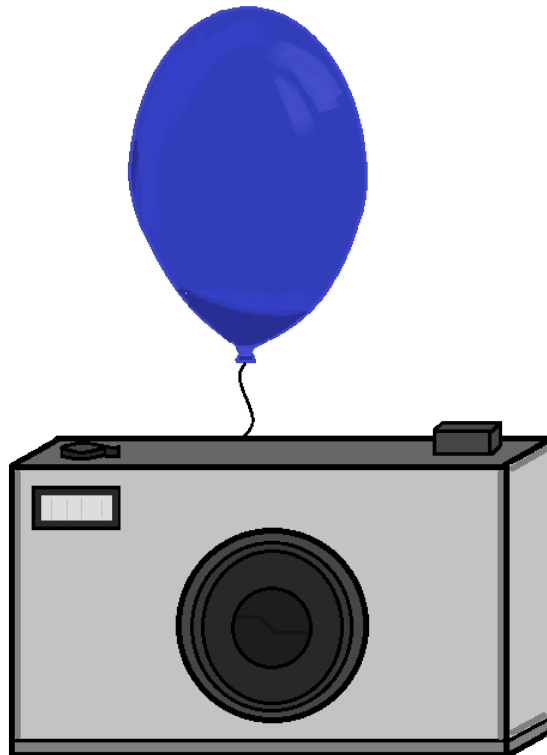


FotoUP

Seminararbeit „Entwicklung mobiler Applikationen – Thema Android“



Name, Matrikelnummer

Nils Wende, 4106128
Lucas Zinkiewicz, 9590629

Kurs

TINF14AI-BC

Datum

29.11.2015

Inhaltsverzeichnis

1	Einleitung	1
2	Spezifikationen.....	2
2.1	Allgemeine Anforderungen	2
2.2	Funktionale Anforderungen	2
2.3	Extras	3
3	Bewertung.....	4
4	Dokumentation	5
4.1	Implementation F-00	5
4.2	Implementation F-10	6
4.3	Implementation F-20	7
4.4	Implementation F-30	8
4.5	Implementation F-40	9
4.6	Implementation F-50	9

1 **Einleitung**

Als Seminararbeit für die Vorlesung „Entwicklung mobiler Applikationen – Thema Android“ soll eine Android-App erstellt werden, die Fotos aufnehmen, speichern und darstellen kann. Es sollen auch existierende Bilder aus der Galerie geladen werden können. Eine genaue Aufschlüsselung der Anforderungen folgt in Kapitel 2.

Die App ist für das API-Level 22 geschrieben, sie ist damit lauffähig auf allen Geräten mit Android 5.1 (Lollipop) und später. Ältere Versionen werden nicht unterstützt.

Getestet wurde über den Android Emulator – welcher mit API-Level 23 in Android Studio enthalten ist – und über ein Samsung Galaxy S II mit CyanogenMod 12.1, was der Android-Version 5.1.1 und damit dem API-Level 22 entspricht.

