

U14-Wetterfrosch

Zuletzt bearbeitet von Alexander Bazo

Die Wetterfrosch-App

In dieser Übungsaufgabe planen und implementieren Sie eine einfache Wetter-App. Nach dieser letzten Aufgabe schließen wir die Übungen des Semesters an und wechseln in die Projektarbeit. Vorher sollen Ihnen diese App die Möglichkeit geben, wesentliche Teile der anstehenden Projektarbeit vorab auszuprobieren. Einen Lösungsvorschlag veröffentlichen wir in zwei Wochen und werden Ihnen vorher im Rahmen der Übungen Gelegenheit geben, Fragen und Probleme bei der Umsetzung mit uns zu besprechen. **Gerne können Sie diese Aufgabe bereits als Gruppenarbeit in Ihrem Projektteam lösen!**

Die Aufgabe

Implementieren Sie auf Basis des Starterpakets eine einfache Android-App, die es NutzerInnen erlaubt, das aktuelle Wetter sowie eine Wetterschau für beliebige Städte anzuzeigen. Datengrundlage dafür ist die Open Weather API, die Sie kostenlos nutzen können und die wir Ihnen in diesem Handout noch genauer vorstellen werden. Ihre App soll im wesentlichen diese Funktionen erfüllen:

- Auf dem Startbildschirm bzw. in der Start-Activity können NutzerInnen eine Sammlung von Städten pflegen. Die Städte werden im UI als scrollbare Liste angezeigt, zu jeder Stadt wird mindestens der Name, die aktuelle Temperatur und die aktuelle Wetterbeschreibung angezeigt. NutzerInnen können neue Städte über die Eingabe des Stadtnamens hinzufügen.
- Einzelne Städte können aus der Liste ausgewählt werden. In einer zweiten Activity werden dann genauere Informationen zur aktuellen Wetterlage sowie ein Ausblick auf die nächsten Stunden bzw. Tage angezeigt.

Die API

Die Wetterdaten beziehen Sie über die Open Weather API. Die API erlaubt registrierten NutzerInnen das kostenlose Abfragen von globalen Wetterinformationen. Der Zugriff erfolgt über eine HTTP-Schnittstelle, ein Prinzip, das Sie bereits aus der Mensa-App kennen. Eine ausführliche Beschreibung der API finden Sie hier. Für den schnellen Einstieg sollten Sie diese Informationen beachten:

1. Erstellen Sie einen kostenlosen Account auf der API-Webseite
2. Sobald Sie auf der Webseite eingeloggt sind, können Sie einen API-Key erstellen. Dieser muss bei jeder Anfrage an die API mitgeschickt werden und identifiziert Ihre Anwendung, bzw. Ihren Account.

3. Das aktuelle Wetter für eine Stadt erhalten Sie als Antwort auf eine HTTP-Anfrage an diese URL `https://api.openweathermap.org/data/2.5/weather?q={CITY}&appid={KEY}`. {CITY} und {KEY} ersetzen Sie dabei durch die gewünschte Stadt bzw. Ihren API-Key. Die fertige Anfrage könnte dann z.B. so aussehen: `https://api.openweathermap.org/data/2.5/weather?q=`
4. Eine nützliche Funktion für diese Anwendung ist die One Call API. Über diese Anfrage erhalten Sie mit nur einer API-Anfrage alle Informationen um das aktuelle Wetter sowie die Wettervorhersagen für eine bestimmte Stadt anzuzeigen. Für die Anfrage müssen Sie die jeweilige Stadt über deren geographische Position identifizieren, d.h. Sie benötigen die Längen- und Breitengrade des Orts. Diese sind z.B. in der Antwort auf die Anfrage zur Ermittlung des aktuellen Wetters beinhaltet.

Achtung:: Die Anfragen an die API unterliegen einem Rate limit. D.h., über einen kostenlosen Account können pro Zeitintervall nur eine bestimmten Anzahl an Anfragen an die API gestellt werden. Die genauen Informationen dazu können Sie hier nachlesen. Die erlaubten 60 Anfragen pro Minute reichen für eine solche App sicherlich aus. Während der Entwicklung, beim Testen, können Sie dieses Limit aber schnell erreichen - behalten Sie das im Hinterkopf. Eine gute Idee, nicht nur aufgrund des *Rate Limits*, ist es, Ergebnisse von Anfragen zwischenspeichern, statt ständig neu zu erfragen. Wettervorhersagen für die nächsten Tage müssen z.B. nicht minütlich aktualisiert werden.

Das Starterpaket

Im Starterpaket finden Sie eine weitgehend leere Android-App. Wir haben Ihnen dort die `HttpRequest`-Klassen aus dem Mensa-Beispiel bereitgestellt, die Sie zur Umsetzung der HTTP-Anfragen an die API verwenden können.

Hinweise zum Vorgehen

- Sammeln Sie stichpunktartig alle Funktionen der Anwendung und leiten Sie daraus alle Aktionen ab, die NutzerInnen mit Ihrer Anwendung durchführen können
- Skizzieren Sie die beiden zentralen Bildschirme der Anwendung. Arbeiten Sie dabei iterativ und beginnen Sie erst dann mit der Umsetzung des *User Interface*, wenn Sie für alle im Vorfeld definierten Funktionen und Aktionen konkrete UI-Elemente zu deren Umsetzung bzw. Ausführung definiert haben. Ein Beispiel: Wenn NutzerInnen neue Städte hinzufügen können sollen, benötigen Sie einen entsprechenden Formular-Bereich, in der z.B. der Stadtname eingegeben wird. Ggf. ergeben sich auch weitere Abhängigkeiten oder Konsequenzen, z.B. die Notwendigkeit, NutzerInnen zu informieren, wenn zum eingegebenen Stadtnamen keine Wetterdaten gefunden werden konnten.
- Planen Sie das *Software Design* Ihrer Anwendung. Skizzieren Sie dazu grob die relevanten Komponenten, die Sie zur Umsetzung der Funktionalität benötigen. Stellen Sie sicher, dass Sie wissen, wie Informationen innerhalb der Anwendung abgebildet und zwischen den Komponenten ausgetauscht werden sollen.
- Beginnen Sie dann mit der systematischen Implementierung der App. Eine gute Strategie ist dabei die Umsetzung von *Vertical Slices*. Versuchen Sie möglichst schnell eine rudimentäre aber nutzbare Version Ihrer Anwendung herzustellen. Verzichten Sie z.B. zu Beginn auf bestimmte Fehlerbehandlungen oder bestimmtes *User Feedback*. Stellen Sie noch nicht alle verfügbaren Wetterdaten oder Vorhersagen dar. Nutzen Sie textuelle Darstellung statt Grafiken. Sobald Sie wissen, dass die zentrale *Säule* der App steht und funktioniert, können Sie damit beginnen, die einzelnen Abschnitte zu optimieren und auszubauen.