

Institute of Technology, Carlow

Computing

Course: BSc (Hons) Software Development Year 4

Author: Eamonn Gaynor

Student ID: C00197458

Tutor: Mr. Nigel Whyte

Document: Technical Document

Plant Disease Identification Application

Institiúid Teicneolaíochta Cheatharlach



INSTITUTE *of*
TECHNOLOGY

CARLOW

At the heart of South Leinster

Installation Guidelines	4
Code	5
2.1 Android Application	5
2.1.1 AndroidManifest.xml	5
2.1.2 cameraDiagnosis.java	6
2.1.3 Constants.java	13
2.1.4 data.java	14
2.1.5 library.java	17
2.1.6 LoginRequest.java	20
2.1.7 LoginScreen.java	20
2.1.8 RegisterRequest.java	24
2.1.9 registerScreen.java	24
2.1.10 StartScreen.java	27
2.1.11 UserAreaActivity.java	29
2.2 Web Application	33
2.2.1 login.php	33
2.2.2 home.php	36
2.2.3 logout.php	38
2.2.4 groupSelection.php	38
2.2.5 manageIndex.php	40
2.2.6 mark.php	42
2.2.7 report.php	43
2.2.8 reportIndex.php	49
2.2.9 weatherD.php	50
2.2.10 analyze.php	52
2.2.11 change.php	53
2.2.12 chooseImage.php	55
2.2.13 config.php	57
2.2.14 csv.php	58
2.2.15 delete.php	59
2.2.16 edit.php	60
2.2.17 editor.php	61
2.2.18 fusioncharts.php	64

2.3 Application Programming Interface (API)	66
2.3.1 upload.php	66
2.3.2 Register.php	72
2.3.3 login.php	72
2.3.4 getLibrary.php	73
2.3.5 getGroup.php	74
2.3.6 getData.php	75
2.3.7 dbDetails.php	76
2.3.8 colours.inc.php	76

1. Installation Guidelines

1. To install the mobile application, please visit the link below on an **Android** device:

<http://glasnost.itcarlow.ie/~softeng4/C00197458/index.html>

- a. Scroll to 'Documents'.
 - b. Select 'APK Download'.
 - c. Once downloaded on the device, please allow your mobile to install third party applications. **NOTE:** This can be done by visiting the settings on your mobile device, please refer to manual of your device as they all differ.
 - d. Upon launch of the application, please select 'Get Started'. Please select 'Allow' on the alert box for the mobile device to access your location information.
 - e. To use the application, you must register an account with the system. To do so, please select the 'Register here' link. Enter the appropriate details and select 'Register'.
 - f. Once registered, please use the details provided previous to login to the system.
2. In order to the web application, please visit the link below on your web browser:
<http://www.c00197458.candept.com/PlantVision/login.php>
 - a. To use the web application, you must have a registered account with the system (If you have already registered on the mobile application, you do not need to register, just log in).
 - b. Please select the 'Create an account' link. Enter the appropriate details and select 'Register'.
 - c. Once registered, please use the details provided previous to login to the system.

2. Code

2.1 Android Application

2.1.1 AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.plantvision">

    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"/>
    <uses-feature android:name="android.hardware.location.gps" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity
            android:name=".StartScreen"
            android:label="@string/app_name"
            android:theme="@style/AppTheme.NoActionBar">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".UserAreaActivity"
            android:label="@string/title_activity_user_area"
            android:theme="@style/AppTheme.NoActionBar" />
        <activity
            android:name=".MainActivity"
            android:label="PlantVision"
            android:theme="@style/AppTheme.NoActionBar" />

        <provider
            android:name="android.support.v4.content.FileProvider"
            android:authorities="com.example.plantvision.fileprovider"
            android:exported="false"
            android:grantUriPermissions="true">
            <meta-data
                android:name="android.support.FILE_PROVIDER_PATHS"
                android:resource="@xml/file_paths" />
        </provider>

        <activity android:name=".cameraDiagnosis" />
        <activity android:name=".capturedDiagnosis" />
        <activity android:name=".LoginScreen" />
        <activity android:name=".registerScreen" />
        <activity android:name=".data" />
        <activity android:name=".library"></activity>
```

```
</application>  
</manifest>
```

2.1.2 cameraDiagnosis.java

```
package com.example.plantvision;  
  
import android.Manifest;  
import android.content.Context;  
import android.content.DialogInterface;  
import android.content.Intent;  
import android.content.SharedPreferences;  
import android.content.pm.PackageManager;  
import android.database.Cursor;  
import android.net.Uri;  
import android.os.AsyncTask;  
import android.os.Build;  
import android.provider.MediaStore;  
import android.support.annotation.NonNull;  
import android.support.design.widget.BottomNavigationView;  
import android.support.v4.app.ActivityCompat;  
import android.support.v4.content.ContextCompat;  
import android.support.v7.app.AlertDialog;  
import android.support.v7.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.AdapterView;  
import android.widget.Button;  
import android.widget.CompoundButton;  
import android.widget.EditText;  
import android.widget.ImageView;  
import android.graphics.Bitmap;  
import android.widget.ListView;  
import android.widget.RadioButton;  
import android.widget.RadioGroup;  
import android.widget.Spinner;  
import android.widget.Switch;  
import android.widget.TextView;  
import android.widget.Toast;  
  
import net.gotev.uploadservice.MultipartUploadRequest;  
import net.gotev.uploadservice.UploadNotificationConfig;  
  
import org.json.JSONArray;  
import org.json.JSONException;  
import org.json.JSONObject;  
  
import java.io.BufferedReader;  
import java.io.IOException;  
import java.io.InputStreamReader;  
import java.net.HttpURLConnection;  
import java.net.URL;  
import java.util.UUID;  
  
import static com.example.plantvision.LoginScreen.MyPREFERENCES;  
import static java.lang.Boolean.FALSE;  
import static java.lang.Boolean.TRUE;
```

```

public class cameraDiagnosis extends AppCompatActivity {

    //Declaring views
    private Button buttonUpload;
    private Button buttonGoBack;
    private ImageView imageView;
    private EditText editText;
    private EditText editText2;
    private Boolean radio;
    //Image request code
    private int PICK_IMAGE_REQUEST = 1;
    //storage permission code
    private static final int STORAGE_PERMISSION_CODE = 123;
    //Bitmap to get image from gallery
    private Bitmap bitmap;
    //Uri to store the image uri
    private Uri filePath;
    ListView listView;
    private Switch mySwitch;
    //TextView latitudeText;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_camera_diagnosis);
        showFileChooser();
        //Requesting storage permission
        requestStoragePermission();

        buttonGoBack = (Button) findViewById(R.id.returnB);
        buttonUpload = (Button) findViewById(R.id.upload);
        imageView = (ImageView) findViewById(R.id.cameraDiagnosis);
        editText = (EditText) findViewById(R.id.editTextName);
        editText2 = (EditText) findViewById(R.id.editGroupName);
        //latitudeText = (TextView) findViewById(R.id.textView4);
        listView = (ListView) findViewById(R.id.listView);
        getJSON("http://www.c00197458.candept.com/getGroup.php");

        mySwitch = (Switch) findViewById(R.id.mySwitch);
        //set the switch to ON
        mySwitch.setChecked(false);
        //attach a listener to check for changes in state
        mySwitch.setOnCheckedChangeListener(new
CompoundButton.OnCheckedChangeListener(){
            @Override
            public void onCheckedChanged(CompoundButton buttonView, boolean isChecked) {
                if(isChecked){
                    radio = TRUE;
                    AlertDialog.Builder builder;
                    if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
                        builder = new AlertDialog.Builder(cameraDiagnosis.this,
android.R.style.Theme_Material_Dialog_Alert);
                    } else {
                        builder = new AlertDialog.Builder(cameraDiagnosis.this);
                    }
                    builder.setTitle("Please Note")
                        .setMessage("Automatic detection is based off prior
selection of disease as part of plant group on web application")
                        .setPositiveButton(android.R.string.yes, new

```

```

DialogInterface.OnClickListener() {
    public void onClick(DialogInterface dialog, int which) {
        // continue
    }
})
    .setNegativeButton(android.R.string.no, new
DialogInterface.OnClickListener() {
    public void onClick(DialogInterface dialog, int which) {
        radio = FALSE;
        mySwitch.setChecked(false);
    }
})
    .setIcon(android.R.drawable.ic_dialog_alert)
    .show();
}
}
});

//-----Navigation Bar functions
-----
BottomNavigationView bottomNavigationView =
(BottomNavigationView)findViewById(R.id.bottom_navigation);
bottomNavigationView.setOnNavigationItemSelectedListener(new
BottomNavigationView.OnNavigationItemSelectedListener() {
    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case R.id.action_library:
                Toast.makeText(cameraDiagnosis.this, "Coming soon",
Toast.LENGTH_SHORT).show();
                return true;

            case R.id.action_account:
                Intent loginScreen = new Intent(cameraDiagnosis.this,
LoginScreen.class);
                cameraDiagnosis.this.startActivity(loginScreen);
                finish();
                return true;

            case R.id.action_home:
                Intent homeScreen = new Intent(cameraDiagnosis.this,
StartScreen.class);
                cameraDiagnosis.this.startActivity(homeScreen);
                finish();
                return true;
        }
        return false;
    }
});

//-----END-----
-----

buttonUpload.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        uploadMultipart();

        if(mySwitch.isChecked() == FALSE){
            Intent homeScreen = new Intent(cameraDiagnosis.this,

```



```

UserAreaActivity.class);
        startActivity(homeScreen);
        Toast.makeText(cameraDiagnosis.this, "Image uploaded",
Toast.LENGTH_SHORT).show();
        finish();
    }

    if(mySwitch.isChecked() == TRUE){

        AlertDialog.Builder builder;
        if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
            builder = new AlertDialog.Builder(cameraDiagnosis.this,
android.R.style.Theme_Material_Dialog_Alert);
        } else {
            builder = new AlertDialog.Builder(cameraDiagnosis.this);
        }
        builder.setTitle("View result of image?")
            .setMessage("Results based on prior selections")
            .setPositiveButton(android.R.string.yes, new
DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int which) {
data.class);

                    startActivity(cDiagnosis);
                    Toast.makeText(cameraDiagnosis.this, "Image
uploaded", Toast.LENGTH_SHORT).show();
                    finish();
                }
            })
            .setNegativeButton(android.R.string.no, new
DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int which) {
UserAreaActivity.class);

                    startActivity(homeScreen);
                    Toast.makeText(cameraDiagnosis.this, "Image
uploaded", Toast.LENGTH_SHORT).show();
                    finish();
                }
            })
            .setIcon(android.R.drawable.ic_dialog_alert)
            .show();
    }
});

buttonGoBack.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent cDiagnosis = new Intent(cameraDiagnosis.this,
UserAreaActivity.class);
        startActivity(cDiagnosis);
        finish();
    }
});

}

private void getJSON(final String urlWebService) {

    class GetJSON extends AsyncTask<Void, Void, String> {

        @Override
        protected void onPreExecute() {
            super.onPreExecute();

```

```

    }

    @Override
    protected void onPostExecute(String s) {
        super.onPostExecute(s);
        //Toast.makeText(getApplicationContext(), s, Toast.LENGTH_SHORT).show();
        try {
            loadIntoListView(s);
        } catch (JSONException e) {
            e.printStackTrace();
        }
    }

    @Override
    protected String doInBackground(Void... voids) {
        try {
            URL url = new URL(urlWebService);
            HttpURLConnection con = (HttpURLConnection) url.openConnection();
            StringBuilder sb = new StringBuilder();
            BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(con.getInputStream()));
            String json;
            while ((json = bufferedReader.readLine()) != null) {
                sb.append(json + "\n");
            }
            return sb.toString().trim();
        } catch (Exception e) {
            return null;
        }
    }
}
GetJSON getJSON = new GetJSON();
getJSON.execute();
}

private void loadIntoListView(String json) throws JSONException {
    JSONArray jsonArray = new JSONArray(json);
    String[] heroes = new String[jsonArray.length()];
    for (int i = 0; i < jsonArray.length(); i++) {
        JSONObject obj = jsonArray.getJSONObject(i);
        heroes[i] = obj.getString("name");
    }
    ArrayAdapter<String> arrayAdapter = new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1, heroes);
    listView.setAdapter(arrayAdapter);
}

/*
// ----- Radio button for upload -----
public void onRadioButtonClicked(View view) {
    // Is the button now checked?
    boolean checked = ((RadioButton) view).isChecked();
    // Check which radio button was clicked
    switch(view.getId()) {
        case R.id.radio_yes:
            if (checked)
                radio = TRUE;
            AlertDialog.Builder builder;
            if (Build.VERSION.SDK_INT >= Build.VERSION_CODES.LOLLIPOP) {
                builder = new AlertDialog.Builder(cameraDiagnosis.this,
android.R.style.Theme_Material_Dialog_Alert);
            } else {
                builder = new AlertDialog.Builder(cameraDiagnosis.this);
            }

```

```

        builder.setTitle("Please Note")
            .setMessage("Automatic detection is based off prior selection of
disease as part of plant group on web application")
            .setPositiveButton(android.R.string.yes, new
DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int which) {
                    // continue with delete
                }
            })
            .setNegativeButton(android.R.string.no, new
DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int which) {
                    // do nothing
                }
            })
            .setIcon(android.R.drawable.ic_dialog_alert)
            .show();
        break;
    case R.id.radio_no:
        if (checked)
            radio = FALSE;
        break;
    }
}
// ----- end / Radio button -----
*/

/*
* This is the method responsible for image upload
* We need the full image path and the name for the image in this method
* */
public void uploadMultipart() {

    String radioAnswer = "No";
    /*
    if(radio == TRUE){
        radioAnswer = "Yes";
    }
    else if(radio ==FALSE){
        radioAnswer = "No";
    }
    */
    if(mySwitch.isChecked() == TRUE){
        radioAnswer = "Yes";
    }
    else if(mySwitch.isChecked()==FALSE){
        radioAnswer = "No";
    }
    //getting name for the image
    String name = editText.getText().toString().trim();
    //getting group name for image
    String group = editText2.getText().toString().trim();
    //getting the actual path of the image
    String path = getPath(filePath);

    SharedPreferences sharedPreferences = getSharedPreferences(MyPREFERENCES,
Context.MODE_PRIVATE);
    final String login = sharedPreferences.getString("Name","No name defined" );
    final Float latitude = sharedPreferences.getFloat("Lat",0 );
    final Float longitude = sharedPreferences.getFloat("Long",0 );
    final String address = sharedPreferences.getString("fulladdress","no name
defined");

    //Uploading code

```

```

try {
    String uploadId = UUID.randomUUID().toString();

    //Creating a multi part request
    new MultipartUploadRequest(this, uploadId, Constants.UPLOAD_URL)
        .addFileToUpload(path, "image") //Adding file
        .addParameter("name", name) //Adding text parameter to the request
        .addArrayParameter("username", login)
        .addParameter("group", group) //Adding text parameter to the request
        .addParameter("radio", radioAnswer) //Adding text parameter to the
request
        .addParameter("address", address)
        .addParameter("longitude", String.valueOf(longitude))
        .addParameter("latitude", String.valueOf(latitude))
        .setNotificationConfig(new UploadNotificationConfig())
        .setMaxRetries(2)
        .startUpload(); //Starting the upload

