

Andorid Studio:

Tutorial – Praktischer Teil meiner Präsentation

Vorbereitung und Einführung in die Oberfläche von Android-Stuido

In dieser Beschreibung wird ausschließlich auf eine Android Studio Version 2.2.2 auf einem Windows-Betriebssystem eingegangen. Andere Betriebssysteme oder Softwareversionen weichen unter Umständen von dieser Beschreibung ab.

Stellen Sie sicher, dass Jave DK 8 und Android Studio auf Ihrem Computer installiert sind. Ist dies nicht der Fall, installieren Sie die Programme in der oben genannten Reihenfolge.

Starten Sie Android Studio

Ist bereits ein Projekt geöffnet, schließen Sie dieses. Wählen Sie hierzu „File“ und „Close Project“.

Nun befinden Sie sich auf der Startseite von Android Studio. Wählen Sie „Start a new Android Studio project“ um ein neues Projekt zu starten.

Im nächsten Dialogfeld haben Sie die Möglichkeit der App einen frei wählbaren Namen zu geben. Achten Sie bei der Wahl der Company Domain darauf, dass diese für eine Veröffentlichung der App weltweit einzigartig sein muss.

Aus dem Name und der Company Domain ergibt sich der Package name.

Unter „Project location“ haben Sie die Möglichkeit den Speicherort der Applikation zu verändern. Klicken Sie nach der Eingabe auf „Next“.

In diesem Fenster besteht die Möglichkeit die Mindestanforderungen der App an die Hardware zu definieren.

Für das folgende Beispiel wählen Sie bitte nur „Phone and Tablet“ mit einem Minimum SDK „API 15: Android 4.0.3 (IceCreamSandwich) aus.

Stellen Sie sicher, dass keine der anderen Optionen ausgewählt ist.

Klicken Sie auf „Next“ um fortzufahren.

Wählen Sie im nächsten Fenster „Empty Activity“ aus und bestätigen Sie Ihre Eingabe mit „Next“.

Lassen Sie die bereits voreingestellten Werte im nächsten Dialogfenster unverändert und klicken Sie auf „Finish“.

Nach einer kurzen Wartezeit gelangen Sie auf die Oberfläche von Android Studio. Diese ist in den Verzeichnisbaum auf der linken Bildschirmseite und den Editor auf der rechten Seite aufgeteilt.

Zum Verzeichnisbaum:

Dieser beinhaltet die geladene App mit drei Unterpunkten:

- manifest
- java
- res

Unter „manifest“ befindet sich die „AndroidManifest.xml“ diese ist für jede Android-App zwingend erforderlich. Sie beinhaltet Eigenschaften und Informationen der App wie zum Beispiel den Name oder die Zugriffsberechtigungen auf dem Endgerät.

Bitte nehmen Sie hier keine Einstellungen vor, da dies für das Programm nicht notwendig ist.

Der Unterordner „java“ beinhaltet wiederum Unterordner, welche nach der gewählten Company Domain benannt sind. Wählen Sie den Unterorder, welcher nicht mit „androidTest“ oder „test“ gekennzeichnet und ausgegraut ist.

In diesem Ordner befindet sich die „MainActivity.java“. Diese enthält den Java-Code der Anwendung.

Im Unterorder „res“ (für Ressourcen) werden Grafiken, Bilder, Strings, Anleitungen oder Interface-Informationen abgelegt.

Unter anderem befindet sich dort der Unterorder „layout“. Hier ist die „activity_main.xml“ hinterlegt. Diese Datei beschreibt das Aussehen der App. Android-Studio ermöglicht es diese entweder per Designer-Vorschau oder per XML-Code anzupassen.

Der Unterorder „values“ beheimatet die Datei „strings.xml“. Diese Datei ermöglicht es über eine ID Strings an die gewünschte Stelle einer App zu referenzieren. Vorteil hierbei ist, dass die Einbindung mehrerer Sprachen so möglich ist.

Layout und Code der App erstellen

Nun soll die App gestaltet werden. In diesem Beispiel wird eine einfache App geschrieben. Sie erstellt ein sogenanntes „Toast“. Ein Toast ist eine Nachricht an den Nutzer, welche auf dem Bildschirm erscheint und nach einer vordefinierten Zeit wieder ausgeblendet wird.

Öffnen Sie „res\layout\activity-main.xml“. Sie erhalten eine Vorschau der App. Bewegen Sie das bereitgestellte Textfeld „Hello World!“ durch Drag and Drop.

Klicken Sie auf das Textfeld. An der rechten Bildschirmseite erscheint ein Formular. Tragen Sie unter „ID“ einen einzigartigen, möglichst sprechenden Namen für dieses Feld ein.

Klicken Sie bei „text“ hinter „Hello World!“ auf die drei Punkte.

Ein neues Fenster öffnet sich. Wählen Sie in diesem in der rechten, oberen Ecke „Add new resource“ und anschließend „New String Value“.

Tragen Sie unter „Resource name“ eine möglichst sprechende Bezeichnung und unter „Resource Value“ den anzuzeigenden Text ein.

Bestätigen Sie Ihre Eingabe mit „OK“.

WelchesIn Sie nun in die „MainActivity.java“, diese befindet sich im Ordner „java“.

