**Nutze**

Menschen die viel an neuen Orten unterwegs sind kennen das Problem sicher: Man hat keine Ahnung wo genau man gerade ist, und man braucht dringendst eine Toilette. Bis jetzt hieß es: Beine zusammenkneifen und warten, bis sich eine Gelegenheit ergibt. Dieses Problem soll unser Projekt lösen. Mit dem Toilet-Tracker soll es möglich sein öffentliche Toilette in der Nähe über eine Karte zu finden, und nach eigenem Ermessen zu Bewerten. So können sich die Benutzer sicher sein, dass die WCs, die sie aufsuchen, auch Ihren Standards gerecht werden. Um immer über aktuelle Daten zu verfügen sollen die WCs durch Benutzer eingetragen werden.

**APP**

Die Android App wird mit Hilfe von Java und Android Studio erstellt. Die Kommunikation mit dem Server erfolgt über REST.

Über die App soll es möglich sein sich einen Account zu erstellen und sich mit diesem einzuloggen. Ein eingeloggter User soll über die mit der Google Maps API integrierten Karte Toiletten finden, bewerten und einfügen können. Es soll möglich sein, sich genauere Informationen über die Toiletten anzeigen zu lassen. Beispiele dafür wären ob die Toilette Behindertengerecht ist, ob, und wenn ja, wie viel das Benutzen der Toilette kostet, wie die Benutzer die Sauberkeit der Toilette bewertet haben und vieles mehr.

Toiletten sollen über die User hinzugefügt werden. Da dies das Risiko bietet das ein Nutzer sozusagen aus „Spaß“ WCs hinzufügt die nicht existieren, soll es ein Limit an hinzufügbaren Toiletten pro Tag geben. Auch soll es ein Bewertungssystem geben, in dem Nutzer angeben können ob eine Toilette existiert, oder nicht. Wenn eine Toilette zu oft mit „Nicht Vorhanden“ bewertet wird, so wird diese von der Datenbank gelöscht.

Da es bei unserer App wichtig ist eine aktive Benutzerbasis zu haben, soll das Benutzen der App mit Punkten belohnt werden. So gibt es zum Beispiel eine Bestimmte Anzahl an Punkten für das Bewerten und Hinzufügen einer Toilette. Eine weitere Möglichkeit die Benutzer zu belohnen wären Achievements. So könnte es zum Beispiel Achievements geben für das mehrmalige Besuchen einer Toilette, das Besuchen von Toiletten in verschiedenen Ländern und so weiter. Die Nutzer könnten dann in einer Rangliste angezeigt werden.

**Chitter API**

Die API, die Daten von der Datenbank holt und verarbeitet wird von mir in Typescript mithilfe von NODE.js programmiert. An alle die es nicht wissen mit Node.js lässt sich Javasctipt Code Server seitig ausführen. Hinter Node.js verbirgt sich die JavaScript-Laufzeitumgebung "V8" die auch in Google Chrome verwendet wird. Javascript ist eine Eventgesteuerte Sprache. Dies hat im Server seitigen Betrieb den Vorteil, dass pro bestehender Verbindung weniger Arbeitsspeicher verbraucht wird als bei vergleichbaren Anwendungen die für jede Verbindung einen eigenen Thread starten. Und da unser Rest Service auf JSON basiert ist Node.js eine sehr gute wahl, da sich JSON in JavaScript nativ verarbeiten lässt. Außerdem ist Node.js mit Express.js perfekt geeignet um HTTP basierte Webanwendungen zu entwickeln und lässt sich sehr gut skalieren. Mit dem Node Package Manager kurz npm lassen sich sehr leicht sehr umfangreiche Module hinzufügen. Mit dem NPM lassen sich mittlerweile über 400.000 Pakete hinzufügen. Eines dieser Packages ist Express.js und dient als Web-Framework.

Wieso verwenden wir Typescript? Typescript hat einige Vorteile wie statische Typisierung. Statisch Typisierte Sprachen vermindern die Fehler und IDEs können dir bessere vorschläge bieten. Natürlich muss man ein bisschen mehr tippen und Typen sind optional in Typescript. Doch wenn man sich daran hält vereinfacht es das zusammenarbeiten enorm. Außerdem wieso sollte man es nicht verwenden jeder JavaScript Code ist gültiger Typescript Code. Somit ist das Umsteigen auf Typescript auch keine riesen Hürde.

Unsere Errors werden mithilfe von Sentry geloggt. Sentry ist eine moderne Error logging und Aggregations Plattform. Das Sentry Package ist eigentlich nur ein WebServer mit einem WebInterface. Für Node.js und viele andere Sprachen gibt es auch eine sehr gute SDK mit der man die Errors sehr leicht loggen kann.

WEBINTERFACE HERZEIGEN.

VLLT über JWT reden

**Chitter Website**

Unsere Website wird mithilfe des JavaScript Frameworks Angular geschrieben.