2 Spezifikationen

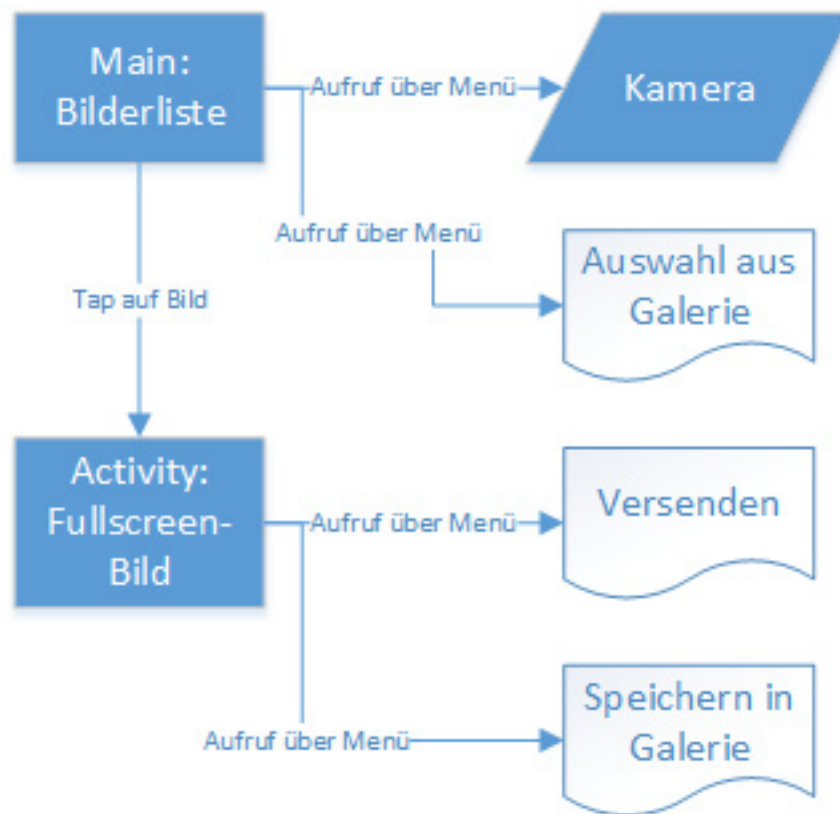
2.1 Allgemeine Anforderungen

- Gruppenarbeit mit 2 – 3 Personen
- Ergebnis: Ausführliche Dokumentation der Implementation (10 – 12 Seiten)
- Kompilierfähiger Quellcode
- Android Studio als IDE
- Abgabe bis 29.11.15

2.2 Funktionale Anforderungen

ID	Beschreibung
F-00	Erstellen einer Android-App, die Fotos über ein Menü aufnimmt oder aus der Fotogalerie lädt
F-10	Ein aufgenommenes Foto soll in einem speziellen Verzeichnis mit dem Aufnahmedatum und -uhrzeit gespeichert werden
F-20	Die aufgenommenen Fotos sollen in einer Liste dargestellt werden, zusammen mit dem Aufnahmedatum und Namen des jeweiligen Fotos
F-30	Ein Tap auf ein Foto in der Liste startet eine zweite Activity, in der das gewählte Bild in einem ImageView dargestellt wird
F-40	Eine Option soll das Versenden des Fotos per Mail ermöglichen
F-50	Eine Option soll das Speichern des Fotos in der Fotogalerie ermöglichen

Diese Anforderungen führen zu folgender Struktur der App:



2.3 Extras

ID	Beschreibung
E-00	Die App besitzt ein eigenes Icon
E-10	Verschiedene Funktionalitäten werden per Toast quittiert
E-20	Statt eines Menüs nutzt die App einen Navigation Drawer

3 Bewertung

Beschreibung	Punkte
Aufgabenstellung – Umsetzung – Dokumentation	10
Programmierung – Umsetzung – Objektorientierung – Fehlerbehandlung – Wiederverwendung	20
Professionalität	5
Extras	5
Gesamt:	40

4 Dokumentation

4.1 Implementation F-00

Um die App zu erstellen wurde in Android Studio ein neues Projekt angelegt, welches eine leere *Activity* beinhaltet.

FotoUP nutzt einen sogenannten *Navigation Drawer* als Menü. Hier kann zwischen den verschiedenen Seiten der App gewechselt werden:

- Main, die als Einleitungsbildschirm das Logo der App in groß zeigt sowie die Autoren nennt
- Gallery, welche die Bildergalerie beinhaltet
- Import, wo der Nutzer ein Bild aus der Android-Galerie in die App importieren kann

