CSE 162 Mobile Computing

Lab 4 Voice UI

Hua Huang
Department of Computer Science and Engineering
University of California, Merced, CA

Goal: achieve the following features

- Speech to text
- Search local data
- Text to speech
- Display the location in the map

Voice interface

- Especialy useful for devices with small or no screens
  - such as a smartwatch

Create a blank activity

```
public class MainActivity extends WearableActivity {
    private TextView mTextView;
    ArrayList<String> mylist;
    ArrayList<LatLng> response;
    boolean command_found;
    LatLng res;
    TextToSpeech t1;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        mTextView = (TextView) findViewById(R.id.text);

        // Enables Always-on
        setAmbientEnabled();
    }
}
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="@dimen/inner_frame_layout_padding"
    android:orientation="vertical"
    app:boxedEdges="all">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="start"
        android:onClick="start"/>

    <TextView
        android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text=" " />

</LinearLayout>
```

Free-form voice-to-text conversion

- generate an intent to invoke the Andoird speech recognition module
- RecognizerIntent

```
public void start(View view){
   displaySpeechRecognizer();
}

private static final int SPEECH_REQUEST_CODE = 0;

// Create an intent that can start the Speech Recognizer activity
private void displaySpeechRecognizer() {
   Intent intent = new Intent(RecognizerIntent.ACTION_RECOGNIZE_SPEECH);
   intent.putExtra(RecognizerIntent.EXTRA_LANGUAGE_MODEL,
         RecognizerIntent.LANGUAGE_MODEL_FREE_FORM);
   // This starts the activity and populates the intent with the speech text.
   startActivityForResult(intent, SPEECH_REQUEST_CODE);
}
```

- This callback is invoked when the Speech Recognizer returns.
- spokenText is what you have spoken.

```
// This callback is invoked when the Speech Recognizer returns.
// This is where you process the intent and extract the speech text from the intent.
@Override
protected void onActivityResult(int requestCode, int resultCode,
                                Intent data) {
   if (requestCode == SPEECH_REQUEST_CODE && resultCode == RESULT_OK) {
      List<String> results = data.getStringArrayListExtra(
            RecognizerIntent.EXTRA_RESULTS);
      String spokenText = results.get(0);
      // Do something with spokenText.
      Log.d("TAG",spokenText);
   }
   super.onActivityResult(requestCode, resultCode, data);
}
```

Use the text to search for information

In OnCreate

```
mylist = new ArrayList<>();
response=new ArrayList<>();
mylist.add("Toronto");
LatLng TORONTO=new LatLng(43.6532, -79.3832);
response.add(TORONTO);
mylist.add("Sydney");
LatLng SYDNEY = new LatLng(-33.85704, 151.21522);
response.add(SYDNEY);
```

```
// Do something with spokenText.
command_found=false;
for (String curVal : mylist){
   if (curVal.contains(spokenText)){
      command_found=true;
      Log.d("TAG","found");
      break;
   }
}

if (command_found){
   Log.d("TAG","enter");
   int idx=mylist.indexOf(spokenText);
   res=response.get(idx);
   mTextView.setText(spokenText);
   t1.speak(spokenText, TextToSpeech.QUEUE_FLUSH, null);

   Intent intent = new Intent(this, MapActivity.class);
   Bundle extras = new Bundle();

   extras.putDouble("long", res.longitude);
   extras.putDouble("lat", res.latitude);
   intent.putExtras(extras);
   startActivity(intent);
}
```

Read texts

In OnCreate

```
t1=new TextToSpeech(getApplicationContext(), new TextToSpeech.OnInitListener() {
    @Override
    public void onInit(int status) {
        if(status != TextToSpeech.ERROR) {
            t1.setLanguage(Locale.US);
        }
    }
});
```

// Do something with spokenText.

```
 t1.speak(spokenText, TextToSpeech.QUEUE_FLUSH, null);
```

Show a map view

Register the map api

- https://developers.google.com/maps/documentation/android-sdk/get-api-key

- In manifest file (**in the application section**):

```
<meta-data android:name="com.google.android.geo.API_KEY"
    android:value="YOUR KEY"/>
```

- Create an **empty** activity: MapActivity

```
public class MapActivity extends WearableActivity implements
OnMapReadyCallback, GoogleMap.OnMapLongClickListener{

    public LatLng LOC;
    GoogleMap mMap;

    private MapFragment mMapFragment;
    private DismissOverlayView mDismissOverlay;
```

