package com.example.covid\_19alertapp.activities;

import androidx.annotation.Nullable;

import androidx.appcompat.app.AlertDialog;

import androidx.appcompat.app.AppCompatActivity;

import androidx.lifecycle.Observer;

import androidx.work.Constraints;

import androidx.work.PeriodicWorkRequest;

import androidx.work.WorkInfo;

import androidx.work.WorkManager;

import android.content.DialogInterface;

import android.content.Intent;

import android.os.Bundle;

import android.util.Log;

import android.view.View;

import android.widget.Button;

import com.example.covid\_19alertapp.R;

import com.example.covid\_19alertapp.extras.Constants;

import com.example.covid\_19alertapp.extras.LogTags;

import com.example.covid\_19alertapp.services.BackgroundWorker;

import com.example.covid\_19alertapp.sharedPreferences.MiscSharedPreferences;

import java.util.List;

import java.util.concurrent.TimeUnit;

public class MenuActivity extends AppCompatActivity {

/\*

starter activity to test and get the permissions + all time running start worker

overwrite or edit this later, keeping the permission codes

\*/

Button home\_btn;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

home\_btn = findViewById(R.id.home\_button\_menu);

home\_btn.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

finish();

}

});

// start background worker for always

startWorker();

}

private void startWorker() {

if(!MiscSharedPreferences.getBgWorkerStatus(this)){

Constraints constraints = new Constraints.Builder()

.setRequiresBatteryNotLow(true)

.setRequiresCharging(false)

.build();

PeriodicWorkRequest promptNotificationWork =

new PeriodicWorkRequest.Builder(BackgroundWorker.class, 30, TimeUnit.MINUTES)

.setConstraints(constraints)

.addTag(Constants.background\_WorkerTag)

.build();

WorkManager.getInstance(getApplicationContext()).getWorkInfoByIdLiveData(promptNotificationWork.getId())

.observe(this, new Observer<WorkInfo>() {

@Override

public void onChanged(@Nullable WorkInfo workInfo) {

if (workInfo != null && workInfo.getState() == WorkInfo.State.ENQUEUED) {

Log.d(LogTags.Worker\_TAG, "onChanged: worker is enqueued");

// set shared preference true

MiscSharedPreferences.setBgWorkerStatus(MenuActivity.this, true);

}

if (workInfo != null && workInfo.getState() == WorkInfo.State.CANCELLED) {

Log.d(LogTags.Worker\_TAG, "onChanged: worker was stopped. why?");

// set shared preference false

MiscSharedPreferences.setBgWorkerStatus(MenuActivity.this, false);

}

}

});

WorkManager.getInstance(getApplicationContext())

.enqueue(promptNotificationWork);

}

}

public void uploadClick(View view) {

if(!MiscSharedPreferences.getUploadStatus(this)) {

Intent intent = new Intent(this, UploadLocationsActivity.class);

startActivity(intent);

}

else{

// show dialog and prevent

AlertDialog.Builder builder = new AlertDialog.Builder(this);

builder.setMessage(getText(R.string.cant\_upload\_twice\_message))

.setCancelable(false)

.setPositiveButton(getText(R.string.permissions\_dialogbox\_positive), new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.dismiss();

}

})

.setNegativeButton("Override", new DialogInterface.OnClickListener() {

@Override

public void onClick(DialogInterface dialog, int which) {

dialog.dismiss();

// TODO: remove this

Intent intent = new Intent(MenuActivity.this, UploadLocationsActivity.class);

startActivity(intent);

}

});

AlertDialog alertDialog = builder.create();

alertDialog.show();

}

}

public void startNewsFeed(View view)

{

startActivity(new Intent(getApplicationContext(),NewsFeedActivity.class));

}

public void openSettingsClick(View view) {

Intent intent = new Intent(this, TrackerSettingsActivity.class);

startActivity(intent);

}

public void showMatchedLocationsClick(View view) {

Intent intent = new Intent(getApplicationContext(), ShowMatchedLocationsActivity.class);

startActivity(intent);

}

public void startMyLocationsMap(View view) {

startActivity( new Intent(this, MyLocationsMapsActivity.class) );

}

}