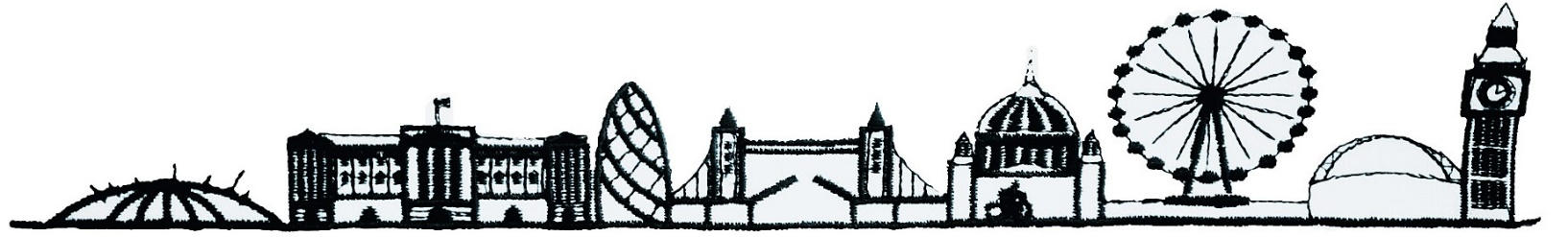
**INF 397 – Mobile Computing**

**Project 1: Android Virtual Travel App**

**London Edition**

developed by **Katalina Dimitrova** and **Nikola Toshev**

21 March 2017

Our team has **successfully** developed the Android project and implemented all of the suggested features in the specifications.

**Additionally**, we implemented a News feature, which fetches the 10 most popular news about London from reddit.com.

*We hereby declare that the submitted project software is all our own work. We did not copy any software from anyone else or anywhere else. No one but us developed the project software.*

*We also declare that the contribution declared for each group member in this report is truthful and correct.*

Katalina Dimitrova \_\_\_\_\_\_\_\_\_\_\_\_\_

Nikola Toshev \_\_\_\_\_\_\_\_\_\_\_\_\_

22 March, 2017

Functionalities:

* Information page – joint work
* Google Map – joint work
* Xml files – joint work
* News functionality – mostly Nikola
* Music service – mostly Nikola
* Navigation bar – mostly Nikola
* History – mostly Katalina
* Fun Facts – mostly Katalina
* Gallery – mostly Katalina
* UX and design (icons, colours, images) – mostly Katalina

The application has one Main Activity which includes the navigation bar. The News, History, Fun Facts, Gallery functionality are developed in separate fragments, which load into the Main Activity when called from the Navigation Bar. The Google Map, however, is called in a new activity.

onCreate() from MainActivity.java

@Override  
**protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
 Toolbar toolbar = (Toolbar) findViewById(R.id.***toolbar***);  
 setSupportActionBar(toolbar);  
  
 *// A media player object, which plays a song* **mediaPlayer** = MediaPlayer.*create*(**this**, R.raw.***london\_hymn***);  
 **mediaPlayer**.setLooping(**true**);  
  
 *// Playing and pausing the song when the floating button is pressed* FloatingActionButton fab = (FloatingActionButton) findViewById(R.id.***fab***);  
 fab.setOnClickListener(**new** View.OnClickListener() {  
 @Override  
 **public void** onClick(View view) {  
 **if**(**mediaPlayer**.isPlaying()){  
 **mediaPlayer**.pause();  
 }  
 **else** {  
 **mediaPlayer**.start();  
 }  
 }  
 });  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 ActionBarDrawerToggle toggle = **new** ActionBarDrawerToggle(  
 **this**, drawer, toolbar, R.string.***navigation\_drawer\_open***, R.string.***navigation\_drawer\_close***);  
 drawer.setDrawerListener(toggle);  
 toggle.syncState();  
  
