



Entwicklung mobiler Anwendungen

Projekt Fahrtenbuch
Dokumentation

Jannik Ostermayer

Matrikelnummer 672330

16. Januar 2019

Inhaltsverzeichnis

1 Idee

Meine Idee ist eine Fahrtenbuchapp zum leichten Übertragen der Fahrten in ein physisches Fahrtenbuch. Da es noch keine verifizierte Fahrtenbuchapp des Finanzamtes gibt und so auch keine Rechtssicherheit herrscht. Zum leichteren übertragen sollte es auch eine Exportfunktion geben oder die Möglichkeit sich eine Email zu schicken. Da das übertragen in ein Fahrtenbuch zeitnahgeschehen (innerhalb der nächsten drei Tage) erfolgen muss sollte es auch Push Benachrichtungen zur Erinnerung geben. Die App kann durch einen Button gestartet werden, aber auch alternativ durch eine Bluetooth Verbindung oder einen NFC Chip. Die erfassten Fahrten sollten auch auf einer Karte angezeigt werden, da man bei größeren Umwegen, diese genau beschreiben muss.

2 Beschreibung

Diese App vereinfacht das führen eines Fahrtenbuches enorm. Alle Fahrten werden erfasst und du wirst Benachrichtigt, dass du dein Fahrtenahrtenbuch zeitnah führen kannst. Außerdem gibt es Export-Funktionen das du dir deine erfasseten Fahrten auch bequem auf deinem PC anschauen kannst.

3 Zielgruppe

Der potentielle Markt für eine Fahrtenbuch App erstreckt sich über alle gewerblich genutzt Fahrzeuge in Deutschland. Das sind laut einer Statistik des KBA (Kraft Fahrt Bundesamtes) aus dem Jahr 2017 64,4% der zugelassenen Fahrzeuge in Deutschland. In absoluten Zahlen sind das 2.215.208 Gewerblich neu zugelassen Fahrzeuge. Meine App richtet sich aber spezial an Personen die einen Firmenwagen fahren, wie Vertreter. Da es bei den Statistiken nur eine unterscheidung zwischen geschäftlicher und privater Nutzung gibt, existieren keine genauen Zahlen über Anzahl der Firmenwagen unter den gewerblich genutzten Fahrzeugen. Man geht von ca 10% der gesamten Fahrzeuge aus. Laut KBA vom Stand 1. Januar 2018 gibt es in Deutschland 46,5 Millionen zugelessene PKWs. Das bedeutet wenn man von 10% ausgeht, gibt es in Deutschland 4,65 Millionen Firmenfahrzeuge. Da sich meine App an Fahrer eines Firmenwagens richtet die Steuern sparen wollen und eine erleichterung beim führen eines Fahrtenbuches suchen, trifft meine App auf einen potentiellen Markt von 4,65 Millionen nutzern. Bei dem Führen von Fahrtenbüchern gibt es starke unterschiede zwischen den Berufsgruppen, so muss zum Beispiel ein Handwerker für jede Fahrt den Grund der Fahrt angeben. Dahingegen müssen Handelsvertreter und Kundendienstmonteure keinen Grund der Fahrt angeben, wenn der Grund plausibel ist, hier reicht wenn der Kunde angegeben wird. Bei diesen Berufsgruppen die nicht alles angeben müssen spricht man von einem *erleichterten* Fahrtenbuch. In erste Linie soll sich meine App an Personen die ein solches *erleichtertes* Fahrtenbuch führen richten. Im nächsten Schritt, wenn meine App ausgebaut wird würde ich auch gerne die anderen Berufsgruppen, die ein *normales* Fahrtenbuch führen in meine Zielgruppe mitaufnehmen.

4 Marktanalyse

Die meisten Apps im Play Store sollen ein Fahrtenbuch ersetzen. Diese bieten nur eingeschränkte Funktionalitäten in der kostenlosen Version. Meist muss ein Abo abge-

schlossen werden für 5-10€ pro Monat, dafür gibt es meistens noch eine Web-Lösung und diemöglichkeiten in Manipulationssichere Formate zu exportieren. Exemplarisch werde ich ein paar Apps näher Analysieren.

4.1 GPS Zeiterfassung + Fahrtenbuch

Unter den Apps die ich analysiert habe wurde diese am häufigsten heruntergeladen mit über 100.000 Downloads. Die App wurde durchschnittlich mit 4,6 Sternen bewertet bei 1.644 Rezessionen. Es gibt eine kostenlose Variante und eine Pro Version für 9,99€.

4.1.1 Design und Funktionalität

Auf ?? sehen Sie den Startbildschirm der App. In der obere Hälfte des Bildschirm befinden sich Buttons „Abfahrt“ und „Ankunft“ um eine Aufzeichnung zu starten und zu beende. Es gibt einen Button um den Kilometerstand des Autos zu korrigiert. Außerdem kann man Kosten für das Auto und wie viel man getankt festhalten, was für meine App aber eher uninteressant ist. In der unteren Hälfte des Bildschirm befindet sich eine Liste der erfassten fahrten und rechts unten ein optionen Button (siehe ??). Wenn man auf eine erfassete fahrt klickt, siehe ??, werden neue Buttons in der oberen hälfte des Bildschirms geladen. Man hat die Möglichkeit die Fahrt als „Dienst“, „Privat“ oder „Sonst.“ zu markieren, den Favoriten hinzuzufügen, den Eintrag löschen, Fahrt auf einer Karte anzusehen (siehe ??) oder das hinzuzufügen von Notizen oder eine Trennlinie in der Listen ansicht.

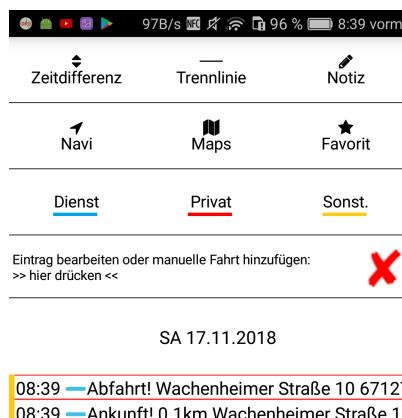


Abbildung 1: GPS Z+F Startbildschirm.



Abbildung 2: GPS Z+F Fahrt ausgewählt.

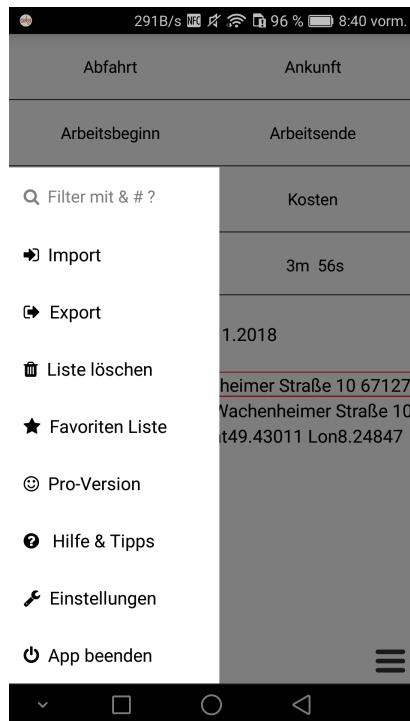


Abbildung 3: GPS Z+F Einstellungen.

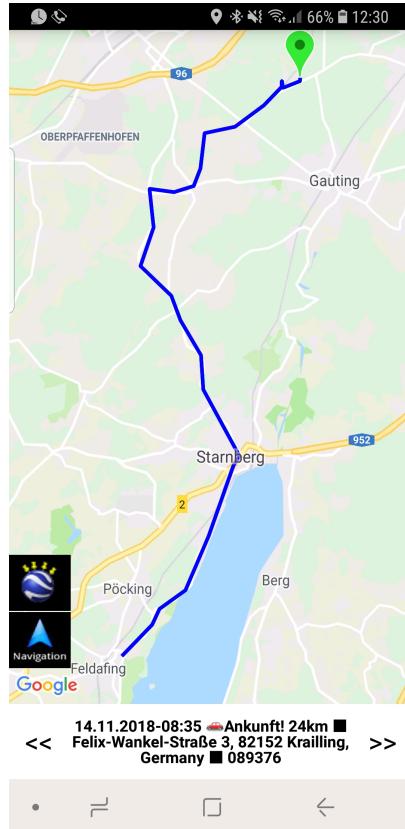


Abbildung 4: GPS Z+F aufgezeichnete Fahrt auf Karte.

4.1.2 Kommentare im Play Store

1. Der automatische Start und Ende der Aufzeichnungen ist sehr gut.
2. Nicht Finanzamt konform da Änderungen nicht nachvollziehbar sind (nur in der Pro Version).
3. Bedienung nicht intuitiv.

4.1.3 Fazit

Diese App ist im punkte Funktionalität woll die beste Lösung die es ohne Abo gibt. Aber designtechnisch sehe ich noch Verbesserungspotential. Die Hälfte des Bildschirms besteht nur aus Buttons, die Liste der erfassten Fahrten ist sehr klein und der optionen Button im rechten unteren Eck entspricht nicht den Style-Guides. Außerdem wird der Eintrag sofort gelöscht wenn man auf löschen klickt. Gut ist die Ansicht der Routen auf einer Karte.

4.2 kfz fahrtenbuch

Diese App wurde über 10.000 mal heruntergeladen und befindet sich damit im mittelfeld von den von mir analysierten Apps. Durchschnittlich wurde die App mit 2,9 Sternen bei 52 Rezessionen bewertet. Diese App wird in Verbindung mit einer Cloud Lösung mit einer Web Anwendung angeboten und ist mit einem monatlichen Abo verbunden. Wenn man die kostenlose Version benutzt kann man nur Fahrten per Hand eintragen.

4.2.1 Design und Funktionalität

Auf 5 sehen Sie den Startbildschirm. Auf diesem gibt es drei Buttons, *Fahrt aufzeichnen*, *Fahrt erfassen* und *Fahrten bearbeiten*. Die möglichkeit eine Fahr aufzuzeichnen gibt es nur wenn man ein Abo abgeschlossen hat. 6 zeigt einen Scrennshot aus dem Play Store wie die Oberfläche einer solchen Aufzeichnung aussieht. Da ich keine möglichkeite hatte die Funktion selbst zu testen werde ich hier nicht näher drauf eingehen. Klickt man auf den Button *Fahrt erfassen* sieht man die Oberfläche aus 7. Ganz Oben kann man die Art der Fahr auswählen. In den unteren Zeilen kann man Start-, Zieladresse, Distanz und Kilometerstand eingeben. Klickt man auf dem Startbildschirm auf *Fahrten bearbeiten* kommt man zum Verlauf (siehe 8). Hier gibt es eine wie ein Zeitstrahl aufgebauten Ansicht aller fahrten. Bei Bedarf kann man erfasste Fahrten vervollständigen, dazu öffnet sich wieder die selbe Oberfläche wie zum erfassen einer Fahrt.



