

EX NO 11

MINI PROJECT – ISLAMIC QURAN & PRAYER APP

DATE :

AIM :

To build an Islamic – Quran & Prayer android application, using Java in Android Studio.

STEPS :

1] Create a New Project

2] First let's setup the layout for the app. So, go to App -> Res -> Layout -> activity_main.xml section.

3] Now design the layout for the app, the main navigation side bar, and each navigator should lead to specific page.

4] Now, go to MainActivity.java section, Now connect the buttons to the Java code.

PROBLEM STATEMENT :

Design and develop an Islamic App for android devices using Android studio. The objective of this app is to provide users a compass to find prayer direction , a digital copy of the quran and provide prayer timings according to their location.

Requirements :

1. Navigation Bar
 - The navigation bar helps in switching between 4 different sections of the application. The Home page, Prayer timings section, prayer direction section and Quran section.
2. Prayer Timings Section
 - The prayer timings section will provide the user timings for the 5 daily prayer based on their location.
3. Prayer Direction Section
 - The prayer direction section will provide the user with the direction to pray from their current location.
4. Digital Copy of Quran
 - The application also provides the user with a digital copy of the Quran, so they can read it whenever possible.

Deliverables:

- Android Studio project files containing the complete source code of the Islamic application.
- Documentation covering the design, implementation details, and usage instructions

for the application.

Constraints:

- The application should be developed using Java or Kotlin programming languages.
- Compatibility: The application should be compatible with Android devices running Android OS version 4.4 (KitKat) or higher.

Evaluation Criteria:

- Correctness and functionality of logic.
- User interface design and usability.
- Error handling and edge case scenarios.
- Compliance with Android development best practices and guidelines.

IMPLEMENTATION :

XML CODE :

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="net.a6te.lazycoder.muslim_pro_islamicremainders">
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION"
/>
    <uses-permission android:name="android.permission.ACCESS_GPS" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"
/>
    <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission
android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
    <uses-permission
android:name="android.permission.READ_EXTERNAL_STORAGE" />
    <uses-feature android:name="android.hardware.location.gps" />
    <uses-permission android:name="android.permission.WAKE_LOCK"></uses-
permission>

    <application
        android:configChanges="layoutDirection|locale"
        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=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER"
/>
            </intent-filter>
        </activity>

        <!-- Register the Alarm Receiver -->
```

```

        <receiver android:name=".Remainder.AlarmReceiver"
            android:enabled="true">
            <intent-filter>
                <action android:name="android.intent.action.BOOT_COMPLETED"
            />
            </intent-filter>
        </receiver>

        <meta-data android:name="com.facebook.sdk.ApplicationId"
            android:value="@string/facebook_app_id"/>
        <provider
            android:authorities="com.facebook.app.FacebookContentProvider179442992713579"
            android:name="com.facebook.FacebookContentProvider"
            android:exported="true" />
        <meta-data
            android:name="io.fabric.ApiKey"
            android:value="27b695160327127fcd19210e585688fd69fa5d22" />

        <service
            android:name=".DownloadData"
            android:exported="false" />
        <provider
            android:name="android.support.v4.content.FileProvider"
            android:authorities="${applicationId}.provider"
            android:exported="false"
            android:grantUriPermissions="true">
            <meta-data
                android:name="android.support.FILE_PROVIDER_PATHS"
                android:resource="@xml/provider_paths"/>
        </provider>

    </application>
</manifest>

```

JAVA CODE:

HOME:

```

package net.a6te.lazycoder.muslim_pro_islamicremainders.fragments;

import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.Intent;
import android.content.IntentFilter;
import android.graphics.Bitmap;
import android.media.MediaPlayer;
import android.os.Build;
import android.os.Bundle;
import android.os.Environment;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.v4.app.Fragment;
import android.support.v4.content.LocalBroadcastManager;
import android.support.v4.widget.TextViewCompat;
import android.text.Html;
import android.util.Log;
import android.view.LayoutInflater;

```

```

import android.view.View;
import android.view.ViewGroup;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.RelativeLayout;
import android.widget.TextView;

import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.HomePresenter;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.MVPPresenter;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.MVPView;
import net.a6te.lazycoder.muslim_pro_islamicremainders.R;
import
net.a6te.lazycoder.muslim_pro_islamicremainders.Remainder.AlarmReceiver;
import
net.a6te.lazycoder.muslim_pro_islamicremainders.Remainder.NotificationScheduler;
import net.a6te.lazycoder.muslim_pro_islamicremainders.Utills;

import java.io.BufferedOutputStream;
import java.io.File;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;

import static android.app.Activity.RESULT_OK;

