

HOCHSCHULE FURTWANGEN

AIB

Android SS 2012

Prof. Dr. Lothar Piepmeyer

Dokumentation

BlueTrack

– App zur Erhebung statistischer Daten über Fortbewegung –



vorgelegt von:

Tim Schmiedl, Milos Babic

Furtwangen, den 20. Juni 2012

Inhalt

1	BLUETRACK.....	1
2	ZIELE.....	1
2.1	MUSSZIELE.....	1
2.2	KANNZIELE.....	1
2.3	FUNKTIONALITÄTEN ALS USE CASE DIAGRAMM.....	2
3	FEATURES.....	3
4	BENUTZERHILFE.....	3
5	PROGRAMM-ARCHITEKTUR	4
6	PROBLEME & SCHWIERIGKEITEN.....	5
6.1	GPS.....	5
6.2	SENSOREN.....	6
7	WEITERENTWICKLUNG.....	6
8	STATISTIKEN.....	7
8.1	LINES OF CODES.....	7
8.2	SONSTIGES.....	7
9	MÖGLICHE VERBESSERUNGEN.....	7

1 BlueTrack

Mit BlueTrack werden Routen, die zu Fuß, mit Fahrrad oder mit Auto zurückgelegt werden, per GPS aufgenommen. Diese Tracks werden mit vielen weiteren Informationen wie z.B Anzahl der Schritte bei Joggen oder Durchschnittsgeschwindigkeit in der Datenbank abgelegt, welche dann jederzeit zur späteren Analyse verwendet werden können. Jogger und Fahrradfahrer können somit ihre Trainingserfolge beobachten. Weiterhin kann man sich die genau zurückgelegte Strecke später mithilfe von Google-Maps metergenau anzeigen lassen um so auch eventuell beim nächsten Mal neue Strecken zu erkunden.

2 Ziele

2.1 Mussziele

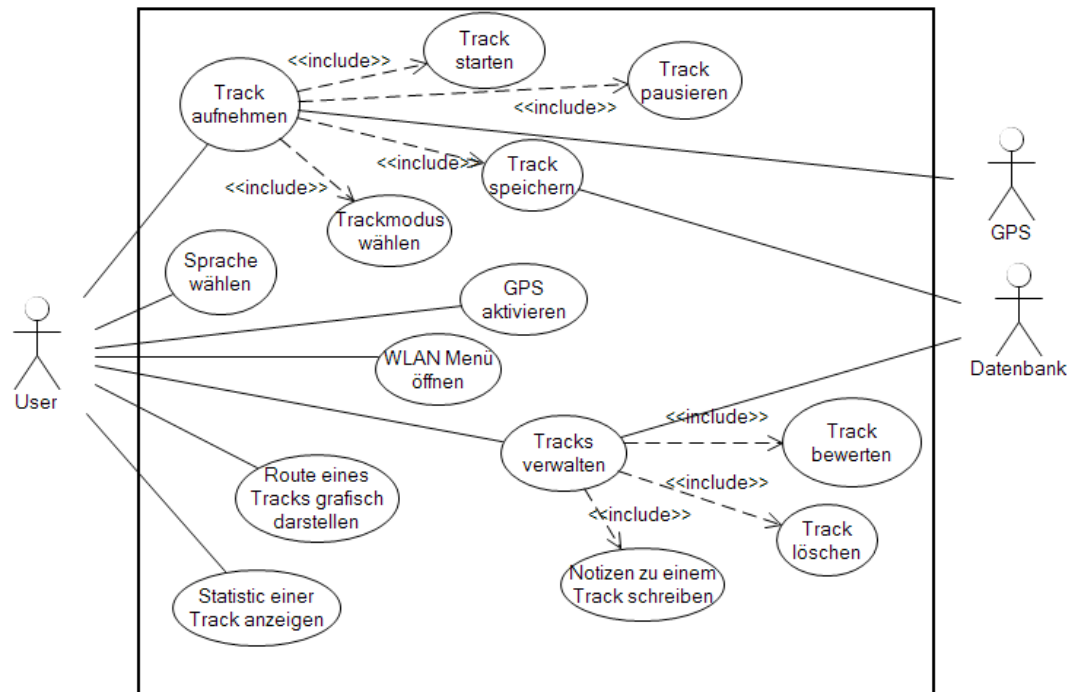
- Protokollierung der Standorte, zurückgelegte Strecken und Höhen für jede Trainingseinheit.
- Trainingseinheiten speicherbar und für spätere Analyse wiederverwendbar.
- Grafisch ansprechende Statistiken in der App mit grafischer Darstellung der zurückgelegten Strecke (mithilfe von Googlemaps).
- Schrittzähler implementieren.

