

**Just do it**

**Pendenzen-Verwaltung**

Version 1.0.0, 25. Mai 2020 | Klemens Raduner



# Inhalt

<b>1</b>	<b>Abstract .....</b>	<b>2</b>
<b>2</b>	<b>Mockups .....</b>	<b>3</b>
<b>2.1</b>	<b>MainActivity .....</b>	<b>3</b>
<b>2.2</b>	<b>PendenzActivity .....</b>	<b>3</b>
<b>3</b>	<b>Technische Realisierung .....</b>	<b>4</b>
<b>3.1</b>	Unterschiede zum Mockup.....	4
<b>3.2</b>	MainActivity.....	4
<b>3.3</b>	PendenzActivity .....	4
<b>3.4</b>	Datenpersistenz .....	4
<b>4</b>	<b>Testing.....</b>	<b>5</b>
<b>4.1</b>	Automatische Tests .....	5
<b>4.2</b>	Manuelle UI-Tests .....	5
<b>4.3</b>	Testauswertung .....	7
<b>5</b>	<b>Fazit.....</b>	<b>8</b>

# 1 Abstract

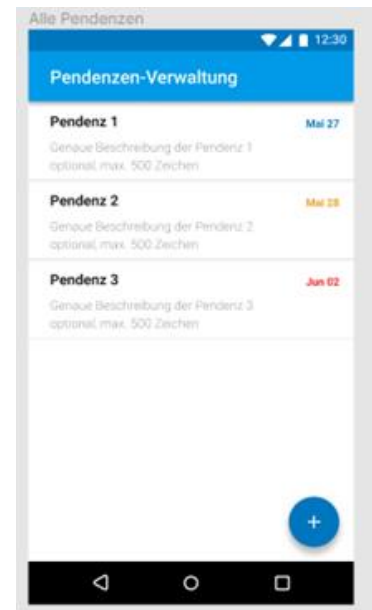
Für den üK 335 wurde ein Projekt umgesetzt. Es gab drei mögliche Aufgabenstellungen zur Auswahl, von denen eines ausgewählt und realisiert werden musste. Ziel dieses Projekts ist es, eine Android-App für Smartphones zu erstellen, über welche ein Lernender Pendenzen erstellen kann und dann in einer Liste (oder ähnlichem) alle erfassten Pendenzen angezeigt werden. In diesem Dokument werden das Vorgehen und der Ablauf des Projektes dokumentiert.

Alle was für diesen üK erstellt wurde findet man hier: [https://github.com/KlemensRaduner/uek\\_335](https://github.com/KlemensRaduner/uek_335)