Schreiben Sie folgenden Code unter die Funktion „onCreate“.

```
public void onButtonTap(View v)
{
    Toast myToast = Toast.makeText
    (
        getApplicationContext(),
        "HelloWorld!",
        Toast.LENGTH_LONG
    );
    myToast.show();
}
```

Die Schlüsselwörter „View“ und „Toast“ werden rot unterstrichen. Klicken Sie auf das jeweilige Wort und drücken Sie „Alt“- + „Enter“-Taste um diese zu importieren.

Nun ist der Code der App vollständig. Gehen Sie zurück zur „activity_main.xml“. Wählen Sie im Designer-Fenster einen Button aus und ziehen Sie ihn an eine beliebige Stelle auf der App-Oberfläche.

Nun nehmen wir die Einstellungen des Buttons vor. Klicken Sie hierzu auf den Doppelpfeil in der rechten, oberen Bildschirmcke neben Properties.

Klicken Sie auf das leere Feld von „onClick“. Rechts neben dem Feld erscheint ein Pfeil nach unten. Klicken Sie auf den Pfeil und wählen Sie „onButtonTap (MainActivity)“.

Nun ist ein Drücken des Buttons mit dem Aufruf der Funktion onButtonTap verbunden. Diese Funktion erstellt ein Toast „myToast“, welches den hinterlegten Text lädt. „myToast.show“ ruft myToast auf und gibt den Text aus.

Die App ist nun fertig gestellt.

Testen der App auf einem virtuellen Gerät

Zum Testen der App wählen Sie in der Menü-Leiste den AVD-Manager. Es ist das fünfte Symbol von rechts. Starten Sie das voreingestellte Endgerät mit dem grünen Play-Button.

Warten Sie, bis das virtuelle Gerät vollständig gestartet ist. Klicken Sie dann auf den grünen Play-Button in der Menü-Leiste von Android Studio (elftes Symbol von rechts).

Wählen Sie im anschließenden Dialogfeld, das von Ihnen gestartete Endgerät aus und bestätigen Sie mit „OK“.

Die App startet auf dem virtuellen Endgerät.

Übertragen der App auf ein physisches Endgerät

Um die App auf Ihrem persönlichen Endgerät zu installieren benötigen Sie den USB-Treiber für das Endgerät.

Falls Sie Ihren Treiber herunterladen möchten, suchen Sie in folgender [Tabelle](#) nach dem entsprechenden Eintrag.

Aktivieren Sie nun die Entwickleroptionen in Ihrem Endgerät. Dies erreichen Sie indem Sie in den Einstellungen\Über das Telefon sieben mal auf die Build-Number tippen.

Nun erhalten Sie Zugriff auf einen neuen Einstellungspunkt „Entwickleroptionen“ Wählen Sie hier „Kein Standby des Bildschirms beim Laden“ und „USB Debugging“ aus.

Verbinden Sie Ihre Gerät mit einem passenden Kabel mit Ihrem Computer.

Zur Überprüfung der Verbindung schauen Sie im Geräte-Manager nach. Ist dort ein Eintrag mit „ADB Interface“, wurde Ihr Gerät richtig erkannt. Ist dies nicht der Fall überprüfen Sie den installierten Treiber, die Telefoneinstellungen und die Kabelverbindung.

Klicken Sie nun in Android-Studio erneut auf den grünen Play-Button.

Wird ihr eigenes Gerät in der Übersicht angezeigt, wählen Sie dieses aus und bestätigen Sie mit „OK“. Nun wird Ihre App auf Ihr Endgerät geladen.

Wird Ihr Gerät nicht angezeigt, starten Sie in dem Ordner „AndroidSDK\platform-tools“ eine Kommandozeile.

Tippen Sie „adb.exe devices“ um eine Liste der verbundenen Geräte zu erhalten.

Hier wird eine Nummer und „unauthorized“ angezeigt.

Tippen Sie „adb.exe kill-server“ und „adb.exe start-server“.

Bestätigen Sie anschließend die Verbindung an Ihrem Gerät.

Tippen Sie nun „adb.exe devices“

Es wird erneut die Nummer angezeigt. Dieses mal jedoch mit „device“.

Nun können Sie in Android-Studio über den Play-Button Ihr Endgerät auswählen und somit Ihre App auf das Endgerät übertragen.

Bereitstellung der App als APK-Datei

Klicken Sie in Android-Studio auf „BuildVariants“. Dieser Punkt befindet sich in der linken, unteren Bildschirmcke.

Klicken Sie in der Auswahl auf „debug“ und wählen Sie den Punkt „release“ aus.

Nun klicken Sie in der Menü-Leiste unter „Build“ auf „Generate Signed APK“.

Wählen Sie im neuen Dialogfeld „Create new...“ um einen neuen Key store anzulegen.

Tragen Sie nun Speicherort und Passwort des Key-Stores in die entsprechenden Felder ein.

Im Unterpunkt „Alias“ tragen Sie eine Beschreibung für den Schlüssel ein. Dies kann zum Beispiel der Name der App sein.

Das erneut geforderte Passwort ist für den Key selbst.

Tätigen Sie abschließend mindestens eine Eingabe unter „Certificate“.

Klicken Sie nun auf „OK“.

Ihre Eingaben werden im folgenden Fenster zusammengefasst. Klicken Sie „Next“ um fortzufahren.

Im nächsten Dialogfeld überprüfen Sie, dass als „Build Type“ „release“ ausgewählt ist.

Unter „APK Destination Folder“ haben Sie die Möglichkeit den Speicherort der zu erstellenden APK-Datei zu definieren.

Klicken Sie anschließend auf „Finish“ um die APK zu erstellen.

Die signierte APK kann nun veröffentlicht werden.