    } catch (Exception exc) {
        Toast.makeText(this, exc.getMessage(), Toast.LENGTH_SHORT).show();
    }
}

//method to show file chooser
private void showFileChooser() {
    Intent intent = new Intent();
    intent.setType("image/*");
    intent.setAction(Intent.ACTION_GET_CONTENT);
    startActivityForResult(Intent.createChooser(intent, "Select Picture"),
PICK_IMAGE_REQUEST);
}

//handling the image chooser activity result
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);

    if (requestCode == PICK_IMAGE_REQUEST && resultCode == RESULT_OK && data != null
&& data.getData() != null) {
        filePath = data.getData();
        try {
            bitmap = MediaStore.Images.Media.getBitmap(getContentResolver(),
filePath);
            imageView.setImageBitmap(bitmap);

        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}

//method to get the file path from uri
public String getPath(Uri uri) {
    Cursor cursor = getContentResolver().query(uri, null, null, null, null);
    cursor.moveToFirst();
    String document_id = cursor.getString(0);
    document_id = document_id.substring(document_id.lastIndexOf(":") + 1);
    cursor.close();

    cursor = getContentResolver().query(
        android.provider.MediaStore.Images.Media.EXTERNAL_CONTENT_URI,
        null, MediaStore.Images.Media._ID + " = ? ", new String[]{document_id},
null);
    cursor.moveToFirst();
    String path =
cursor.getString(cursor.getColumnIndex(MediaStore.Images.Media.DATA));
}

```

```

        cursor.close();

        return path;
    }

    //Requesting permission
    private void requestStoragePermission() {
        if (ContextCompat.checkSelfPermission(this,
Manifest.permission.READ_EXTERNAL_STORAGE) == PackageManager.PERMISSION_GRANTED)
            return;

        if (ActivityCompat.shouldShowRequestPermissionRationale(this,
Manifest.permission.READ_EXTERNAL_STORAGE)) {
            //If the user has denied the permission previously your code will come to
this block
            //Here you can explain why you need this permission
            //Explain here why you need this permission
        }
        //And finally ask for the permission
        ActivityCompat.requestPermissions(this, new
String[]{Manifest.permission.READ_EXTERNAL_STORAGE}, STORAGE_PERMISSION_CODE);
    }

    //This method will be called when the user will tap on allow or deny
    @Override
    public void onRequestPermissionsResult(int requestCode, @NonNull String[]
permissions, @NonNull int[] grantResults) {

        //Checking the request code of our request
        if (requestCode == STORAGE_PERMISSION_CODE) {

            //If permission is granted
            if (grantResults.length > 0 && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                //Displaying a toast
                Toast.makeText(this, "Permission granted now you can read the storage",
Toast.LENGTH_LONG).show();
            } else {
                //Displaying another toast if permission is not granted
                Toast.makeText(this, "Oops you just denied the permission",
Toast.LENGTH_LONG).show();
            }
        }
    }
}

```

2.1.3 Constants.java

```

package com.example.plantvision;

/**

```

```

* Created by Eamo on 11/02/2018.
*/

public class Constants {
    public static final String UPLOAD_URL =
"http://www.c00197458.candept.com/upload.php";
    public static final String IMAGES_URL =
"http://www.c00197458.candept.com/getImages.php";
}

```

2.1.4 data.java

```

package com.example.plantvision;

import android.content.Intent;
import android.os.AsyncTask;
import android.support.annotation.NonNull;
import android.support.design.widget.BottomNavigationView;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.PopupMenu;
import android.widget.TextView;
import android.widget.Toast;

import com.example.plantvision.R;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;

import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;

public class data extends AppCompatActivity {

    //ListView listView;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_data);
        //listView = (ListView) findViewById(R.id.listView);
        // ----- Tool bar and camera button -----
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        // ----- End/ Tool bar and camera button
        -----
        getJSON("http://www.c00197458.candept.com/getData.php");
    }
}

```

```

//-----Navigation Bar functions
-----
    BottomNavigationView bottomNavigationView =
(BottomNavigationView)findViewById(R.id.bottom_navigation);
    bottomNavigationView.setOnNavigationItemSelectedListener(new
BottomNavigationView.OnNavigationItemSelectedListener() {
        @Override
        public boolean onNavigationItemSelectedListener(@NonNull MenuItem item) {
            switch (item.getItemId()){
                case R.id.action_library:
                    //Toast.makeText(data.this, "Coming soon",
Toast.LENGTH_SHORT).show();
                    Intent library = new Intent(data.this, library.class);
                    data.this.startActivity(library);
                    return true;

                case R.id.action_account:
                    Intent loginScreen = new Intent(data.this, LoginScreen.class);
                    data.this.startActivity(loginScreen);
                    finish();
                    return true;

                case R.id.action_home:
                    Intent homeScreen = new Intent(data.this,
UserAreaActivity.class);
                    data.this.startActivity(homeScreen);
                    finish();
                    return true;
            }
            return false;
        }
    });
});

//-----END-----
-----

//-----Camera button action listener-----
final FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        final PopupMenu popupMenu = new PopupMenu(data.this, fab);
        popupMenu.getMenuInflater().inflate(R.menu.popup_menu,
popupMenu.getMenu());

        //----- Popup Menu
        -----
        popupMenu.setOnMenuItemClickListener(new
PopupMenu.OnMenuItemClickListener() {
            @Override
            public boolean onMenuItemClick(MenuItem menuItem) {
                //-----
                switch (menuItem.getItemId()){
                    //----- Functionality for take photo option
                    -----
                    /*
                    case R.id.take:
                        Intent captureDiagnosis = new Intent(library.this,
capturedDiagnosis.class);

```

```

        startActivity(captureDiagnosis);
        return true;
        //----- end
    }
    /*
    //----- Functionality for diagnose photo
option -----
        case R.id.choose:
            Intent cDiagnosis = new Intent(data.this,
cameraDiagnosis.class);
            startActivity(cDiagnosis);
        }
        return false;
    }
    });
    popupMenu.show();
}
});
//----- END
}

private void getJSON(final String urlWebService) {

    class GetJSON extends AsyncTask<Void, Void, String> {

        @Override
        protected void onPreExecute() {
            super.onPreExecute();
        }

        @Override
        protected void onPostExecute(String s) {
            super.onPostExecute(s);
            //Toast.makeText(getApplicationContext(), s, Toast.LENGTH_SHORT).show();
            try {
                loadIntoListView(s);
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }

        @Override
        protected String doInBackground(Void... voids) {
            try {
                URL url = new URL(urlWebService);
                HttpURLConnection con = (HttpURLConnection) url.openConnection();
                StringBuilder sb = new StringBuilder();
                BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(con.getInputStream()));
                String json;
                while ((json = bufferedReader.readLine()) != null) {
                    sb.append(json + "\n");
                }
                return sb.toString().trim();
            } catch (Exception e) {
                return null;
            }
        }
    }
}

```



```

    }
}
GetJSON getJSON = new GetJSON();
getJSON.execute();
}

private void loadIntoListView(String json) throws JSONException {
    JSONArray jsonArray = new JSONArray(json);
    String[] heroes = new String[jsonArray.length()];
    for (int i = 0; i < jsonArray.length(); i++) {
        JSONObject obj = jsonArray.getJSONObject(i);
        heroes[i] = obj.getString("name");
    }
    //ArrayAdapter<String> arrayAdapter = new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1, heroes);
    //listView.setAdapter(arrayAdapter);
    String disease = heroes[0];
    final TextView diseaseText = (TextView) findViewById(R.id.diseaseText);
    diseaseText.setText(disease + "%");
}
}
}

```

2.1.5 library.java

```

package com.example.plantvision;
import android.content.Intent;
import android.os.AsyncTask;
import android.support.annotation.NonNull;
import android.support.design.widget.BottomNavigationView;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.support.v7.widget.Toolbar;
import android.view.MenuItem;
import android.view.View;
import android.widget.ArrayAdapter;
import android.widget.ListView;
import android.widget.PopupMenu;

import org.json.JSONArray;
import org.json.JSONException;
import org.json.JSONObject;
import java.io.BufferedReader;
import java.io.InputStreamReader;
import java.net.HttpURLConnection;
import java.net.URL;

public class library extends AppCompatActivity {

    ListView listView;

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

        listView = (ListView) findViewById(R.id.listView);
        getJSON("http://www.c00197458.candept.com/getLibrary.php");
    }
}

```

```

// ----- Tool bar and camera button -----
Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
setSupportActionBar(toolbar);
// ----- End/ Tool bar and camera button
-----

//-----Navigation Bar functions
-----
BottomNavigationView bottomNavigationView =
(BottomNavigationView)findViewById(R.id.bottom_navigation);
bottomNavigationView.setOnNavigationItemSelectedListener(new
BottomNavigationView.OnNavigationItemSelectedListener() {
    @Override
    public boolean onNavigationItemSelected(@NonNull MenuItem item) {
        switch (item.getItemId()){
            case R.id.action_library:
                //Toast.makeText(UserAreaActivity.this, "Coming soon",
Toast.LENGTH_SHORT).show();
                Intent library = new Intent(library.this, library.class);
                library.this.startActivity(library);
                return true;

            case R.id.action_account:
                Intent loginScreen = new Intent(library.this,
LoginScreen.class);
                library.this.startActivity(loginScreen);
                finish();
                return true;

            case R.id.action_home:
                Intent homeScreen = new Intent(library.this,
UserAreaActivity.class);
                library.this.startActivity(homeScreen);
                finish();
                return true;
        }
        return false;
    }
});

//-----END-----
-----

//-----Camera button action listener-----
final FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        final PopupMenu popupMenu = new PopupMenu(library.this, fab);
        popupMenu.getMenuInflater().inflate(R.menu.popup_menu,
popupMenu.getMenu());

        //----- Popup Menu
        -----
        popupMenu.setOnMenuItemClickListener(new
PopupMenu.OnMenuItemClickListener() {
            @Override
            public boolean onMenuItemClick(MenuItem menuItem) {
                //-----
                switch (menuItem.getItemId()){
                    //----- Functionality for take photo option

```

```

-----
        /*
        case R.id.take:
            Intent captureDiagnosis = new Intent(library.this,
capturedDiagnosis.class);
                startActivity(captureDiagnosis);
                return true;
        ----- end
        -----
        */
        ----- Functionality for diagnose photo
option -----
        case R.id.choose:
            Intent cDiagnosis = new Intent(library.this,
cameraDiagnosis.class);
                startActivity(cDiagnosis);
            }
            return false;
        }
    });
    popupMenu.show();
}
});
----- END
-----

}
private void getJSON(final String urlWebService) {

    class GetJSON extends AsyncTask<Void, Void, String> {

        @Override
        protected void onPreExecute() {
            super.onPreExecute();
        }

        @Override
        protected void onPostExecute(String s) {
            super.onPostExecute(s);
            //Toast.makeText(getApplicationContext(), s, Toast.LENGTH_SHORT).show();
            try {
                loadIntoListView(s);
            } catch (JSONException e) {
                e.printStackTrace();
            }
        }
        @Override
        protected String doInBackground(Void... voids) {
            try {
                URL url = new URL(urlWebService);
                HttpURLConnection con = (HttpURLConnection) url.openConnection();
                StringBuilder sb = new StringBuilder();
                BufferedReader bufferedReader = new BufferedReader(new
InputStreamReader(con.getInputStream()));
                String json;
                while ((json = bufferedReader.readLine()) != null) {
                    sb.append(json + "\n");
                }
                return sb.toString().trim();
            } catch (Exception e) {
                return null;
            }
        }
    }
}
}

```

```

        GetJSON getJSON = new GetJSON();
        getJSON.execute();
    }

    private void loadIntoListView(String json) throws JSONException {
        JSONArray jsonArray = new JSONArray(json);
        String[] heroes = new String[jsonArray.length()];
        for (int i = 0; i < jsonArray.length(); i++) {
            JSONObject obj = jsonArray.getJSONObject(i);
            heroes[i] = obj.getString("name");
        }
        ArrayAdapter<String> arrayAdapter = new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1, heroes);
        listView.setAdapter(arrayAdapter);
    }
}
}

```

2.1.6 LoginRequest.java

```

package com.example.plantvision;

import com.android.volley.Response;
import com.android.volley.toolbox.StringRequest;

import java.util.HashMap;
import java.util.Map;

public class LoginRequest extends StringRequest {
    private static final String LOGIN_REQUEST_URL =
"http://www.c00197458.candept.com/login.php";
    private Map<String, String> params;

    public LoginRequest(String username, String password, Response.Listener<String>
listener) {
        super(Method.POST, LOGIN_REQUEST_URL, listener, null);
        params = new HashMap<>();
        params.put("username", username);
        params.put("password", password);
    }

    @Override
    public Map<String, String> getParams() {
        return params;
    }
}
}

```

2.1.7 LoginScreen.java

```

package com.example.plantvision;

import android.app.AlertDialog;
import android.app.ProgressDialog;

```

```

import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.support.annotation.NonNull;
import android.support.design.widget.BottomNavigationView;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.PopupMenu;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.toolbox.Volley;

import org.json.JSONException;
import org.json.JSONObject;

public class LoginScreen extends AppCompatActivity {

    public static final String MyPREFERENCES = "MyPrefs" ;
    public static final String Name = "nameKey";
    SharedPreferences sharedPreferences;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login_screen);
        final EditText etUsername = (EditText) findViewById(R.id.inputUsername);
        final EditText etPassword = (EditText) findViewById(R.id.inputPassword);
        final Button bLogin = (Button) findViewById(R.id.loginButton);
        TextView textView = (TextView) findViewById(R.id.registerLink);

        sharedPreferences = getSharedPreferences(MyPREFERENCES, Context.MODE_PRIVATE);

        bLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                final String username = etUsername.getText().toString();
                final String password = etPassword.getText().toString();

                // Response received from the server
                Response.Listener<String> responseListener = new
                Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {

                        if (TextUtils.isEmpty(username)) {
                            AlertDialog.Builder builder = new
                            AlertDialog.Builder(LoginScreen.this);
                            builder.setMessage("Username or Password cannot be blank")
                                .setNegativeButton("Retry", null)
                                .create()
                                .show();
                        } else {
                            try {

```

```

        JSONObject jsonResponse = new JSONObject(response);
        boolean success = jsonResponse.getBoolean("success");

        if (success) {

            ProgressDialog dialog = new
ProgressDialog(LoginScreen.this); // this = YourActivity
            dialog.setProgressStyle(ProgressDialog.STYLE_SPINNER);
            dialog.setMessage("Loading. Please wait...");
            dialog.setIndeterminate(true);
            dialog.setCanceledOnTouchOutside(false);
            dialog.show();
            String name = jsonResponse.getString("name");
            Intent intent = new Intent(LoginScreen.this,
UserAreaActivity.class);

            intent.putExtra("name", name);
            intent.putExtra("username", username);

