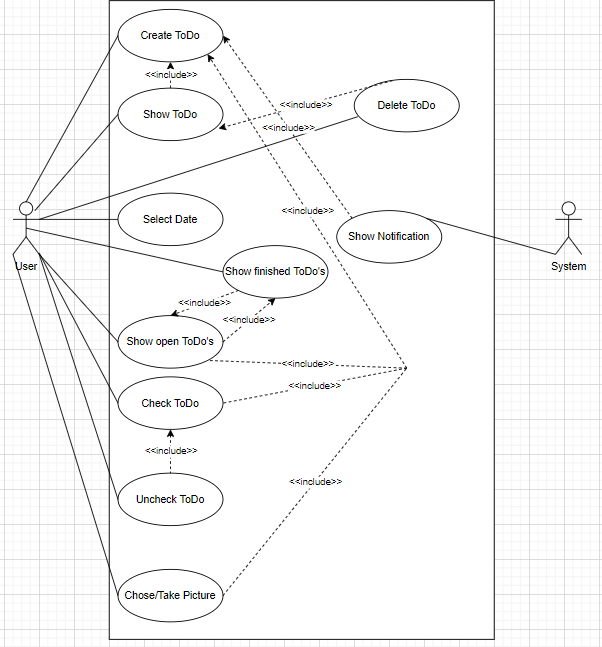
Technischer Entwurf

# Funktionen:

## Funktionale Anforderungen (Use Case):



### Akteure:

#### User:

Der User, ist der normale Benutzer, der sich die App installiert hat und benutzen möchte.

#### System:

Das System, ist anders gesagt das Handy, welches nur die Funktion hat Notifikationen zu erstellen, anzuzeigen und die Daten zu speichern sowie anzuzeigen.

## Nicht Funktionale Anforderungen:

|  |  |
| --- | --- |
| **FURPS** | **Funktion** |
| **P**erformance | Die Ladezeiten sollen nicht länger als 3 Sekunden dauern, um auf eine neue Action zu gelangen. |
| **U**sability | Die abgehackten ToDo’s sollen einen grünen Hintergrund bekommen, um besser sehen zu können, ob die View erledigt worden ist. |
| **R**eliability | Bei einem Fehler, soll das Programm nicht abstürzen, sondern die gleiche Activity soll noch einmal angezeigt werden. |

# Testkonzept:

## Womit wird die App getestet:

Die App wird mit dem Android Emulator des Smartphones Google Pixel 5 und der Version API 27 Android 8.1 Oreo getestet.

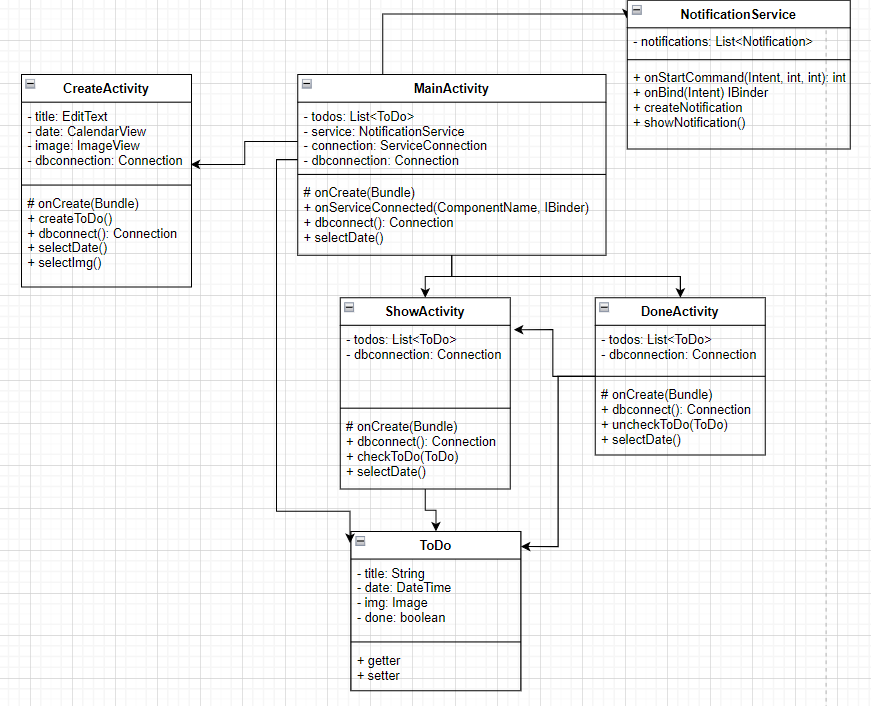
## Wie wird die App getestet:

Die App wird mit manuellen eingaben getestet, und zwar wie oben schon erwähnt mit einem Emulator.

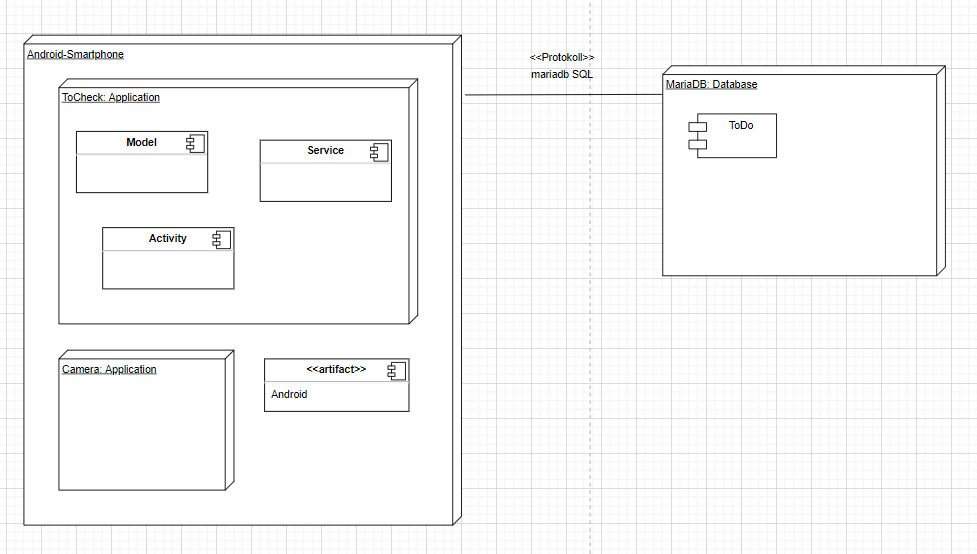
## Testfälle:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Vorbedingung** | **Vorgehen** | **Erwartetes Resultat** |
| 1 | * App wurde gestartet * „+“ oben rechts wurde ausgewählt | 1. Titel = „ToDo“ 2. Datum=20.01.21 3. Button „Create“ betätigen | ToDo wird erstellt und in der Datenbank gespeichert. Man wird direkt zu dem Menu (Main) Activity weitergeleitet und sieht dort nun die neue ToDo. |
| 2 | * App wurde gestartet * Eine ToDo wurde bereits erstellt | 1. Auf show bei der erstellten ToDo klicken. | Die ToDo wird mit ihren Informationen (Titel, Datum, Bild(falls vorhanden)) angezeigt. |
| 3 | * App wurde gestartet * Eine ToDo wurde bereits erstellt * Die ToDo wird angezeigt | 1. Button „Delete“ betätigen | Die ToDo wird aus der Datenbank gelöscht und man wird direkt zum Menu (Main) Activity weitergeleitet wo die ToDo nicht mehr zu sehen ist. |
| 4 | * App wurde gestartet | 1. Button „Date (aktuell ausgewähltes Datum)“ betätigen. 2. Datum: 25.02.21 wählen. | Man sieht wieder alle ToDo’s die für den 25.02.21 eingeplant worden sind. |
| 5 | * App wurde gestartet. * Offene ToDo’s werden angezeigt. | 1. Button „Done“ betätigen. | Die abgehackten ToDo’s für das eingestellte Datum werden angezeigt. |
| 6 | * App wurde gestartet. * Abgehackte ToDo’s werden angezeigt. | 1. Button „Open“ betätigen. | Die offenen ToDo’s für das eingestellte Datum werden angezeigt. |
| 7 | * App wurde gestartet * Offene ToDo’s werden angezeigt. | 1. Button „Check“ betätigen. | ToDo wird als erledigt markiert und verschwindet aus dieser Action und ist auf der „Done“ Action zu sehen. |
| 8 | * App wurde gestartet. * ToDo wurde erstellt. * ToDo wurde abgehackt. * Abgehackte ToDo’s werden angezeigt. | 1. Button „Uncheck“ betätigen. | ToDo wird als unerledigt angesehen und ToDo verschwindet aus der „Done“ Action und taucht in der „Menu“ Action auf. |
| 9 | * Kamera Zugriff wurde erlaubt. * App wurde gestartet. * „+“ oben rechts wurde betätigt. * Titel und Datum wurden angegeben. | 1. Button „Take pic“ betätigen. 2. Foto auswählen/aufnehmen. | Foto wird in der Vorschau angezeigt. |
| 10 | * ToDo wurde für das heutige Datum erstellt. * Notificationen wurden eingeschaltet. | 1. Nichts (System macht das) | Notifikation wird angezeigt. |

# Klassendiagramm:



# Deployment Diagramm:



## Schnittstellen:

Die Schnittstelle befindet sich zwischen der ToCheck Applikation und der Kamera Applikation, die bereits vorhanden ist. Diese müssen kommunizieren, um das Bild in der ToCheck Applikation anzeigen zu können.