jdbc.Driver");
    Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/karate","root","password");
    PreparedStatement ps=con.prepareStatement("Insert into
user(username,password,email,phoneno,address) Values(?,?,?, ?,?)");
    ps.setString(1, username);
    ps.setString(2, password);
    ps.setString(3, email);
    ps.setString(4, mobile);
    ps.setString(5, address);
    ps.executeUpdate();
}
catch(ClassNotFoundException | SQLException e){
    System.out.println(e);
}
return "success";

}

```

```

@WebMethod(operationName = "verify")
public String verify(@WebParam(name="data") String data){

    System.out.println(data);
    try{
        Class.forName("com.mysql.jdbc.Driver");
        Connection
con=DriverManager.getConnection("jdbc:mysql://localhost:3306/karate","root","password");
        Statement st=con.createStatement();
        ResultSet rs=st.executeQuery("Select * from data where registerno='"+data+"' ");
        if(rs.next())
        {
            System.out.println("Success");

```

```

        System.out.println(data);
        return "success";
    }
    else
    {
        System.out.println("Invalid");
        return "Invalid Data";
    }
}
catch(ClassNotFoundException | SQLException e){
    System.out.println(e);
}
return "internal server error";
}
}

```

[4/22, 2:15 PM] Pantech 2: package com.example.karate;

```

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.ActionBar;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

```

```

import android.Manifest;
import android.annotation.SuppressLint;
import android.app.AlertDialog;
import android.content.ContentValues;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.graphics.Bitmap;

```

```

import android.graphics.drawable.BitmapDrawable;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
import android.provider.MediaStore;
import android.util.Log;
import android.util.SparseArray;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.Toast;

import com.google.android.gms.vision.Frame;
import com.google.android.gms.vision.text.TextBlock;
import com.google.android.gms.vision.text.TextRecognizer;
import com.theartofdev.edmodo.cropper.CropImage;
import com.theartofdev.edmodo.cropper.CropImageView;

public class Camera extends AppCompatActivity {

    private EditText mResultEt;
    private ImageView mPreviewIv;
    Button button;
    String text;
    //Permission Code
    private static final int CAMERA_REQUEST_CODE = 200;
    private static final int STORAGE_REQUEST_CODE = 400;
    private static final int IMAGE_PICK_GALLERY_CODE = 1000;
    private static final int IMAGE_PICK_CAMERA_CODE = 2001;

```

```

String cameraPermission[];
String storagePermission[];

Uri image_uri;

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

    ActionBar actionBar = getSupportActionBar();
    actionBar.setSubtitle("Click + button to insert image");

    mResultEt = findViewById(R.id.resultEt);
    mPreviewLv = findViewById(R.id.imageLv);

    //camera permission
    cameraPermission = new String[] {Manifest.permission.CAMERA,
        Manifest.permission.WRITE_EXTERNAL_STORAGE};
    //storage permission
    storagePermission = new String[]
{Manifest.permission.WRITE_EXTERNAL_STORAGE};

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

    button.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            Toast.makeText(getApplicationContext(), text, Toast.LENGTH_SHORT).show();
        }
    });
}

```

```

        if(text.matches("")){
            Toast.makeText(getApplicationContext(),"Text Values are
empty",Toast.LENGTH_LONG).show();
        }
        else{
            Activetask task = new Activetask();
            task.execute();
        }
    }
});
}

```

@Override

```

public boolean onCreateOptionsMenu(Menu menu) {
    getMenuInflater().inflate(R.menu.main_menu, menu);
    return true;
}

```

@Override

```

public boolean onOptionsItemSelected(@NonNull MenuItem item) {
    switch (item.getItemId()) {
        case R.id.search_button:
            showImageImportDialog();
            Log.d("Button","Error");
            break;

        case R.id.about:
            dialogAbout();
            break;
    }
}

```

```

        return true;
    }

    private void dialogAbout() {
        new AlertDialog.Builder(this)
            .setTitle("About App")
            .setMessage("Image Processing Part")
            .setPositiveButton("CLOSE", new DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog, int which) {
                    dialog.dismiss();
                }
            })
            .show();
    }

    private void showImageImportDialog() {
        String[] items = {"Camera", "Gallery"};
        AlertDialog.Builder dialog = new AlertDialog.Builder(this);

        dialog.setTitle("Select Image");
        dialog.setItems(items, new DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog, int which) {
                if (which == 0) {
                    if (!checkCameraPermission()) {
                        //camera permission not allowed, request it
                        requestCameraPermission();
                    } else {
                        //permission allowed, take picture
                        pickCamera();
                    }
                }
            }
        })
    }

```

```

        if (which == 1) {
            if (!checkStoragePermission()) {
                //storage permission not allowed, request it
                requestStoragePermission();
            } else {
                //permission allowed, take picture
                pickGallery();
            }
        }
    }
    });
    dialog.create().show();
}

```

```

private void pickGallery() {
    //intent to pick image from gallery
    Intent intent = new Intent(Intent.ACTION_PICK);
    //set intent type to image
    intent.setType("image/*");
    startActivityForResult(intent, IMAGE_PICK_GALLERY_CODE);
}

```

```

private void pickCamera() {
    //intent to take image from camera, it will also be save to storage to get high quality
    image
    ContentValues values = new ContentValues();
    values.put(MediaStore.Images.Media.TITLE, "NewPick"); //title of the picture
    values.put(MediaStore.Images.Media.DESCRPTION, "Image To Text"); //title of the
    picture
    image_uri =
    getContentResolver().insert(MediaStore.Images.Media.EXTERNAL_CONTENT_URI,
    values);
}

```