            String n = username;
            SharedPreferences.Editor editor =
sharedpreferences.edit();

            editor.putString("Name", n);
            editor.commit();
            //String login = sharedpreferences.getString("Name", "No
name defined" );

            LoginScreen.this.startActivity(intent);
            finish();
        } else {
            AlertDialog.Builder builder = new
AlertDialog.Builder(LoginScreen.this);
            builder.setMessage("Login Failed")
                .setNegativeButton("Retry", null)
                .create()
                .show();
        }

    } catch (JSONException e) {
        e.printStackTrace();
    }
}
}

});

    LoginRequest loginRequest = new LoginRequest(username, password,
responselister);
    RequestQueue queue = Volley.newRequestQueue(LoginScreen.this);
    queue.add(loginRequest);
}

});

//-----Navigation Bar functions
-----
    BottomNavigationView bottomNavigationView =
(BottomNavigationView)findViewById(R.id.bottom_navigation);
    bottomNavigationView.setOnNavigationItemSelectedListener(new
BottomNavigationView.OnNavigationItemSelectedListener() {
        @Override
        public boolean onNavigationItemSelected(@NonNull MenuItem item) {
            switch (item.getItemId()){
                case R.id.action_library:
                    Toast.makeText(LoginScreen.this, "Please login to use this
feature", Toast.LENGTH_SHORT).show();
                    return true;
            }
        }
    });

```

```

        case R.id.action_account:
            Intent loginScreen = new Intent(LoginScreen.this,
LoginScreen.class);
            LoginScreen.this.startActivity(loginScreen);
            return true;

        case R.id.action_home:
            Intent homeScreen = new Intent(LoginScreen.this,
StartScreen.class);
            LoginScreen.this.startActivity(homeScreen);
            return true;
    }
    return false;
}
});

//-----END-----
-----

//-----Register button action listener -----
textView.setOnClickListener(new View.OnClickListener() {
    public void onClick(View v) {
        Intent registerIntent = new Intent(LoginScreen.this,
registerScreen.class);
        LoginScreen.this.startActivity(registerIntent);
    }
});
//-----END-----

/*
//-----Camera button action listener-----
final FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.fab);
fab.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {

        final PopupMenu popupMenu = new PopupMenu(LoginScreen.this, fab);
        popupMenu.getMenuInflater().inflate(R.menu.popup_menu,
popupMenu.getMenu());

        //----- Popup Menu
        -----
        popupMenu.setOnMenuItemClickListener(new
PopupMenu.OnMenuItemClickListener() {
            @Override
            public boolean onMenuItemClick(MenuItem menuItem) {
                //-----
                switch (menuItem.getItemId()){
                    //----- Functionality for take photo option
                    -----
                    case R.id.take:
                        Intent captureDiagnosis = new Intent(LoginScreen.this,
capturedDiagnosis.class);
                        startActivity(captureDiagnosis);
                        return true;
                    //----- end
                    -----
                    //----- Functionality for diagnose photo
option -----

```

```

        case R.id.choose:
            Intent cDiagnosis = new Intent(LoginScreen.this,
cameraDiagnosis.class);
                startActivity(cDiagnosis);
            }
            return false;
        }
    });
    popupMenu.show();
}
});
//----- END
*/
}
}

```

2.1.8 RegisterRequest.java

```

package com.example.plantvision;

import com.android.volley.Response;
import com.android.volley.toolbox.StringRequest;

import java.util.HashMap;
import java.util.Map;

public class RegisterRequest extends StringRequest {
    private static final String REGISTER_REQUEST_URL =
"http://www.c00197458.candept.com/Register.php";
    private Map<String, String> params;

    public RegisterRequest(String email, String name, String username, String password,
Response.Listener<String> listener) {
        super(Method.POST, REGISTER_REQUEST_URL, listener, null);
        params = new HashMap<>();
        params.put("email", email);
        params.put("name", name);
        params.put("username", username);
        params.put("password", password);
    }

    @Override
    public Map<String, String> getParams() {
        return params;
    }
}

```

2.1.9 registerScreen.java

```

package com.example.plantvision;

import android.app.AlertDialog;
import android.content.Intent;
import android.support.annotation.NonNull;

```



```

import android.support.design.widget.BottomNavigationView;
import android.support.design.widget.FloatingActionButton;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.text.TextUtils;
import android.view.MenuItem;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.PopupMenu;
import android.widget.Toast;

import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.toolbox.Volley;

import org.json.JSONException;
import org.json.JSONObject;

public class registerScreen extends AppCompatActivity {

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

        final EditText etEmail = (EditText) findViewById(R.id.inputEmail);
        final EditText etName = (EditText) findViewById(R.id.inputName);
        final EditText etUsername = (EditText) findViewById(R.id.inputUsername);
        final EditText etPassword = (EditText) findViewById(R.id.inputPassword);

        final Button bRegister = (Button) findViewById(R.id.registerButton);

        bRegister.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                final String email = etEmail.getText().toString();
                final String name = etName.getText().toString();
                final String username = etUsername.getText().toString();
                final String password = etPassword.getText().toString();

                if (TextUtils.isEmpty(username)) {
                    AlertDialog.Builder builder = new
AlertDialog.Builder(registerScreen.this);
                    builder.setMessage("Please enter a Username")
                        .setNegativeButton("Retry", null)
                        .create()
                        .show();
                }

                else if (TextUtils.isEmpty(email)) {
                    AlertDialog.Builder builder = new
AlertDialog.Builder(registerScreen.this);
                    builder.setMessage("Please enter E-Mail")
                        .setNegativeButton("Retry", null)
                        .create()
                        .show();
                }

                else if (TextUtils.isEmpty(name)) {
                    AlertDialog.Builder builder = new
AlertDialog.Builder(registerScreen.this);

```

```

        builder.setMessage("Please enter Name")
            .setNegativeButton("Retry", null)
            .create()
            .show();
    }

    else if (TextUtils.isEmpty(password)) {
        AlertDialog.Builder builder = new
AlertDialog.Builder(registerScreen.this);
        builder.setMessage("Please enter Password")
            .setNegativeButton("Retry", null)
            .create()
            .show();
    }
    else {
        Response.Listener<String> responseListener = new
Response.Listener<String>() {
            @Override
            public void onResponse(String response) {
                try {
                    JSONObject jsonResponse = new JSONObject(response);
                    boolean success = jsonResponse.getBoolean("success");
                    if (success) {
                        Toast.makeText(registerScreen.this, "Account created
successfully", Toast.LENGTH_SHORT).show();
                        Intent intent = new Intent(registerScreen.this,
LoginScreen.class);
                        registerScreen.this.startActivity(intent);
                    } else {
                        AlertDialog.Builder builder = new
AlertDialog.Builder(registerScreen.this);
                        builder.setMessage("Register Failed")
                            .setNegativeButton("Retry", null)
                            .create()
                            .show();
                    }
                } catch (JSONException e) {
                    e.printStackTrace();
                }
            }
        };

        RegisterRequest registerRequest = new RegisterRequest(email, name,
username, password, responseListener);
        RequestQueue queue = Volley.newRequestQueue(registerScreen.this);
        queue.add(registerRequest);
    }
});
//-----Navigation Bar functions
-----
        BottomNavigationView bottomNavigationView =
(BottomNavigationView)findViewById(R.id.bottom_navigation);
        bottomNavigationView.setOnNavigationItemSelectedListener(new
BottomNavigationView.OnNavigationItemSelectedListener() {
            @Override
            public boolean onNavigationItemSelectedListener(@NonNull MenuItem item) {
                switch (item.getItemId()){
                    case R.id.action_library:
                        Toast.makeText(registerScreen.this, "Please log in to use this
feature", Toast.LENGTH_SHORT).show();
                        return true;

                    case R.id.action_account:

```

```

        Intent loginScreen = new Intent(registerScreen.this,
LoginScreen.class);
        registerScreen.this.startActivity(loginScreen);
        return true;

        case R.id.action_home:
        Intent homeScreen = new Intent(registerScreen.this,
StartScreen.class);
        registerScreen.this.startActivity(homeScreen);
        return true;
    }
    return false;
}
});
//-----END-----
}
}

```

2.1.10 StartScreen.java

```

package com.example.plantvision;

import android.Manifest;
import android.app.ProgressDialog;
import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.content.pm.PackageManager;
import android.location.Address;
import android.location.Geocoder;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.support.v4.app.ActivityCompat;
import android.support.v4.content.ContextCompat;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import java.util.List;
import java.util.Locale;

public class StartScreen extends AppCompatActivity implements LocationListener{

    Button getLocationBtn;
    //TextView locationText;

```

```

//TextView lat1;
//TextView lat2;
LocationManager locationManager;
public static final String MyPREFERENCES = "MyPrefs" ;
//public static final Float Lat = Float.valueOf(0);
SharedPreferences sharedPreferences;

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    this.requestWindowFeature(Window.FEATURE_NO_TITLE);
    this.getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,
WindowManager.LayoutParams.FLAG_FULLSCREEN);
    setContentView(R.layout.activity_start_screen);

    getLocationBtn = (Button)findViewById(R.id.startButton);
    //locationText = (TextView)findViewById(R.id.textView2);
    //lat1 = (TextView)findViewById(R.id.textView3);
    //lat2 = (TextView)findViewById(R.id.long);
    sharedPreferences = getSharedPreferences(MyPREFERENCES, Context.MODE_PRIVATE);

    if (ContextCompat.checkSelfPermission(getApplicationContext(),
        android.Manifest.permission.ACCESS_FINE_LOCATION) !=
PackageManager.PERMISSION_GRANTED
        && ActivityCompat.checkSelfPermission(getApplicationContext(),
        android.Manifest.permission.ACCESS_COARSE_LOCATION) !=
PackageManager.PERMISSION_GRANTED) {
        ActivityCompat.requestPermissions(this, new
String[]{android.Manifest.permission.ACCESS_FINE_LOCATION,
android.Manifest.permission.ACCESS_COARSE_LOCATION}, 101);
    }
    getLocationBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            ProgressDialog dialog = new ProgressDialog(StartScreen.this); // this =
YourActivity
            dialog.setProgressStyle(ProgressDialog.STYLE_SPINNER);
            dialog.setMessage("Loading. Please wait...");
            dialog.setIndeterminate(true);
            dialog.setCancelable(false);
            dialog.show();
            getLocation();
        }
    });
}
void getLocation() {
    try {
        locationManager = (LocationManager)
getSystemService(Context.LOCATION_SERVICE);
        locationManager.requestLocationUpdates(LocationManager.NETWORK_PROVIDER,
5000, 5, this);
    }
    catch(SecurityException e) {
        e.printStackTrace();
    }
}

@Override
public void onLocationChanged(Location location) {
    //locationText.setText("Latitude: " + location.getLatitude() + "\n Longitude: "
+ location.getLongitude());

    try {
        Geocoder geocoder = new Geocoder(this, Locale.getDefault());
        List<Address> addresses = geocoder.getFromLocation(location.getLatitude(),

```

```

location.getLongitude(), 1);
    //locationText.setText(locationText.getText() +
"\n"+addresses.get(0).getAddressLine(0)+", "+
    //    addresses.get(0).getAddressLine(1)+",
"+addresses.get(0).getAddressLine(2));

    Double latDouble = location.getLatitude();
    Double longDouble = location.getLongitude();
    Float longs =Float.valueOf(String.valueOf(longDouble));
    Float lat = Float.valueOf(String.valueOf(latDouble));

    String FullAddress = addresses.get(0).getAddressLine(0)+", "+
        addresses.get(0).getAddressLine(1)+",
"+addresses.get(0).getAddressLine(2);

    SharedPreferences.Editor editor = sharedPreferences.edit();

    editor.putFloat("Lat", lat);
    editor.putFloat("Long", longs);
    editor.putString("fulladdress", FullAddress);

    editor.commit();
    //Float lats = sharedPreferences.getFloat("Lat", 0);
    //Float longtitude = sharedPreferences.getFloat("Long", 0);
    //String rAddress = sharedPreferences.getString("fulladdress", "no name
defined");
    //String mytext =Float.toString(lats);
    //String mytext2 =Float.toString(longtitude);

    //lat1.setText(mytext);
    //lat2.setText(rAddress);
    Intent library = new Intent(StartScreen.this, LoginScreen.class);
    StartScreen.this.startActivity(library);
    finish();
}catch(Exception e)
{

}

}
@Override
public void onProviderDisabled(String provider) {
    Toast.makeText(StartScreen.this, "Please Enable GPS and Internet",
Toast.LENGTH_SHORT).show();
}

@Override
public void onStatusChanged(String provider, int status, Bundle extras) {

}

@Override
public void onProviderEnabled(String provider) {

}
}
}

```

2.1.11 UserAreaActivity.java

```

package com.example.plantvision;

import android.content.Context;
import android.content.Intent;
import android.content.SharedPreferences;
import android.location.Location;
import android.os.Bundle;
import android.provider.MediaStore;
import android.support.annotation.NonNull;
import android.support.design.widget.BottomNavigationView;
import android.support.design.widget.FloatingActionButton;
import android.support.design.widget.Snackbar;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.Toolbar;
import android.util.Log;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.PopupMenu;
import android.widget.TextView;
import android.widget.Toast;

import static com.example.plantvision.LoginScreen.MyPREFERENCES;

public class UserAreaActivity extends AppCompatActivity {

    final int REQUEST_IMAGE_CAPTURE = 1;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_user_area);
        final TextView welcomeMessage = (TextView) findViewById(R.id.welcomeMessage);
        //final TextView textView2 = (TextView) findViewById(R.id.textView2);
        //Intent intent = getIntent();
        //String name = intent.getStringExtra("name");

        SharedPreferences sharedPreferences = getSharedPreferences(MyPREFERENCES,
Context.MODE_PRIVATE);
        String login = sharedPreferences.getString("Name","No name defined" );

        // Display user details
        String message = login + ", welcome to your user area";
        welcomeMessage.setText(message);

        // ----- Tool bar and camera button -----
        Toolbar toolbar = (Toolbar) findViewById(R.id.toolbar);
        setSupportActionBar(toolbar);
        // ----- End/ Tool bar and camera button
        -----

        //-----Navigation Bar functions
        -----
        BottomNavigationView bottomNavigationView =
(BottomNavigationView)findViewById(R.id.bottom_navigation);
        bottomNavigationView.setOnNavigationItemSelectedListener(new
BottomNavigationView.OnNavigationItemSelectedListener() {
            @Override
            public boolean onNavigationItemSelected(@NonNull MenuItem item) {
                switch (item.getItemId()){
                    case R.id.action_library:
                        //Toast.makeText(UserAreaActivity.this, "Coming soon",
Toast.LENGTH_SHORT).show();

```



```
        return false;
    }
    });
    popupMenu.show();
}
});
//----- END
}
}
```


2.2 Web Application

2.2.1 login.php

