

Praktikum GUI-Development – Aufgabe 2

Prof. Dr. Matecki

Ausgabe des Aufgabenblatts: Montag, 17.12.2018

Abgabe per ILIAS bis Sonntag, 20.01.2019 12:00 Uhr

Abnahme: an dem Praktikumstermin nach dem Abgabetermin

- Zulässig zur Bearbeitung dieses Aufgabenblatts sind Teams mit 1-3 Studierenden.
- Die Programme werden am Abgabetag in der Praktikumsstunde von jedem Team am Rechner vorgeführt und von den Lehrenden abgenommen.
- Entwürfe müssen
 - o als wissenschaftliche Ausarbeitung formuliert sein
 - o von **allen** Mitgliedern der einzelnen Teams am Abgabetag erklärt werden können.
- Eine wissenschaftliche Ausarbeitung enthält in diesem Praktikum immer
 - o Ein Deckblatt mit Fach und den Namen der Team-Mitglieder
 - o Ein Inhaltsverzeichnis vor Beginn der Ausarbeitung
 - o Ein Kapitel für jede Aufgabe mit sinnvollen Kapitelüberschriften
 - Ein Unterkapitel für jede Unteraufgabe
 - ... mit sinnvollen Kapitelüberschriften !!
 - o Auf der Fußzeile die Seitenzahl
- Selbst erstellte Programme müssen von **allen** Mitgliedern der einzelnen Teams am Abgabetag erklärt und am Rechner vorgeführt werden können.
- Rechnen Sie am Abgabetag damit, kleine Codesequenzen programmieren zu müssen!
- Tipp: Die in der Vorlesung besprochenen Programmbeispiele sind im Rahmen Ihrer Lösungen zur Nutzung erlaubt!
- Abgenommen werden von den Lehrenden
 - o Ihre Dokumentation mit Ihrer SW-Architektur
 - o Ihre SW / Ihr Quellcode
- Die Implementierung erfolgt unter Android Studio
- Mindest-Version: API-Level 23 oder höher (Android 6)

Entwicklung einer Android-App für die Anzeige von Bildern

Verhalten der App:

Ihre App hat als Haupt-Activity eine **AppCompatActivity** mit einer **ListView**, die die beiden Bildverzeichnisse für Fotos und Screenshots zur Auswahl stellt. Als dritte (noch funktionslose) Möglichkeit kann der Nutzer ein anderes Verzeichnis wählen.

Nach Auswahl eines der beiden Verzeichnisse wird eine SubActivity (ebenfalls **AppCompatActivity**) geöffnet, die das erste Bild des gewählten Verzeichnisses zeigt. Falls das Verzeichnis keine Bilder enthält, so muss dies in geeigneter Form abgefangen werden. Eine Wischbewegung nach rechts zeigt das nächste Bild, usw. . Mit Wischbewegung nach links wird wieder zurück geblättert.

Haupt-Activity erbt von AppCompatActivity (wegen Toolbar mit Menü)



ToolBar mit Menü
Button (so lange kein ListView-Eintrag gewählt: Insensitiv)
ListView mit 3 Einträgen.
 Die Einträge sind mit geeigneten Shapes umrahmt.

ImageView mit passendem Hintergrundbild (von Ihnen zu erstellen)



Button sensitiv, sobald ein ListView – Eintrag gewählt wurde.
 Er öffnet die Sub-Activity mit der Galerie.
 Diese zeigt das erste Bild im Verzeichnis an.

Sub-Activity erbt von AppCompatActivity (wegen Toolbar mit Menü)



ToolBar mit Menü

ImageView: Sie zeigt mit Links-Wisch das nächste Bild an.
 Rechts-Wisch: Vorheriges Bild.

Die Menüs der beiden Activities enthalten in der Grundversion nur die Option „Beenden“. Diese beendet die jeweilige Activity.