## 2 Mockups

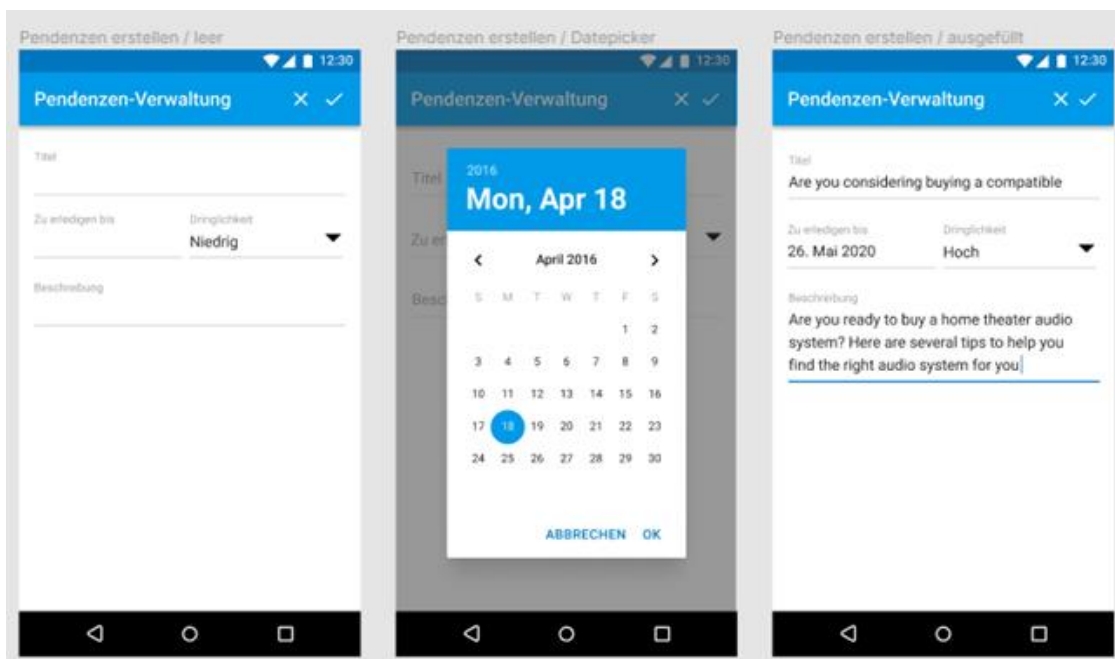
### 2.1 MainActivity

Innerhalb der MainActivity werden alle Pendenzen aufgelistet. Von den einzelnen Pendenzen wird jeweils der Title, die Beschreibung übereinander dargestellt. Das Datum befindet sich am rechten Rand und ist blau, orange oder rot gefärbt, je nach Priorität. Die einzelnen Pendenzen werden von einer feinen Linie getrennt. Mit einem Klick auf eine Pendenz wird die 2. Activity gestartet, in welcher die ausgewählte Pendenz bearbeitet werden kann. Am unteren rechten Rand in der MainActivity befindet sich ein Button mit einem Pluszeichen. Mit einem Klick auf den Button wird auch die 2. Activity gestartet. Alle Felder sind aber noch leer.



### 2.2 PendenzActivity

Mit der PendenzActivity kann eine neue Pendenz erstellt, oder bereits existierende Pendenzen bearbeitet werden. Oben in der ActionBar gibt es zwei Buttons. Ein Häkchen, um zu speichern und ein Kreuz, um die vorgenommenen Änderungen zu verwerfen. Es gibt zwei normale Textfelder für den Titel und die Beschreibung. Für das Datum gibt es auch ein Textfeld. Wenn man aber auf das Datums Feld klickt, wird anstelle von der Tastatur ein Datum Picker geöffnet. Für die Priorität gibt es ein Spinner mit den drei Optionen niedrig, normal und hoch als Dropdown. Wird gespeichert, bevor alle Felder ausgefüllt sind, wird ein Toast mit einer hilfreichen Fehlermeldung angezeigt und der Fokus springt zum obersten der leeren Felder.



## 3 Technische Realisierung

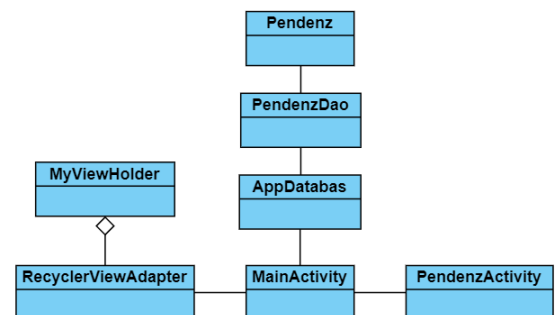
### 3.1 Unterschiede zum Mockup

Vom Mockup zum fertigen Produkt gibt es zwei kleine Unterschiede. Zum einen wurde der Name zu von "Pendenzen-Verwaltung" zu "Just du it" geändert. Zum anderen wurde in der PendenzActivity ein Button mit einem Mülleimer hinzugefügt. Dieser Button erscheint nur, wenn man eine existierende Pendenz bearbeitet. Mit einem Klick auf den Butten wird diese Pendenz gelöscht.



### 3.2 MainActivity

Die MainActivity besteht aus einem einfachen ConstraintLayout mit einem Button und einer RecyclerView. Alle Pendenzen werden in der "onStart" Method aus der Datenbank geladen. Aus jeder geladenen Pendenz wird mit Hilfe der "RecyclerViewAdapter" Klasse ein Fragment erstellt und in die RecyclerView eingefügt. Bei einem Klick auf eine der Pendenzen wird die "onItemClick" Methode ausgeführt. Sie startet die PendenzActivity und gibt die geladene Pendenz im Intent mit. Bei einem Klick auf den schwebenden Button wird die "createNewPendenz" Methode ausgeführt. Dabei wird ebenfalls die PendenzActivity gestartet. Es wird aber keine Pendsenz im Intent mitgegeben.



### 3.3 PendenzActivity

Das Root Element der PendenzActivity ist ebenfalls ein ConstraintLayout. Die einzelnen Input Felder sind jeweils LinearLayouts mit einem Text Feld zur Beschreibung und dem eigentlichen Input darunter. In der "onCreate" Methode wird die Pendenz, falls sie existiert aus dem Intent ausgelesen und dann werden Listener für alle Input Felder gesetzt. Für den Datum Picker werden zwei extra Methoden gebraucht. "openDatePicker" um beim Öffnen das aktuelle Datum auszuwählen, und "onDateSet" um die Auswahl zu verarbeiten. Die "onOptionsItemSelected" Methode gibt den Buttons in der ActionBar ihre Funktionen.

### 3.4 Datenpersistenz

Die erstellten Pendenzen werden in der SQLite Datenbank von Android gespeichert. Dafür werden zwei Klassen und ein Interface gebraucht. Ein Pojo, "Pendenz", das alle Daten enthält. "PendenzDao" gibt alle verwendeten Abfragen als Methoden mit den entsprechenden Annotationen an. "AppDatabase" initialisiert eine Instanz von "PendenzDao" und stellt diese in der gesamten Applikation zur Verfügung.