- in the activity_map.xml

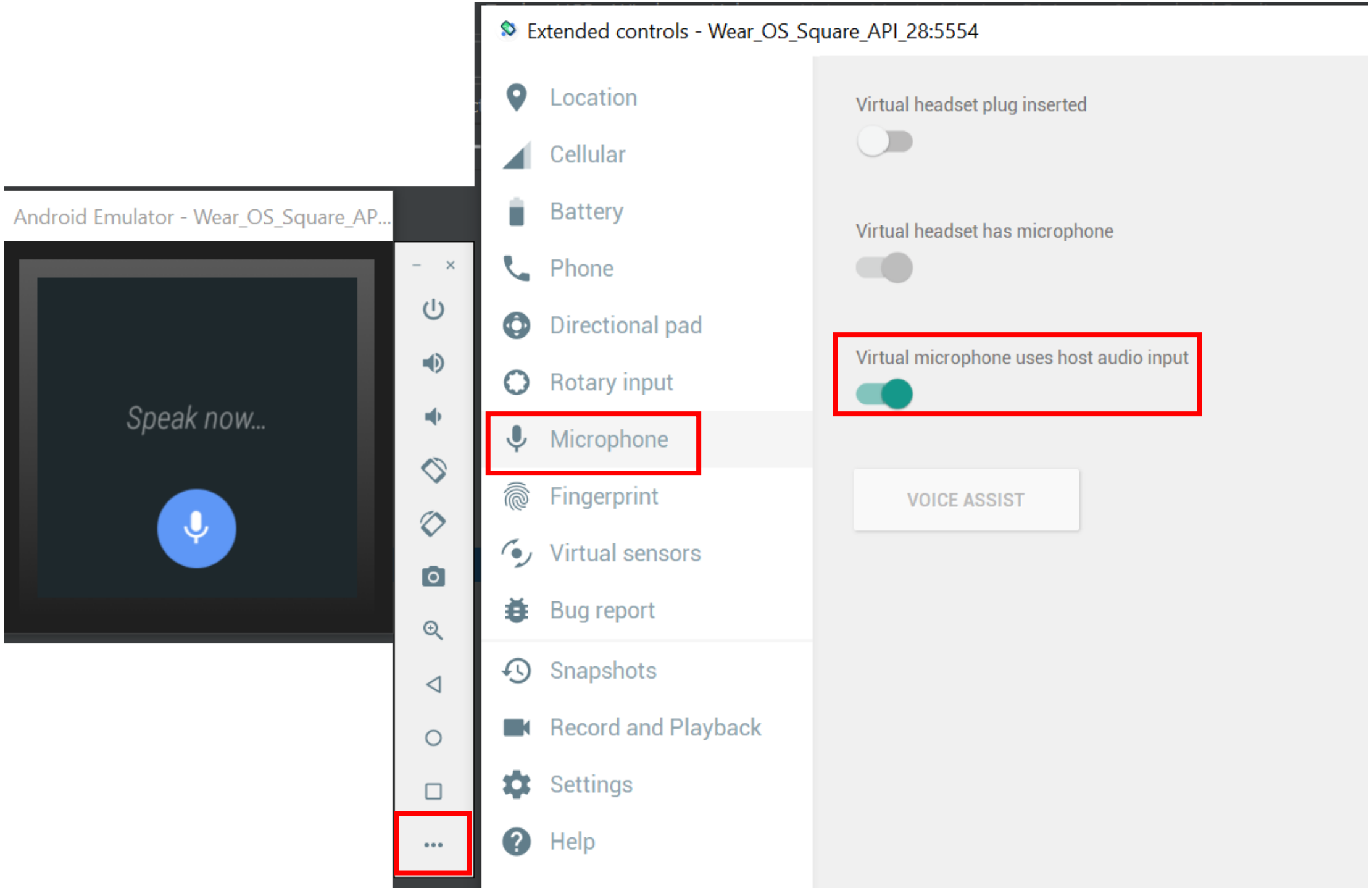
```
<?xml version="1.0" encoding="utf-8"?>
<FrameLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
   xmlns:map="http://schemas.android.com/apk/res-auto"
   android:id="@+id/root_container"
   android:layout_height="match_parent"
   android:layout_width="match_parent">

   <FrameLayout
      android:id="@+id/map_container"
      android:layout_width="match_parent"
      android:layout_height="match_parent">

      <fragment
         android:id="@+id/map"
         android:layout_width="match_parent"
         android:layout_height="match_parent"
         android:name="com.google.android.gms.maps.MapFragment"/>

   </FrameLayout>

   <android.support.wearable.view.DismissOverlayView
      android:id="@+id/dismiss_overlay"
      android:layout_height="match_parent"
      android:layout_width="match_parent"/>
</FrameLayout>
```

- In MapActivity.java, firstly prepare for the map view. Then read the coordinate sent by the other activity.