/**
 * A simple {@link Fragment} subclass.
 */

public class Home extends Fragment implements View.OnClickListener,
MVPView.HomeView{

    private View view;
    private TextView autoSizeTv;

    private RelativeLayout createImageRL;//relative layout that we will
convert to an image bitmap
    private ImageView shareIvBtn;
    private Button createNewImageBtn;
    private String appName;
    private File imageDirectory;
    private String imageName;
    private MVPPresenter.HomePresenter presenter;
    private MediaPlayer ring;
    public static final int SHARE_IMAGE_REQUEST_CODE=101;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
Bundle savedInstanceState) {

        // Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment_home, container, false);
        initializeAll();

        return view;
    }
}

```

```

private void initializeAll() {
    autoSizeTv = view.findViewById(R.id.atkharTv);
    createImageRL = view.findViewById(R.id.createImageRL);
    shareIvBtn = view.findViewById(R.id.shareIvBtn);
    createNewImageBtn = view.findViewById(R.id.createNewImageBtn);

    appName = "Muslim Athkar";
    imageDirectory = new File(Environment.getExternalStorageDirectory()
+ "/" + appName + "/");
    imageName = "Athkar.jpg";

    presenter = new HomePresenter(this);

    shareIvBtn.setOnClickListener(this);
    createNewImageBtn.setOnClickListener(this);
    ring= MediaPlayer.create(getContext(),R.raw.shared_thank_you);
    TextViewCompat.setAutoSizeTextTypeWithDefaults(autoSizeTv,
TextViewCompat.AUTO_SIZE_TEXT_TYPE_UNIFORM);

//
LocalBroadcastManager.getInstance(getActivity()).registerReceiver(connectionStatusReceiver
//
    ,new IntentFilter(Utils.BROADCAST_CONNECTION_STATUS));
//

LocalBroadcastManager.getInstance(getContext()).registerReceiver(messageReceiver
    ,new IntentFilter(Utils.BROADCAST_ACTION));

}

/**
 * after view create initialize remainders and also show athkar/verse
on home page
 * */
@Override
public void onViewCreated(@NonNull View view, @Nullable Bundle
savedInstanceState) {
    super.onViewCreated(view, savedInstanceState);

    //initialize remainder
    presenter.initializeRemainder();
    presenter.prepareAtkhar();
}

/**
 * This method will be call from presenter class
 * */
@Override
public void updateRemainder(Context context,int hour, int mint, long
interval){
    NotificationScheduler.setReminder(context, AlarmReceiver.class,
hour, mint,interval);
}

@Override
public void onClick(View v) {

```

```

        switch (v.getId()) {
            case R.id.shareIvBtn:
                presenter.createBitmap(createImageRL); //this will create
new image
                shareImageBtn();
                break;
            case R.id.createNewImageBtn:
                presenter.prepareAtkharBtnPress();
                break;
        }
    }

    /**
     * when user press the share button this method will show user the
available social media
     */
    private void shareImageBtn() {
        File filePath = new File(imageDirectory, "/" + imageName);

        presenter.createIntentToShareImage(filePath);
    }

    //this is method will call from presenter it will take a intent then it
will make share event
    @Override
    public void shareImage(Intent shareIntent) {

        Intent intent2 = Intent.createChooser(shareIntent,
getString(R.string.share_via));
        /** From version 24 we need to take file read permission */
        if (Build.VERSION.SDK_INT >= 24) {
            intent2.addFlags(Intent.FLAG_GRANT_READ_URI_PERMISSION);
        }

        startActivityForResult(intent2, SHARE_IMAGE_REQUEST_CODE);
    }

    @Override
    public void onActivityResult(int requestCode, int resultCode, Intent
data) {
        super.onActivityResult(requestCode, resultCode, data);

        if (requestCode == SHARE_IMAGE_REQUEST_CODE) {
            if (resultCode == RESULT_OK) {
                playSound();
            }
        }
    }

    /**
     * show athkar/ verse*/
    @Override
    public void setTodayImage(String data) {
        if (android.os.Build.VERSION.SDK_INT >=
android.os.Build.VERSION_CODES.N) {
autoSizeTv.setText(Html.fromHtml(data, Html.FROM_HTML_MODE_LEGACY));
        } else {
            autoSizeTv.setText(Html.fromHtml(data));
        }
    }

```

```

    }

    /*
    * Store athkar image to user phone
    * */
    @Override
    public void storeBitMapImage(Bitmap bitmap){

        File folderDirectory = imageDirectory;
        //create storage directories, if they don't exist
        folderDirectory.mkdirs();
        try {
            String filePath = folderDirectory.toString() + "/" + imageName;
            FileOutputStream fileOutputStream = new
FileOutputStream(filePath);

            BufferedOutputStream bos = new
BufferedOutputStream(fileOutputStream);
            //choose another format if PNG doesn't suit you
            bitmap.compress(Bitmap.CompressFormat.JPEG, 100, bos);

            bos.flush();
            bos.close();

