

Chapter 7 - Networking in Android

[Group 7] Ngo Sy Tung Lam

Chapter Objective

- Remind some network concepts
- Know how to make a HTTP request and process its JSON response

Permissions

- Privacy is an important aspect of Android programming.
- One of the ways to implement or improve security and privacy is to request permissions to several actions.
- The app will be stopped if it does not have proper permission.
- The action sensitivity is classified into 2 levels:
 - Normal level: has no effect on user privacy, e.g. internet access, set wallpaper, ...
 - Dangerous level: has some effect on user privacy, e.g. access contact list, access SMS, read from or write to external storage.
- Define the permission in the manifest:

```
<uses-permission android:name="your_permission" />
```

Embedded package

- Create URL from string:

```
URL url = new URL( 'your_url' );
```
- Make a request to server:

```
URLConnection connection =  
    (URLConnection) url.openConnection();  
connection.setRequestMethod( 'GET' );  
connection.setDoInput( true );  
connection.connect();
```

- Receive response:

```
int response = connection.getResponseCode();
InputStream is = connection.getInputStream();
```

- Process response:

Different response type - different ways to process

For JSON and XML: parsing

Image: decode to bitmap

```
Bitmap bitmap = BitmapFactory.decodeStream(is);
```

Close connection after all:

```
connection.disconnect();
```

- Limitation: a lot of codes, no queue, no cache

External Library: Volley

- Volley is an Android HTTP Client library
- It is very simple, powerful and extendable.
- In order to use Volley, clone it from its GitHub repository and add it as module.

- Create request queue:

```
RequestQueue queue = Volley.newRequestQueue(context);
```

- Create request:

```
ImageRequest imageRequest = new ImageRequest(...)
```

- Create listeners:

```
Response.Listener<Bitmap> listener =
    new Response.Listener<Bitmap>() {
    @Override
    public void onResponse(Bitmap response) {
        ...
    }
};
```

- Add request to queue:

```
queue.add(imageRequest);
```