```

        Intent cameraIntent = new Intent(MediaStore.ACTION_IMAGE_CAPTURE);
        cameraIntent.putExtra(MediaStore.EXTRA_OUTPUT, image_uri);
        startActivityForResult(cameraIntent, IMAGE_PICK_CAMERA_CODE);
    }

    private void requestStoragePermission() {
        ActivityCompat.requestPermissions(this, storagePermission,
        STORAGE_REQUEST_CODE);
    }

    private boolean checkStoragePermission() {
        boolean result = ContextCompat.checkSelfPermission(this,
            Manifest.permission.WRITE_EXTERNAL_STORAGE) ==
        (PackageManager.PERMISSION_GRANTED);
        return result;
    }

    private void requestCameraPermission() {
        ActivityCompat.requestPermissions(this, cameraPermission,
        CAMERA_REQUEST_CODE);
    }

    private boolean checkCameraPermission() {
        boolean result = ContextCompat.checkSelfPermission(this,
            Manifest.permission.CAMERA) ==
        (PackageManager.PERMISSION_GRANTED);

        boolean result1 = ContextCompat.checkSelfPermission(this,
            Manifest.permission.WRITE_EXTERNAL_STORAGE) ==
        (PackageManager.PERMISSION_GRANTED);
        return result && result1;
    }

```

```

//handle permission result
@SuppressLint("MissingSuperCall")
@Override
public void onRequestPermissionsResult(int requestCode, @NonNull String[]
permissions, @NonNull int[] grantResults) {
    switch (requestCode){
        case CAMERA_REQUEST_CODE:
            if (grantResults.length > 0) {
                boolean cameraAccepted = grantResults[0] ==
                    PackageManager.PERMISSION_GRANTED;
                boolean writeStorageAccepted = grantResults[0] ==
                    PackageManager.PERMISSION_GRANTED;

                if (cameraAccepted && writeStorageAccepted) {
                    pickCamera();
                } else {
                    Toast.makeText(this, "Permission denied",
Toast.LENGTH_SHORT).show();
                }
            }
            break;

        case STORAGE_REQUEST_CODE:
            if (grantResults.length > 0) {
                boolean writeStorageAccepted = grantResults[0] ==
                    PackageManager.PERMISSION_GRANTED;

                if (writeStorageAccepted) {
                    pickGallery();
                } else {
                    Toast.makeText(this, "Permission denied",
Toast.LENGTH_SHORT).show();

```

```

        }
    }
    break;
}
}

//handle image result
@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (resultCode == RESULT_OK) {
        if (requestCode == IMAGE_PICK_GALLERY_CODE) {
            //got image from gallery now crop it
            CropImage.activity(data.getData())
                .setGuidelines(CropImageView.Guidelines.ON) //enable image guid lines
                .start(Camera.this);
        }

        if (requestCode == IMAGE_PICK_CAMERA_CODE) {
            //got image from camera now crop it
            CropImage.activity(image_uri)
                .setGuidelines(CropImageView.Guidelines.ON) //enable image guid lines
                .start(Camera.this);
        }
    }
}

//get cropped image
if (requestCode == CropImage.CROP_IMAGE_ACTIVITY_REQUEST_CODE) {
    CropImage.ActivityResult result = CropImage.getActivityResult(data);
    if (resultCode == RESULT_OK) {
        Uri resultUri = result.getUri(); //get image uri
        //set image to image view
        mPreviewIv.setImageURI(resultUri);
    }
}

```

```

        //get drawable bitmap for text recognition
        BitmapDrawable bitmapDrawable = (BitmapDrawable)
mPreviewIv.getDrawable();
        Bitmap bitmap = bitmapDrawable.getBitmap();

        TextRecognizer recognizer = new
TextRecognizer.Builder(getApplicationContext()).build();

        if (!recognizer.isOperational()) {
            Toast.makeText(this, "Error", Toast.LENGTH_SHORT).show();
        } else {
            Frame frame = new Frame.Builder().setBitmap(bitmap).build();
            SparseArray<TextBlock> items = recognizer.detect(frame);
            StringBuilder sb = new StringBuilder();
            //get text from sb until there is no text
            for (int i = 0; i < items.size(); i++) {
                TextBlock myItem = items.valueAt(i);
                sb.append(myItem.getValue());
                sb.append("\n");
            }

            //set text to edit text
            mResultEt.setText(sb.toString());
            text = sb.toString();
            //Toast.makeText(getApplicationContext(), text,
Toast.LENGTH_SHORT).show();

        }
    } else if (resultCode ==
CropImage.CROP_IMAGE_ACTIVITY_RESULT_ERROR_CODE) {
        //if there is any error show it
        Exception error = result.getError();

```

```

        Toast.makeText(this, "" + error, Toast.LENGTH_SHORT).show();
    }

}

}

private class Activetask extends AsyncTask<String,Void,Void> {
    String s1 =text;
    String res=null;

    protected void onPreExecute() {
        //Toast.makeText(getApplicationContext(), "Please wait... ",
Toast.LENGTH_LONG).show();
    }

    protected Void doInBackground(String... params) {
        res=callservice.verifyservice(s1,"verify");
        return null;
    }

    protected void onPostExecute(Void result) {
        if(res.equals("success")) {
            Intent intent=new Intent(getBaseContext(),Belttypes.class);
            //intent.putExtra("data", text);
            startActivity(intent);
        }

        else {
            Intent in = new Intent(getBaseContext(), Contact.class);
            Toast.makeText(getApplicationContext(),res,Toast.LENGTH_LONG).show();
            startActivity(in);
        }
    }
}

```

}