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

    }

    //
    // BroadcastReceiver connectionStatusReceiver = new BroadcastReceiver()
    {
    //
    // @Override
    // public void onReceive(Context context, Intent intent) {
    //     Bundle bundle = intent.getExtras();
    //     String message = bundle.getString(Utils.CONNECTION_STATUS);
    //
    //     if (bundle.getInt(Utils.STATUS_CODE) ==
Utils.ALL_CONNECTED){
    //
    Toast.makeText(context,message,Toast.LENGTH_SHORT).show();
    //
    //     }else if (bundle.getInt(Utils.STATUS_CODE) ==
Utils.NO_CONNECTION_CODE){
    //
    Toast.makeText(context,message,Toast.LENGTH_SHORT).show();
    //
    //     }
    //
    // };

    public void playSound(){
        ring= MediaPlayer.create(getContext(),R.raw.shared_thank_you);
        ring.start();

    }

```

```

@Override
public void onResume() {
    ring= MediaPlayer.create(getContext(),R.raw.shared_thank_you);
    super.onResume();
}

@Override
public void onPause() {
    ring.stop();
    super.onPause();
}

/*
 * This method will be call after API call
 * */
//broadcast receiver
BroadcastReceiver messageReceiver = new BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        boolean isUpdateData =
intent.getBooleanExtra(Utils.EXTENDED_IS_UPDATE_DATA,false);
        //new data update
        if (isUpdateData){
            presenter.prepareAtkhar();
        }

        Log.d("Test", "onReceive: "+isUpdateData);
    }
};
}

```

PrayerTime :

```

package net.a6te.lazycoder.muslim_pro_islamicremainders.fragments;

import android.Manifest;
import android.app.AlertDialog;
import android.content.BroadcastReceiver;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.IntentFilter;
import android.content.pm.PackageManager;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.provider.Settings;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.v4.app.ActivityCompat;
import android.support.v4.app.Fragment;
import android.support.v4.content.ContextCompat;
import android.support.v4.content.LocalBroadcastManager;
import android.support.v4.widget.SwipeRefreshLayout;
import android.support.v7.widget.LinearLayoutManager;
import android.support.v7.widget.RecyclerView;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

```



```

import android.widget.TextView;
import android.widget.Toast;

import
net.a6te.lazycoder.muslim_pro_islamicremainders.CheckInternetConnection;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.MVPPresenter;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.MVPView;
import
net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.PrayerTimePresenter;
import net.a6te.lazycoder.muslim_pro_islamicremainders.R;
import net.a6te.lazycoder.muslim_pro_islamicremainders.SavedData;
import net.a6te.lazycoder.muslim_pro_islamicremainders.Utils;
import
net.a6te.lazycoder.muslim_pro_islamicremainders.adapters.PrayerTimeAdapter;

/**
 * A simple {@link Fragment} subclass.
 */
public class PrayerTime extends Fragment implements MVPView.PrayerTimeView{

    private RecyclerView prayerTimeRV;
    private View v;
    private MVPPresenter.PrayerTimePresenter presenter;
    private TextView cityTv;
    private TextView errorNoInternetTv;
    private SwipeRefreshLayout refreshLayout;
    private CheckInternetConnection internetConnectionTest;
    private MediaPlayer ring;
    public static final int MY_PERMISSIONS_REQUEST_LOCATION = 100;
    private SavedData savedData;

    public PrayerTime() {
        // Required empty public constructor
    }

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        v = inflater.inflate(R.layout.fragment_prayer_time, container,
false);
        initializeAll();
        return v;
    }

    /*
     * initialize all necessary variable or initialize object etc
     */
    private void initializeAll() {
        presenter = new PrayerTimePresenter(this);
        prayerTimeRV = v.findViewById(R.id.prayerTimeRV); //prayerTime
recyclerView (RV)
        prayerTimeRV.setHasFixedSize(true);
        prayerTimeRV.setLayoutManager(new
LinearLayoutManager(getContext(),LinearLayoutManager.VERTICAL,false));

        cityTv = v.findViewById(R.id.cityNameTv);

```

```

        errorNoInternetTv = v.findViewById(R.id.errorNoInternetTv);

        refreshLayout = v.findViewById(R.id.refreshLayout);
        refreshLayout.setOnRefreshListener(refreshListener);
        internetConnectionTest = new CheckInternetConnection();
        ring= MediaPlayer.create(getContext(),R.raw.prayer_allahu_akbar);

LocalBroadcastManager.getInstance(getActivity()).registerReceiver(connectionStatusReceiver
        ,new IntentFilter(Utils.BROADCAST_CONNECTION_STATUS));

        savedData = new SavedData(getContext());
    }

    @Override
    public void onViewCreated(@NonNull View view, @Nullable Bundle savedInstanceState) {
        super.onViewCreated(view, savedInstanceState);

        callPresenter();
        playSound();//it will play allhu akbar sound
    }

    private void callPresenter() {

        /** We have old lat and long so we can calculate by old report */
        if (savedData.getLong() != 0 && savedData.getLat() != 0){
            presenter.startCalculationPrayerTime();
        }
        if (internetConnectionTest.netCheck(getContext())){

            if (checkLocationPermission()){
                presenter.startCalculationPrayerTime();
                unVisibleErrorTv();
            }else {