Abbildung 5: kfz Fahrtenbuch:
Startbildschirm

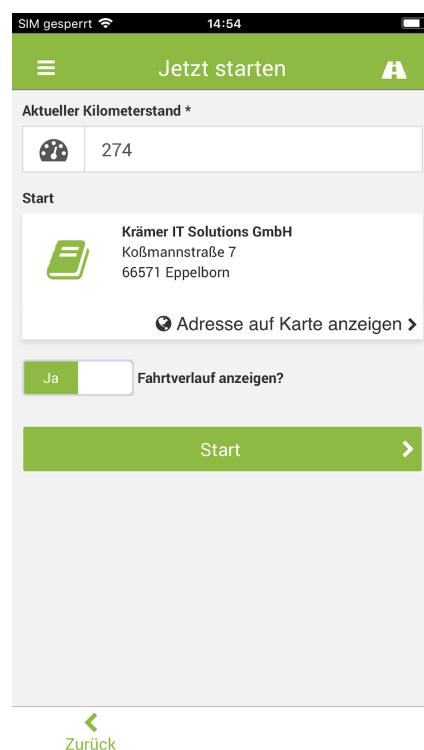


Abbildung 6: kfz Fahrtenbuch:
fahrt aufzeichnen

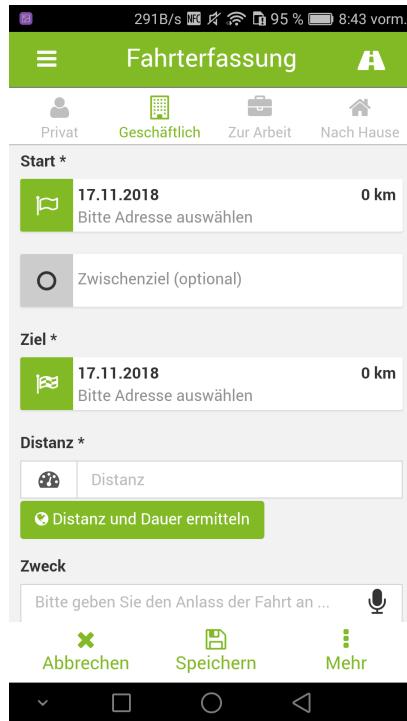


Abbildung 7: kfz Fahrtenbuch:
fahrt erfassen

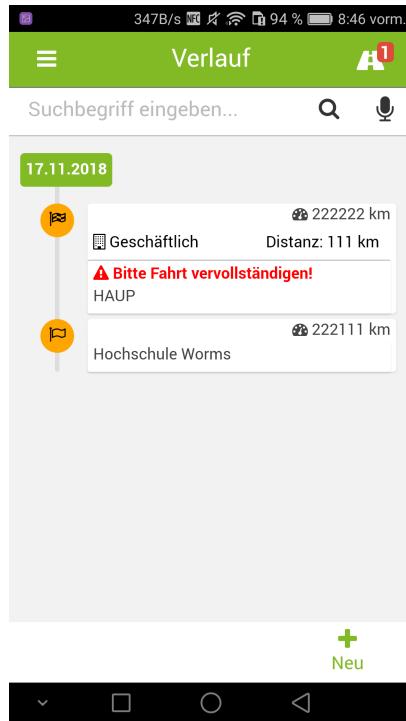


Abbildung 8: kfz Fahrtenbuch:
fahrt bearbeiten

4.2.2 Kommentare im Play Store

Pries Leistung Verhältnis der Cloud LÖsung wird kritisiert

1.

4.2.3 Fazit

Die Funktionalität in der kostenlosen Version ist sehr eingeschränkt und bietet nicht einmal die Möglichkeit der Aufzeichnung durch GPS. Das Design der Oberfläche ist moderner und übersichtlicher im Vergleich zu der ersten App. Aber wenn man die eingeschränkte Funktionalität in der kostenlosen Version berücksichtigt halte ich diese App im Vergleich für die schlechteste, was auch die Bewertungen im Play Store wiederspiegeln.

4.3 SquareTrip

Square Trip hat durchschnittlich 4,1 Sterne bei 118 Rezessionen bekommen und wurde über 10.000 mal heruntergeladen. Es gibt eine kostenlose und eine Pro Version.

4.3.1 Design und Funktionalität

9 zeigt den Startbildschirm der App beim ersten Starten und 10 die Ansicht wenn schon Fahrten erfasst wurden. Im ersten oberen drittel sieht es ganz links einen Button um

zwischen den Ansichten zu wechseln. Wenn man auf den Stern klickt kann man erfasste fahrten zuordnen, wie in 11, ob es eine Private, Geschäftlichefahrt oder Arbeitsweg war. In der Mitte gibt es ein Icon mit dem man seine Autos verwalten kann. Zum Verwalten der Autos kann man ein Bild hinzufügen, Name/Modell angeben, Nummerschild und den Kilometerstand. Daneben kann man Fahrten anhand eines Filters suchen. Mit dem + Symbol kann man eine neue Fahrt anlegen, siehe 12. Die App erfasst fahrten automatisch, das bedeutet sie läuft permanent im Hintergrund und wartet auf ein GPS-Signal. Außerdem bittet die App die möglichkeit die letzte Parkposition seines Auto anzuzeigen und die laufenden Kosten des Autos zu erfassen, da das für meine App aber eher uninteressant ist habe ich mir diese Funktionalitäten nicht näher angeschaut.

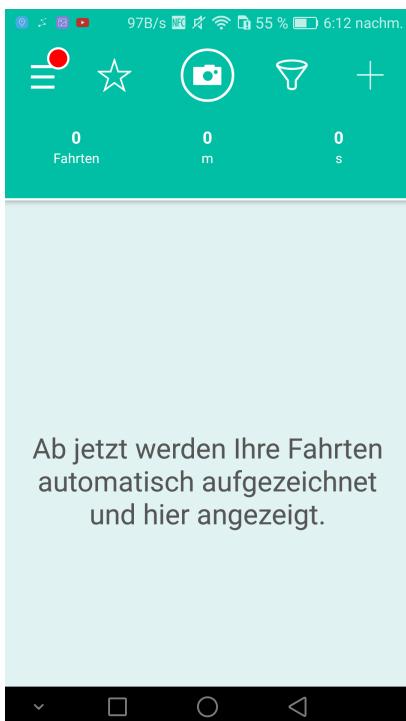


Abbildung 9: GPS Z+F Startbildschirm.

