

CS5530

Mobile/Wireless Systems

GPS/Location in Android

Yanyan Zhuang

Department of Computer Science

<http://www.cs.uccs.edu/~yzhuang>

Android Location Interface

1. Request permission in `AndroidManifest.xml` and code (Marshmallow+)
2. Create a Location Manager
3. Register a listener with the Location Manager to receive location updates
4. Define the listener that responds to location updates (similar to `IntentFilter` and `registerReceiver`)



Android Location Interface

- Permission

- `<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />`
`<!-- Needed only if your app targets Android 5.0 (API level 21) or higher. -->`
`<uses-feature android:name="android.hardware.location.gps" />`
- If app targets Android 5.0 (API level 21) or higher, must declare that your app uses `android.hardware.location.gps` hardware feature in the manifest file



Android Location Interface

- Create a location manager
 - `LocationManager locationManager = (LocationManager) this.getSystemService(Context.LOCATION_SERVICE);`
- Before requesting location updates, must check permission
 - ```
if (checkPermission(this)) {
 locationManager.requestLocationUpdates(....);
}
else{
 ActivityCompat.requestPermissions(this, ...);
}
```



# Android Location Interface

---

- Before requesting location updates, checking permission

- ```
public static boolean checkPermission(final Context context) {  
    return ActivityCompat.checkSelfPermission(context,  
        Manifest.permission.ACCESS_FINE_LOCATION)  
        == PackageManager.PERMISSION_GRANTED;  
}
```

- ```
if (checkPermission(this)) {
```

```
 locationManager.requestLocationUpdates(LocationManager.GPS_PROVIDER,
 MIN_TIME_BW_UPDATES, MIN_DISTANCE_CHANGE_FOR_UPDATES,
 locationManager);
```

```
 locationManager.addGpsStatusListener(gpsStatusListener);
```

```
}else{
```

```
 ActivityCompat.requestPermissions(this, PERMS_INITIAL, 111);
```

```
}
```



# Android Location Interface

---

- Register a listener (to request location updates)
  - `locationManager.requestLocationUpdates(  
 locationManager.GPS_PROVIDER,  
 MIN_TIME_BW_UPDATES,  
 MIN_DISTANCE_CHANGE_FOR_UPDATES, locationListener);`
  - `// The minimum distance to change Updates in meters  
private static final long MIN_DISTANCE_CHANGE_FOR_UPDATES = 10; // 10 m  
// The minimum time between updates in milliseconds  
private static final long MIN_TIME_BW_UPDATES = 1000 * 60 * 1; // 1 minute`



# Android Location Interface

---

- Define the listener that responds to location updates

- `LocationListener locationListener = new LocationListener() {`

```
@Override
public void onLocationChanged(Location location) {
 if (location != null)
 myLocation = location;
}
```

```
@Override
public void onStatusChanged(String provider, int status, Bundle extras) {
}
```

```
@Override
public void onProviderEnabled(String provider) {
}
```

```
@Override
public void onProviderDisabled(String provider) {
}
};
```



# Android Location Interface

---

- Get information about satellites
  - Register a satellite listener
    - ▶ `locationManager.addGpsStatusListener(gpsStatusListener)`
  - Implement `GpsStatus.Listener` to respond to satellite status changes
    - ▶ 

```
private GpsStatus.Listener gpsStatusListener = new GpsStatus.Listener() {
 @Override
 public void onGpsStatusChanged(int event) {

 }
};
```





# Android Location Interface

---

- GPS satellite listener

- ```
private GpsStatus.Listener gpsStatusListener = new GpsStatus.Listener() {
    @Override
    public void onGpsStatusChanged(int event) {
        if (event == GpsStatus.GPS_EVENT_SATELLITE_STATUS){
            if (checkPermission(getApplicationContext())) {
                GpsStatus gpsstatus = locationManager.getGpsStatus(null);
                Iterable<GpsSatellite> gpsit = gpsstatus.getSatellites();
                int numsat = 0;
                for (GpsSatellite sat : gpsit) {
                    Log.v(TAG, Float.toString(sat.getAzimuth()));
                    Log.v(TAG, Float.toString(sat.getElevation()));
                    Log.v(TAG, Integer.toString(sat.getPrn()));
                    Log.v(TAG, Float.toString(sat.getSnr()));
                    numsat++;
                }
            }
        }
    }
};
```



Android Location Interface

- `float getAzimuth()`
 - Returns the azimuth of the satellite in degrees. The azimuth can vary between 0 and 360
- `float getElevation()`
 - Returns the elevation of the satellite in degrees. The elevation can vary between 0 and 90
- `int getPrn()`
 - Returns the PRN (pseudo-random number) for the satellite
- `float getSnr()`
 - Returns the signal to noise ratio for the satellite.