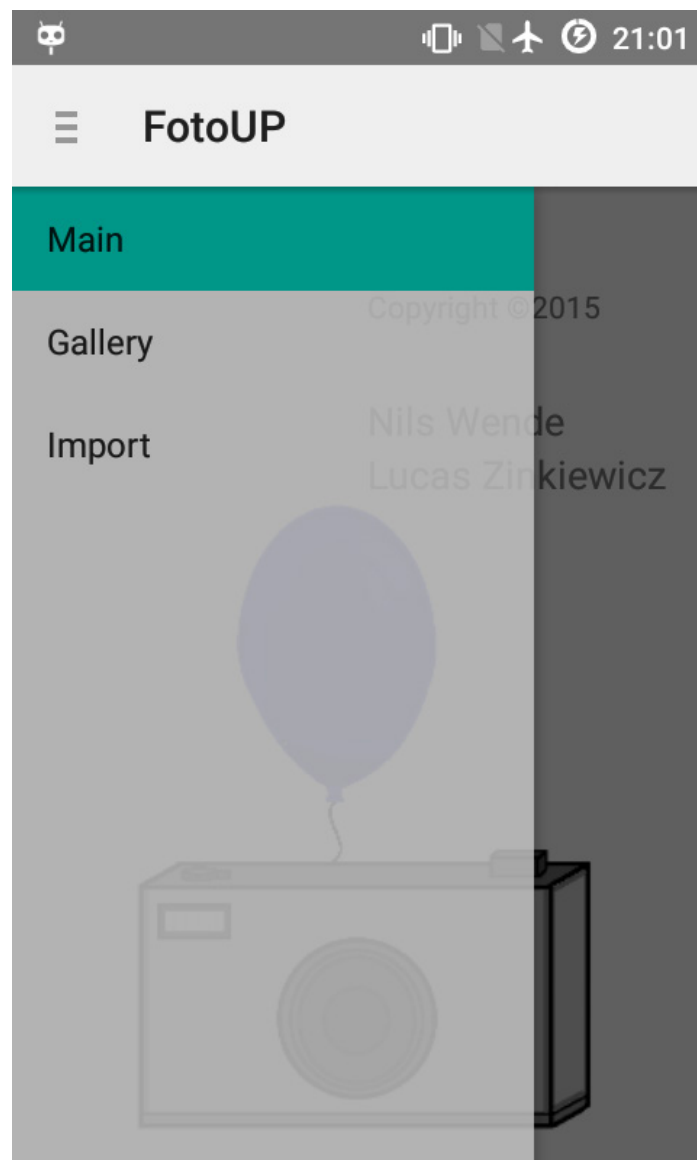


Abbildung 1: Navigation Drawer

Um den Hauptzweck der App – das Aufnehmen von Fotos – zu erleichtern, wurde die Option, ein Foto aufzunehmen, nicht in ein ausklappbares Menü gepackt, sondern in einen Button, der direkt in die obere Menüleiste integriert ist. Dies sorgt für eine schnellere Aufrufbarkeit der Kamera.

Dieser Button startet über einen *Intent* die in Android integrierte Kamera-App. Wird ein Bild gemacht und angenommen, werden das aufgenommene Foto und das zugehörige Thumbnail im für die App reservierten Teil des internen Speichers gespeichert. Gleichzeitig wird ein *Toast* angezeigt, der das erfolgreiche Speichern anzeigt und den Speicherort des Fotos angibt. Sollte das Speichern nicht klappen, weil z. B. nicht mehr genug Speicher vorhanden ist, wird ein entsprechender Error-Toast angezeigt.

Um ein Bild aus der Android-Galerie zu laden, muss der Navigation Drawer aufgerufen werden. Unter dem Punkt „Import“ findet man dort die Möglichkeit, ein Bild auszuwählen und es dann zu importieren oder abzubrechen.

Wählt man ein Bild aus der Galerie, wird es zusammen mit einem Toast, der den Speicherort des Bildes angibt, in der App angezeigt. Der Nutzer hat dann die Möglichkeit, das Ganze abzubrechen, was das Bild wieder aus der Anzeige entfernt, oder das Bild zu importieren, was es der App-eigenen Galerie hinzufügt. Ein Toast bestätigt das erfolgreiche Importieren.

4.2 Implementation F-10

Wird ein Bild aufgenommen oder importiert, wird es im App-eigenen internen Speicher gespeichert. Dazu werden bei App-Start – sofern noch nicht vorhanden – die Ordner „BigPictures“ und „SmallPictures“ angelegt.

BigPictures enthält alle Bilder in Originalauflösung, während SmallPictures die jeweiligen Thumbnails – also kleine Vorschauversionen der eigentlichen Bilder – enthält.

Aufnahmedatum und -uhrzeit werden nicht eigens abgespeichert, zur Anzeige in der In-App-Galerie wird das „last modified“-Datum – also das Datum, an dem das Bild das letzte Mal geändert wurde – herangezogen. Dieses Datum stellt gleichzeitig auch das Erstellungsdatum dar, da ein Bild im App-internen Speicher nur bei der Erstellung geändert wird.