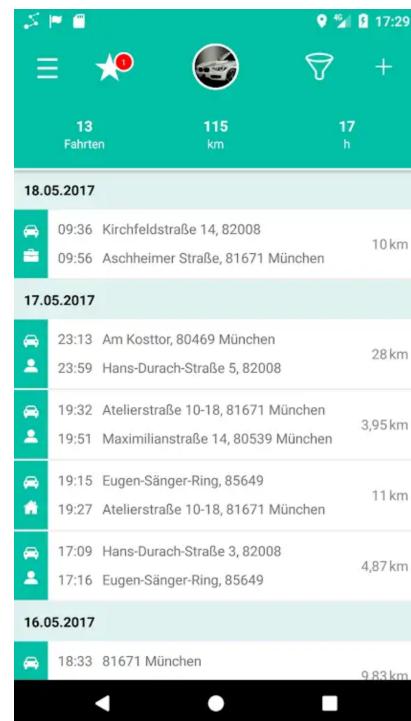


Abbildung 10: GPS Z+F Fahrt ausgewählt.

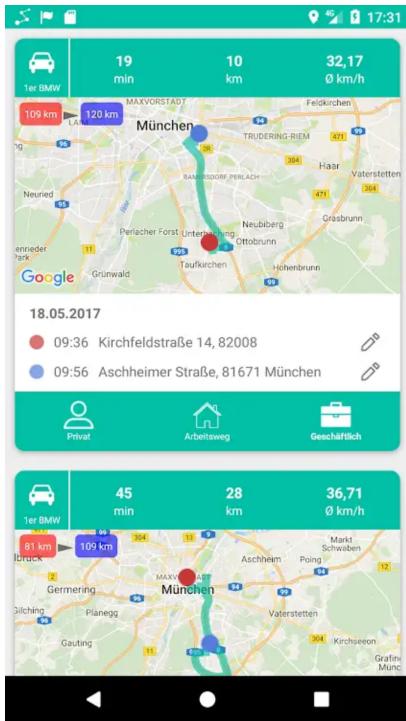


Abbildung 11: GPS Z+F Einstellungen.

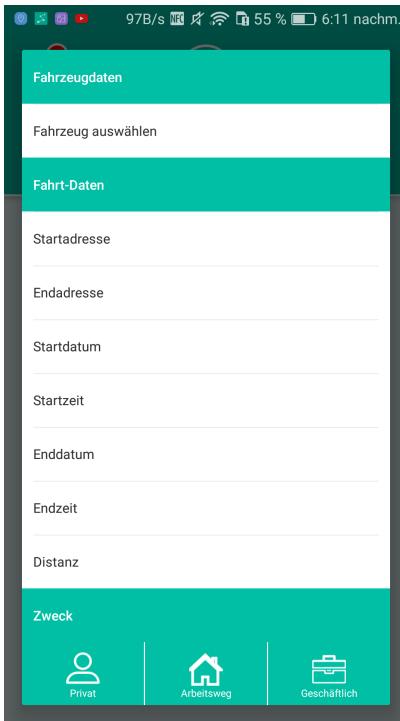


Abbildung 12: GPS Z+F aufgezeichnete Fahrt auf Karte.

4.3.2 Kommentare im Play Store

1. Die meisten Bewertungen sind zufrieden mit der App

4.3.3 Fazit

Die Icons sind nicht unbedingt selbst erklärend. Das Design gefällt mir gut, die Design Guidelines wurden eingehalten. Die App bietet auch die Möglichkeit mehrere Autos zu verwalten, was aber für meine Zielgruppe eher uninteressant ist. Problematisch ist das die App permanent im Hintergrund läuft und auf ein GPS-Signal wartet, was sich negativ auf die Akkuleistung auswirkt und außerdem gibt es keinen Button zum starten einer Fahrt. Gut ist auch das die gefahrene Strecke auf einer Karte angezeigt wird. Exportfunktionen gibt es nur in der Pro Version.

4.4 Driverslog Pro 2 - Fahrtenbuch

Diese App hat eine durchschnittliche Bewertung von 3.9 Sternen bei 60 Rezensionen und über 5.000 Downloads. Es gibt keine kostenlose Version dieser Anwendung, nur eine Abo Version und eine einmonatige Testlizenz beim Download. Beim ersten Start ist die App abgestürzt und nicht mehr gestartet. Nach einem Neustart des Handys konnte man die App benutzen.

4.4.1 Design und Funktionalität

13 zeigt die Übersicht der fahrten. Beim start der App wird man auch auf diese Seite geleitet. Wählt man eine Fahrt aus der Liste aus so gelangt man auf ???. Auf 14 sieht man die Oberfläche wenn man eine neue Fahrt hinzufügen bzw. erfassen möchte. Klickt man auf den Button manuell wird man auf 16 geleitet. Wählt man erfassen kommt man auf 15. Hier wird die aktuelle Position auf einer Karte angezeigt, die zurückgelegten Kilometer und dafür benötigte Zeit. Klickt man auf abschließen landet man wieder auf 16 nur das einige Felder schon durch die gewonnenen GPS Information ausgefüllt wurden.