2.2 Kannziele

- Googlemaps verbinden um weitere Rad/Joggingwege zu finden.
- Statistik auf Computer übertragen.
- Liveanzeige der Strecke.
- Anpassung der App an unterschiedliche Auflösungen (Tablet).

Die Mussziele wurden alle eingehalten, auch die Kannziele (mit Ausnahme der Übertragung auf den Computer) wurden erfüllt. Zur Übertragung an den Computer siehe Weiterentwicklung.

2.3 Funktionalitäten als Use Case Diagramm

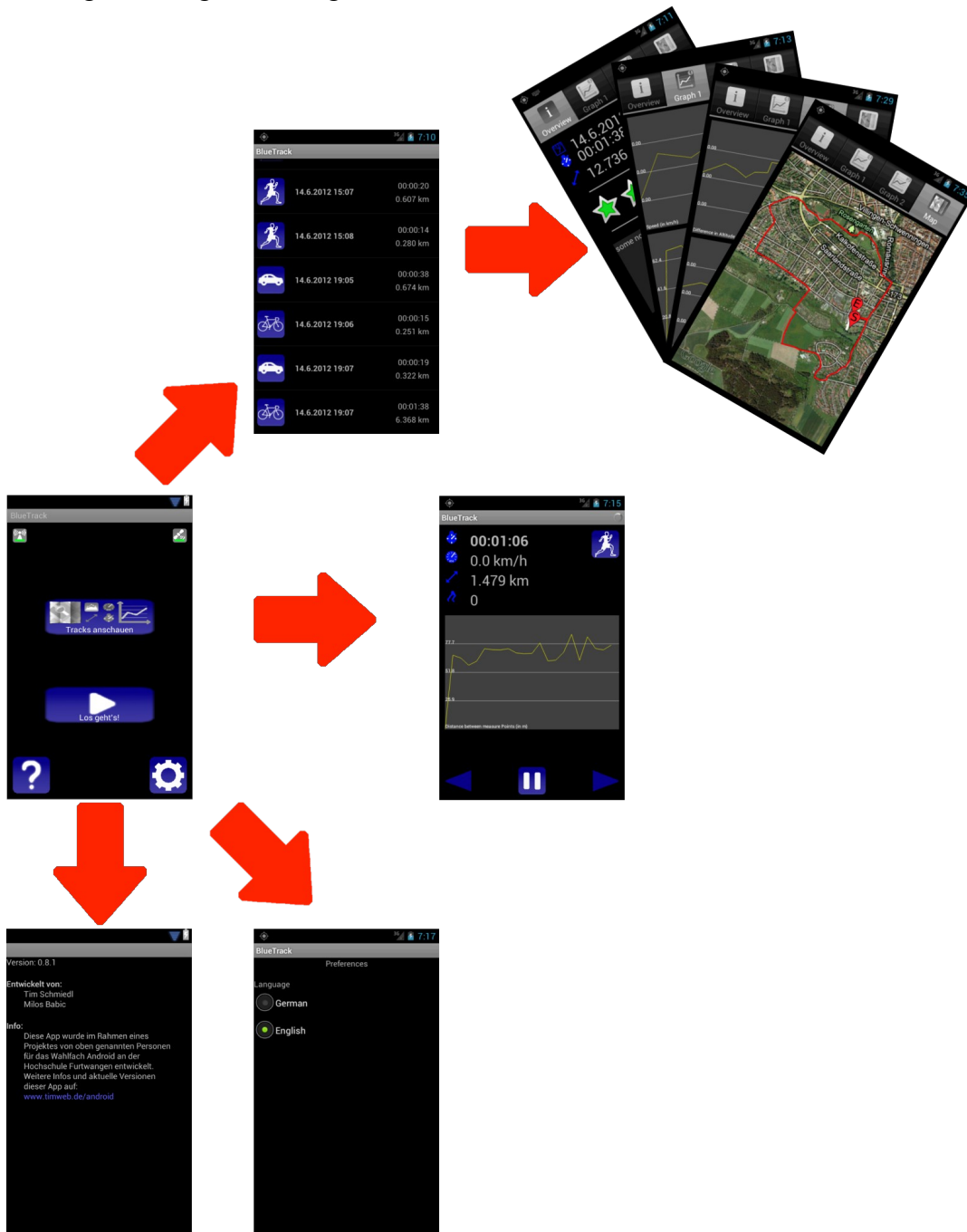


3 Features

- Modi: Joggen, Fahrrad und Auto fahren.
- Live-Diagramme über Geschwindigkeit, Höhenunterschied, Distanz zwischen Messpunkten und Länge der Schritten.
- Statistik samt Live-Diagramme zu jedem aufgenommenem Track abrufbar.
- Tracking ist pausierbar
- Genauigkeit der Darstellung der zurückgelegten Strecke auf GoogleMap einstellbar.
- Eintragen von Rating und Notiz zu den einzelnen Tracks.
- Sprachpakete: Deutsch und Englisch.
- automatisiertes Senden von Absturzberichten (ACRA („Application Crash Report for Android“)- <https://code.google.com/p/acra/>)