Toast.makeText(getContext(),getResources().getString(R.string.gps_setting_message), Toast.LENGTH_SHORT).show();

visibleErrorTv(getContext().getResources().getString(R.string.gps_setting_message));
            }

        }else {

Toast.makeText(getContext(),getResources().getString(R.string.required_data_connection), Toast.LENGTH_SHORT).show();

errorNoInternetTv.setText(getContext().getResources().getString(R.string.required_data_connection));
            errorNoInternetTv.setVisibility(View.VISIBLE);
        }
    }

    /*
    * set Recycler view adapter
    */
    @Override
    public void initializeRecyclerView(PrayerTimeAdapter adapter) {

```

```

        if (adapter != null) {
            prayerTimeRV.setAdapter(adapter);
        }
    }

    public void visibleErrorTv(String message){
        errorNoInternetTv.setVisibility(View.VISIBLE);
        errorNoInternetTv.setText(message);
    }
    public void unVisibleErrorTv(){
        errorNoInternetTv.setVisibility(View.GONE);
    }

    @Override
    public void setCityName(String cityName) {
        cityTv.setText(cityName);
    }

    /*
     * if GPS is not turn on this method will show a alert dialog to user to
    setting enable GPS
     */
    @Override
    public void showGpsSettingAlert(){
        AlertDialog.Builder alertDialog = new
        AlertDialog.Builder(getContext());

        // Setting Dialog Title
        alertDialog.setTitle(R.string.gps_setting_title);

        // Setting Dialog Message
        alertDialog.setMessage(R.string.gps_setting_message);

        // On pressing Settings button
        alertDialog.setPositiveButton(R.string.settings, new
        DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog,int which) {
                Intent intent = new
                Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
                getContext().startActivity(intent);
            }
        });

        // on pressing cancel button
        alertDialog.setNegativeButton(R.string.cancel, new
        DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });

        visibleErrorTv(getContext().getString(R.string.gps_setting_message));
    }

    // Showing Alert Message
    alertDialog.show();
}

```

```

SwipeRefreshLayout.OnRefreshListener refreshListener = new
SwipeRefreshLayout.OnRefreshListener() {
    @Override
    public void onRefresh() {

        callPresenter();
        refreshLayout.setRefreshing(false);
    }
};

@Override
public void onResume() {
    super.onResume();
}

public void playSound(){
    if (!ring.isPlaying()) {
        ring.start();
    }
}
@Override
public void onPause() {
    ring.stop();
    super.onPause();
}

/*
 * Taking runtime permission for location
 * */

public boolean checkLocationPermission() {

    final String[] PERMISSIONS =
{Manifest.permission.ACCESS_FINE_LOCATION,Manifest.permission.WRITE_EXTERNAL_STORAGE};

    if(!hasPermissions(getContext(), PERMISSIONS)){

        ActivityCompat.requestPermissions(getActivity(),
            PERMISSIONS,
            MY_PERMISSIONS_REQUEST_LOCATION);
        return false;
    } else {
        return true;
        //permission already granted
        //like android version < 5(lollipop) don,t need runtime
permission
    }

}

/*
 * check is already permission granted or not
 * */
public static boolean hasPermissions(Context context, String...
permissions) {
    if (context != null && permissions != null) {
        for (String permission : permissions) {
            if (ActivityCompat.checkSelfPermission(context, permission)
!= PackageManager.PERMISSION_GRANTED) {

```

```

        return false;
    }
}

return true;
}

@Override
public void onRequestPermissionsResult(int requestCode,
                                       String permissions[], int[]
grantResults) {
    switch (requestCode) {
        case MY_PERMISSIONS_REQUEST_LOCATION: {
            // If request is cancelled, the result arrays are empty.
            if (grantResults.length > 0
                && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                if (ContextCompat.checkSelfPermission(getContext(),
Manifest.permission.ACCESS_FINE_LOCATION)
                    == PackageManager.PERMISSION_GRANTED) {
                    //permission granted
                }
            } else {
                permissionDenied();
            }
            return;
        }
    }

    /*
    * if user deny to give permission this method will be called
    * */
    private void permissionDenied() {
        Toast.makeText(getContext(),
R.string.permission_denied, Toast.LENGTH_SHORT).show();
    }

    BroadcastReceiver connectionStatusReceiver = new BroadcastReceiver() {
        @Override
        public void onReceive(Context context, Intent intent) {
            Bundle bundle = intent.getExtras();
            String message = bundle.getString(Utils.CONNECTION_STATUS);

            if (bundle.getInt(Utils.STATUS_CODE) == Utils.ALL_CONNECTED) {
                unVisibleErrorTv();
                presenter.startCalculationPrayerTime();
            } else if (bundle.getInt(Utils.STATUS_CODE) ==
Utils.NO_CONNECTION_CODE) {
                visibleErrorTv(message);
            }
        }
    };
}

```

QIBLA:

