

Teamseeker

Thema und Motivation

Das Thema von Teamseeker ist das Kennenlernen neuer Leute. Die Idee für diese App kam aus der Frustration heraus, dass Kommilitonen durchgehend verhindert waren und deshalb nie eine gemeinsame Unternehmung zustande kam. Das kann daran liegen, dass jüngere Studenten meist öfter die Eltern oder Heimat besuchen, als ältere Studenten. Wir als Team wollten eine App erschaffen, mit der man schnell neue Leute kennenlernen kann, obwohl der gewohnte Kreis gerade verhindert ist. Die Idee für Teamseeker war geboren: Man geht auf eine Karte und kann in seiner Umgebung, beispielsweise in einer Bar, einem Restaurant oder auch bei sich zu Hause ein Event erstellen und für andere sichtbar machen. Die anderen Studenten können das Event auf ihrer Karte mitverfolgen und sich dafür anmelden – und plötzlich hat man einen gemeinschaftlichen Abend organisiert.

Aufgabenverteilung

Das Entwicklerteam hinter Teamseeker besteht aus **Fabian Schebera**, **Matthias Zerniekel** und **Nicole Schönwerth**. Nach einigen Meetings zum Thema Konzept, Design und Umsetzung war das Projekt gestartet. Fabian nahm sich vor, sich um die Persistenz-Aspekte zu kümmern, das heißt alles was mit Google Firebase und dergleichen zu tun hat, wurde von ihm in Angriff genommen. Matthias nahm sich das Thema Map mit den dazugehörigen Aspekten vor. Nicole hat sich um die Design-Aspekte gekümmert, dazu gehören Icons und Layouts für Activities und Fragments.

Umsetzung

Die Umsetzung erfolgte während der vorlesungsfreien Zeit, wobei die einzelnen Teammitglieder bei jeder Arbeit am Projekt mit den Anderen im Sprachchat waren. Somit konnte man sich bei Fragen helfen und sich beraten. So wurden auch Designpunkte in offener Diskussion besprochen. Wir benutzten BitBucket als Git-Client.

Zum Speichern der Events wurde Google Firebase (Firestore) verwendet. Die Datenbank erlaubt Echtzeit Updates, und schnelles hoch-/runterladen und „Queryn“ von Event-Dokumenten. Um auf komplexere Integrationen verzichten zu können wurde auch für den Eventchat die Datenbank benutzt um simples Instant-Messaging bereitzustellen. Zum Verknüpfen der Events mit Nutzern wurden die ID's, welche von Firebase Authentication bereitgestellt werden, benutzt. Nutzerprofile werden auch in der Datenbank gespeichert während die dazugehörigen Profilbilder mit der NutzerID in der Firebase Storage gespeichert werden.

Sowohl Nutzerprofile als auch die Profilbilder werden auch lokal gespeichert um Bandbreite zu sparen, bei Events war dies leider nicht praktikabel aufgrund von „möglichen“ ungleichmäßigen Updates vom Chat. Zur lokalen Speicherung von Event-IDs (Events an denen der Nutzer teilnimmt) werden die Dokument-Referenzen in den Shared Preferences gespeichert.

Zur Navigation innerhalb der App entschieden wir uns für den optisch ansprechenden Navigation Drawer (Burger Menü). Dieser ist die Grundlage jeder Activity, die layouts der Activity werden programmatisch unter den Navigation Drawer (verbunden mit der Action Bar) inflatet. Der Navigation Drawer benutzt eine Expandable List View zur Darstellung der Menü-Elemente um Untermenüs zu erlauben, hierzu zählt aktuell nur die „My Events“ Liste.