Bitte beachten Sie einige technische Details:

Ihre App besteht aus

- Einer **Haupt-Activity**: Sie enthält in einem vertikal angeordneten **LinearLayout**:
 - Einen **ToolBar** mit dem Menüeintrag
 - „Beenden“, der die Activity beendet.
 - Ein Button „Auswählen“. Er ist insensitiv, so lange der Nutzer keinen der Listeneinträge gewählt hat.
 - Eine Liste mit gültigen in einer **ListView**. Die Verzeichnisse selbst werden nicht angezeigt – die App merkt sie sich aber hinter den Einträgen. Ein dritter Eintrag „Anderes Verzeichnis“ ist im Moment nur als Mock vorhanden
 - Die **ListView** enthält die Auflistung der Wahlmöglichkeiten in **CheckedTextView**-Elementen. Aus ihr kann eines der Verzeichnisse selektiert werden.
 - Diese sind mit einer geeigneten Shape umrahmt.
 - Die Haupt-Activity ist wegen der ToolBar-Darstellung von **AppCompatActivity** abgeleitet.
- Einer **Sub-Activity**: Sie enthält in einem **vertikal** angeordneten **LinearLayout**:
 - Eine **ToolBar** mit
 - einem „Zurück“-Eintrag, der veranlaßt, dass die App wieder in die Haupt-Activity zurück springt. (ruft finish() für die SubActivity auf)
 - Eine **ImageView**, die zunächst das erste Bild aus dem gewählten Bildverzeichnis anzeigt.
 - Mit Wisch-Bewegung (Fling) nach rechts kann das vorherige Bild angezeigt werden
 - Mit Wisch-Bewegung nach links kann das nächste Bild angezeigt werden.
 - Zu einer Zeit kann nur ein Bild in der **ImageView** angezeigt werden.
- Die Dateizugriffe auf die Bilddateien und Verzeichnisse werden in eine **eigene** Klasse **FileIO** ausgelagert. Eine Grundversion hiervon wird Ihnen zur Verfügung gestellt.
- **Nicht vergessen:** Ab Android 6 müssen Dateizugriffsberechtigungen nicht nur im Manifest eingetragen werden, sondern auch innerhalb der Activity abgefragt werden!
- Wenn die Bilderverzeichnisse leer sind, so ist dies geeignet in Ihrer App abzufangen. Ebenso, wenn ein bestimmtes Verzeichnis nicht (mehr) existiert!

- Die Pfade zu Ihren Foto- und Screenshot-Verzeichnissen können über die Android-Klasse Environment abgefragt werden. Ein Beispiel dazu:

```
private String pathScreenshots =  
    Environment.getExternalStoragePublicDirectory(  
        Environment.DIRECTORY_PICTURES).toString()  
        + "/Screenshots";  
private String pathCamera =  
    Environment.getExternalStoragePublicDirectory(  
        Environment.DIRECTORY_DCIM).toString()  
        + "/Camera";
```

Erweiterung der App (**freiwillig, für diejenigen, die den Pflichtteil sehr schnell fertig haben**):

Ausbau der Wahlmöglichkeit „Anderes Verzeichnis“: Schreiben Sie hierzu eine weitere Activity, die beispielsweise über eine weitere ListView ein weiteres Verzeichnis (ab Wurzelverzeichnis SD-Karte) wählen lässt:

Haupt-Activity:



Button wird gedrückt:

ListView : Anderes Verzeichnis... ist ausgewählt

Eine andere Sub-Activity wird angezeigt: File-Auswahl in einer ListView. Auch diese Sub-Activity erbt von AppCompatActivity



Ein **ListView**-Eintrag mit einem weiteren Verzeichnis öffnet die gleiche Activity erneut:

Diesmal mit aktualisiertem Wurzel-Verzeichnis

Wird ein Datei-Eintrag selektiert, so geschieht einfach nichts ...

Sub-Activity für File-Auswahl ist jetzt mit dem Inhalt des gewählten Verzeichnisses geöffnet:



Dieses Verzeichnis enthält Bilddateien.

Wir drücken den Button „Auswählen“

Sub-Activity für die Bilddarstellung wird mit dem ersten Bild des Verzeichnisses geöffnet



Wir können uns wieder mit Wisch-Geste durch die Bilder bewegen ...

Von dieser Activity aus kann dann ebenfalls die Bilddarstellungs-Activity gestartet werden. Es darf auch gern ein Browser mit komfortableren Einträgen geschrieben werden. Dazu wird aber möglicherweise ein eigener Adapter (abgeleitet von ArrayAdapter) benötigt.