```
<!DOCTYPE html>
<html lang="en" >
<head>

  <title>Plant Vision</title>
  <link rel="stylesheet" href="css/style.css">
</head>

<script>
//----- Scroll function for Sign up and Login--
$('.message a').click(function(){
  $('form').animate({height: "toggle", opacity: "toggle"}, "slow");
});

</script>

<!------- PHP
----->

<?php
//----- LOGIN
-----

if (isset($_POST['login'])){

include("config.php");
session_start();

//Gathering inputs from form
$username=$_POST['username'];
$password=$_POST['password'];
$username = strip_tags($username);
htmlspecialchars($username, ENT_QUOTES, 'UTF-8');
htmlspecialchars($password, ENT_QUOTES, 'UTF-8');

//Initial query on the table
$query = mysqli_query($db, "SELECT * FROM user WHERE username = '$username.'" and
password = '$password.'" );
$result = mysqli_query($db,"SELECT * FROM user WHERE username='$username' ");
//$sessionQuery = mysqli_query($db,"SELECT * FROM activesession WHERE sessionID
='$strForSession' ");
$row = mysqli_fetch_array($result);
//$rowSes = mysqli_fetch_array($sessionQuery);

/*
echo "<br>";
echo "Current attempts: ";
echo $rowSes['counter'];
*/
```

```

/*
if($rowSes['counter'] >= 3){
    echo "<script type='text/javascript'>alert('You are locked out')</script>";
}
else{
*/
    if ($password == $row['password']){
        //Assigning the username to a session
        $_SESSION['login_user']=$username;
        header('Location:
http://www.c00197458.candep.com/PlantVision/home.php');
    }

    else{
        echo "<script type='text/javascript'>alert('Incorrect
Login')</script>";
    }
}
// ----- END/ Login
-----

//-----Register-----
-----
if (isset($_POST['register'])){
include("config.php");
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
//Taking input from POST form
$username=$_POST['uname'];
$username = strip_tags($username);
$password=$_POST['pword'];
$name=$_POST['name'];
$email=$_POST['email'];
$password2=$_POST['pword2'];

htmlspecialchars($username, ENT_QUOTES, 'UTF-8');

$result = mysqli_query($db,"SELECT * FROM user WHERE username='$username' ");

if($result && mysqli_num_rows($result)>0){ //Checking to see if username is already
taken
    echo "<script type='text/javascript'>alert('Username is already taken, user creation
unsuccessful')</script>";
}

else{
    //Checking if password is equal to the re-entered password
    if($password==$password2){
        $pattern = "/^.*(?=.{7,})(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).*$/"; //Password
Pattern.
        $passwordresult = preg_match($pattern,$password); //Checking password
entered against pattern

        if($passwordresult == 0){ //if the password is 0 it does not meet
requirements (did not match pattern)
            echo "<script type='text/javascript'>alert('Password does not meet

```

```

requirements')</script>";
    }

    else{ //if password is ok, proceed and insert the user ot the table
        //$password = password_hash($password, PASSWORD_DEFAULT);
        $sql = "INSERT INTO user (email, name, username, password) VALUES
('$email', '$name', '$username', '$password')"; //Insert query

        if ($conn->query($sql) === TRUE){ //checking connection to the
table/db
            echo "<script type='text/javascript'>alert('User created
successfully. Please Log in.')</script>";
        }

        else{
            echo "Error: " . $sql . "<br>" . $conn->error;} //connect_error
message
        }
    }
    else{
        echo "<script type='text/javascript'>alert('Passwords do not match, user
not created')</script>"; //if the passwords did not match
    }
    $conn->close();
}
}
//----- END/Register
-----
?>

<!------- /PHP
----->

<!------- HTML
----->

<!-- Page BODY -->
<body>
<div class="login-page">
<div class="form">
    <!-- Registration -->
    <form name = "register" id = "register" method ="post" class="register-form" >
        <input name = "name" type="text" placeholder="Name" required />
        <input name = "uname" type="text" placeholder="Username" required />

        <input name = "email" type="email" placeholder="E-mail address" />
        <input name = "pword" type="password" placeholder="Enter your password"
pattern="(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).{8,}"
        title = "Must contain at least one number and one uppercase and lowercase
letter, and at least 8 or more characters" required /> <!-- Pattern for password, but
aslo validated in PHP -->

        <input name = "pword2" type="password" placeholder="Re-enter your password" />
        <button name = "register" type = "submit" >create</button>
        <!-- Login message -->
        <p class="message">Already registered? <a href="#">Sign In</a></p>
    </form>

    <!-- Log in -->

```

```

    <form method="POST" class="login-form" >
        
        <input type="text" name = "username" placeholder="Username" required />
        <input id = "password" type="password" name = "password" placeholder="Password"
pattern="(?!.*\d)(?!.*[a-z])(?!.*[A-Z]).{8,}"
        title = "Must contain at least one number and one uppercase and lowercase
letter, and at least 8 or more characters" required />
        <button name = "login" type = "submit">login</button>
        <!-- Register message -->
        <p class="message">Not registered? <a href="#">Create an account</a></p>
    </form>

</div>
</div>

    <!--Facilitates the scroll for Registration form -->
    <script
src='http://cdnjs.cloudflare.com/ajax/libs/jquery/2.1.3/jquery.min.js'></script>
    <!--Facilitates the JavaScript file -->
    <script type="text/javascript" src="js/index.js"></script>

</body>
</html>

<!------- /HTML
----->

```

2.2.2 home.php

```

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
    <link rel="stylesheet" href="css/style.css">
<title>Plant Vision</title>
</head>
<body>

<!------- PHP
----->

    <?php

include 'config.php';
//Creating active session
session_start();

if(!isset($_SESSION['login_user'])){
    header("Location:login.php");
}

if (isset($_POST['analyze'])){
    header("Location:manageIndex.php");
}

```

```

if (isset($_POST['report'])){
    header("Location:reportIndex.php");
}

if (isset($_POST['change'])){
    header("Location:change.php");
}

?>
<!------- /PHP
----->

<!------- HTML
----->
<br><br><br>
<div class="form">
    <p align = "left" class="message">Welcome<b><font style="color:black;" size="4">
    <?php
    //login session is the user that is currently active in the session
    $login_session=$_SESSION['login_user'];
    echo $login_session;
    ?>
    

    <!-- Option Menu -->
    </font></b>
    </p><br>
    <!-- Analyze Option -->
    <form method="post">
    <button type = "submit" name = "analyze" >Manage Leaves</button><br><br>
    </form>
    <!-- Reporting Option -->
    <form method="post">
    <button type = "submit" name = "report" >Analyze Plant</button><br><br>
    </form>

    <!-- Logout message -->
    <p class="message">Finished? <a href="logout.php">Logout</a></p>

    <!-- Logout message -->
    <p class="message">Need to change your password? <a href="change.php">Change
    Password</a></p>
    </form>

</div>
</body>
</html>
<!------- /HTML
----->

```

2.2.3 logout.php

```
<?php
session_start();
if(session_destroy())
{
header("Location: login.php");
}
?>
```

2.2.4 groupSelection.php

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>Website</title>
  <link rel="stylesheet" href="css/style.css">
</head>
<style>
select#soflow, select#soflow-color {
  -webkit-appearance: button;
  -webkit-border-radius: 2px;
  -webkit-box-shadow: 0px 1px 3px rgba(0, 0, 0, 0.1);
  -webkit-padding-end: 20px;
  -webkit-padding-start: 2px;
  -webkit-user-select: none;
  background-image: url(http://i62.tinypic.com/15xvbd5.png),
  -webkit-linear-gradient(#FAFAFA, #F4F4F4 40%, #E5E5E5);
  background-position: 97% center;
  background-repeat: no-repeat;
  border: 1px solid #AAA;
  color: #555;
  font-size: inherit;
  margin: 20px;
  overflow: hidden;
  padding: 5px 10px;
  text-overflow: ellipsis;
  white-space: nowrap;
  width: 300px;
}
div.polaroid {
  height: 75%;
  width: 100%;
  background-color: white;
  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px 20px 0 rgba(0, 0, 0, 0.19);
  margin-bottom: 25px;
}
div.container {
```

```

    text-align: center;
    padding: 5px 5px;
}
</style>

<?php
include 'config.php';
session_start();
$plant = $_SESSION['plant'];
$imageId = $_SESSION['imageId'];

if(!isset($_SESSION['login_user'])){
    header("Location:login.php");
}

$username = $_SESSION['login_user'];
$result = mysqli_query($db,"SELECT * FROM user WHERE username = '$username'");
$row = mysqli_fetch_array($result);
$user_id = $row[0];

//Connect to our MySQL database using the PDO extension.
$pdo = new PDO('mysql:host=mysql3792.cp.blacknight.com;dbname=db1480761_plantvision',
'u1480761_Eamonn', 'Fabregas@2019');
//Our select statement. This will retrieve the data that we want.
$sql = "SELECT distinct plant_group FROM db_images WHERE user_id = '$user_id'";
//Prepare the select statement.
$stmt = $pdo->prepare($sql);
//Execute the statement.
$stmt->execute();
//Retrieve the rows using fetchAll.
$plants = $stmt->fetchAll();

if (isset($_POST['analyze'])){
    $group = $_POST["group"];
    //$plantN = $_POST["plantN"];
    $imageId = $_SESSION['imageId'];

    $query = mysqli_query($db, "UPDATE db_images SET plant_group = '$group' WHERE id
= '$imageId'");
    //$query = mysqli_query($db, "UPDATE db_images SET name = '$plantN' WHERE id =
'$imageId'");
    echo "<script type='text/javascript'>alert('Details successfully
changed')</script>";
    header('Location: http://www.c00197458.candept.com/PlantVision/manageIndex.php');
}

$username = $_SESSION['login_user'];
$result = mysqli_query($db,"SELECT * FROM user WHERE username = '$username'");
$row = mysqli_fetch_array($result);
$user_id = $row[0];

$result = mysqli_query($db,"SELECT * FROM db_images WHERE name = '$plant' AND user_id =
'$user_id'");
$row = mysqli_fetch_array($result);
$image = $row['url'];
$_SESSION['imageId'] = $row['id'];
?>
<!-- -----HTML----- -->
<body>

    <div class="form">

        <form method="POST" class="login-form" >
            <input type="text" name = "plantN" placeholder = "Plant name: <?php echo

```

```

$row['name']?>"disabled />
    <select id = "soflow" name="group" >
        <?php foreach($plants as $plant): ?>
            <option value="<?=$plant['plant_group']; ?>"><?=$plant['plant_group'];
?></option>
        <?php endforeach; ?>
    </select> <br><br>
    <input type="text" placeholder= "Time taken: <?php echo $row['datetime']?> "
disabled />

    <div class="polaroid">
    <img src= "<?php echo $image?>" alt="" style="width:100%">
    <div class="container">
    <p><?php echo $row['name']?>, <?php echo $row['plant_group']?></p>
    </div>
    </div>
    <button name = "analyze" type = "submit">Confirm</button>
    <p class="message">Changed your mind? <a href="chooseImage.php"> Go back</a></p>
    </form>
</div>
</body>
</html>

```

2.2.5 manageIndex.php

```

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
    <link rel="stylesheet" href="css/style.css">
<title>Plant Vision</title>
</head>
<body>
<br><br><br><br><br><br><br><br><br><br><br><br><br><br>

<style>
select#soflow, select#soflow-color {
    -webkit-appearance: button;
    -webkit-border-radius: 2px;
    -webkit-box-shadow: 0px 1px 3px rgba(0, 0, 0, 0.1);
    -webkit-padding-end: 20px;
    -webkit-padding-start: 2px;
    -webkit-user-select: none;
    background-image: url(http://i62.tinypic.com/15xvbd5.png),
-webkit-linear-gradient(#FAFAFA, #F4F4F4 40%, #E5E5E5);
    background-position: 97% center;
    background-repeat: no-repeat;
    border: 1px solid #AAA;
    color: #555;
    font-size: inherit;
    margin: 20px;
    overflow: hidden;
    padding: 5px 10px;
    text-overflow: ellipsis;
    white-space: nowrap;
    width: 300px;
}

#home{

```



```

        position: absolute;
        top: 3%;
        left: 2%;
    }
</style>
<?php
include 'config.php';
session_start();
if(!isset($_SESSION['login_user'])){
    header("Location:login.php");
}

if (isset($_POST['analyze'])){
    $option = $_POST['plant'];
    $_SESSION['group'] = $option;
    header("Location:chooseImage.php");
}

$username = $_SESSION['login_user'];
$result = mysqli_query($db, "SELECT * FROM user WHERE username= '$username'");
$row = mysqli_fetch_array($result);
$id = $row[0];

//Connect to our MySQL database using the PDO extension.
$pdo = new PDO('mysql:host=mysql13792.cp.blacknight.com;dbname=db1480761_plantvision',
'u1480761_Eamonn', 'Fabregas@2019');
//Our select statement. This will retrieve the data that we want.
$sql = "SELECT DISTINCT plant_group FROM db_images WHERE user_id = '$id'";
//Prepare the select statement.
$stmt = $pdo->prepare($sql);
//Execute the statement.
$stmt->execute();
//Retrieve the rows using fetchAll.
$plants = $stmt->fetchAll();

?>

<div id="home">
<a href = "home.php">

</a>
</div>
<form method="post" name = "plant">
<div class="form">
<font size = "2"> Please select <b> Group </b> to which Leaf belongs </font>
<select id = "soflow" name="plant" >
<?php foreach($plants as $plant): ?>
<option value="<?=$plant['plant_group']; ?>"><?=$plant['plant_group']; ?></option>
<?php endforeach; ?>
</select> <br><br>
<button type = "submit" name = "analyze" >Analyze</button><br><br>
</form>

<!-- Logout message -->
<p class="message">Something else? <a href="home.php">Go back</a></p>

</div>
</body>
</html>

```

2.2.6 mark.php

```
<?php
include 'config.php';
session_start();
$id = $_GET['id'];
echo $id;

$dbname = "db1480761_plantvision";
$conn = mysqli_connect("mysql3792.cp.blacknight.com", "u1480761_Eamonn",
"Fabregas@2019", $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
$result = mysqli_query($db,"SELECT * FROM imageResult WHERE id = '$id'");
$row = mysqli_fetch_array($result);
$imageI = $row[1];

$mark = "Diseased";
$result = mysqli_query($db,"SELECT * FROM imageResult WHERE imageID = '$imageI' AND
diseaseM = '$mark'");
$row = mysqli_fetch_array($result);
$queryR = $row[5];
>ID2 = $row[0];
echo "<br>" . $queryR;
echo "<br>" . $ID2;
if($queryR == "Diseased"){
    $mark = "";
    $result = mysqli_query($db,"UPDATE imageResult SET diseaseM = '$mark' WHERE id =
>ID2");
}
$mark = "Diseased";
$result = mysqli_query($db,"UPDATE imageResult SET diseaseM = '$mark' WHERE id = $id");

//Insert into markedtable 'mark'
$query = mysqli_query($db,"SELECT * FROM db_images WHERE id = '$imageI'");
$groupA = mysqli_fetch_array($query);
$group = $groupA[4];
echo "<br>" . $group;
$user = $_SESSION['login_user'];
$userQuery = mysqli_query($db,"SELECT * FROM user WHERE username = '$user'");
$userA = mysqli_fetch_array($userQuery);
$user = $userA[0];
echo "<br>" . $user;
$diseaseQuery = mysqli_query($db,"SELECT * FROM imageResult WHERE id = $id");
$diseaseA = mysqli_fetch_array($diseaseQuery);
$colourCode = $diseaseA[2];
echo "<br>" . $colourCode;

$delete = mysqli_query($db,"DELETE FROM markedtable WHERE user_id = $user AND
plant_group = '$group' AND mark = 'Marked'");

$insert = mysqli_query($db,"INSERT INTO markedtable
(user_id, plant_group, colourCode, mark) VALUES ($user, '$group', $colourCode
,'Marked')");

header('Location: editor.php');
?>
```