```
package net.a6te.lazycoder.muslim_pro_islamicremainders.fragments;

import android.Manifest;
import android.app.AlertDialog;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.media.MediaPlayer;
import android.os.Bundle;
import android.provider.Settings;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.design.widget.Snackbar;
import android.support.v4.app.ActivityCompat;
import android.support.v4.app.Fragment;
import android.support.v4.content.ContextCompat;
import android.support.v4.widget.SwipeRefreshLayout;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.widget.RelativeLayout;
import android.widget.TextView;
import android.widget.Toast;

import net.a6te.lazycoder.muslim_pro_islamicremainders.DrawCompass;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.MVPPresenter;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.MVPView;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.QiblaPresenter;
import net.a6te.lazycoder.muslim_pro_islamicremainders.R;

/**
 * A simple {@link Fragment} subclass.
 */
public class Qibla extends Fragment implements MVPView.QiblaView{

    private RelativeLayout directionContainer;
    private DrawCompass rose;
    View view;
    TextView qiblaDistance, qiblaDegree;

    private MVPPresenter.QiblaPresenter presenter;
    private SwipeRefreshLayout refreshLayout;
    public static final int MY_PERMISSIONS_REQUEST_LOCATION = 99;
    private Context context;
    private TextView errorTv;
    private MediaPlayer ring;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
                             Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment_qibla, container,
false);
        initializeAll();
        return view;
    }
}
```

```

@Override
public void onCreateView(@NonNull View view, @Nullable Bundle
savedInstanceState) {
    super.onCreateView(view, savedInstanceState);

    directionContainer.addView(rose);
    rose.invalidate();

    refreshLayout.setOnRefreshListener(refreshListener);

    //check is all type of run time permission enabled or not if not
enabled then show again
    checkPermission();
    presenter.startCalculatingLocation();

    playSound();
}

private void initializeAll() {
    qiblaDistance = view.findViewById(R.id.idDistance);
    qiblaDegree = view.findViewById(R.id.idDegree);

    directionContainer = view.findViewById(R.id.cantainer_layout);
    rose = new DrawCompass(context);
    presenter = new QiblaPresenter(this, context);
    refreshLayout = view.findViewById(R.id.refreshLayout);
    errorTv = view.findViewById(R.id.errorTv);
    ring= MediaPlayer.create(getContext(),R.raw.prayer_allahu_akbar);
}

@Override
public void onAttach(Context context) {
    super.onAttach(context);
    this.context = context;
}

/*
 * initialize qibla current status(view update)
 */
@Override
public void setQiblaInfo(String qiblaDegree, String qiblaDistance) {
    this.qiblaDegree.setText(qiblaDegree);
    this.qiblaDistance.setText(qiblaDistance);
}

//this will generate new compass direction
@Override
public void changeCompassDirection(float directionsNorth, float
directionsQibla, float degree) {
    rose.setDirections(directionsNorth, directionsQibla, degree);
}

@Override
public void notifyNoInternetConnection() {
    Toast.makeText(context, R.string.no_internet_connection,

```

```

Toast.LENGTH_SHORT).show();
    }

    @Override
    public void showSensorNotAvailable() {
        Snackbar.make(view,
R.string.magnetic_sensor_not_available, Snackbar.LENGTH_SHORT).show();
        errorTv.setVisibility(View.VISIBLE);
    }

    /*
     * This method will check is GPS is turned on or off if off then this
    method will show a dialog
     */
    @Override
    public void notifyNotEnabledGPS() {

        AlertDialog.Builder alertDialog = new AlertDialog.Builder(context);
        // Setting Dialog Title

        alertDialog.setTitle(getContext().getResources().getString(R.string.gps_set
ting_title));

        // Setting Dialog Message

        alertDialog.setMessage(getContext().getResources().getString(R.string.gps_s
etting_message));

        // On pressing Settings button

        alertDialog.setPositiveButton(getContext().getResources().getString(R.strin
g.settings), new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {
                Intent intent = new
Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS);
                startActivity(intent);
            }
        });
        // on pressing cancel button

        alertDialog.setNegativeButton(getContext().getResources().getString(R.strin
g.cancel), new DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int which) {
                dialog.cancel();
            }
        });

        // Showing Alert Message
        alertDialog.show();
    }

    SwipeRefreshLayout.OnRefreshListener refreshListener = new
SwipeRefreshLayout.OnRefreshListener() {
        @Override
        public void onRefresh() {

            presenter.startCalculatingLocation();
            refreshLayout.setRefreshing(false);
        }
    };

```



