

Wochenberichte Projekt Hunde-Gassi-App

26.09.18 – 3.10.18 → Woche 1

Erstellen eines Git Repositories mit dem Ordner Dokumente. In diesem Ordner befindet sich das ebenso erstellte Pflichtenheft, sowie die wöchentlichen Berichte bezüglich des Projektes sowie die Internetrecherchen. Ebenso wurde Android Studio installiert und am Emulator Hello World ausgegeben.

03.10.18 – 10.10.18 → Woche 2

Es wurde ein Server eingerichtet, welcher die Mitteilungen an den Google Server schickt und diese dann weiterleitet zum Emulator des Android Studios.

10.10.18 – 17.10.18 → Woche 3

Es wurde mittels Google Firebase ermöglicht Pushmitteilungen per Knopfdruck zu senden. Client für Pushmitteilungen wurde in Android Studio geschrieben.

17.10.18 – 24.10.18 → Woche 4

Es wurde eine graphische Oberfläche zum Auswählen von Datum und Uhrzeit der Pushmitteilung eingebaut. Dabei wird die Notification in eine ArrayList gespeichert in welcher dann jede Minute der current Timestamp mit dem Timestamp der Nachricht verglichen wird. Ist der der aktuelle Timestamp größer gleich dem Timestamp zu dem die Nachricht geschickt werden soll, wird die Nachricht an den Emulator geschickt. Ansonsten verweilt diese solange in der Warteschleife bis der vorher beschriebene Fall eintritt.

31.10.18 – 7.11.18 → Woche 5

Es wurde die graphische XML Oberfläche überarbeitet, so dass bei einem Reiter in der die GoogleMaps Karte platziert wird. Es wurde mithilfe der Googlemaps API und einem Tutorial für Android der aktuelle Standort des Emulators mit einem blauen Punkt auf der Karte angezeigt. Es wurde ebenfalls aufgrund von Mehrfachverwendung auf abstrakte Klassen und Interfaces zurückgegriffen.

7.11.18 – 14.11.18 → Woche 6

Es wurde in der Datenbank eine neue Tabelle, die den Längengrad und den Breitengrad für jeden Punkt speichert. Ebenso wurde eine Routentabelle erstellt, in welcher die Startzeit und die Endzeit für die gegangene Route gespeichert werden.

Ebenso wurde ein Button erstellt, mit dem eine Testroute erzeugt und gestoppt werden kann. Mittels einer Datenbankmethode wird die gegangene Route samt Punkte in die Datenbank übernommen. Ebenfalls ist möglich mit der Hilfe der App und einer weiteren Methode das Datum sowie die Dauer der letzten 15 Gassigänge in der App mittels Dropdownmenü anzuzeigen

14.11.18 – 21.11.18 → Woche 7

Es wurde mittels einer Methode geschrieben, mit welcher die Distanz der gegangenen Route am Emulator ausgegeben werden kann. Dabei werden mit der Hilfe der vorgegebenen Methode DistanceBetween die Distanz zwischen dem jetzigem und dem nachfolgenden Punkten gemessen und diese dann in ein Array gespeichert. Die Gesamtanzahl der Strecke wird in Meter dann am Emulator angezeigt.

Weiters wurde ein Login- bzw. Registriertab erstellt, welcher sich beim Starten der App öffnet. Es wurden Servlets und Datenbankmethoden fürs Registrieren bzw. zum Einloggen erstellt. Nach erfolgreichem Einloggen befindet man sich im Hauptmenü der App, in dem man sich Erinnerungen selbst schicken kann bzw. neue Routen aufzeichnen und sich anzeigen lassen kann.

Der Login kann mittels Email oder Benutzername und durch Passwort erfolgen. Bei falschen Daten wird eine Meldung geöffnet, die den User berichtet, dass die Anmeldedaten nicht korrekt sind.

Beim Registrieren werden überprüft, ob Username bzw. Emailadresse bereits existiert. Trifft dies zu, wird wiederum eine Meldung mit der entsprechenden Nachricht geöffnet. Ebenfalls wird überprüft, ob der sich registrierende User einen anderen fcmToken vorweist, wie bereits registrierte User. Trifft dieser Fall ein wird ebenfalls eine Meldung geöffnet, die den User darauf hinweist, dass Registrierungen mit demselben fcmToken nicht möglich seien.

21.11.18 – 28.11.18 → Woche 8

Es wurde eine Settingsseite in XML designt, in welcher die IP-Adresse der Datenbank, der Port und der Name des Servers angegeben werden.

Dazu wurde ein asynchroner Task geschrieben der dann eine Verbindung mit der Datenbank, dem Port und dem Server aufnimmt.

Nachdem diese Parameter in den Einstellungen getroffen wurden, kann man sich Anmelden bzw. Registrieren.

28.11.18 – 5.12.18 → Woche 9

Keine Meilensteine bearbeitet, da vorausgearbeitet wurde

5.12.18 – 12.10.18 → Woche 10

Es wurde ein Contactstab in XML designt, in welcher mittels der Datenbankmethode getAllUsers alle registrierten User angezeigt werden. Diese Methode wird im GetFriendsServlet aufgerufen und dann als JSON zum Android Gerät übermittelt. Im GetMyFriendsTask wird das JSON Objekt wieder in ein GSON Objekt umgewandelt und dann im Contactstab angezeigt. Mit einem Klick auf den entsprechenden User wird dann die Pushnachricht sofort übermittelt.