2.2.7 report.php

```
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
<link rel="stylesheet" href="css/style.css">
<link rel="stylesheet" href="css/circle.css">
<script type="text/javascript" src="fusioncharts/js/fusioncharts.js"></script>
<script type="text/javascript"
src="fusioncharts/js/themes/fusioncharts.theme.ocean.js"></script>
<title>Plant Vision</title>
</head>
<style>

.div {
    width: 1400px;
    padding: 2px;
    height: 680px;
    border: 2px solid gray;
    background: white;
    position: absolute;
    top: 3%;
    left: 4%;
}

#home{
    position: absolute;
    top: 3%;
    left: 2%;
}

#homeText{
    position: absolute;
    top: 11%;
    left: 2.5%;
}

#CSVText{
    position: absolute;
    top: 24%;
    left: 1%;
}

#plantText{
    position: absolute;
    top: 36%;
    left: 0.2%;
}

#csv{
    position: absolute;
    top: 15%;
    left: 2%;
}

#plant{
    position: absolute;
    top: 28%;
}
```

```

    left: 2%;
}

#title{
    position:absolute;
    top: 8%;
    left: 20%;
}

#displayP{
    position:absolute;
    top: 10%;
    left: 13%;
}

#chart-1{
    position:absolute;
    top: 3%;
    left: 54%;
}

#chart2{
    position:absolute;
    top: 50%;
    left: 55%;
}

#chart-3{
    position:absolute;
    top: 55%;
    left: 9%;
}

</style>

<?php
include 'config.php';
session_start();
$group = $_SESSION['group'];
$_SESSION['group'] = $group;
//$group = "Plant Group A";
//echo $group;
$username = $_SESSION['login_user'];
if(!isset($_SESSION['login_user'])){
    header("Location:login.php");
}
include 'fusioncharts.php';

$hostdb = "mysql13792.cp.blacknight.com"; // MySQL host
$userdb = "u1480761_Eamonn"; // MySQL username
$passdb = "Fabregas@2019"; // MySQL password
$namebdb = "db1480761_plantvision"; // MySQL database name

// Establish a connection to the database
$dbhandle = new mysqli($hostdb, $userdb, $passdb, $namebdb);

$result = mysqli_query($db,"SELECT * FROM user WHERE username = '$username'");
$row = mysqli_fetch_array($result);
$user_id = $row[0];
$_SESSION['user_id'] = $user_id;
//echo $user_id;

```

```

// sql to delete a record
$sql = "DELETE FROM report WHERE user_id = '$user_id'";

if ($db->query($sql) === TRUE) {
    //echo "Record deleted successfully";
} else {
    echo "Error deleting record: " . $db->error;
}

$result = mysqli_query($db,"SELECT * FROM db_images WHERE plant_group = '$group' AND
user_id = '$user_id'");
$i = 0;
$counter = 0;
while ($row = mysqli_fetch_array($result)) {
    $id[$i] = $row[0];
    $name[$i] = $row[2];
    $time[$i] = $row[5];
    $i++;
    $counter++;
}
$i = 0;
while ($i < $counter){
    $result = mysqli_query($db,"SELECT * FROM imageResult WHERE diseaseM = 'Diseased'
AND user_id = '$user_id' AND imageID = '$id[$i]'");
    $row = mysqli_fetch_array($result);
    $diseasePer[$i] = $row[3];
    //echo "<br>" . $diseasePer[$i];
    $i++;
}

$i = 0;
while($i < $counter){
    $sql = "INSERT INTO report (user_id, leaf_name, plant_group, disease_per, time)
VALUES ('$user_id', '$name[$i]', '$group', '$diseasePer[$i]', '$time[$i]')";

    if($diseasePer[$i] == "")
    {
        echo "<script type='text/javascript'>alert('Please note:
Some samples were not assigned a disease and have been left out from the
report.')

```

```

$result = $dbh->query($strQuery) or exit("Error code ({ $dbh->errno }):
{ $dbh->error}");

// If the query returns a valid response, prepare the JSON string
if ($result) {
    // The `$arrData` array holds the chart attributes and data
    $arrData = array(
        "chart" => array(
            "caption" => "Disease % of " . $group,
            "paletteColors" => "#0075c2",
            "bgColor" => "#ffffff",
            "borderAlpha" => "20",
            "canvasBorderAlpha" => "0",
            "usePlotGradientColor" => "0",
            "plotBorderAlpha" => "10",
            "palettecolors" => "FF5904,0372AB,FF0000,3fe1a5, a7e813, bb18af,
4a21e9",
            "xAxisName" => "Title of Leaf",
            "yAxisName" => "Disease (%)",
            "showXAxisLine" => "1",
            "xAxisLineColor" => "#999999",
            "showValues" => "0",
            "divlineColor" => "#999999",
            "divLineIsDashed" => "1",
            "showAlternateHGridColor" => "0"
        )
    );

    $arrData["data"] = array();

    // Push the data into the array
    while($row = mysqli_fetch_array($result)) {
        array_push($arrData["data"], array(
            "label" => $row["leaf_name"],
            "value" => $row["disease_per"]
        )
    );
    }

    /*JSON Encode the data to retrieve the string containing the JSON representation of
the data in the array. */
    $jsonEncodedData = json_encode($arrData);
    $columnChart = new FusionCharts("column3d", "myFirstChart" , 600, 300, "chart-1",
"json", $jsonEncodedData);
    // Render the chart
    $columnChart->render();
}

// Form the SQL query that returns the top 10 most populous countries
$strQuery = "SELECT * FROM report WHERE plant_group = '$group' AND user_id =
'$user_id'";
// Execute the query, or else return the error message.
$result = $dbh->query($strQuery) or exit("Error code ({ $dbh->errno }):
{ $dbh->error}");

// If the query returns a valid response, prepare the JSON string
if ($result) {
    // The `$arrData` array holds the chart attributes and data
    $arrData2 = array(
        "chart" => array(
            "caption" => "Disease % of " . $group. " over time",
            "paletteColors" => "#0075c2",
            "bgColor" => "#ffffff",

```

```

        "borderAlpha"=> "20",
        "canvasBorderAlpha"=> "0",
        "usePlotGradientColor"=> "0",
        "plotBorderAlpha"=> "10",
        "xAxisName" => "Time Period",
        "palettecolors" => "3fe1a5, a7e813, bb18af, 4a21e9",
        "yAxisName" => "Disease (%)",
        "showXAxisLine"=> "1",
        "xAxisLineColor" => "#999999",
        "showValues" => "0",
        "divlineColor" => "#999999",
        "divLineIsDashed" => "1",
        "showAlternateHGridColor" => "0"
    )
);

$arrData2["data"] = array();

// Push the data into the array
while($row = mysqli_fetch_array($result)) {
    array_push($arrData2["data"], array(
        "label" => $row["time"],
        "value" => $row["disease_per"]
    )
);
}
/*JSON Encode the data to retrieve the string containing the JSON representation of
the data in the array. */
$jsonEncodedData2 = json_encode($arrData2);
$columnChart2 = new FusionCharts("line", "myFirstChart2" , 600, 300, "chart2",
"json", $jsonEncodedData2);
// Render the chart
$columnChart2->render();
}

// Form the SQL query that returns the top 10 most populous countries
$strQuery = "SELECT * FROM report WHERE plant_group = '$group' AND user_id =
'$user_id'";
// Execute the query, or else return the error message.
$result = $dbhandle->query($strQuery) or exit("Error code ({$dbhandle->errno}):
{$dbhandle->error}");

// If the query returns a valid response, prepare the JSON string
if ($result) {
    // The `arrData` array holds the chart attributes and data
    $arrData3 = array(
        "chart" => array(
            "caption" => "Disease % of " . $group,
            "paletteColors" => "#0075c2",
            "bgColor" => "#ffffff",
            "borderAlpha"=> "20",
            "canvasBorderAlpha"=> "0",
            "usePlotGradientColor"=> "0",
            "plotBorderAlpha"=> "10",
            "palettecolors" => "FF5904,0372AB,FF0000,3fe1a5, a7e813, bb18af, 4a21e9",
            "xAxisName" => "Title of Leaf",
            "yAxisName" => "Disease (%)",
            "showXAxisLine"=> "1",
            "xAxisLineColor" => "#999999",
            "showValues" => "0",
            "divlineColor" => "#999999",
            "divLineIsDashed" => "1",
            "showAlternateHGridColor" => "0"
        )
    );
}

```

```

    )
);

$arrData3["data"] = array();

// Push the data into the array
while($row = mysqli_fetch_array($result)) {
    array_push($arrData3["data"], array(
        "label" => $row["leaf_name"],
        "value" => $row["disease_per"]
    )
);
}

/*JSON Encode the data to retrieve the string containing the JSON representation of
the data in the array. */
$jsonEncodedData3 = json_encode($arrData3);
$columnChart3 = new FusionCharts("pie3d", "myFirstChart3" , 500, 300, "chart-3",
"json", $jsonEncodedData3);
// Render the chart
$columnChart3->render();
// Close the database connection
$dbhandle->close();
}
?>
<body>
<div class = "div">
<div id="title">
    <b><font color = "black" size = "3">Highest value of Disease %</font></b>
</div>
<div id = "displayP">
<div class="c100 p10 orange big">
    <span><?php echo max($diseasePer);?></span>
    <div class="slice">
        <div class="bar"></div>
        <div class="fill"></div>
    </div>
</div>
</div>
</div>
<div id="homeText">
<b><font size = "2">Home
</font>
</b>
</div>
<div id="CSVText">
<b><font size = "2">Download Data
</font>
</b>
</div>
<div id="plantText">
<b><font size = "2">Choose Anoter Plant
</font>
</b>
</div>
<div id="home">
<a href = "home.php">

</a>
</div>
<div id="csv">
<a href = "csv.php">

</a>
</div>
<div id="plant">
<a href = "reportIndex.php">

```



```


</a>
</div>
<div id="chart-1"><!-- Fusion Charts will render here--></div>
<div id="chart2"><!-- Fusion Charts will render here--></div>
<div id="chart-3"><!-- Fusion Charts will render here--></div>
</div>
</body>
</html>

```

2.2.8 reportIndex.php

```

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
  <link rel="stylesheet" href="css/style.css">
<title>Plant Vision</title>
</head>
<body>
<br><br><br><br><br><br><br><br><br><br><br>

<style>
select#soflow, select#soflow-color {
  -webkit-appearance: button;
  -webkit-border-radius: 2px;
  -webkit-box-shadow: 0px 1px 3px rgba(0, 0, 0, 0.1);
  -webkit-padding-end: 20px;
  -webkit-padding-start: 2px;
  -webkit-user-select: none;
  background-image: url(http://i62.tinypic.com/15xvbd5.png),
  -webkit-linear-gradient(#FAFAFA, #F4F4F4 40%, #E5E5E5);
  background-position: 97% center;
  background-repeat: no-repeat;
  border: 1px solid #AAA;
  color: #555;
  font-size: inherit;
  margin: 20px;
  overflow: hidden;
  padding: 5px 10px;
  text-overflow: ellipsis;
  white-space: nowrap;
  width: 300px;
}
</style>

<?php
include 'config.php';
session_start();
if(!isset($_SESSION['login_user'])){
  header("Location:login.php");
}

if (isset($_POST['analyze'])){
  $option = $_POST['plant'];
  $_SESSION['group'] = $option;
  header("Location:report.php");
}

```

```

$username = $_SESSION['login_user'];
$result = mysqli_query($db, "SELECT * FROM user WHERE username= '$username'");
$row = mysqli_fetch_array($result);
$id = $row[0];
//Connect to our MySQL database using the PDO extension.
$pdo = new PDO('mysql:host=mysql3792.cp.blacknight.com;dbname=db1480761_plantvision',
'u1480761_Eamonn', 'Fabregas@2019');
//Our select statement. This will retrieve the data that we want.
$sql = "SELECT DISTINCT plant_group FROM db_images WHERE user_id = '$id'";
//Prepare the select statement.
$stmt = $pdo->prepare($sql);
//Execute the statement.
$stmt->execute();
//Retrieve the rows using fetchAll.
$plants = $stmt->fetchAll();

?>

<form method="post" name = "plant">
<div class="form">
<font size = "2"> Please select <b> Group </b> to which Leaf belongs </font>
<select id = "soflow" name="plant" >
<?php foreach($plants as $plant): ?>
<option value="<?=$plant['plant_group']; ?>"><?=$plant['plant_group']; ?></option>
<?php endforeach; ?>
</select> <br><br>
<button type = "submit" name = "analyze" >Analyze</button><br><br>
</form>

<!-- Logout message -->
<p class="message">Something else? <a href="home.php">Go back</a></p>
</form>

</div>
</body>
</html>

```

2.2.9 weatherD.php

```

<!DOCTYPE html>
<html lang="en" >
<head>
<meta charset="UTF-8">
<title>Plant Vision</title>
<link rel="stylesheet" href="css/style.css">
</head>
<?php
include 'config.php';
session_start();
//$plant = $_SESSION['plant'];
if(!isset($_SESSION['login_user'])){
header("Location:login.php");
}
$imageId = $_SESSION['imageId'];
$result = mysqli_query($db,"SELECT * FROM weather WHERE imageId = '$imageId'");
$row = mysqli_fetch_array($result);
$image = $row['icon'];
$temp = $row['temp'];
$hum = $row['hum'];
$wind = $row['wind'];