Abbildung 13: Driverslog Pro 2 - Fahrtenbuch, Listenansicht

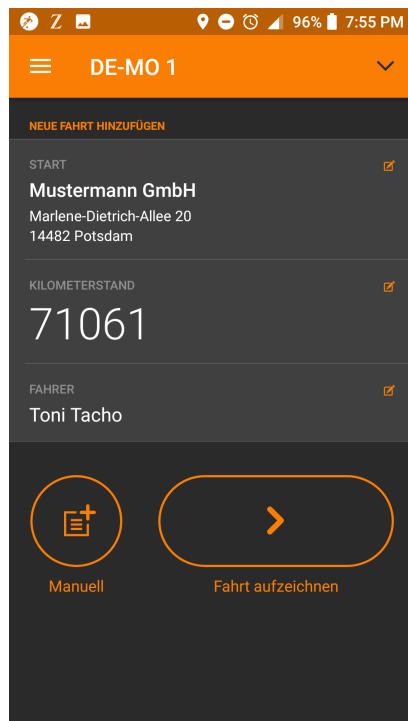


Abbildung 14: Driverslog Pro 2 - Fahrtenbuch, Fahrt hinzufügen



Abbildung 15: Driverslog Pro 2 - Fahrtenbuch, Fahrt aufzeichnen



Abbildung 16: Driverslog Pro 2 - Fahrtenbuch, Fahrt anlegen

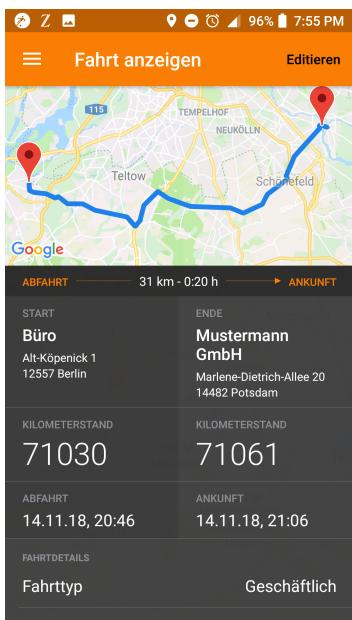


Abbildung 17: Driverslog Pro 2 - Fahrtenbuch, Detailansicht

4.4.2 Kommentare im Play Store

1.

Tabelle 1: Spalten des Fahrtenbuches

Datum	Fahrzeit von - bis	Zweck der Fahrt	Besuchte Person, Firma, Behörde	km-Stand Fahrtbeginn	Gefahr
1	Home	HS	lernen		

Tabelle 2: Aufbau Datenbank

id	start	destination	purpose	mileage _{start}	mileage _{destination}	odometer
----	-------	-------------	---------	--------------------------	--------------------------------	----------

4.4.3 Fazit

Diese App benutzt die selbe Oberfläche um Fahrten zu erfassen, hinzuzügen und für die Detailansicht

4.5 Fazit

Generell lässt sich sagen das es im Play Store eher wenige Apps gibt.

5 Bedarfsanalyse

5.1 Fahrt hinzufügen

In der App soll man in einer Suchleiste die Startadresse eingeben und in einer anderen die Zieladresse. Zur Autovervollständigung der Adresse werde ich die Google Places API verwenden.

5.2 gefahrene Strecke auf Karte anzeigen

Dafür werde ich die Google Maps SDK verwenden.

5.3 Datenbank

5.4 Aufzeichnen Der Fahrt

Um die Fahrt aufzuzeichnen

5.4.1 Position bestimmen

Um die Position unter Android zu ermitteln gibt es drei Wege:

1. LocationManager LocationListener
2. fusion location client
3. fusion location provider

LocationManager LocationListener Bei dieser Möglichkeit werden die Boardmittel von ANDROID verwendet um die Position bestimmen. Diese Methode hat den größten Akkuverbrauch und wird von Google nicht empfohlen.

Fusion Location client Bei dieser Methode wird der Google Location Service verwendet der sowieso im Hintergrund läuft. Bei dieser Methode startet man ein update

Request bei dem man die Rahmenbedingungen wie Priorität min und max Zeit zwischen den Updates einstellen kann. Diese Methode ist Akkuschernder.

Fusion Location Provider Diese Methode ist im Prinzip eine verbesserte und vereinfachte Form des Fusion Location Client und wird von Google empfohlen.