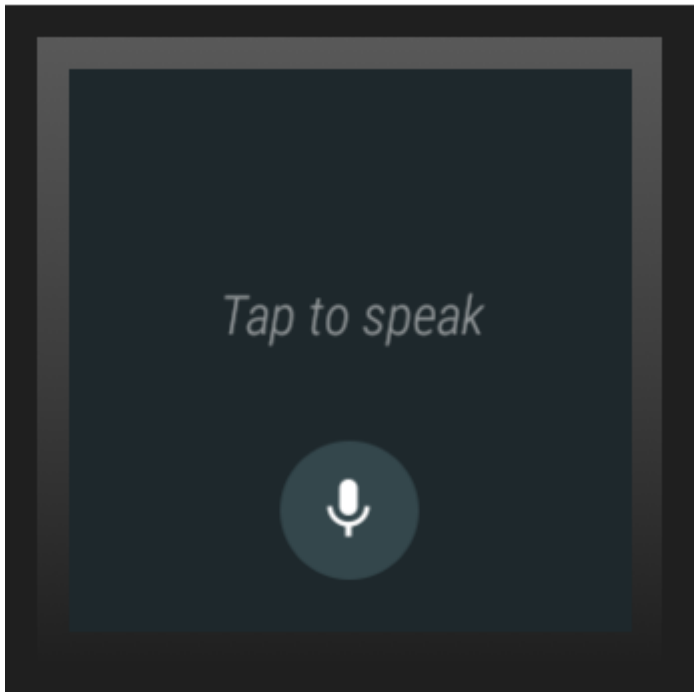
```
@Override
  protected void onCreate(Bundle savedInstanceState) {
     super.onCreate(savedInstanceState);
     setContentView(R.layout.activity_map);

     Intent intent = getIntent();
     Bundle extras=intent.getExtras();
     double longitude=extras.getDouble("long");
     double latitude=extras.getDouble("lat");

     LOC=new LatLng(latitude, longitude);


     mDismissOverlay =
           (DismissOverlayView) findViewById(R.id.dismiss_overlay);
     mDismissOverlay.setIntroText("test");
     mDismissOverlay.showIntroIfNecessary();

     mMapFragment = (MapFragment) getFragmentManager()
           .findFragmentById(R.id.map);
     mMapFragment.getMapAsync(this);


  }
```

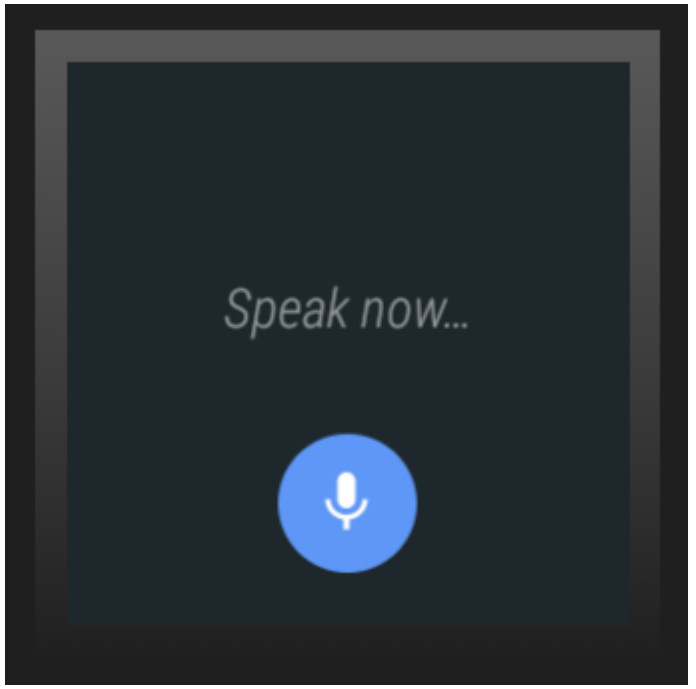
- Display the map with the appropriate view