```

```

$pressure = $row['pressure'];
$cloud = $row['cloud'];
$time = $row['time'];
$desc = $row['description'];

$result2 = mysqli_query($db,"SELECT * FROM gpsLocation WHERE imageId = '$imageId'");
$row2 = mysqli_fetch_array($result2);
$address = $row2['address'];
$long = $row2['longitude'];
$lat = $row2['latitude'];

?>
<style>
.div {
    width: 830px;
    padding: 2px;
    height: 470px;
    border: 2px solid gray;
    background: white;
    position:absolute;
    top:0;
    bottom: 0;
    left: 0;
    right: 0;
    margin: auto;
}

.mapContainer{
    position:absolute;
    top:200px;
    left: 300px;
    background: black;
}
</style>
<!-------HTML----->
<body>

    <div class="form">
        <form method="POST" class="login-form" >
            <img src = "<?php echo $image?>" alt="" style="width:70% height:70%">
                <h1 align = "left"><u><font color = "darkgreen" >Weather
Conditions</font></u></h1>
                <h3 align = "left">Description: <font size="5" color = "grey" ><?php echo
$desc;?></font></h3>
                <h3 align = "left">Time taken: <font size="5" color = "grey" ><?php echo
$time;?></font></h3>
                <h3 align = "left">Temperature: <font size="5" color = "grey" ><?php echo
$temp;?></font><font size="3" color = "grey"> &deg;C</font></h3>
                <h3 align = "left">Humidity: <font size="5" color = "grey" ><?php echo
$hum;?></font><font size="3" color = "grey"> %</font></h3>
                <h3 align = "left">Wind: <font size="5" color = "grey" ><?php echo
$wind;?></font><font size="3" color = "grey"> km/h</font></h3>
                <h3 align = "left">Cloud coverage: <font size="5" color = "grey" ><?php echo
$cloud;?></font><font size="3" color = "grey"> %</font></h3>
                <h1 align = "left"><u><font color = "darkgreen" >Image
Location</font></u></h1>
                <h3 align = "left">Full Address: <font size="5" color = "grey" ><?php echo
$address;?></font></h3>
                <div id="map" style="width:400px;height:400px;background:yellow"></div>
                <p class="message">All good?<a href="analyze.php"> Go back</a></p>
            </form>
            <script>
function myMap() {
    var long = "<?php echo $long ?>";
    var lat = "<?php echo $lat ?>";

```

```

var myLatLng = {lat: 53.2, lng: 64.3};

var map = new google.maps.Map(document.getElementById('map'), {
  zoom: 15,
  center: new google.maps.LatLng(lat, long),
  mapTypeId: google.maps.MapTypeId.HYBRID,
});

var marker = new google.maps.Marker({
  position: new google.maps.LatLng(lat, long),
  map: map,
  title: 'Plant location!'
});
}
</script>
</div>
<script
src="https://maps.googleapis.com/maps/api/js?key=AIzaSyAnqWwhEdWXYP0nzbULQvXyfhjP6KvmWzQ
&callback=myMap"></script>
</body>
</html>

```

2.2.10 analyze.php

```

<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>Plant Vision</title>
  <link rel="stylesheet" href="css/style.css">
</head>

<style>
div.polaroid {
  height: 75%;
  width: 100%;
  background-color: white;
  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px 20px 0 rgba(0, 0, 0, 0.19);
  margin-bottom: 25px;
}

div.container {
  text-align: center;
  padding: 5px 5px;
}

#home{
  position: absolute;
  top: 3%;
  left: 2%;
}
</style>
<?php
include 'config.php';
session_start();
$plant = $_SESSION['plant'];

```

```

if(!isset($_SESSION['login_user'])){
    header("Location:login.php");
}

if (isset($_POST['analyze'])){
    header("Location:editor.php");
}

$username = $_SESSION['login_user'];
$result = mysqli_query($db,"SELECT * FROM user WHERE username = '$username'");
$row = mysqli_fetch_array($result);
$user_id = $row[0];

$result = mysqli_query($db,"SELECT * FROM db_images WHERE name = '$plant' AND user_id = '$user_id'");
$row = mysqli_fetch_array($result);
$image = $row['url'];
$_SESSION['imageId'] = $row['id'];
?>
<!-- -----HTML----->
<body>

<div id="home">
<a href = "home.php">

</a>
</div>
    <div class="form">

        <form method="POST" class="login-form" >
            <p align = "right" class="message"><a href="edit.php">Edit Plant Name</a></p>
            <input type="text" placeholder = "Plant name: <?php echo $row['name']?>"disabled
/>
                <p align = "right" class="message"><a href="edit.php">Edit Plant
Group</a></p>
            <input type="text" placeholder= "Plant group: <?php echo $row['plant_group']?> "
disabled />
            <input type="text" placeholder= "Time taken: <?php echo $row['datetime']?> "
disabled />

            <div class="polaroid">
                <img src = "<?php echo $image?>" alt="" style="width:100%">
                <div class="container">
                    <p><?php echo $row['name']?>, <?php echo $row['plant_group']?></p>
                </div>
                </div>
                <button name = "analyze" type = "submit">Analyze</button>
                <p class="message">Need more details about the plant? <a href="weatherD.php">Check
weather and location</a></p>
                <p class="message">Not the correct plant? <a href="chooseImage.php">Choose
another</a></p>
            </form>
        </div>
</body>
</html>

```

2.2.11 change.php

```

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">

```

```

<head>
<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
    <link rel="stylesheet" href="css/style.css">
<title>Plant Vision</title>
</head>
<body>
<!------- PHP
----->

    <?php
    //Checking Session----
    session_start();
    //Checking to see if user is authenticated to access pages
    if(!isset($_SESSION['login_user'])){
        header("Location:login.php");
    }

    if (isset($_POST['changeP'])){
        include("config.php");
        // Create connection
        $conn = new mysqli($servername, $username, $password, $dbname);
        // Check connection
        if ($conn->connect_error){
            die("Connection failed: " . $conn->connect_error);
        }
        //Grabbing variables from form via POST
        $user=$_SESSION['login_user'];
        $oPassword=$_POST['oPWord'];
        $nPassword=$_POST['nPWord'];
        $nPassword1=$_POST['nPWord1'];
        $result = mysqli_query($conn,"SELECT * FROM user WHERE username='$user' ");
        $row = mysqli_fetch_array($result);

        //checking old password in the database to ensure it is correct
        if($oPassword == $row['password']){
            //checking new password against re-entered password
            if($nPassword==$nPassword1){
                $pattern = "/^(?=.*{7,})(?=.*\d)(?=.*[a-z])(?=.*[A-Z]).*$/";
                //Password Pattern.
                $passwordresult = preg_match($pattern,$nPassword); //Checking
                password against pattern

                if($passwordresult == 0){ //if the password is 0 it does not
                meet requirements/did not match pattern
                    echo "<script type='text/javascript'>alert('Password does
                    not meet requirements')</script>";
                }

                else{ //if ok, go ahead and insert to table the new user
                    //$nPassword = password_hash($nPassword, PASSWORD_DEFAULT);
                    $sql = "UPDATE user SET password = '$nPassword' WHERE
                    username='$user'"; //Update sql statement
                    if ($conn->query($sql) === TRUE){ //checking connection to
                    the table/db
                        echo "<script type='text/javascript'>alert('Password
                        successfully changed. Please log in using your new credentials.')</script>";
                        session_destroy();
                        echo "<script language='javascript'
                        type='text/javascript'>
                        location.href='http://www.c00197458.candept.com/PlantVision/login.php' </script>";
                    }

                    else{

```

```

                                echo "Error: " . $sql . "<br>" . $conn->error;}
                            }
                        }
                    }
                }
            }
        }
    }
}
?>
<!------- /PHP
----->
<!------- HTML
----->
----->
<br><br><br>
<div class="form">

    <!-- Uploading Option -->
    <form method="post">
        <input name = "oPWord" type="password" placeholder="Old Password" required />

        <input name = "nPWord" type="password" placeholder="New Password"
pattern="(?!.*\d)(?!.*[a-z])(?!.*[A-Z]).{8,}"
        title = "Must contain at least one number and one uppercase and lowercase
letter, and at least 8 or more characters" required /> <!-- Pattern for password, but
aslo validated in PHP -->

        <input name = "nPWord1" type="password" placeholder="Re-enter New Password"
pattern="(?!.*\d)(?!.*[a-z])(?!.*[A-Z]).{8,}"
        title = "Must contain at least one number and one uppercase and lowercase
letter, and at least 8 or more characters" required /> <!-- Pattern for password, but
aslo validated in PHP -->

        <button name = "changeP" type = "submit" >Change Password</button>
        <p class="message">Remembered it? <a href="home.php">Go Back</a></p>
    </form>

</div>
</body>
</html>
<!------- /HTML
----->

```

2.2.12 chooseImage.php

```

<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml">
<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
    <link rel="stylesheet" href="css/style.css">

```

```

<title>Plant Vision</title>
</head>
<body>
<br><br><br><br><br><br><br><br><br><br><br>

<style>

select#soflow, select#soflow-color {
  -webkit-appearance: button;
  -webkit-border-radius: 2px;
  -webkit-box-shadow: 0px 1px 3px rgba(0, 0, 0, 0.1);
  -webkit-padding-end: 20px;
  -webkit-padding-start: 2px;
  -webkit-user-select: none;
  background-image: url(http://i62.tinypic.com/15xvbd5.png),
  -webkit-linear-gradient(#FAFAFA, #F4F4F4 40%, #E5E5E5);
  background-position: 97% center;
  background-repeat: no-repeat;
  border: 1px solid #AAA;
  color: #555;
  font-size: inherit;
  margin: 20px;
  overflow: hidden;
  padding: 5px 10px;
  text-overflow: ellipsis;
  white-space: nowrap;
  width: 300px;
}

#home{
  position:absolute;
  top: 3%;
  left: 2%;
}

</style>
<?php
include 'config.php';
session_start();
$group = $_SESSION['group'];
if(!isset($_SESSION['login_user'])){
  header("Location:login.php");
}

if (isset($_POST['analyze'])){
  $option = $_POST['plant'];
  $_SESSION['plant'] = $option;
  header("Location:analyze.php");
}

$username = $_SESSION['login_user'];
$result = mysqli_query($db,"SELECT * FROM user WHERE username= '$username'");
$row = mysqli_fetch_array($result);
$id = $row['user_id'];

//Connect to our MySQL database using the PDO extension.
$pdo = new PDO('mysql:host=mysql3792.cp.blacknight.com;dbname=db1480761_plantvision',
'u1480761_Eamonn', 'Fabregas@2019');
//Our select statement. This will retrieve the data that we want.
$sql = "SELECT name FROM db_images WHERE user_id = '$id' AND plant_group = '$group'";
//Prepare the select statement.
$stmt = $pdo->prepare($sql);
//Execute the statement.

```



```

$stmt->execute();

//Retrieve the rows using fetchAll.
$plants = $stmt->fetchAll();

?>
<div id="home">
<a href = "home.php">

</a>
</div>

<form method="post" name = "plant">
<div class="form">
<font size = "2"> Please select <b> Name </b> of leaf</font>
<select id = "soflow" name="plant" >
<?php foreach($plants as $plant): ?>
<option value="<?=$plant['name']; ?>"><?=$plant['name']; ?></option>
<?php endforeach; ?>
</select> <br><br>
<button type = "submit" name = "analyze" >Analyze</button><br><br>
</form>

<!-- Logout message -->
<p class="message">Something else? <a href="manageIndex.php">Go back</a></p>
</form>

</div>
</body>
</html>

```

2.2.13 config.php

```

<?php
//Defining the Database settings
define('DB_SERVER', 'mysql13792.cp.blacknight.com');
define('DB_USERNAME', 'u1480761_Eamonn');
define('DB_PASSWORD', 'Fabregas@2019');
define('DB_DATABASE', 'db1480761_plantvision');
$db = mysqli_connect(DB_SERVER,DB_USERNAME,DB_PASSWORD,DB_DATABASE);

//This is set for register form.
$servername = 'mysql13792.cp.blacknight.com';
$username = 'u1480761_Eamonn';
$password = 'Fabregas@2019';
$dbname = 'db1480761_plantvision';
?>

```

2.2.14 csv.php

```
<?php
include 'config.php';
session_start();
$group = 'Plant Group A';
$id = $_SESSION['user_id'];
$group = $_SESSION['group'];

// Creates a new csv file and store it in tmp directory

// output headers so that the file is downloaded rather than displayed
header('Content-Type: text/csv; charset=utf-8');
header('Content-Disposition: attachment; filename=myPlantDetails.csv');
// create a file pointer connected to the output stream
$output = fopen('php://output', 'w');
$result = mysqli_query($db, "SELECT id, name, plant_group FROM db_images WHERE
plant_group = '$group' AND user_id = $id");
$count = 0;
$i = 0;
while ($aRow = mysqli_fetch_array($result)) {
    $imageID[$i] = $aRow[0];
    $count++;
    $i++;
}
fputcsv($output, array('Leaf Name', 'Plant Group', 'Disease %', 'Time Taken'));
$rows4 = mysqli_query($db, "SELECT leaf_name, plant_group, disease_per, time FROM report
WHERE plant_group = '$group' AND user_id = $id");
while($row4 = mysqli_fetch_assoc($rows4)){
    fputcsv($output, $row4);
}
fputcsv($output, array('Image ID', 'Name of Leaf', 'Plant Group'));
$rows3= mysqli_query($db, "SELECT id, name, plant_group FROM db_images WHERE plant_group
= '$group' AND user_id = $id");
while($row3 = mysqli_fetch_assoc($rows3)){
    fputcsv($output, $row3);
}
$i = 0;
fputcsv($output, array('Image ID', 'Temperature(C)', 'Humidity %', 'Wind Km/h',
'Pressure(Pa)', 'Cloud Coverage %', 'Time Taken', 'Description'));
while($i < $count){
    $rows2 = mysqli_query($db, "SELECT imageId, temp, hum, wind, pressure, cloud, time,
description FROM weather WHERE imageId = $imageID[$i]");
    $i++;
    while($row2 = mysqli_fetch_assoc($rows2)){
        fputcsv($output, $row2);
    }
}
$i = 0;
fputcsv($output, array('Image ID', 'Address', 'Longitude', 'Latitude'));
while($i < $count){
    $rows5 = mysqli_query($db, "SELECT imageId, address, longitude, latitude FROM
gpsLocation WHERE imageId = $imageID[$i]");
    $i++;
    while($row5 = mysqli_fetch_assoc($rows5)){
        fputcsv($output, $row5);
    }
}
?>
```

2.2.15 delete.php

```
<?php
include 'config.php';
session_start();
$id = $_GET['id'];
$plant = $_SESSION['plant'];

$dbname = "db1480761_plantvision";
$conn = mysqli_connect("mysql3792.cp.blacknight.com", "u1480761_Eamonn",
"Fabregas@2019", $dbname);
// Check connection
if (!$conn) {
    die("Connection failed: " . mysqli_connect_error());
}
$result = mysqli_query($db,"SELECT * FROM imageResult WHERE id = '$id'");
$row = mysqli_fetch_array($result);
$imageI = $row[1];
//echo "<br>" . $imageI;
$sql = mysqli_query($conn, "SELECT * FROM imageResult WHERE id = $id");
$row = mysqli_fetch_array($sql);
echo $row[0];
$colourCode = $row[2];
echo "<br>" . $colourCode;
$userID = $row[6];
echo "<br>" . $userID;
$sql = mysqli_query($conn, "SELECT * FROM db_images WHERE user_id = '$userID' AND name =
'$plant'");
$row = mysqli_fetch_array($sql);
$plant_group = $row[4];
echo "<br>" . $plant_group;
$sql = mysqli_query($conn, "INSERT into markedtable (user_id, plant_group, colourCode,
mark) VALUES ($userID, '$plant_group', $colourCode, '')");
// sql to delete a record
$sql = mysqli_query($conn, "DELETE FROM imageResult WHERE id = $id");
$result = mysqli_query($db,"SELECT * FROM imageResult WHERE imageID = '$imageI'");
$i = 0;
$counter = 0;
while ($row = mysqli_fetch_array($result)){
    $count[$i] = $row[4];
    $imageID[$i] = $row[0];
    //echo "<br>" . $count[$i];
    //echo "<br>" . $imageID[$i];
    $i++;
    $counter ++;
}
$i = 0;
$totalCount = 0;
while($i < $counter){
    $totalCount = $totalCount + $count[$i];
    $i++;
}
//echo "<br>" . $totalCount;
$i = 0;
while($i < $counter){
    $per = $count[$i]/$totalCount*100;
    $per = round($per, '2');
    //echo "<br>" . $per;
    $sql = mysqli_query($conn, "UPDATE imageResult SET colourPer = '$per' WHERE id =
```

```
'$imageID[$i]' AND imageID = '$imageI");
    $i++;
}
header('Location: editor.php');
?>
```