5.4.2 Entfernung bestimmen

Die bestimmten Positionen werden gesammelt bis das Aufzeichnen der Fahrt beendet wird. Um die zurückgelegte Entfernung zu bestimmen, muss die Entfernung zwischen den einzelnen Punkten bestimmt werden. Dafür hat die Klasse 'Location' schon die Funktion 'distanceTo(Location)'. Ein Codeausschnitt um die Entfernung zu bestimmten könnte wie folgt aussehen

```
““ fun getDistance(locations: List<Location>): Int for (l in locations.indices) if (l + 1 < locations.size) distance += locations[l].distanceTo(locations[l + 1]) return distance.toInt() / 1000 //meter to kilometer ““
```

5.4.3 Start- und Zieladresse ermitteln

Bevor man die Fahrt in die Datenbank schreiben kann benötigt man außer den GPS-Koordinaten, Ankunftszeit, Abfahrtszeit und Entfernung auch die Adresse der ersten und letzten Position. Um diese zu ermitteln benötigt man eine API die eine GPS-Position entgegennimmt und eine Adresse zurückgibt. Da ich eh schon die Google Maps API verwende, habe ich mich für Geocoding entschieden. Das ist eine Google API die genau das macht, GPS-Koordinaten einer Adresse zuordnen und umgekehrt.

5.4.4 Foreground Service

Um das Aufzeichnen der Fahrt sollte auch funktionieren wenn die App nicht im Vordergrund läuft. Deswegen sollte das Aufzeichnen der Fahrt im Hintergrund laufen. Seit Android Oreo gibt es Einschränkungen für "Background Threads" und "Background Location Limits". Diese besagen das wenn man die Position mehr als ein paar mal pro Stunde Abfragt, dann ist das nur in einem 'Foreground Service' möglich. Außerdem wurden 'Foreground Services' auch limitiert, diese haben nach dem Start 5 Sekunden Zeit eine Notifaction anzuzeigen sonst werden diese vom System gekillt.

5.4.5 Darstellung der Fahrt auf Karte

In der Detailansicht sollte es eine Karte geben auf der die Start und Zielposition bei hinzugefügten Fahrten angezeigt werden und bei aufgezeichneten Fahrten außerdem noch der Weg. Für die Darstellung auf der Karte werde ich die Google Maps SDK verwenden. Die Start und Zieladresse werden als Marker dargestellt. Bei einer aufgezeichneten Fahrt wird dann noch eine Verbindungsline zwischen allen erfassten Positionen, mithilfe von 'Polylines'. Das hinzufügen eines Markes und Polylienes könnte dann wie folgt ausschauen.

```
““ map.addMarker( MarkerOptions() .position(destinationLatLng).title(getString(R.string.destinationAddress) + ":" + destination.address) )
if (routes != null) wenn es GPS-Position zwischen Start und Zieladresse gibt val options = PolylineOptions().width(8f) .color(ContextCompat.getColor(applicationContext, R.color.colorPrimaryDark))
routes.forEach options.add(it) map.addPolyline(options) ““
```

5.5 Bluetooth

Die App soll das aufzeichnen Automatisch starten wenn eine Bluetoothverbindung aufgebaut wird. Dafür benötige ich einen Broadcastreciever der getriggert wird wenn eine Bluetoothverbindung aufgebaut oder beendet wird. Das auslösen eines Recievers wenn eine Bluetoothverbindung aufgebaut wird geschieht über android.bluetooth.device.action.AC_LCONNECTED". undd
““ <receiver android:name=".receiver.StartActivityOnBluetooth> <intent-filter> <action android:name=android.bluetooth.device.action.AC_LCONNECTED >< action android:name = android.bluetooth.device.action.AC_LDISCONNECTED >< /intent-filter >< /receiver > ““

5.6 NFC

Meine App ist nach dem MVVM Pattern und dem Single-Activity-Pattern aufgebaut

5.6.1 MVVM

Das steht für Model, ViewModel, View.

5.7 Einstellungen

Der Benutzer soll die Möglichkeit haben einzustellen wann er Benachrichtigt gespeichert in den Shared Preferences

5.8 View

6 Architectur

Da ich die App auch gern in Zukunft weiter entwickeln möchte wollte ich die App so moder wie möglich gestalten. Android Jetpack

7 Features

7.1 Kernfeatures

Diese Features müssen auf jeden Fall umgesetzt werden, damit man die App benutzen kann.

