

## Aufgabe 1 - Erste Schritte

### a) ADK

Lade das ADK herunter: <http://developer.android.com/sdk/index.html>

Das zip enthält einen eclipse-Ordner, in welchem ein komplett vorkonfiguriertes Eclipse zu finden ist.

### b) Hallo Welt (2 Punkte)

Erstelle in Eclipse ein neues, leeres Android-Projekt und führe es aus. Wo kommt der Text "Hello World" her? Beschreibe kurz, wie er sich an die Sprache des Benutzers anpassen lässt. Eine Abgabe dieses Zwischenstandes ist nicht erforderlich!

## Aufgabe 2 - Download-App

Im Folgenden soll nun eine kleine App entstehen, die auf Knopfdruck Dateien aus dem Internet lädt. Die Version für Masterstudenten soll zusätzlich den Fortschritt anzeigen.

### a) Button (2 Punkte)

Füge zuerst einen Button zum Layout hinzu, die Beschriftung soll nicht hardcoded sein (Eclipse zeigt sonst auch die Warnung "Hardcoded string..."). Er soll die volle Breite des Layouts einnehmen und so hoch sein wie nötig.

### b) onClick (2 Punkte)

Die einfachste Methode, einen Button mit einer Funktion zu versehen, ist das xml-Attribut onClick. Logge auf diese Weise jeden Klick auf den Button und lasse dir das Ergebnis im LogCat anzeigen.

### c) Download (3 Punkte)

Schreibe nun eine Methode, die eine Datei herunterlädt. Du benötigst folgende Importe:

- `import org.apache.http.HttpResponse;`
- `import org.apache.http.client.HttpClient;`
- `import org.apache.http.client.methods.HttpPost;`
- `import org.apache.http.impl.client.DefaultHttpClient;`

Der HTTP-Request kann so eingeleitet werden:

```
HttpClient httpclient = new DefaultHttpClient();
HttpPost httppost = new HttpPost("http://mimuc.de");
HttpResponse response = httpclient.execute(httppost);
```

Den Inhalt der Antwort ist hier nicht relevant, man kann sie jedoch so extrahieren:

```
String responseText = org.apache.http.util.EntityUtils.toString(response.getEntity());
```

Was passiert, wenn man diese Methode nach einem Buttonklick aufruft? Warum ist das in Android so umgesetzt?

### d) AsyncTask (4 Punkte)

Verschaffe dir anhand der Dokumentation einen Überblick über die Funktionsweise der Klasse AsyncTask. Warum wird der AsyncTask hier benötigt? Verwende nun einen AsyncTask, um auf Knopfdruck den Download anzustoßen und gebe danach eine beliebige Art von Feedback (z.B. ein Toast, eine Notification...). Vermutlich bekommst du anfangs eine IllegalStateException im LogCat. Was hast du vergessen?

## Aufgabe 3 - Erweiterung für Masterstudenten

Diese Aufgaben sind nur für Masterstudenten verpflichtend.

### e) ProgressBar (2 Punkte)

Füge nun in das Layout eine ProgressBar mit dem Attribut `style="@android:style/Widget.ProgressBar.Horizontal"` hinzu. Wie sieht die ProgressBar ohne dieses Attribut aus?

### f) AsyncTask 2 (2 Punkte)

Der AsyncTask bietet auch Methoden, um den Fortschritt des Tasks weiterzugeben. Lade nun mehrere Dateien herunter (darf auch mehrmals die gleiche sein) und ändere den Fortschritt der ProgressBar nach jedem Download anteilig. Eine Referenz für die ProgressBar erhält man in der Activity über ihre id: `ProgressBar progressBar = (ProgressBar) findViewById(R.id.progressbar);`.

## Abgabe

Die Lösungen bitte als komplettes Projekt zippen und über Uniworx abgeben. Die Antworten zu den Fragen könnt ihr in das Wurzelverzeichnis legen.