ROAD WIZARD

Presented by:
Megan Bishop & Courtney Valentine

Background

- Early navigators relied on landmarks, major constellations, and the sun's position in the sky to determine latitude and longitude
- Now we have location-based services to determine where we are, where we have been, and where we are going.

What is Road Wizard?

- User-based traffic condition Information application
- Users create tags at specified locations they want to report
- Road Wizard uses geo-coding and reverse geo-coding to find a location specified by the user.
- Users have the capability to add and delete their tags

Implementation

- Resources Used :
  - Eclipse IDE with the Android Development Tools(ADT) plugin built in.
    - Languages used-Java and XML
  - Android Software Development Toolkit (SDK)
  - Android Virtual Device Manager- Managers your emulator and connected devices
  - Google Play Services
  - Google Maps API Version 2
  - Motorola Droid Razr M

Location-Based Services

What Are They?

- Services that take in a mobile device's location data to show exact geographical position.
- A software application using LBS must request permission from users.

Location Based Services

There are two main methods that are most frequently used to find location

- The first is Control Plane Locating (positioning system)
  - It uses the service providers to get a location based off of radio signal delay off of cell-phone towers
- The second is Global Positioning System (GPS)
  - Network of 24 satellites that orbit the Earth.
  - GPS receivers locate at least 4 of these to determine its distance to each one.(Tri-lateration)
  - This gives location.

Geocoding

- What is it?
  - Is the translating of address into latitude and longitude coordinates to display a location on a map.
  - Ex. 1050 Union University Drive, Jackson, TN gives us the coordinates: Latitude = 35.681924, Longitude = -88.858222

Reverse Geocoding

- What is it?
  - It is the translating of latitude and longitude coordinates into an address that is displayed on the map.
  - Ex. Latitude: 35.681924,Longitude:-88.858222

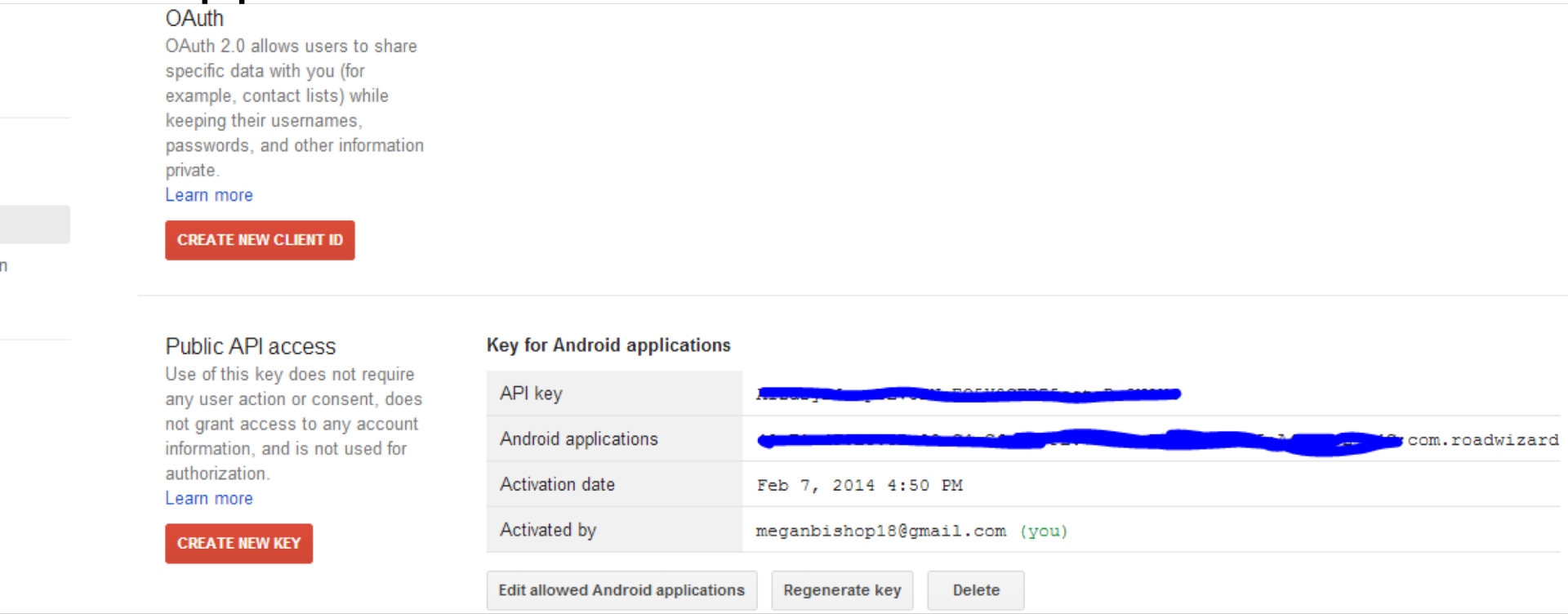
coordinates give us 1050 Union University Drive.