```

    public boolean checkPermission() {

        final String[] PERMISSIONS =
{Manifest.permission.ACCESS_FINE_LOCATION,Manifest.permission.WRITE_EXTERNAL_STORAGE};

        if(!hasPermissions(getContext(), PERMISSIONS)){

            ActivityCompat.requestPermissions(getActivity(),
                PERMISSIONS,
                MY_PERMISSIONS_REQUEST_LOCATION);

            return false;
        } else {
            //permission already granted
            //like android version < 5(lollipop) don,t need runtime
permission

            return true;

        }

    }

    public static boolean hasPermissions(Context context, String...
permissions) {
        if (context != null && permissions != null) {
            for (String permission : permissions) {
                if (ActivityCompat.checkSelfPermission(context, permission)
!= PackageManager.PERMISSION_GRANTED) {
                    return false;
                }
            }
        }
        return true;
    }

    @Override
    public void onRequestPermissionsResult(int requestCode,
String permissions[], int[]
grantResults) {
        switch (requestCode) {
            case MY_PERMISSIONS_REQUEST_LOCATION: {
                // If request is cancelled, the result arrays are empty.
                if (grantResults.length > 0
                    && grantResults[0] ==
PackageManager.PERMISSION_GRANTED) {
                    if (ContextCompat.checkSelfPermission(getContext(),
Manifest.permission.ACCESS_FINE_LOCATION)
                        == PackageManager.PERMISSION_GRANTED) {
                        //permission granted
                    }

                } else {
                    permissionDenied();
                }
                return;
            }

        }
    }

```

```

    }

    private void permissionDenied() {
        Toast.makeText(getApplicationContext(),
R.string.permission_denied, Toast.LENGTH_SHORT).show();
    }

    public void playSound() {
        if (!ring.isPlaying()) {
            ring.start();
        }
    }

    @Override
    public void onPause() {
        presenter.onPause();
        ring.stop();
        super.onPause();
    }

    @Override
    public void onResume() {
        presenter.onResume();
        super.onResume();
    }
}

```

QURAN:

```

package net.a6te.lazycoder.muslim_pro_islamicremainders.fragments;

import android.media.MediaPlayer;
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.v4.app.Fragment;
import android.text.Editable;
import android.text.TextWatcher;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;

import com.github.barteksc.pdfviewer.PDFView;
import com.github.barteksc.pdfviewer.listener.OnLoadCompleteListener;
import com.github.barteksc.pdfviewer.listener.OnPageChangeListener;
import com.github.barteksc.pdfviewer.scroll.DefaultScrollHandle;

import com.mancj.materialsearchbar.MaterialSearchBar;
import com.shockwave.pdfium.PdfDocument;

import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.MVPPresenter;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.MVPView;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MVP.QuranPresenter;
import net.a6te.lazycoder.muslim_pro_islamicremainders.MainActivity;
import net.a6te.lazycoder.muslim_pro_islamicremainders.R;
import net.a6te.lazycoder.muslim_pro_islamicremainders.adapters.CustomSuggestionsA

```

```

dapter;
import
net.a6te.lazycoder.muslim_pro_islamicremainders.interfaces.OnSearchItemClic
k;
import net.a6te.lazycoder.muslim_pro_islamicremainders.model.Surah;

import java.util.ArrayList;
import java.util.List;

import static android.content.Context.LAYOUT_INFLATER_SERVICE;

/**
 * A simple {@link Fragment} subclass.
 */
public class Quran extends Fragment implements MVPView.QuranView,
OnPageChangeListener, OnLoadCompleteListener, OnSearchItemClick {

    private View view;
    public static final String SAMPLE_FILE = "quraan.pdf";
    PDFView pdfView;
    String pdfFileName;
    private ArrayList<Surah> surahs;
    private LayoutInflater inflater;

    private MaterialSearchBar searchBar;
    private CustomSuggestionsAdapter customSuggestionsAdapter;
    private MVPPresenter.QuranPresenter presenter;
    private MediaPlayer ring;

    @Override
    public View onCreateView(LayoutInflater inflater, ViewGroup container,
        Bundle savedInstanceState) {
        // Inflate the layout for this fragment
        view = inflater.inflate(R.layout.fragment_quran, container,
false);
        initializeAll();
        playSound();//it will play allhu akbar sound

        return view;
    }

    private void initializeAll() {
        ring= MediaPlayer.create(getContext(),R.raw.prayer_allahu_akbar);
        pdfView = view.findViewById(R.id.pdfView);
        surahs = new ArrayList<>();

        inflater = (LayoutInflater)
getContext().getSystemService(LAYOUT_INFLATER_SERVICE);
        searchBar = view.findViewById(R.id.searchBar);

        presenter = new QuranPresenter(this);
    }

    @Override
    public void onViewCreated(@NonNull View view, @Nullable Bundle
savedInstanceState) {
        super.onViewCreated(view, savedInstanceState);

        displayFromAsset(SAMPLE_FILE,0);//default pdf will be start from 0

```

```

page

    /*
    * search bar default settings*/
    try {
        searchBar.setMaxSuggestionCount(2);
        searchBar.setHint(getString(R.string.find_surah));

        searchBar.addTextChangedListener(textWatcher);