 NavigationView navigationView = (NavigationView) findViewById(R.id.***nav\_view***);  
 navigationView.setNavigationItemSelectedListener(**this**);  
 navigationView.setCheckedItem(R.id.***nav\_information***);  
  
 *// Check that the activity is using the layout version with  
 // the fragment\_container FrameLayout* **if** (findViewById(R.id.***fragment\_container***) != **null**) {  
  
 *// However, if we're being restored from a previous state,  
 // then we don't need to do anything and should return or else  
 // we could end up with overlapping fragments.* **if** (savedInstanceState != **null**) {  
 **return**;  
 }  
  
 *// Create a new Fragment to be placed in the activity layout* **currentFragment** = **new** InformationFragment();  
  
 *// In case this activity was started with special instructions from an  
 // Intent, pass the Intent's extras to the fragment as arguments* **currentFragment**.setArguments(getIntent().getExtras());  
  
 *// Add the fragment to the 'fragment\_container' FrameLayout* getSupportFragmentManager().beginTransaction()  
 .add(R.id.***fragment\_container***, **currentFragment**).commit();  
 }

}

Navigation from MainActivity.java

@Override  
**public boolean** onNavigationItemSelected(MenuItem item) {  
 *// Handle navigation view item clicks here.* **int** id = item.getItemId();  
  
 **switch** (id) {  
 **case** R.id.***nav\_information***:  
 *// Create a new Fragment to be placed in the activity layout* **currentFragment** = **new** InformationFragment();  
  
 *// Add the fragment to the 'fragment\_container' FrameLayout* getSupportFragmentManager().beginTransaction()  
 .replace(R.id.***fragment\_container***, **currentFragment**).commit();  
  
 **break**;  
 **case** R.id.***nav\_news***:  
 *// Create a new Fragment to be placed in the activity layout* **currentFragment** = **new** NewsFragment();  
  
 *// Add the fragment to the 'fragment\_container' FrameLayout* getSupportFragmentManager().beginTransaction()  
 .replace(R.id.***fragment\_container***, **currentFragment**).commit();  
  
 **break**;  
 **case** R.id.***nav\_map***:  
 *// Starting a Google Maps activity* Intent intent = **new** Intent(**this**, GoogleMapsActivity.**class**);  
 startActivity(intent);  
  
 **...**

**case** R.id.***nav\_gallery***:  
 *// Create a new Fragment to be placed in the activity layout* **currentFragment** = **new** GalleryFragment();  
  
 *// Add the fragment to the 'fragment\_container' FrameLayout* getSupportFragmentManager().beginTransaction()  
 .replace(R.id.***fragment\_container***, **currentFragment**).commit();  
 **break**;  
 }  
  
 DrawerLayout drawer = (DrawerLayout) findViewById(R.id.***drawer\_layout***);  
 drawer.closeDrawer(GravityCompat.***START***);  
 **return true**;  
}

The xml file associated with the MainActivity has Drawer Layout and includes the layout of the navigation bar.

*<?***xml version="1.0" encoding="utf-8"***?>*<**android.support.v4.widget.DrawerLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:id="@+id/drawer\_layout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:fitsSystemWindows="true"  
 tools:openDrawer="start"**>  
  
 <**include  
 layout="@layout/app\_bar\_main"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"** />  
  
 <**android.support.design.widget.NavigationView  
 android:id="@+id/nav\_view"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="match\_parent"  
 android:layout\_gravity="start"  
 android:fitsSystemWindows="true"  
 app:headerLayout="@layout/nav\_header\_main"  
 app:menu="@menu/activity\_main\_drawer"** />  
  
</**android.support.v4.widget.DrawerLayout**>

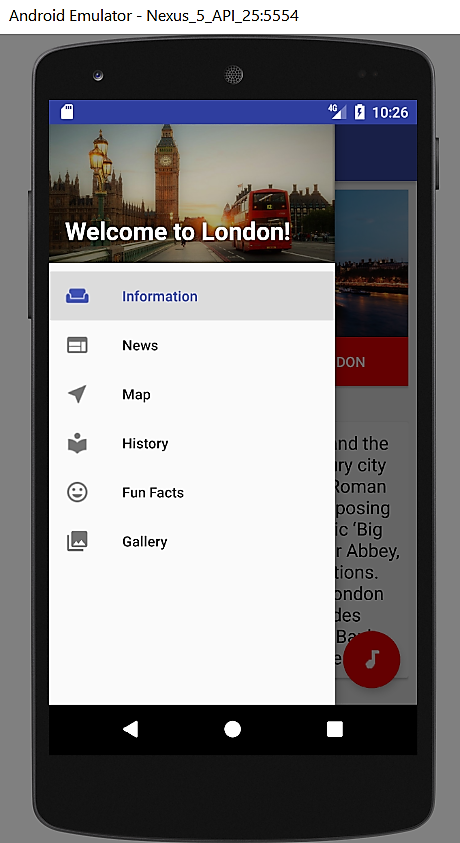
The app\_bar\_main loads the content of the activity as well as the floating icon, which plays music. Additionally, three xml files are associated with the side navigation bar. Their contents restrict the checkable behaviour and define the names and icons associated with the different tabs.

From activity\_main\_drawer.xml

<**group android:checkableBehavior="single"**>  
 <**item  
 android:id="@+id/nav\_information"  
 android:icon="@drawable/ic\_weekend\_black\_24dp"  
 android:title="@string/nav\_menu\_information\_option"** />  
 <**item  
 android:id="@+id/nav\_news"  
 android:icon="@drawable/ic\_web\_black\_24dp"  
 android:title="@string/nav\_menu\_news"** />

...

</**group**>



The Navigation bar with custom icons and pictures.

The information at the first page is loaded in the Information fragment. We are using ConstraintLayout, which was recently introduced by Google, since it provides improved performance and easier way to define alignment rules.

From fragment\_information.xml

<**android.support.v4.widget.NestedScrollView …**>  
  
 <**android.support.constraint.ConstraintLayout…**>  
  
 <**ImageView…** />  
  
 <**Button…** />  
  
 <**TextView…** />  
  
 <**android.support.v7.widget.CardView…**>

<**TextView…** />  
  
 </**android.support.v7.widget.CardView**>

</**android.support.constraint.ConstraintLayout**>

</**android.support.v4.widget.NestedScrollView**>

The java file simply inflates the layout for the fragment. Additionally, it handles the clicks on the button and image and opens a video about London. This happens in the onViewCreated method.

From InformationFragment.java

@Override  
**public void** onViewCreated(View view, @Nullable Bundle savedInstanceState) {  
 **super**.onViewCreated(view, savedInstanceState);  
  
  
 **btnVideo** = (Button) view.findViewById(R.id.***btnVideo***);  
 **londonMainImage** = (ImageView) view.findViewById(R.id.***london\_main\_image***);  
  
 View.OnClickListener videoClickListener = **new** View.OnClickListener() {  
  
 **public void** onClick(View v) {  
 startActivity(**new** Intent(Intent.***ACTION\_VIEW***, Uri.*parse*(**"https://www.youtube.com/watch?v=jiz0uJaFFII"**)));  
 Log.*i*(**"Video"**, **"Video Playing...."**);  
  
 }  
 };  
  
 **btnVideo**.setOnClickListener(videoClickListener);  
 **londonMainImage**.setOnClickListener(videoClickListener);  
}

3. A third page where you are to describe in detail what each member of your group did for the project. It is expected and required that each student contributes to approximately 50% of the design and implementation of the app.

A breakdown of tasks and functionalities undertaken by each student is required.

4. Extra pages with screen shots of your app in action. There are to be enough screen shots to demonstrate all features of the app. Annotate with text explanations each of your screenshots. Each screenshot is to occupy at least half a page.

