



Group 2: Night Out App

Kyle Defler, Joe Clements,
Adrian Carideo, Kendall Conley



Overview

- Android App
- Plan an itinerary of a 'Night Out'
 - Local Restaurants
 - Local Bars
 - Local Activities
- Show planned places on a map



Map

```
public class NightOutMap extends FragmentActivity implements OnMapReadyCallback {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_night_out_map);
        // Obtain the SupportMapFragment and get notified when the map is ready to be used.
        SupportMapFragment mapFragment = (SupportMapFragment) getSupportFragmentManager()
            .findFragmentById(R.id.map);
        assert mapFragment != null;
        mapFragment.getMapAsync(this);
    }

    @Override
    public void onMapReady(GoogleMap googleMap) {
        Bundle getLocation = getIntent().getExtras();
        assert getLocation != null;
        double usr_lat = getLocation.getDouble("latString");
        double usr_long = getLocation.getDouble("lonString");
        // Add a marker for the user's choice and move the camera
        LatLng usr_choice = new LatLng(usr_lat, usr_long);
        googleMap.addMarker(new MarkerOptions().position(new LatLng(usr_lat, usr_long)).title("Marker"));
        googleMap.moveCamera(CameraUpdateFactory.newLatLng(usr_choice));
    }
}
```

Side Bar

```
//defines and adds items to the drawer, sets demo listeners
private void addDrawerItems() {
    String[] osArray = {"Login", "History", "Destination Options", "Settings"}; //str array of items
    ArrayAdapter<String> mAdapter; //adapts strings to drawer items
    mAdapter = new ArrayAdapter<>( context: this, android.R.layout.simple_list_item_1, osArray);
    mDrawerList.setAdapter(mAdapter);

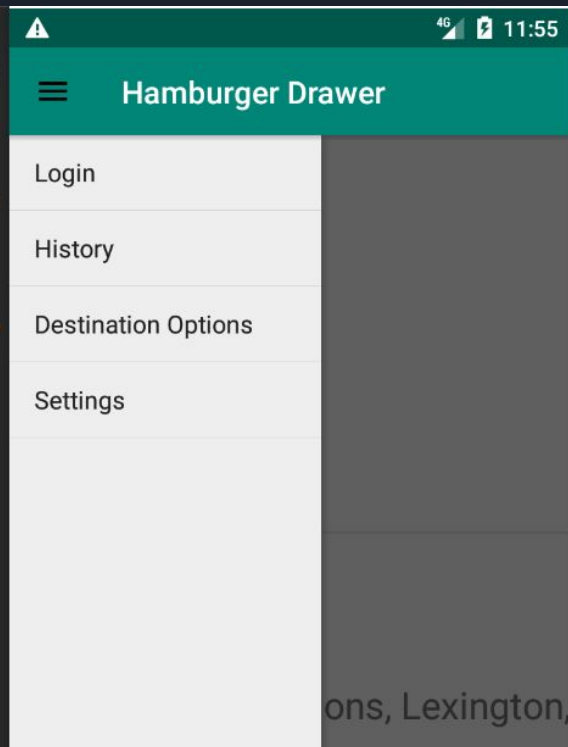
    mDrawerList.setOnItemClickListener((parent, view, position, id) -> {
        Toast.makeText( context: MainActivity.this, text: "Page NYI!", Toast.LENGTH_SHORT).show();
    });
}

//changes the title bar based on drawer status
private void setupDrawer() { //this activity, drawer's layout, drawable icon, accessibility strings
    mDrawerToggle = new ActionBarDrawerToggle( activity: this, mDrawerLayout, "Open navigation drawer",

        //called when a drawer has settled in a completely open state
        public void onDrawerOpened(View drawerView) {
            super.onDrawerOpened(drawerView);
            getSupportActionBar().setTitle("Hamburger Drawer");
            invalidateOptionsMenu(); // creates call to onPrepareOptionsMenu()
        }

        //Called when a drawer has settled in a completely closed state
        public void onDrawerClosed(View view) {
            super.onDrawerClosed(view);
            getSupportActionBar().setTitle(mActivityTitle);
            invalidateOptionsMenu(); // creates call to onPrepareOptionsMenu()
        }
    );

    mDrawerToggle.setDrawerIndicatorEnabled(true); //enables the drawer button
    mDrawerLayout.addDrawerListener(mDrawerToggle); //registers this listener
}
```





Filter Pages

Cost

\$ \$\$

\$\$\$

Distance

CLOSE MID

FAR

SOULFOOD

SUSHI

RANDOMIZE

APPLY



Inter-Activity Communication

```
button_apply.setOnClickListener((v) -> {  
    Intent filtersIntent = new Intent();  
    filtersIntent.putExtra( name: "foodFilters", value: costStr+", "+distStr+", "+typeStr);  
    setResult(RESULT_OK, filtersIntent);  
    finish();  
});
```

```
@Override  
public void onBackPressed() {  
    Intent filtersIntent = new Intent();  
    filtersIntent.putExtra( name: "foodFilters", value: costStr+", "+distStr+", "+typeStr);  
    setResult(RESULT_OK, filtersIntent);  
    finish();  
}
```



Inter-Activity Communication

```
//listeners to move to each filtering activity
Button food_btn = findViewById(R.id.foodSelect);
food_btn.setOnClickListener({v} -> {
    startActivityForResult(new Intent( packageContext MainActivity.this, FoodFilters.class),
        SET_FOOD_FILTERS_REQUEST); //expect a result
});
```

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (requestCode == SET_FOOD_FILTERS_REQUEST) {
        String choiceStr = data.getStringExtra( name: "foodFilters");
        if (choiceStr == null || resultCode != RESULT_OK) {
            choiceStr = "No food filters set!";
        }
        foodChoice.setText(choiceStr);
    }
    if (requestCode == SET_DRINK_FILTERS_REQUEST) {
```



Location

```
//check if permission has been granted by user
@Override
public void onRequestPermissionsResult(int requestCode,
                                       @NonNull String permissions[], @NonNull int[] grantResults) {
    switch (requestCode) {
        case MY_PERMISSIONS_REQUEST_FINE_LOCATION : {
            if (grantResults.length > 0
                && grantResults[0] == PackageManager.PERMISSION_GRANTED) {
                getLatLng(); //grabs lat long
            } else {
                Toast.makeText(MainActivity.this, "Location usage required, defaulting to Google HQ",
                               Toast.LENGTH_SHORT).show(); //app cannot function without location, tell user that
            }
        }
    }
}
```




Location

```
//gets lat and long from Location data
private void getLatLong() {
    LocationManager lm = (LocationManager) getSystemService(Context.LOCATION_SERVICE);
    @SuppressWarnings("MissingPermission") Location location = lm.getLastKnownLocation(LocationManager.GPS_PROVIDER);
    if (location != null) {
        longitude = location.getLongitude();
        latitude = location.getLatitude();
    } else {
        latitude = 38.0406;
        longitude = -84.5037;
    }

    String loc = getAddress(latitude, longitude);
    address_text = findViewById(R.id.address_text);
    address_text.setText(loc);
}
```

Yelp API Integration

- Allows for 5000 API calls per day.
- Search by categories, price range, coordinates, and distance from said coordinates.
- Runs asynchronously and updates screen with result.

```
{
  "id": "cQ0F5tDQ9CZ2X1RdZ--13w",
  "alias": "tonys-of-lexington-lexington",
  "name": "Tony's of Lexington",
  "image_url": "https://s3-media1.fl.yelpcdn.com/bphoto/qQxCbePoKt0RoX4otUuGw/o.jpg",
  "is_closed": false,
  "url": "https://www.yelp.com/biz/tonys-of-lexington-lexington?adjust_creative=IAAQ-Nu07GsHzYuM5XGQXQ&utm_campaign=yelp_api_v3&utm_medium=api_v3_business_search&utm_source=IAAQ-Nu07GsHzYuM5XGQXQ",
  "review_count": 287,
  "categories": [
    {
      "alias": "steak",
      "title": "Steakhouses"
    },
    {
      "alias": "seafood",
      "title": "Seafood"
    },
    {
      "alias": "wine_bars",
      "title": "Wine Bars"
    }
  ],
  "rating": 4.5,
  "coordinates": {
    "latitude": 38.049353030595,
    "longitude": -84.5003329838937
  },
  "transactions": [],
  "price": "$$$",
}
```

Yelp API Integration

```
public Request buildRequest(double lat, double lon, String item){
    String[] choices = item.split(",");
    String price = getPriceString(choices[0]);
    int radius = getDistance(choices[1].substring(1));
    String term = choices[2].substring(1);

    if (price == null){
        return new Request.Builder()
            .url("https://api.yelp.com/v3/businesses/search?open_now=true&categories=" + term + "&latitude=" + lat + "&longitude=" + lon + "&radius=" + radius)
            .get()
            .addHeader("Authorization", "Bearer C7UvU1P5eSCdaKyC0T5mjE8EXnhPm4lu_1GFDFAI_vzIzUkNNbgyl6pnaZMYFcU4p2E-93JFKP20URmNfBbcl2sKJ-lugBGY6FxyCRFUFjMuBFm9n3")
            .addHeader("Content-Type", "application/x-www-form-urlencoded")
            .addHeader("cache-control", "no-cache")
            .addHeader("Postman-Token", "fb69fb24-4846-4a93-9e4e-1199fa4693ff")
            .build();
    }

    else{
        return new Request.Builder()
            .url("https://api.yelp.com/v3/businesses/search?open_now=true&categories=" + term + "&price=" + price + "&latitude=" + lat + "&longitude=" + lon + "&radius=" + radius)
            .get()
            .addHeader("Authorization", "Bearer C7UvU1P5eSCdaKyC0T5mjE8EXnhPm4lu_1GFDFAI_vzIzUkNNbgyl6pnaZMYFcU4p2E-93JFKP20URmNfBbcl2sKJ-lugBGY6FxyCRFUFjMuBFm9n3")
            .addHeader("Content-Type", "application/x-www-form-urlencoded")
            .addHeader("cache-control", "no-cache")
            .addHeader("Postman-Token", "fb69fb24-4846-4a93-9e4e-1199fa4693ff")
            .build();
    }
}
```



Yelp API Integration

```
@SuppressWarnings("StaticFieldLeak")
private class downloadHandler extends AsyncTask<Request, JSONObject, JSONObject> {
    OkHttpClient client = new OkHttpClient();
    @Override
    protected JSONObject doInBackground(Request... requests) {
        try {
            Response response = client.newCall(requests[0]).execute();
            assert response.body() != null;
            JSONObject jsonObject = new JSONObject(response.body().string().trim());
            if(jsonObject.getInt("total") != 0){ //results found
                JSONArray myResponse = (JSONArray) jsonObject.get("businesses");
                Random rand = new Random();
                int objNum = rand.nextInt(myResponse.length());
                return (myResponse.getJSONObject(objNum));
            }
            return null;
        } catch (IOException e) {
            e.printStackTrace();
            return null;
        } catch (JSONException e) {
            e.printStackTrace();
            return null;
        }
    }
}
```

The Result

Happy China

1301 Winchester Rd Lexington, KY

38.044114, -84.458999

Winchell's

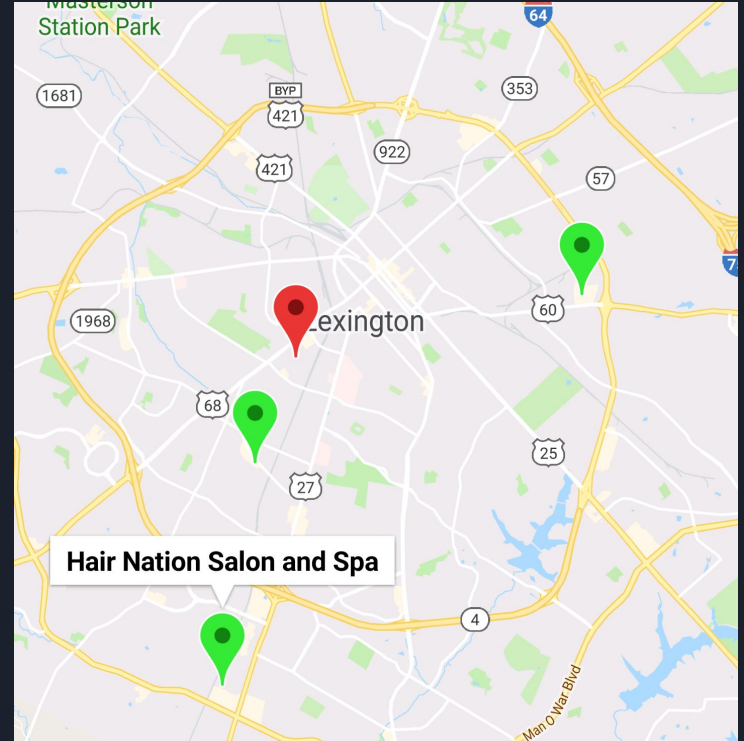
348 Southland Dr Lexington, KY

38.0169233514957, -84.5261542809509

Hair Nation Salon and Spa

152 W Tiverton Way Lexington, KY

37.9809026113987, -84.5328727889549





User Stories

Completed

- Yelp Integration
- Location Services
- Map
- Side Bar
- User Selection Pages
- Refresh Options

Still To-Do

- Login
- History/Favorites
- Dietary Restrictions
- Routes on Map

A decorative graphic on the left side of the slide. It consists of a blue parallelogram and a light green parallelogram, both tilted at an angle. The blue shape is in the foreground, and the green shape is partially behind it. They are set against a dark blue background with subtle diagonal stripes.

Demo