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

}

/*
* initialize PDF view [PDF is located on asset folder]
* */
private void displayFromAsset(String assetFileName, int pageNumber) {
    pdfFileName = assetFileName;

    pdfView.fromAsset(SAMPLE_FILE)
        .defaultPage(pageNumber)
        .enableSwipe(true)
        .swipeHorizontal(false)
        .onPageChange(this)
        .enableAnnotationRendering(true)
        .onLoad(this)
        .scrollHandle(new DefaultScrollHandle(this.getContext()))
        .load();

}

@Override
public void onPageChanged(int page, int pageCount) {
    getActivity().setTitle(String.format("%s %s / %s", pdfFileName,
page + 1, pageCount));
}

@Override
public void loadComplete(int nbPages) {
    printBookmarksTree(pdfView.getTableOfContents(), "-");

}

public void printBookmarksTree(List<PdfDocument.Bookmark> tree, String
sep) {
    for (PdfDocument.Bookmark b : tree) {

        surahs.add(new Surah(b.getTitle(), b.getPageIdx()));
        if (b.hasChildren()) {
            printBookmarksTree(b.getChildren(), sep + "-");
        }
    }

    presenter.prepareSearchAdapter(inflater, surahs);
}

```

```

    }

    TextWatcher textWatcher = new TextWatcher() {
        @Override
        public void beforeTextChanged(CharSequence charSequence, int i, int
i1, int i2) {
        }

        @Override
        public void onTextChanged(CharSequence charSequence, int i, int i1,
int i2) {
            try {
                // send the entered text to our filter and let it manage
everything
customSuggestionsAdapter.getFilter().filter(searchBar.getText());
            } catch (Exception e) {
                e.printStackTrace();
            }
        }

        @Override
        public void afterTextChanged(Editable editable) {

        }
    };

    @Override
    public void initializeSearchView(CustomSuggestionsAdapter adapter) {
        if (adapter != null) {
            try {
                searchBar.setCustomSuggestionAdapter(adapter);
                this.customSuggestionsAdapter = adapter;
            } catch (Exception e) {
                e.printStackTrace();
            }
        }
    }

    @Override
    public void onSearchItemClick(String indexNo) {

        searchBar.disableSearch();
        pdfView.jumpTo(Integer.parseInt(indexNo));

    }

    public void playSound(){
        if (!ring.isPlaying()) {
            ring.start();
        }
    }