2.2.16 edit.php

```
<!DOCTYPE html>
<html lang="en" >
<head>
  <meta charset="UTF-8">
  <title>Plant Vision</title>
  <link rel="stylesheet" href="css/style.css">
</head>

<style>
div.polaroid {
  height: 75%;
  width: 100%;
  background-color: white;
  box-shadow: 0 4px 8px 0 rgba(0, 0, 0, 0.2), 0 6px 20px 0 rgba(0, 0, 0, 0.19);
  margin-bottom: 25px;
}

div.container {
  text-align: center;
  padding: 5px 5px;
}
</style>

<?php
include 'config.php';
session_start();
$plant = $_SESSION['plant'];
if(!isset($_SESSION['login_user'])){
  header("Location:login.php");
}

if (isset($_POST['analyze'])){
  $group = $_POST["group"];
  $plantN = $_POST["plantN"];
  echo $group;
  echo $plantN;
  $imageId = $_SESSION['imageId'];

  if($group != "" && $plantN != ""){
    $query = mysqli_query($db, "UPDATE db_images SET plant_group = '$group'
WHERE id = '$imageId'");
    $query = mysqli_query($db, "UPDATE db_images SET name = '$plantN' WHERE id
= '$imageId'");
    echo "<script type='text/javascript'>alert('Details successfully
changed')</script>";
    header('Location:
http://www.c00197458.candept.com/PlantVision/chooseImage.php');
  }
  else{
    echo "<script type='text/javascript'>alert('Changes cannot be
```

```

blank')</script>";
    }
}
$username = $_SESSION['login_user'];
$result = mysqli_query($db,"SELECT * FROM user WHERE username = '$username'");
$row = mysqli_fetch_array($result);
$user_id = $row[0];

$result = mysqli_query($db,"SELECT * FROM db_images WHERE name = '$plant' AND user_id =
'$user_id'");
$row = mysqli_fetch_array($result);
$image = $row['url'];
$_SESSION['imageId'] = $row['id'];
?>
<!-- -----HTML----- -->
<body>
  <div class="form">
    <form method="POST" class="login-form" >
      <input type="text" style = "border: 2px solid red;" name = "plantN" placeholder =
      "Plant name: <?php echo $row['name']?>"enabled />
      <p align = "right" class="message"><a href="groupSelection.php">Choose
      from pre-defined group</a></p>
      <input type="text" style = "border: 2px solid red;"name = "group" placeholder=
      "Plant group: <?php echo $row['plant_group']?> " enabled />
      <input type="text" placeholder= "Time taken: <?php echo $row['datetime']?> "
      disabled />
      <div class="polaroid">
        <img src= "<?php echo $image?>" alt="" style="width:100%">
        <div class="container">
          <p><?php echo $row['name']?>, <?php echo $row['plant_group']?></p>
        </div>
        </div>
        <button name = "analyze" type = "submit">Confirm</button>
        <p class="message">Changed your mind? <a href="chooseImage.php"> Go back</a></p>
      </form>
    </div>
  </body>
</html>

```

2.2.17 editor.php

```

<!DOCTYPE html>
<html>
<head>

<meta http-equiv="Content-Type" content="text/html; charset=ISO-8859-1" />
<link rel="stylesheet" href="css/style.css">
<link rel="stylesheet" href="css/circle.css">
<title>Plant Vision</title>
</head>

<style>

BUTTON{
  background: yellow;

```

```

        color: white;
    }
    .div {
        width: 920px;
        padding: 2px;
        height: 490px;
        border: 2px solid gray;
        background: white;
        position: absolute;
        top: 20%;
        left: 25%;
    }

    .box1{
        background-color: white;
        border: 5px solid gray;
        width: 350px;
        height: 350px;
        position: absolute;
        top: 15%;
        left: 110%;
    }

    .form3{
        position: fixed;
        top: 10%;
        left: 5%;
    }

    .table {
        position: absolute;
        top: 3%;
        left: 4%;
        background-color: white;
    }
</style>

<?php
session_start();
$plant = $_SESSION['plant'];
$_SESSION['plant'] = $plant;
include 'config.php';
if(!isset($_SESSION['login_user'])){
    header("Location: login.php");
}

$username = $_SESSION['login_user'];
$result = mysqli_query($db, "SELECT * FROM user WHERE username = '$username'");
$row = mysqli_fetch_array($result);
$user_id = $row[0];

$result = mysqli_query($db, "SELECT * FROM db_images WHERE name = '$plant' AND user_id = '$user_id'");
$row = mysqli_fetch_array($result);
$imageIdentifier = $row[0];
$url = $row[1];

$result = mysqli_query($db, "SELECT * FROM imageResult WHERE imageID = '$imageIdentifier'");
$colours = [];
$counter = 0;
$i = 0;
while ($row = mysqli_fetch_array($result)) {

```

```

    $id[$i] = $row[0];
    $colour[$i] = $row[2];
    $colour[$i] = dechex($colour[$i]);
    $percentage[$i] = $row[3];
    $counter++;
    $diseased[$i] = $row[5];
    $i++;
}

$result = mysqli_query($db,"SELECT * FROM imageResult WHERE imageID = '$imageIdentifier'
AND diseaseM = 'Diseased'");
$row = mysqli_fetch_array($result);
$displayDisease = $row[3];

?>
<body>
<div class="c100 p10 orange big">
    <span><?php echo $displayDisease."%";?></span>
    <div class="slice">
        <div class="bar"></div>
        <div class="fill"></div>
    </div>
</div>
<div class = "div">
<div class = "table">
<table border="4">
<tr><th colspan="7"><?php echo $plant ?></th></tr>

<tr><td><b>No.</b></td><td><b>Color</b></td><td><b>Coverage</b></td><td><b>Marked</b></td><td><b>Color value</b></td><td><b>Remove</b></td><td><b>Disease</b></td></tr>
<?php
$number = 1;
for ($i = 0; $i < $counter; $i++)
{
    echo "<tr><td>$number</td><td bgcolor="."$colour[$i]." width=16
height=16></td>&nbsp;<td>".$percentage[$i].
"%</td><td><b>$diseased[$i]</b></td><td>#$colour[$i]</td><td><button><a
href='delete.php?id="."$id[$i]." '>Delete</a></button></td><td><button><a href='mark.php?
id="."$id[$i]." '>Mark As</a></button></td></tr>";
    $number++;
}
?>
</div>
<div class = "box1">

</div>
<p class="message">Not the correct plant? <a href="chooseImage.php">Choose
another</a></p>
</div>
</body>
</html>

```

2.2.18 fusioncharts.php

```
<?php

class FusionCharts {

    private $constructorOptions = array();

    private $constructorTemplate = '
<script type="text/javascript">
    FusionCharts.ready(function () {
        new FusionCharts(__constructorOptions__);
    });
</script>';

    private $renderTemplate = '
<script type="text/javascript">
    FusionCharts.ready(function () {
        FusionCharts("__chartId__").render();
    });
</script>
';

    // constructor
    function __construct($type, $id, $width = 400, $height = 300, $renderAt,
$dataFormat, $dataSource) {
        isset($type) ? $this->constructorOptions['type'] = $type : '';
        isset($id) ? $this->constructorOptions['id'] = $id : 'php-fc-' . time();
        isset($width) ? $this->constructorOptions['width'] = $width : '';
        isset($height) ? $this->constructorOptions['height'] = $height : '';
        isset($renderAt) ? $this->constructorOptions['renderAt'] = $renderAt : '';
        isset($dataFormat) ? $this->constructorOptions['dataFormat'] = $dataFormat :
'';
        isset($dataSource) ? $this->constructorOptions['dataSource'] = $dataSource :
'';

        $tempArray = array();
        foreach($this->constructorOptions as $key => $value) {
            if ($key === 'dataSource') {
                $tempArray['dataSource'] = '__dataSource__';
            } else {
                $tempArray[$key] = $value;
            }
        }

        $jsonEncodedOptions = json_encode($tempArray);

        if ($dataFormat === 'json') {
            $jsonEncodedOptions = preg_replace('/\"__dataSource__\"/',
$this->constructorOptions['dataSource'], $jsonEncodedOptions);
        } elseif ($dataFormat === 'xml') {
            $jsonEncodedOptions = preg_replace('/\"__dataSource__\"/',
'\''__dataSource__\'', $jsonEncodedOptions);
            $jsonEncodedOptions = preg_replace('/__dataSource__/',
$this->constructorOptions['dataSource'], $jsonEncodedOptions);
        } elseif ($dataFormat === 'xmlurl') {
            $jsonEncodedOptions = preg_replace('/__dataSource__/',
$this->constructorOptions['dataSource'], $jsonEncodedOptions);
        } elseif ($dataFormat === 'jsonurl') {
            $jsonEncodedOptions = preg_replace('/__dataSource__/',
$this->constructorOptions['dataSource'], $jsonEncodedOptions);
        }
    }
}
```



```
    }
    $newChartHTML = preg_replace('/__constructorOptions__/',
$jsonEncodedOptions, $this->constructorTemplate);

    echo $newChartHTML;
}

// render the chart created
// It prints a script and calls the FusionCharts javascript render method of
created chart
function render() {
    $renderHTML = preg_replace('/__chartId__/', $this->constructorOptions['id'],
$this->renderTemplate);
    echo $renderHTML;
}

}
?>
```

2.3 Application Programming Interface (API)

2.3.1 upload.php

```
<?php

//importing dbDetails file
require_once 'dbDetails.php';

//this is our upload folder
$upload_path = 'AndroidImageUpload/uploads/';
$upload_path2 = '/AndroidImageUpload/uploads/';

//Getting the server ip
$server_ip = 'www.c00197458.candept.com';

//creating the upload url
$upload_url = 'http://'.$server_ip.$upload_path2;

//response array
$response = array();

if($_SERVER['REQUEST_METHOD']=='POST'){

//checking the required parameters from the request
if(isset($_POST['name']) and isset($_FILES['image']['name'])){

//connecting to the database
$con = mysqli_connect(HOST,USER,PASS,DB) or die('Unable to Connect...');

//getting name from the request
$name = $_POST['name'];

//getting file info from the request
$fileinfo = pathinfo($_FILES['image']['name']);

//getting the file extension
$extension = $fileinfo['extension'];

//file url to store in the database
$file_url = $upload_url . getFileName() . '.' . $extension;

//file path to upload in the server
$file_path = $upload_path . getFileName() . '.' . $extension;

//getting username
$username = $_POST['username'];
//$username = file_get_contents("log.txt");
$result = mysqli_query($con, "SELECT * FROM user WHERE username='$username' ");
$row = mysqli_fetch_array($result);
$userID = $row['user_id'];

//$plant_group = 'No Group Defined';
$plant_group = $_POST['group'];
```

```

date_default_timezone_set("Europe/Dublin"); //getting current timezone
$date = date('Y-m-d H:i:s', time()); //getting current timestamp

//Writing to group.txt for later retrieval
$file = 'group.txt';
file_put_contents($file, $plant_group);

// ----- Plant duplicate name check -----
$i;
$sql = mysqli_query($con, "SELECT * from db_images WHERE user_id = '$userID'");
while ($row = mysqli_fetch_array($sql)) {
    if($name.$i == $row[2]){
        $i++;
        //echo $i;
        //echo "<br>".$name.$i;
    }
}
$name = $name . $i;
//echo "<br>" . $name;
// ----- end / Duplicate name check -----

//trying to save the file in the directory
try{
    //saving the file
    move_uploaded_file($_FILES['image']['tmp_name'],$file_path);
    $sql = "INSERT INTO db_images (id, url, name, user_id, plant_group, datetime) VALUES
(null, '$file_url', '$name', '$userID', '$plant_group', '$date')";

    echo "<br>".$file_url;

//----- Image processing -----

include_once("colors.inc.php");
$ex=new GetMostCommonColors();
$ex->image=$file_url;
$colors=$ex->Get_Color();
$how_many=12;
$colors_key=array_keys($colors);

$i = 0;
$totalPix = 0;
$percentage = [];

// Gathering total amount of pixels
while($i < 12){

    $totalPix = $totalPix + $colors[$colors_key[$i]];
    $colourCount[$i] = $colors[$colors_key[$i]];
    $i ++;
}

// Gatering percentage

```

```

$i = 0;
$p = 0;
while($i < 12){
    $percentage[$p] = round($colors[$colors_key[$i]] / $totalPix * 100, 2);
    $p ++;
    $i ++;
}

$i = 0;
$p = 0;
$mark = "";
//Getting ID of image to insert to 'imageResult'
$idNumber = getFileName();
while($i < 12){
    $hex = $colors_key[$i];
    $hex = hexdec($hex);
    $per = $percentage[$p];
    $result = mysqli_query($con, "INSERT into imageResult (imageID, colourCode,
colourPer, colourCount, diseaseM, user_id) VALUES ($idNumber, $hex, $per,
$colourCount[$i], '$mark', '$userID')");
    $p ++;
    $i ++;
}
//-----

// ----- Getting radio answer / Check for avoided colours
-----
$checkGroup = $_POST['radio'];
$i = 0;
$count = 0;
if($checkGroup == "Yes"){
    //echo "<br>in";
    //echo "<br>" . $userID;
    //echo "<br>" . $plant_group;
    $avoidSQL = mysqli_query($con, "SELECT * from markedtable WHERE user_id =
'$userID' AND plant_group = '$plant_group' AND mark = ''");
    while ($row = mysqli_fetch_array($avoidSQL)) {
        $avoid[$i] = $row[3];
        // $colour[$i] = dechex($colour[$i]);
        //echo "<br>" . $avoid[$i];
        $i++;
        $count++;
    }
    $i = 0;
    while($i < $count){
        $deleteSQL = mysqli_query($con, "DELETE from imageResult WHERE imageID =
'$idNumber' AND user_id = '$userID' AND colourCode= '$avoid[$i]'");
        $i ++;
    }

    $searchSQL = mysqli_query($con, "SELECT * FROM imageResult WHERE imageID =
'$idNumber'");
    $i = 0;
    $counter = 0;
    while ($row2 = mysqli_fetch_array($searchSQL)){
        $countOfColours[$i] = $row2[4];
        $imageIDs[$i] = $row2[0];
        //echo "<br>" . $count[$i];
        //echo "<br>" . $imageID[$i];
        $i++;
        $counter ++;
    }
}

