

# Inhaltsverzeichnis

<b>1</b>	<b>Cheatsheet</b>	<b>2</b>
1.1	Android Manifest . . . . .	2
1.1.1	Application . . . . .	2
1.1.2	Activity + Intent Filter . . . . .	2
1.1.3	Permissions . . . . .	2
1.2	Application . . . . .	2
1.2.1	Application (Authentication + Firebase) . . . . .	2
1.2.2	Firebase . . . . .	3
1.3	Activity . . . . .	4
1.3.1	Broadcast Receiver . . . . .	4
1.3.2	Broadcast Receiver Register/Unregister . . . . .	4
1.3.3	Async Task . . . . .	4
1.3.4	Click Handler . . . . .	4
1.3.5	Fragment Manager . . . . .	5
1.3.6	Layout Inflater . . . . .	5
1.3.7	Start Intent . . . . .	5

# 1 Cheatsheet

## 1.1 Android Manifest

### 1.1.1 Application

```
<application
    android:name=".MocApplication"
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportsRtl="true"
    android:theme="@style/AppTheme">
    <!-- Activity + Intent Filter-->
</application>
```

### 1.1.2 Activity + Intent Filter

```
<activity android:name=".ui.MainActivity">
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>
```

### 1.1.3 Permissions

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

## 1.2 Application

### 1.2.1 Application (Authentication + Firebase)

```
public class MocApplication extends Application {

    private static MocApplication instance;
    public static MocApplication getInstance() { return instance; }
```

```

private static final String FIREBASE_MESSAGING_TOPIC = "moc";

@Override
public void onCreate() {
    super.onCreate();

    instance = this;

    FirebaseApp.initializeApp(this);
    FirebaseMessaging.getInstance().subscribeToTopic(
        FIREBASE_MESSAGING_TOPIC);
}

//AuthToken
private String authenticationToken;
public synchronized String getAuthenticationToken() {
    return authenticationToken;
}
public synchronized void setAuthenticationToken(String
    ↪ authenticationToken) { this.authenticationToken =
    ↪ authenticationToken; }
public synchronized boolean isAuthenticated() {
    return getAuthenticationToken() != null;
}
}

```

## 1.2.2 Firebase

```

@Override
public void onMessageReceived(RemoteMessage remoteMessage) {
    Map<String, String> data = remoteMessage.getData();

    ↪ if(REMOTE_MESSAGE_TYPE_VALUE.equals(data.get(REMOTE_MESSAGE_TYPE_KEY)))
    ↪ {
        String mocId = data.get(REMOTE_MESSAGE_MOC_ID_KEY);
        Intent intent = new Intent(MOC_CHANGED_INTENT_ACTION);
        if(mocId != null) {
            intent.putExtra(MOC_CHANGED_INTENT_MOC_ID, mocId);
        }
        LocalBroadcastManager.getInstance(this).sendBroadcast(intent);
    }
}
}

```

## 1.3 Activity

### 1.3.1 Broadcast Receiver

```
private final BroadcastReceiver broadcastReceiver = new
↳ BroadcastReceiver() {
    @Override
    public void onReceive(Context context, Intent intent) {
        updateSomething();
    }
};
```

### 1.3.2 Broadcast Receiver Register/Unregister

```
LocalBroadcastManager.getInstance(this).registerReceiver(broadcastReceiver,
↳ MyFirebaseMessagingService.mocChangedIntentFilter());
```

```
LocalBroadcastManager.getInstance(this).unregisterReceiver(broadcastReceiver);
```

### 1.3.3 Async Task

```
//String -> Input
//Void -> Progress
//LatLng -> Output
new AsyncTask<String, Void, LatLng>() {
    @Override
    protected LatLng doInBackground(String... strings) {
        // do something
        // DO NOT UPDATE UI HERE

        // Example: return new
        ↳ ServiceProxy().getData(MocManagerApplication.getInstance().getA
    }

    @Override
    protected void onPostExecute(final LatLng latLng) {
        // update UI
    }.execute()
}
```

### 1.3.4 Click Handler

```
element = findViewById(R.id.elementId);
element.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
    public void onClick(View view) {
        // do something
    }
});
```

### 1.3.5 Fragment Manager

```
//GMaps Fragment
fragment = (SupportMapFragment)
↳ getSupportFragmentManager().findFragmentById(R.id.fragmentId);
```

### 1.3.6 Layout Inflater

```
View view =
↳ LayoutInflater.from(parent.getContext()).inflate(R.layout.item,
↳ parent, false);
```

### 1.3.7 Start Intent

```
Intent intent = new Intent(this, Activity.class);
startActivity(intent);
```