4.3 Implementation F-20

Die App-eigene Fotogalerie besteht aus einem *ScrollView*, um vertikal zu scrollen, und einem darin eingebetteten *TableLayout*.

Jedes gespeicherte Bild wird als horizontales *LinearLayout* dargestellt, welches zwei weitere Elemente beinhaltet. Zum einen das Thumbnail als klickbarer Button, zum anderen ein *TextView*, welches das Speicherdatum des Bildes angibt.

Eine Zeile des *TableLayouts* kann maximal drei *LinearLayouts* beinhalten, soll ein weiteres hinzugefügt werden, wird eine neue Zeile erstellt. Die Breite der einzelnen Elemente wird dabei so angepasst, dass genau drei in eine Zeile passen.

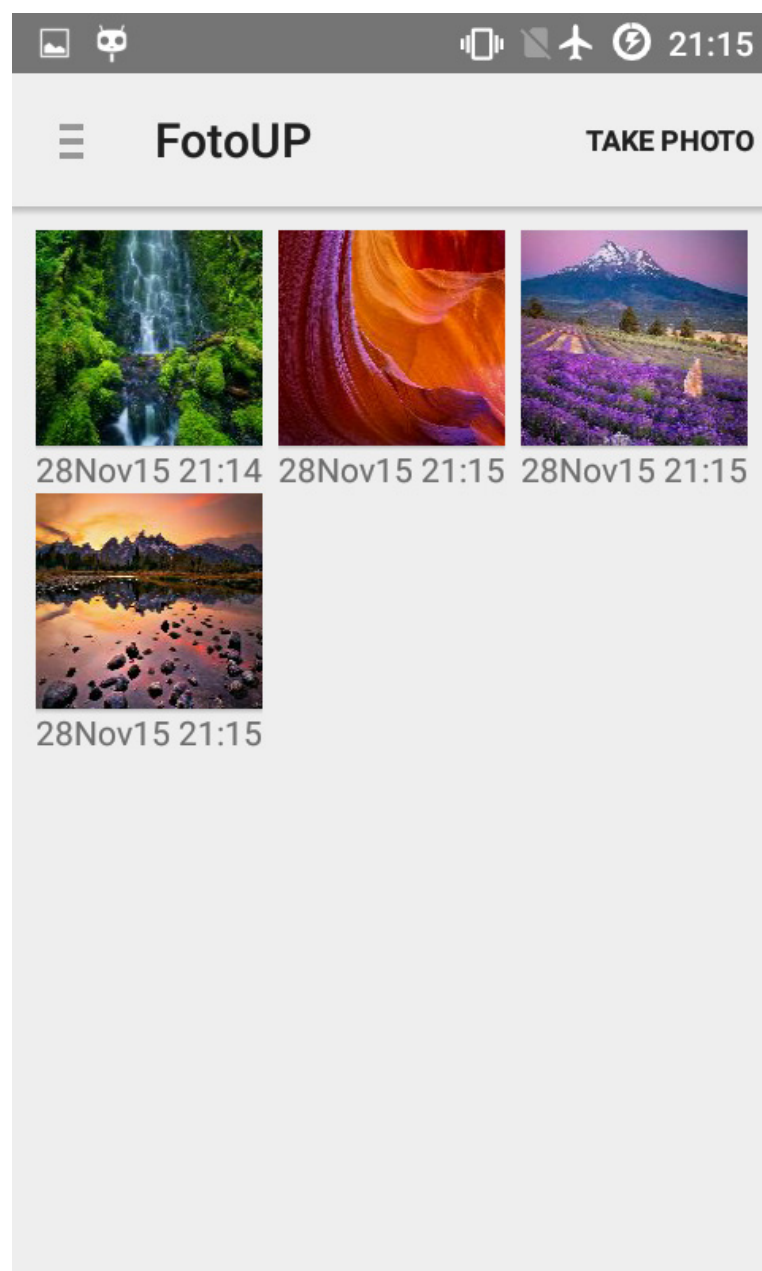


Abbildung 2: Galerie

4.4 Implementation F-30

Ein Tap auf einen der Thumbnail-Buttons öffnet eine neue Activity. Hier wird das ausgewählte Bild größer in einem ImageView angezeigt, zusammen mit den Optionen das Bild in die Android-Galerie zu exportieren oder per E-Mail zu versenden.