Google Maps API v2

- The Google Map API automatically handles the access to Google Map servers, data downloading, map display, and touch gestures.
- This API also enables users to add markers and overlays and to change users' views of the map.
- It is available in the Google Play Services library plugin for Android version 2.2 or higher.

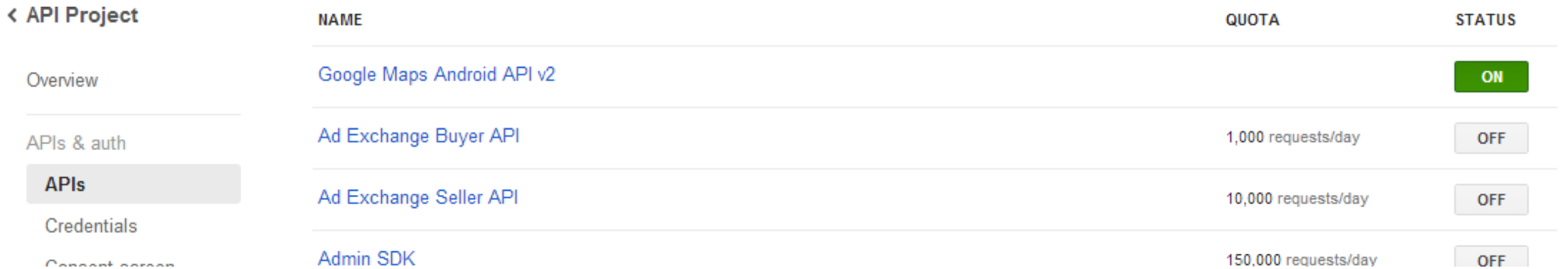
Obtaining API Key

- Needs your application's unique "finger print" (sha1) that Google Maps uses to identify your application.



Activating Google Maps API

- After signing up for the API, you go to the APIs subsection and find Google Maps Android API in the list and click the button to ON.



Using API Key

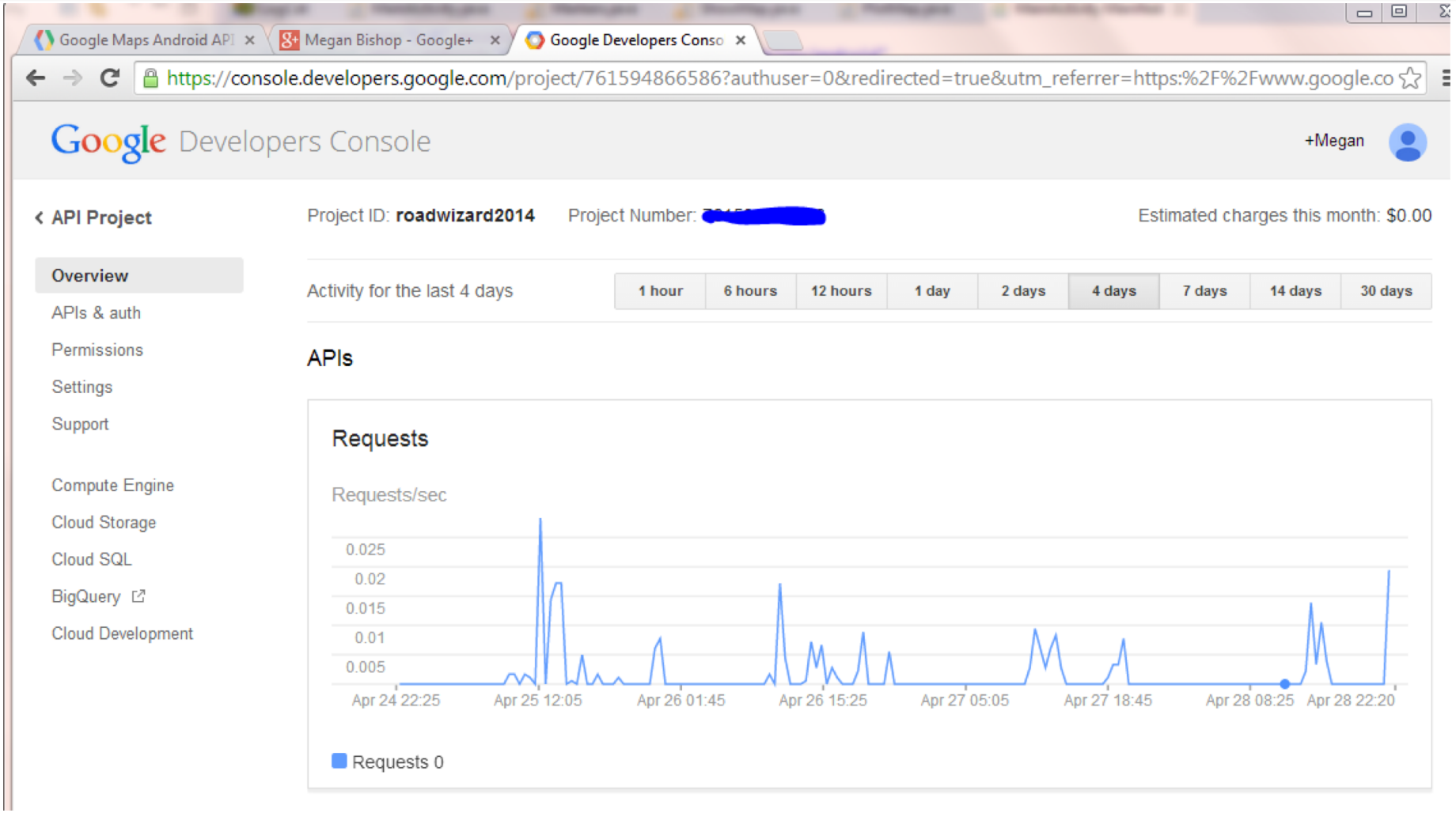
- Belongs in a meta-data block within the Manifest file.

```
    <meta-data
        android:name="com.google.android.maps.v2.API_KEY"
        android:value="[redacted]" />

    <meta-data
        android:name="com.google.android.gms.version"
        android:value="@integer/google_play_services_version" />
</application>

anifest>
```