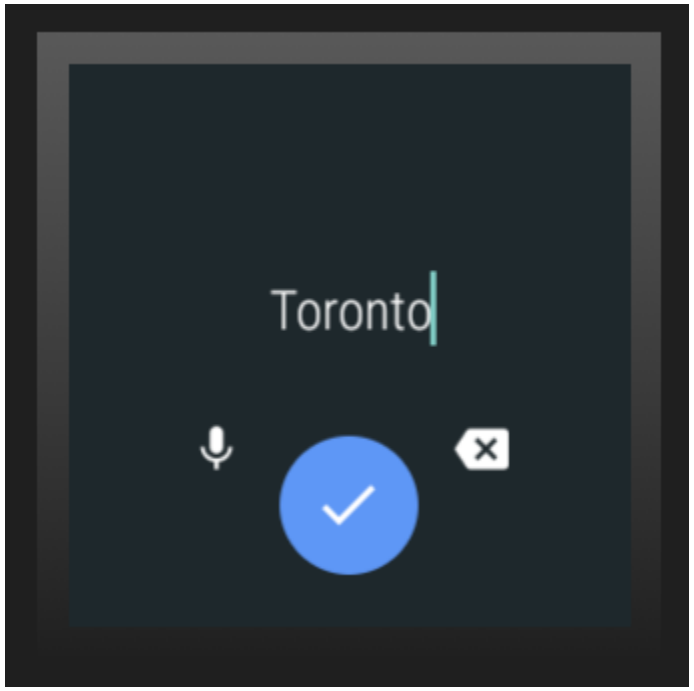
```
@Override
  public void onMapReady(GoogleMap map) {
     mMap = map;
     mMap.addMarker(new MarkerOptions().position(LOC)
           .title(""));

     mMap.moveCamera(CameraUpdateFactory.newLatLngZoom(LOC, 10));
     mMap.setOnMapLongClickListener(this);
  }
@Override
  public void onMapLongClick(LatLng latLng) {
     mDismissOverlay.show();
  }
```


Extended controls - Wear_OS_Square_API_28:5554
Location
Cellular
Battery
Phone
Directional pad
Rotary input
Microphone
Fingerprint
Virtual sensors
Bug report
Snapshots
Record and Playback
Settings
Help
Virtual headset plug inserted
Virtual headset has microphone
Virtual microphone uses host audio input
VOICE ASSIST
Android Emulator - Wear_OS_Square_AP...
Speak now...

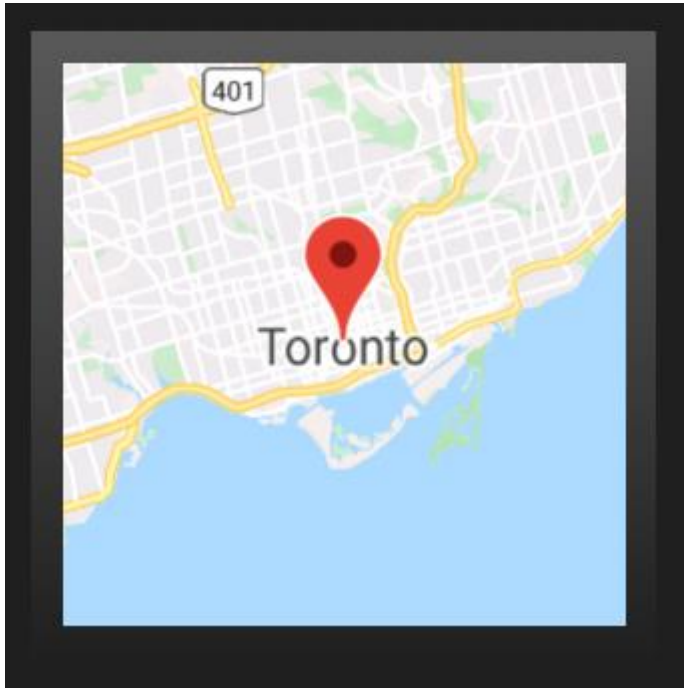

START

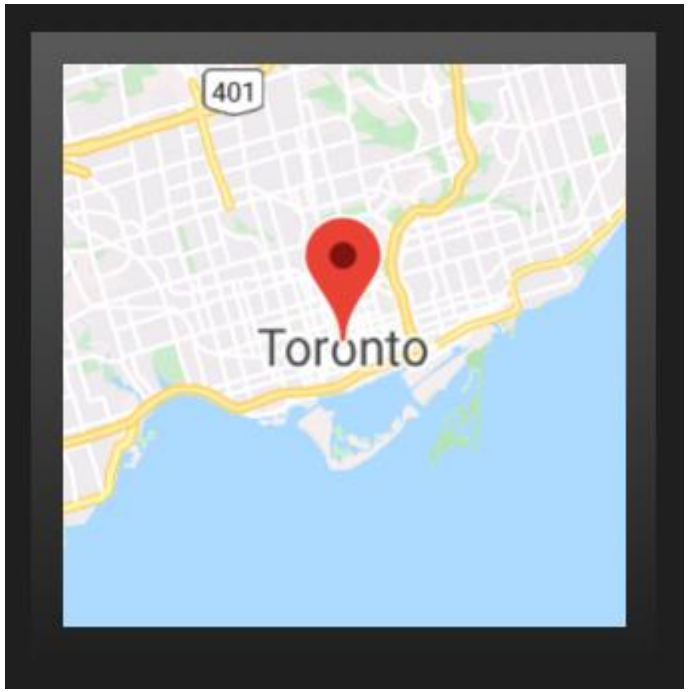

Tap to speak


Speak now...


Toronto


START
Toronto


401
Toronto


401
Toronto


401
Quit