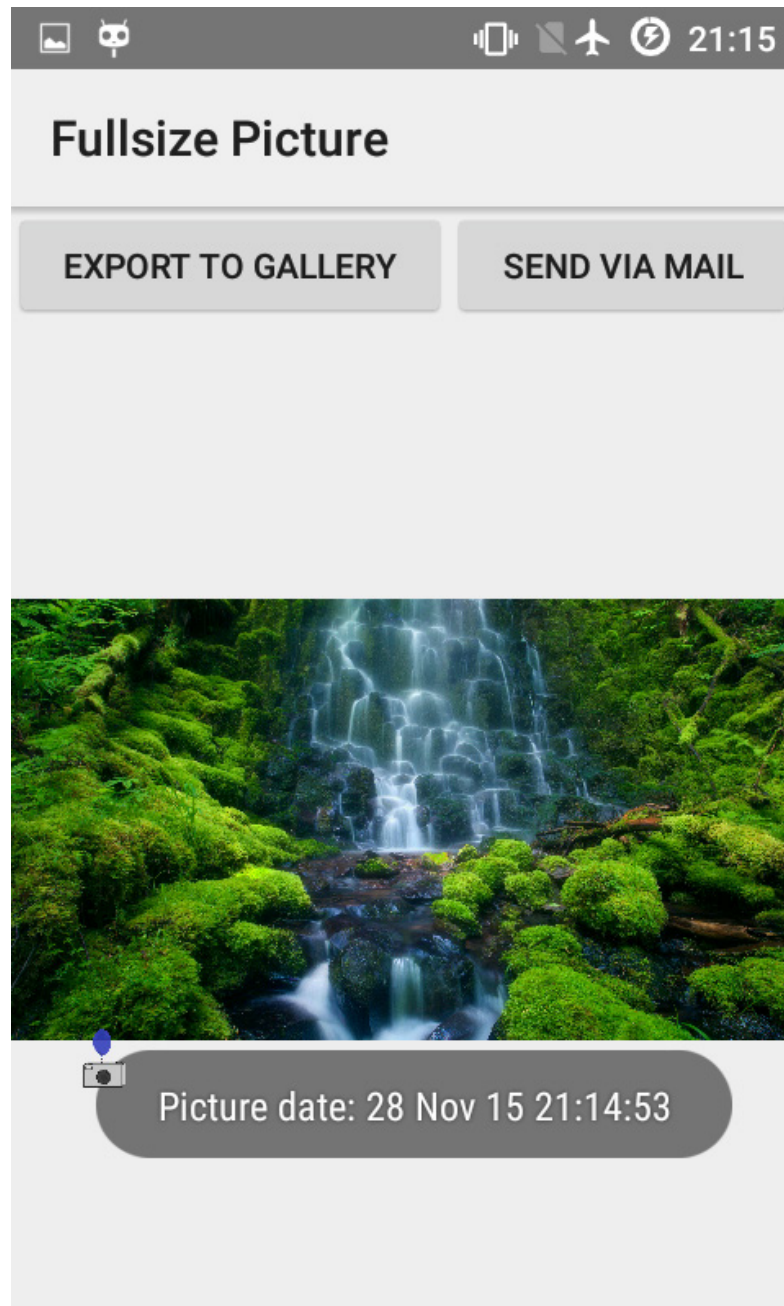


Abbildung 3: Ausgewähltes Bild mit Toast

4.5 Implementation F-40

Ein Tap auf den Button „Send via Mail“ öffnet das installierte E-Mail-Programm. Dazu wird ein Intent erstellt, der im Mailprogramm eine neue Mail erstellt, Betreff und Nachricht ausfüllt und das Bild anhängt.

4.6 Implementation F-50

Wählt man das Exportieren des Bildes in die Android-Galerie über den Button „Export to Gallery“, wird das Bild über die statische API-Methode *MediaStore.Images.Media.insertImage()* veröffentlicht.

Der Titel des Bildes enthält dann die interne ID des Bildes und den Aufnahmezeitpunkt.

Ein Toast quittiert das erfolgreiche Exportieren in die Galerie.