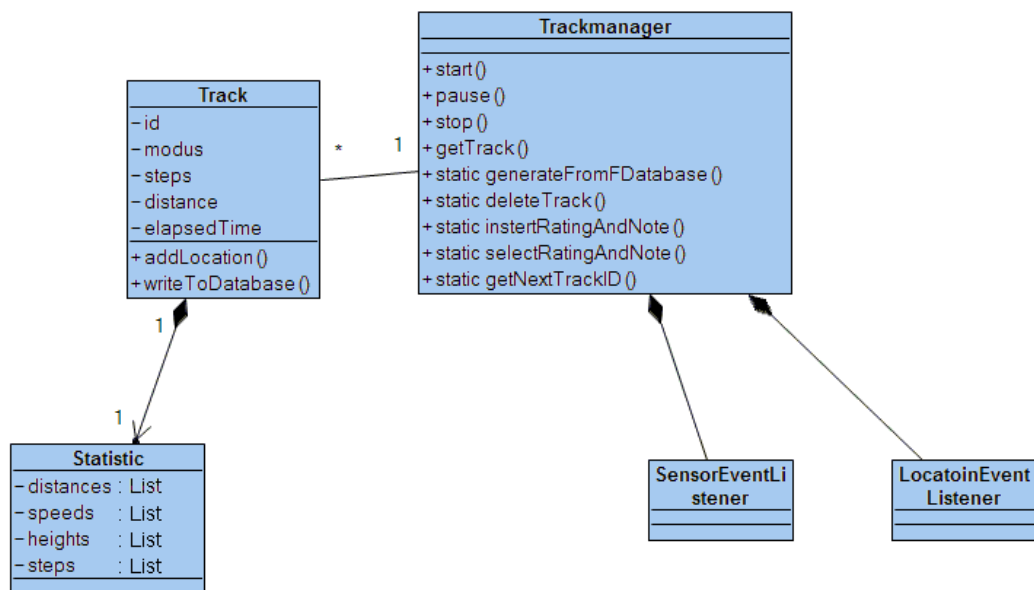
4 Benutzerhilfe

Die Bedienung der App ist möglichst intuitiv gehalten. Um einen kompletten Überblick über alle Activities zu bekommen und so ein Eindruck von der Navigation zu erhalten gibt es folgendes Diagramm.



5 Programm-Architektur

Um die App auch weiterentwickeln zu können, war uns eine solide und modulare Architektur sehr wichtig. Zu den sicherlich wichtigsten Klassen gehören die Klassen "TrackManager" sowie "Track". Die Funktion der Klassen ist vermutlich schon teilweise aus dem Klassendiagramm auszulesen.

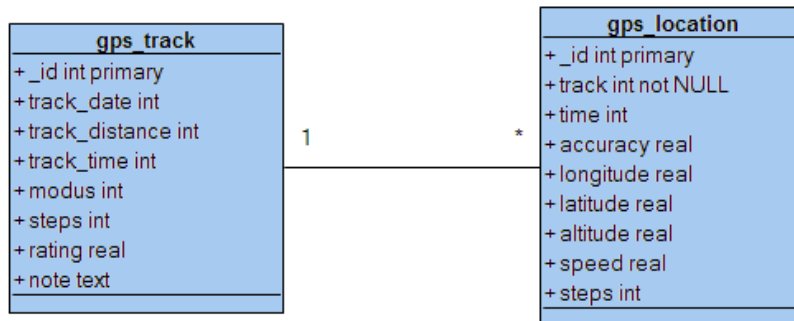


Klasse "Track"

- stellt jeweils eine Laufstecke dar
- wird vom Trackmanager während der Aufnahme mit Daten versorgt (Methode: addLocation, addSteps)
- erstellt in Echtzeit Statistiken (Klasse "Statistics"), die dann auch grafisch angezeigt werden können (Klasse "GraphLiveView")

Klasse "TrackManager"

- verwaltet alles was mit Tracks zu tun hat (erstellen von Track(Live), auslesen aus der Datenbank)
- beinhaltet LocationLister (versorgt Track mit GPS-Daten); SensorEventListener (erkennt Schritte aus den Sensordaten, und sendet Schritte an den Track)
- enthält statische Methoden zum generieren von Tracks aus der Datenbank