```

```

}

$i = 0;
$totalCount = 0;
while($i < $counter){
    $totalCount = $totalCount + $countOfColours[$i];
    $i++;
}
//echo "<br>" . $totalCount;
$i = 0;

while($i < $counter){
    $per = $countOfColours[$i]/$totalCount*100;
    $per = round($per, '2');
    //echo "<br>" . $per;
    $update = mysqli_query($con, "UPDATE imageResult SET colourPer = '$per'
WHERE id = '$imageIDs[$i]' AND imageID = '$idNumber'");
    $i++;
}

// ----- Check for marked disease -----

$markCheck = mysqli_query($con, "SELECT * from markedtable WHERE user_id =
$userID AND plant_group = '$plant_group' AND mark = 'Marked'");
$markCheckRow = mysqli_fetch_array($markCheck);
$markedDisease = $markCheckRow[3];
//echo "<br>" . $markedDisease;
$hex = dechex($markedDisease);
$hex = "#". $hex;
list($r, $g, $b) = sscanf($hex, "%02x%02x%02x");
//echo "<br>";
//echo "$hex -> $r $g $b";
$r = $r+1;
$g = $g+1;
$b = $b+1;

$getResults = mysqli_query($con, "SELECT * from imageResult WHERE user_id =
$userID AND imageID = $idNumber");
$i = 0;
while ($getResultA = mysqli_fetch_array($getResults)){
    $id[$i] = $getResultA[0];
    $hex = $getResultA[2];
    $hex = dechex($hex);
    $hexVal[$i] = "#". $hex;
    //echo "<br>" . $hexVal[$i];

    list($r1, $g1, $b1) = sscanf($hexVal[$i], "%02x%02x%02x");

    $r1 = $r1+1;
    $g1 = $g1+1;
    $b1 = $b1+1;
    //echo "<br>" . $r1;
    //echo "," . $g1;
    //echo "," . $b1;

    $percentageChangeRed = (1- $r/$r1) * 100;
    $percentageChangeGreen = (1- $g/$g1) * 100;
    $percentageChangeBlue = (1- $b/$b1) * 100;

    //echo "<br> Red is " . $percentageChangeRed;
    //echo "<br> Green is " . $percentageChangeGreen;
    //echo "<br> Blue is " . $percentageChangeBlue;

    if($percentageChangeRed < 5 && $percentageChangeRed > -5){
        if($percentageChangeGreen < 5 && $percentageChangeGreen > -5){

```

```

        if($percentageChangeBlue < 5 && $percentageChangeBlue >
-5){
            $target = $hexVal[$i];
            $targetID = $id[$i];
        }
    }
    $i++;
}
//echo "<br><br> The Target is: ".$target;
$updateQuery = mysqli_query($con, "UPDATE imageResult SET diseaseM = 'Diseased'
WHERE user_id = $userID AND id = $targetID");

    //echo "<br>" . $targetID;

    // ----- Inserting in mobileResult -----

    //$deleteMob = mysqli_query($con, "DELETE * from mobileResult WHERE id = 1");
    $result3 = mysqli_query($con, "SELECT * from imageResult WHERE id = $targetID");
    echo "<br>" . $targetID;
    $rArray = mysqli_fetch_array($result3);
    echo "<br>" . $rArray[3];
    $value = $rArray[3];
    echo "<br>" . $value;
    $str = "$value";
    echo "<br>" . $str;
    $mobile = mysqli_query($con, "UPDATE mobileResult SET name = '$str' WHERE id =
1");

    echo "<br> All works";
    // ----- end / mobileResult -----

    // ----- end / check for marked disease -----
}
else if($checkGroup == "No"){
    echo "<br>Not in";
}
// ----- end -----

//----- Getting GPS co -----
$address = $_POST['address'];
$long = $_POST['longitude'];
$lat = $_POST['latitude'];
$gpsLocationQuery = mysqli_query($con, "INSERT into gpsLocation (imageId, address,
longitude, latitude) VALUES ($idNumber, '$address', $long, $lat)");

//----- Weather -----
$ip = $_SERVER['REMOTE_ADDR'];
$details = json_decode(file_get_contents("http://ipinfo.io/{$ip}/json"));
$region = $details -> region;
$country = $details -> country;
$ip = ip2long($ip);
$location = $region. ", " . $country;
if($region == ""){
    $location = "Ireland";
}

```

```

}
$jsonurl =
"http://api.openweathermap.org/data/2.5/weather?q={$location}&appid=fc72c995fe6227c7f1f2
44ca806b300b";
$json = file_get_contents($jsonurl);
$weather = json_decode($json);
$data=json_decode($json,true);

$temp = $weather->main->temp;
$temp = $temp - 273.15;
$humidity = $weather->main->humidity;
$cloud = $weather->clouds->all;
$wind = $weather->wind->speed;
$pressure = $weather->main->pressure;
$icon = $data['weather']['0']['icon'];
$url = 'http://openweathermap.org/img/w/';
$icon = $url.$icon.".png";
$desc = $data['weather']['0']['description'];
date_default_timezone_set("Europe/Dublin");

$weather = mysqli_query($con, "INSERT into weather (imageId, ip, temp, hum, wind,
pressure, cloud, time, description, icon) VALUES ($idNumber, $ip, $temp, $humidity,
$wind, $pressure, $cloud, NOW(), '$desc', '$icon')");

//----- end / Weather
-----

//adding the path and name to database
if(mysqli_query($con,$sql)){

//filling response array with values
$response['error'] = false;
$response['url'] = $file_url;
$response['name'] = $name;
}
//if some error occurred
}catch(Exception $e){
$response['error']=true;
$response['message']=$e->getMessage();
}
//displaying the response
//echo json_encode($response);

//closing the connection
mysqli_close($con);
}else{
$response['error']=true;
$response['message']='Please choose a file';
}
}

/*
We are generating the file name
so this method will return a file name for the image to be upload
*/
function getFileName(){
$con = mysqli_connect(HOST,USER,PASS,DB) or die('Unable to Connect...');
$sql = "SELECT max(id) as id FROM db_images";
$result = mysqli_fetch_array(mysqli_query($con,$sql));
mysqli_close($con);
if($result['id']==null)

```

```

return 1;
else
return ++$result['id'];
}
?>

```

2.3.2 Register.php

```

<?php
    $con = mysqli_connect("mysql3792.cp.blacknight.com", "u1480761_Eamonn",
"Fabregas@2019", "db1480761_plantvision");

    $email = $_POST["email"];
    $name = $_POST["name"];
    $username = $_POST["username"];
    $password = $_POST["password"];

    $statement = mysqli_prepare($con, "INSERT INTO user (email, name, username,
password) VALUES ('$email', '$name', '$username', '$password')");
    mysqli_stmt_bind_param($statement, "siss", $user_id, $email, $name, $username,
$password);
    mysqli_stmt_execute($statement);

    $response = array();
    $response["success"] = true;
    echo json_encode($response);
?>

```

2.3.3 login.php

```

<?php
    $con = mysqli_connect("mysql3792.cp.blacknight.com", "u1480761_Eamonn",
"Fabregas@2019", "db1480761_plantvision");
    $username = $_POST["username"];
    $password = $_POST["password"];
    $file = 'log.txt';
    file_put_contents($file, $username);
    $statement = mysqli_prepare($con, "SELECT * FROM user WHERE username = '$username'
AND password = '$password'");
    mysqli_stmt_bind_param($statement, "ss", $username, $password);
    mysqli_stmt_execute($statement);
    mysqli_stmt_store_result($statement);
    mysqli_stmt_bind_result($statement, $userID, $email, $name, $username, $password);
    $response = array();
    $response["success"] = false;
    while(mysqli_stmt_fetch($statement)){
        $response["success"] = true;
        $response["email"] = $email;
        $response["name"] = $name;
        $response["username"] = $username;
        $response["password"] = $password;
    }
    echo json_encode($response);
?>

```


2.3.4 getLibrary.php

```
<?php
$servername = "mysql3792.cp.blacknight.com";
$username = "u1480761_Eamonn";
$password = "Fabregas@2019";
$dbase = "db1480761_plantvision";
//creating a new connection object using mysqli
$conn = new mysqli($servername, $username, $password, $database);
//if there is some error connecting to the database
//with die we will stop the further execution by displaying a message causing the error
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$username = file_get_contents("log.txt");

//if everything is fine
//creating an array for storing the data
$heroes = array();
//echo $username;
$searchQ = mysqli_query($conn, "SELECT * FROM user WHERE username = '$username'");
$rowUser = mysqli_fetch_array($searchQ);
$idUser = $rowUser[0];
//echo "<br>" . $idUser;
//this is our sql query
//$sql = "SELECT user_id, name FROM user;";
$sql = "SELECT id, name FROM db_images WHERE user_id = '$idUser'";

//creating an statement with the query
$stmt = $conn->prepare($sql);

//executing that statement
$stmt->execute();

//binding results for that statement
$stmt->bind_result($id, $name);

//looping through all the records
while($stmt->fetch()){

    //pushing fetched data in an array
    $temp = [
        'id'=>$id,
        'name'=>$name
    ];

    //pushing the array inside the hero array
    array_push($heroes, $temp);
}

//displaying the data in json format
echo json_encode($heroes);

?>
```

2.3.5 getGroup.php

```
<?php

$servername = "mysql13792.cp.blacknight.com";
$username = "u1480761_Eamonn";
$password = "Fabregas@2019";
$dbase = "db1480761_plantvision";

//creating a new connection object using mysqli
$conn = new mysqli($servername, $username, $password, $database);

//if there is some error connecting to the database
//with die we will stop the further execution by displaying a message causing the error
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}
$username = file_get_contents("log.txt");

//if everything is fine
//creating an array for storing the data
$heroes = array();
//echo $username;
$searchQ = mysqli_query($conn, "SELECT * FROM user WHERE username = '$username'");
$rowUser = mysqli_fetch_array($searchQ);
$idUser = $rowUser[0];
//echo "<br>" . $idUser;

//this is our sql query
// $sql = "SELECT user_id, name FROM user;";
$sql = "SELECT DISTINCT user_id, plant_group FROM db_images WHERE user_id = '$idUser'";

//creating an statement with the query
$stmt = $conn->prepare($sql);

//executing that statement
$stmt->execute();

//binding results for that statement
$stmt->bind_result($id, $name);

//looping through all the records
while($stmt->fetch()){

    //pushing fetched data in an array
    $temp = [
        'id'=>$id,
        'name'=>$name
    ];

    //pushing the array inside the hero array
    array_push($heroes, $temp);
}

//displaying the data in json format
echo json_encode($heroes);

?>
```

2.3.6 getData.php

```
<?php

$servername = "mysql3792.cp.blacknight.com";
$username = "u1480761_Eamonn";
$password = "Fabregas@2019";
$dbase = "db1480761_plantvision";

//creating a new connection object using mysqli
$conn = new mysqli($servername, $username, $password, $database);

//if there is some error connecting to the database
//with die we will stop the further execution by displaying a message causing the error
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

/*
$sqlQuery = mysqli_query($conn, "SELECT * from mobileResult");
$row = mysqli_fetch_array($sqlQuery);
echo "<br>" . $row[1];
*/

//if everything is fine

//creating an array for storing the data
$heroes = array();

//this is our sql query
// $sql = "SELECT user_id, name FROM user;";
$sql = "SELECT id, name FROM mobileResult;";
//creating an statement with the query
$stmt = $conn->prepare($sql);

//executing that statement
$stmt->execute();

//binding results for that statement
$stmt->bind_result($id, $name);

//looping through all the records
while($stmt->fetch()){

    //pushing fetched data in an array
    $temp = [
        'id'=>$id,
        'name'=>$name
    ];

    //pushing the array inside the hero array
    array_push($heroes, $temp);
}

//displaying the data in json format
echo json_encode($heroes);

?>
```

2.3.7 dbDetails.php

```
<?php
define('HOST','mysql3792.cp.blacknight.com');
define('USER','u1480761_Eamonn');
define('PASS','Fabregas@2019');
define('DB','db1480761_plantvision');
?>
```

2.3.8 colours.inc.php

```
<?php
class GetMostCommonColors{
    // The filename of the image (it can be a JPG, GIF or PNG image)
    var $image;
    // Returns the colors of the image in an array, ordered in descending order,
    where the keys are the colors, and the values are the count of the color.
    function Get_Color()
    {
        if (isset($this->image))
        {
            $PREVIEW_WIDTH    = 150; //resize image to 150 both height and
width
            $PREVIEW_HEIGHT   = 150;
            $size = GetImageSize($this->image);
            $scale=1;
            if ($size[0]>0)
            $scale = min($PREVIEW_WIDTH/$size[0], $PREVIEW_HEIGHT/$size[1]);
            if ($scale < 1)
            {
                $width = floor($scale*$size[0]);
                $height = floor($scale*$size[1]);
            }
            else
            {
                $width = $size[0];
                $height = $size[1];
            }
            $image_resized = imagecreatetruecolor($width, $height);
            if ($size[2]==1)
            $image_orig=imagecreatefromgif($this->image);
            if ($size[2]==2)
            $image_orig=imagecreatefromjpeg($this->image);
            if ($size[2]==3)
            $image_orig=imagecreatefrompng($this->image);
            imagecopyresampled($image_resized, $image_orig, 0, 0, 0, 0, $width,
$height, $size[0], $size[1]);
            //nearest neighbour resizing, doesnt alter image colours
            $im = $image_resized;
```

```

        $imgWidth = imagesx($im);
        $imgHeight = imagesy($im);
        for ($y=0; $y < $imgHeight; $y++)
        {
            for ($x=0; $x < $imgWidth; $x++)
            {
                $index = imagecolorat($im,$x,$y);
                $Colors = imagecolorsforindex($im,$index);
                $Colors['red']=intval(((($Colors['red'])+15)/32)*32;
//rounds colours to reduce counts of similar colours
$Colors['green']=intval(((($Colors['green'])+15)/32)*32;
$Colors['blue']=intval(((($Colors['blue'])+15)/32)*32;
                if ($Colors['red']>=256)
                $Colors['red']=240;
                if ($Colors['green']>=256)
                $Colors['green']=240;
                if ($Colors['blue']>=256)
                $Colors['blue']=240;

                $hexarray[]=substr("0".dechex($Colors['red']),-2).substr("0".dechex($Colors['green']),-2
                ).substr("0".dechex($Colors['blue']),-2);
            }
        }
        $hexarray=array_count_values($hexarray);
        natsort($hexarray);
        $hexarray=array_reverse($hexarray,true);
        return $hexarray;
    }
    else die("You must enter a filename! (\$image parameter)");
}
?>

```