# 4 Testing

## 4.1 Automatische Tests

Zusätzlich zu den Manuellen Tests gibt es im Projekt mehrere Espresso Tests. Man findet sie unter "app" und dann "androidTest". Um diese durch zu führen wird ein Emulator oder ein angeschlossenes Android Gerät gebraucht.

## 4.2 Manuelle UI-Tests

Abschnitt	MainActivity
ID	1
Vorbedingungen	In der Datenbank sind keine Pendenzen
Ablauf	1. Die App wird gestartet damit die MainActivity erscheint.
Erwartetes Resultat	In der MainActivity werden keine Pendenzen angezeigt. Es gibt nur die ActionBar und einen Button.

Abschnitt	Validierung Titel
ID	2
Vorbedingungen	keine
Ablauf	1. Die App wird gestartet damit die MainActivity erscheint. 2. Der Button mit dem "Plus" Zeichen wird geklickt damit die PendenzActivity gestartet wird. 3. Der Button mit dem Häckchen in der ActionBar wird geklickt.
Erwartetes Resultat	Es erscheint ein Toast mit der Nachricht "Es wird ein Titel benötigt" und der Fokus springt zum Titelinput Feld.

Abschnitt	Validierung Datum
ID	3
Vorbedingungen	keine
Ablauf	<ol style="list-style-type: none"> <li>1. Die App wird gestartet damit die MainActivity erscheint.</li> <li>2. Der Button mit dem "Plus" Zeichen wird geklickt damit die PendenzActivity gestartet wird.</li> <li>3. Es wird ein Titel eingegeben</li> <li>4. Der Button mit dem Häckchen in der ActionBar wird geklickt.</li> </ol>
Erwartetes Resultat	Es erscheint ein Toast mit der Nachricht "Es wird ein Datum benötigt".

Abschnitt	Validierung Beschreibung
ID	4
Vorbedingungen	keine
Ablauf	<ol style="list-style-type: none"> <li>1. Die App wird gestartet damit die MainActivity erscheint.</li> <li>2. Der Button mit dem "Plus" Zeichen wird geklickt damit die PendenzActivity gestartet wird.</li> <li>3. Es wird ein Title eingegeben.</li> <li>4. Es wird ein Datum ausgewählt.</li> <li>5. Der Button mit dem Häckchen in der ActionBar wird geklickt.</li> </ol>
Erwartetes Resultat	Es erscheint ein Toast mit der Nachricht "Es wird eine Beschreibung benötigt" und der Fokus springt zum Beschreibungsinput Feld.

Abschnitt	Pendenz erstellen
ID	5
Vorbedingungen	keine
Ablauf	<ol style="list-style-type: none"> <li>1. Die App wird gestartet damit die MainActivity erscheint.</li> <li>2. Der Button mit dem "Plus" Zeichen wird geklickt damit die PendenzActivity gestartet wird.</li> <li>3. Es wird ein Title eingegeben</li> <li>4. Es wird ein Datum ausgewählt</li> <li>5. Es wird eine Beschreibung eingegeben</li> <li>6. Der Button mit dem Häckchen in der ActionBar wird geklickt.</li> </ol>
Erwartetes Resultat	Die PendenzActivity wird geschlossen. Die Main Activity erscheint wieder. Es erscheint nun eine Pendenz mit dem eingegebenen Title, Datum und Beschreibung.

Abschnitt	Hohe Priorität ist rot
ID	6
Vorbedingungen	In der Datenbank existiert eine Pendenz mit niedriger Priorität
Ablauf	<ol style="list-style-type: none"> <li>1. Die App wird gestartet damit die MainActivity erscheint.</li> <li>2. Der Button mit dem "Plus" Zeichen wird geklickt damit die PendenzActivity gestartet wird.</li> <li>3. Es wird ein Title eingegeben</li> <li>4. Es wird ein Datum ausgewählt</li> <li>5. Es wird eine Beschreibung eingegeben</li> <li>6. Der Button mit dem Häckchen in der ActionBar wird geklickt.</li> </ol>
Erwartetes Resultat	Die PendenzActivity wird geschlossen. Die Main Activity erscheint wieder. Es erscheint nun eine Pendenz mit dem eingegeben Title, Datum und Beschreibung.

## 4.3 Testauswertung

Durch die Automatisierten und die Manuellen Ui-Tests zusammen werden alle Anforderungen an die App abgedeckt. Alle Tests wurden von Klemens Raduner erfolgreich durchgeführt.



## 5 Fazit

Ich bin sehr zufrieden mit dem Ergebnis. Es sieht genauso aus, wie auf den Mockups. Alle Anforderungen sind umgesetzt und getestet. Bis auf einmal sind beim Implementieren keine Probleme aufgetreten. Dieses Problem konnte mit einem Neustart von Android Studio behoben werden. Ich habe in der Vergangenheit bereits Apps mit Android Studio entwickelt. Seit da hat sich aber einiges geändert. Dass es in Android eine Datenbank gibt, auf die man so einfach zugreifen kann, wusste ich vorher noch nicht. Das hätte ich gerne schon früher gewusst.