1. Alle Fahrten werden in einer Liste angezeigt.
2. Es gibt eine Detailansicht für jede Fahrt.
3. Bei aufgezeichneten Fahrten muss es in der Detailansicht noch eine Karte mit dem eingezeichneten Weg geben.
4. Man muss einzelne Fahrten löschen können

7.2 Features

1. Einträge editieren
2. Einträge als Email exportieren.
3. Die Email ist so formatiert das man den Text der Email einfach in Excel einfügen kann.
4. Alle Einträge vor einem bestimmten Datum löschen.
5. Benachrichtigung als Erinnerung, dass man die Einträge in sein Fahrtenbuch übernimmt.
6. Anpassbare Listenübersicht, man kann selbst bestimmen welche Informationen angezeigt werden.
7. Die App startet das aufzeichen Automatisch wenn man sich per Bluetooth mit dem Auto verbindet.
8. Man kann den Kilometerstand korrigieren.
9. Man kann in den Einstellungen den Tag und den Abstand bestimmen in dem man Benachrichtigt wird.
10. Wenn man einen Eintrag löscht werden die Kilometerstände aller neueren Einträge um die Distanz des gelöschten Eintrags verringert.

7.3 Anforderungen

1. Die Fahrtenbuch Daten sollen in einer Datenbank auf dem Handy gespeichert werden.
2. Alle Einträge der Datenbank sollen in einer Liste dargestellt werden.
3. Die Liste enthält die Information Zeit, Zweck, Start, Ziel
4. Die Liste kann man nach Datum sortieren.
5. Der Benutzer kann in der Liste suchen.
6. Der Benutzer kann auf ein Element der Liste klicken und bekommt eine Detailsansicht.
7. Die App speichert die Daten in der Form wie die vom Finanzamt anerkannten Fahrtenbücher.
8. Die App muss per GPS erfassen wenn der Benutzer losfährt und automatisch Start-, Endpunkt und Entfernung bestimmen und diese eintragen.
9. Die App muss den Benutzer auffordern fehlende Information, wie „Zweck der Fahrt nachzutragen“.
10. Der Benutzer muss Fahrten in der App händisch eintragen können.
11. Der Benutzer muss alle Fahrtenbuch Daten exportieren können als PDF oder HTML(bzw. Jason).

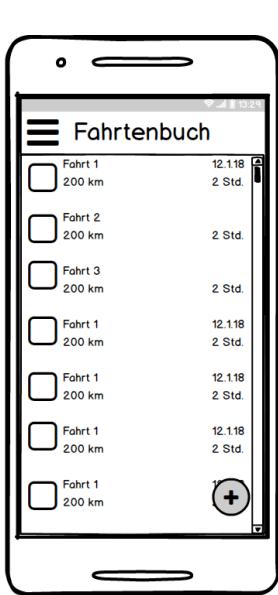


Abbildung 18: , Startbildschirm

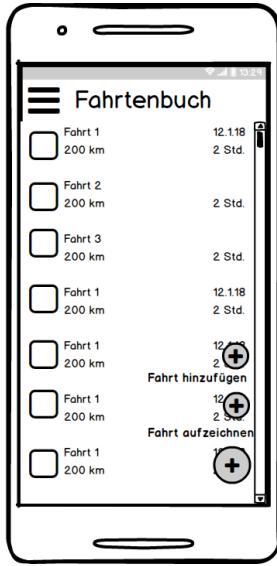


Abbildung 19: Driverslog Pro 2 - Fahrtenbuch, Detailansicht

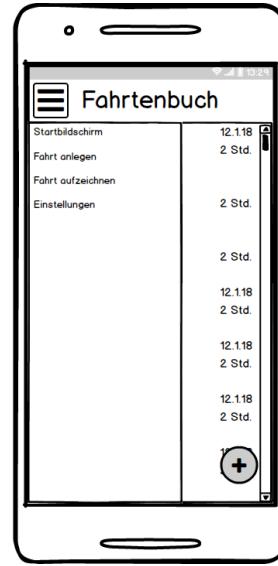


Abbildung 20: Driverslog Pro 2 - Fahrtenbuch, Detailansicht

12. Die App soll automatisch starten wenn das Handy per Bluetooth mit dem Auto gekoppelt wird.
13. Der Benutzer soll Einträge bearbeiten können.
14. Der Benutzer soll Einträge löschen können.

8 User Stories, Personas

9 Mockup, Wireframe

10 UML-Diagramme

11 Code Dokumentation

12 Fazit

13 Aufwand



Abbildung 21: , Startbildschirm



Abbildung 22: Driverslog Pro 2 - Fahrtenbuch, Detailansicht



Abbildung 23: Driverslog Pro 2 - Fahrtenbuch, Detailansicht