

Vorstellung der Thesis:

GraphHopper-Routing mit Maut-Erweiterung

Herr Robert Klemm

M.Sc. Geodatenerfassung und -visualisierung

Robert.Klemm1988@gmail.com

Passau, 23.03.2017



Gliederung

1. Einleitung
2. Problemstellung
3. