

Richtungsanzeiger

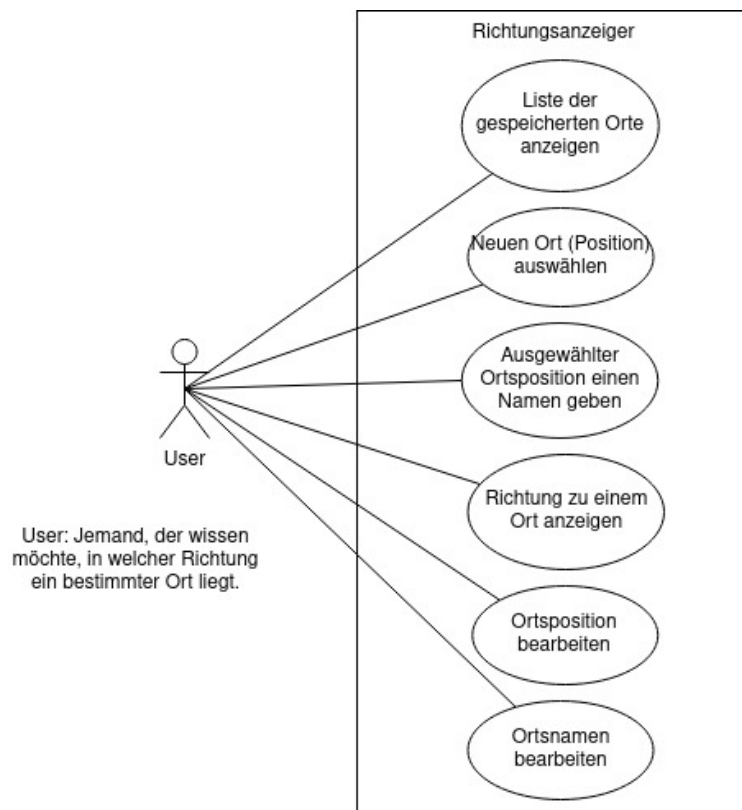
Technischer Entwurf

Jonas Tochtermann

28. Februar 2021

1 Funktionale und nicht-funktionale Anforderungen

1.1 Funktionale Anforderungen



1.2 Nicht-funktionale Anforderungen

- Die Sprache der Benutzeroberfläche ist Deutsch.

- Persistente Daten werden mit SharedPreferences verwaltet.
- Zielplattform ist Android 8.1.
- Die App wird in Java implementiert.

2 Testkonzept

Womit wird die App getestet? Wahrscheinlich allein mit dem Emulator (Nexus 6 mit Android 8.1, API-Level 27); falls sich noch die Möglichkeit ergibt, ein Android-Phone aufzutreiben, würde gegebenenfalls auch dieses zum Einsatz kommen.

Wie wird die App getestet? Manuelle und automatisierte Tests. Bei den hier aufgeführten manuellen Tests wäre es sinnvoll, auch die Möglichkeit der Automatisierung zu berücksichtigen.

2.1 Testfälle

1. Liste der gespeicherten Orte anzeigen

Identifikation showList

Vorbedingungen Bei der Installation werden zwei Orte (Matterhorn und Bundeshaus) mitgegeben, sodass die Liste bereits beim ersten Öffnen einen Inhalt besitzt.

Vorgehen Automatisierter Test zur Überprüfung, ob beim Anzeigen der MainActivity eine Liste mit Inhalt angezeigt wird.

Erwartetes Resultat Die Liste mit den "installierten"Orten (inklusive aller Icons: Kompass und Stift) wird angezeigt.

2. Neuen Ort (Position) auswählen

Identifikation pickPosition

Vorbedingungen Eine KartenApp ist auf dem Test-Phone vorhanden und brauchbar für eine Datenein- und -ausgabe zwischen Activities. Die MainActivity mit der Liste wird angezeigt. Der Testfall showList ist erfolgreich durchgeführt worden.

Vorgehen Manueller Test; Drücken auf den "Ort hinzufügen"-Button, Eingabe "ZLI"im Suchfeld, Drücken auf "Position übernehmen"-Icon.

Erwartetes Resultat Die EditLocationActivity wird angezeigt mit den Koordinaten "47.3598043 N"(Breitengrad) und "8.5210211 O"(Längengrad). Die Angaben dürfen eine Abweichung von +/- 0.001 aufweisen.

3. Ausgewählter Ortsposition einen Namen geben

Identifikation pickName

Vorbedingungen Der Testfall pickPosition wurde ausgeführt und die EditLocationActivity wird angezeigt.

Vorgehen Manueller Test; Im Feld mit dem Platzhalter "Name des Ortes" wird "ZLI" eingegeben, danach wird auf das Icon "Speichern und zurück zur MainActivity" geklickt.

Erwartetes Resultat Die aktualisierte Liste mit dem neuen Eintrag "ZLI" wird angezeigt.

4. Richtung zu einem Ort anzeigen

Identifikation showDirection

Vorbedingungen Der Testfall showList wurde durchgeführt. In den Einstellungen des Emulators wird als Location Basel (47.5596N, 7.5886O) genommen. Die MainActivity wird angezeigt.

Vorgehen Manueller Test; Auf der Liste wird mit Klick auf das entsprechende Kompass-Icon "Matterhorn" ausgewählt.

Erwartetes Resultat Die ShowDirectionActivity wird angezeigt mit dem Kompass und der Richtungsanzeige auf 178°. Auf dem Emulator sollte Norden gegen "Oben" sein.

5. Ortsposition bearbeiten

Identifikation editPosition

Vorbedingungen Der Testfall showList wurde durchgeführt. Die MainActivity wird angezeigt.

Vorgehen Manueller Test; Auf der Liste wird mit Klick auf das entsprechende Stift-Icon "Bundeshaus" ausgewählt. Nun sollte die EditLocationActivity angezeigt werden. Mit Klick auf "Ort auswählen" wird auf die SearchOnMapActivity gewechselt. Im Suchfeld "Freibad Marzili" eingeben. Drücken auf "Position übernehmen"-Icon.

Erwartetes Resultat Die EditLocationActivity wird angezeigt mit den Koordinaten "46.9424203 N" (Breitengrad) und "7.4437811 O" (Längengrad). Die Angaben dürfen eine Abweichung von +/- 0.001 aufweisen.

6. Ortsnamen bearbeiten

Identifikation editName

Vorbedingungen Der Testfall editPosition wurde durchgeführt. Die EditLocationActivity wird angezeigt.

Vorgehen Manueller Test; Die TextView mit dem Namen wird ausgewählt. Anstelle des bisherigen Textes wird "Marzilibad" eingegeben, danach wird auf das Icon "Speichern und zurück zur MainActivity" geklickt.

Erwartetes Resultat Die aktualisierte Liste mit dem geänderten Eintrag "Marzili" wird angezeigt.

3 Aufbau des Systems