Wie aus dem ER-Diagramm deutlich ist, besteht die Datenbank aus nur zwei Tabellen. Die Tabelle "gps_track" enthält die wichtigsten Daten welche schließlich in den Statistiken angezeigt werden. Die Tabelle "gps_location" beinhaltet alle Locations die je bei einem Tracking aufgenommen werden. Dies ist notwendig, da auf der Google-Map der genaue Streckenverlauf aufgezeichnet wird. Außerdem sind die Locations essentiell um einen Track erneut zu generieren. Dies geschieht, indem die Locations genau wie bei der Live-Aufnahme mithilfe der "addLocation"-Methode abgespielt werden. Somit unterscheidet sich ein generierter Track nicht von einem Live-aufgenommenen.

6 Probleme & Schwierigkeiten

6.1 GPS

- die Locations, welche das GPS liefert sind im besten Fall im Radius von 5 m genau
- zu viele Locations auf kleinem Raum liefern genauso schlechte Ergebnisse, wie zu wenig Locations
- wenn der Standort nicht vor Beginn einer Aufnahme durch GPS festgelegt wurde, dauert es bis zu 2 min (Samsung Galaxy S plus) bis eine erste Locations gesendet wird

6.2 Sensoren

- Schritte müssen aus den Erschütterungen (d.h. Beschleunigungssensoren) entlang der Y-Achse des Telefons ausgelesen werden. Die ist bei starken Erschütterungen - z.B. beim Joggen oder Rennen - relativ gut möglich, da die Hochpunkte der Y-Achsen-Beschleunigungssensoren relativ eindeutige Sinus-Wellenformen liefert. Doch bei normalem Laufen sind keine klaren Hochpunkte mehr zu erkennen, es ergibt sich eine ziemlich chaotische Funktionskurve. Dadurch ergibt sich eine starke Ungenauigkeit der Schritte beim Laufen.
- wird das Telefon recht locker in der Tasche gehalten ergibt sich ebenfalls mehr Interferenzen und dadurch Ungenauigkeiten
- wird das Telefon nicht aufrecht gehalten verschieben sich die Y und Z-Achse, was wiederum zu Ungenauigkeiten führt.

7 Weiterentwicklung

Aufgrund der unserer Meinung nach ansprechenden Entwicklung der App haben wir uns schon vor einiger Zeit entschieden, die App auch nach Ablauf des Projektes weiter zu entwickeln. (Die abgelieferte Version entspricht der Version 1.0) Wir werden die Entwicklung genau wie auch schon in unserem Projekt weiterhin mithilfe der Versionsverwaltung "git" und öffentlich zugänglich auf github.com entwickeln.

Hier sind einige der Punkte die wir für künftige Versionen geplant haben:

- integrieren der App in "Google Play"
- kostenlose Version (evtl. mit Werbung) + evtl. kostenpflichtige Pro-Version (zusätzliche Features, z.B. Übertragung der Statistiken auf Computer, mehr Modi auswählbar, mehr Einstellungsmöglichkeiten)
- Überarbeitung aller Icons (verbleibende nicht konforme Icons an die Android Richtlinien anpassen)
- MapLiveView - d.h. Anzeige des aktuellen Tracks auch während der Aufnahme (RunningActivity)

8 Statistiken

Aus eigener Interesse haben wir uns um ein paar statistischen Daten zu der App gekümmert.

Dabei wurden alle Statistiken mithilfe des Eclipse-Plugins „Metrics“ erfasst.

8.1 Lines of Codes

- *insgesamt*: 3088
- *src*: 1870
 - 159
 - track: 493
 - activity: 824
 - util: 394
- *gen*: 248
- *res*: 970
 - layout: 682
 - values: 153
 - values-en: 89
 - menu: 46

8.2 Sonstiges

- 6 Pakete
- 34 Klassen
- 121 Methoden
- 110 Attribute
- 49 Icons

9 Mögliche Verbesserungen

Da die App wie schon erwähnt noch weiterentwickelt werden wird, gibt es noch einige Stellen, an denen eine Verbesserungen möglich sind.

Einige dieser Verbesserungen sind unten aufgeführt.

- ungenauer Schrittzähler (vor allem beim Laufen), ist aber nicht ohne massiven Aufwand verbesserbar
- ungenaue erste Location
- mehr Informationen (Icons) im MapView z.B. höchster Punkt der Strecke, maximale Geschwindigkeit etc.
- bessere Performanz z.B. im MapView