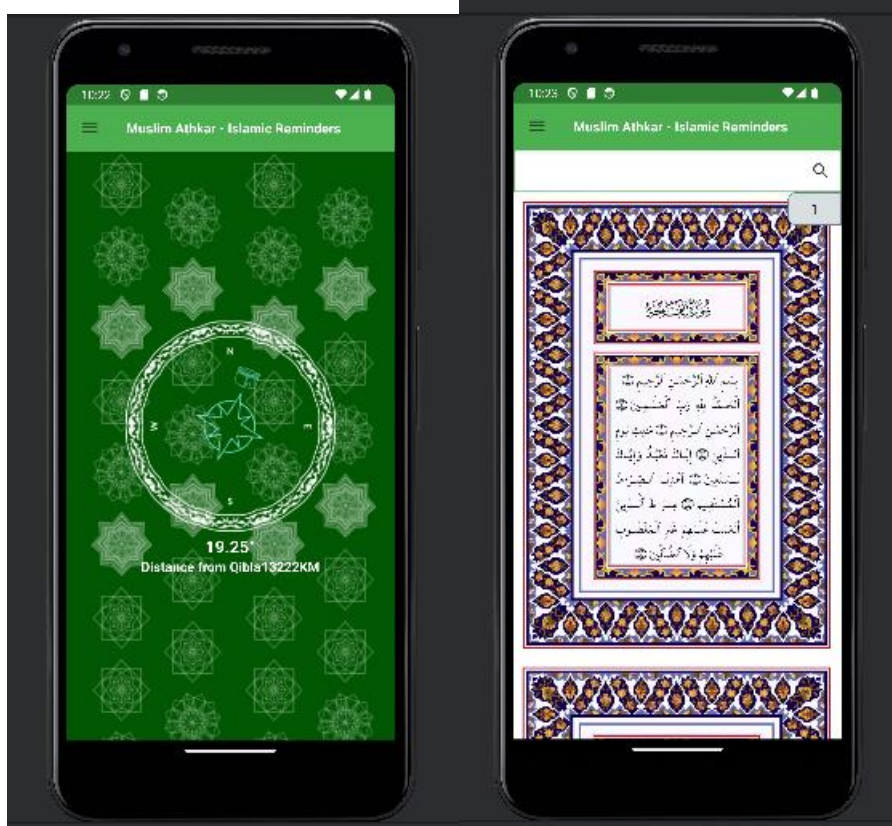
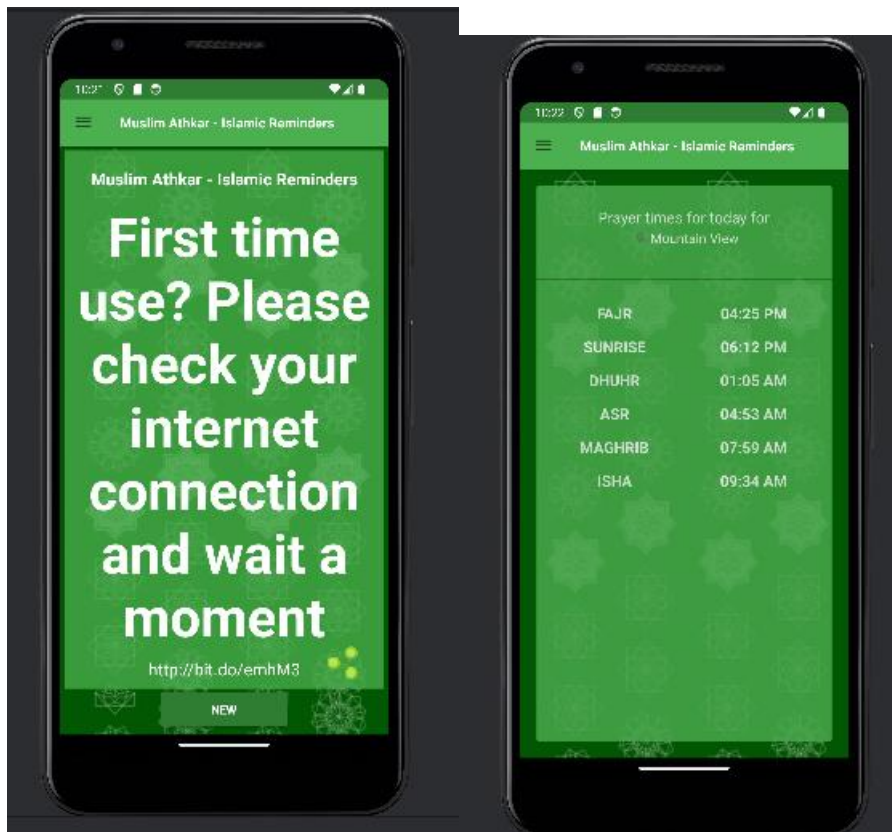
    @Override
    public void onPause() {

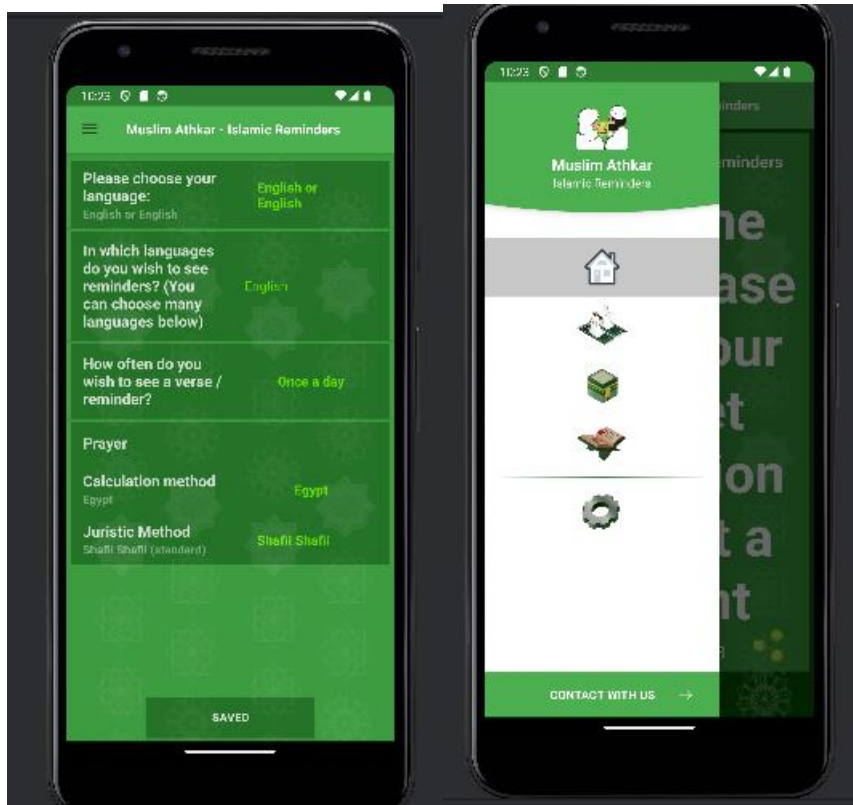
```

```
ring.stop();  
super.onPause();  
}  
}
```

OUTPUT:

210171601004





RESULT :

Therefore, the Islamic Application has been developed successfully using Java in Android Studio.