Road Wizard Stats Overview



Android Virtual Device Manager

- The AVD allows you to model an actual Android device by defining hardware and software options.
- The device created is then used to run and test your Android application.
- The AVD also detects physical devices that are plugged into the b computer via USB and allows the user to run the application on the device if desired.

Permissions Required

- *Manifest*
- *android.permission.INTERNET*
  - *To download map tiles from Google Maps servers*
- android.permission.WRITE_EXTERNAL_STORAGE
  - to cache tile map data in the external storage area.
- com.google.android.providers.gsf.permission.READ_GSERVICES
  - to access Google web-based services.
- android.permission.ACCESS_NETWORK_STATE
  - to check connection to see if data can be downloaded.
- android.permission.ACCESS_COARSE_LOCATION
  - to use of WiFi or mobile data to detect your location
- android.permission.ACCESS_FINE_LOCATION
  - To use GPS to detect your location

Permissions Example Code

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.roadwizard"
    android:versionCode="1"
    android:versionName="1.0" >

    <uses-sdk
        android:minSdkVersion="11"
        android:targetSdkVersion="18" />

    <!-- Permissions that are required for retrieving map data and using the map and location services -->
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
    <uses-permission android:name="com.google.android.providers.gsf.permission.READ_GSERVICES"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-feature android:glEsVersion="0x00020000" android:required="true"/>

    <application
```

Map Views

- Traffic View
  - Shows how congested road traffic is
    - public final void setTrafficEnabled (boolean enabled)
- Satellite View
  - Displays satellite images for the map
  - public static final int MAP_TYPE_SATELLITE()
- Street View
  - Displays panoramic views from the position
  - Private void showStreetView(double lat, double lon)
- Points of Interest for College Bound Students
  - Private college markers
  - Links to college websites in information window

Adding Markers

- We use the addMapMarker method to create markers

```
}
private void addMapMarker(double lat, double lon, float markerColor,
        String title, String snippet){

    if(map != null){
        Marker marker = map.addMarker(new MarkerOptions()
        .title(title)
        .snippet(snippet)
        .position(new LatLng(lat,lon))
        .icon(BitmapDescriptorFactory.defaultMarker(markerColor))
        );
        marker.setDraggable(false);
        marker.showInfoWindow();
    } else {
        Toast.makeText(this, getString(R.string.nomap_error),
                Toast.LENGTH_LONG).show();
    }

}
```

Complications

- Issues getting map to show due to
  - Eclipse Default Android Emulator does not come Google Play Services enabled.
  - Not having the correct API Key
- Issues saving markers
- Issues sending markers
- Time constraints

Future Plans for RoadWizard

- Saved markers
- Sending markers
- Alerts and notifications
- More added points of interest
- Check for a Street View's Existence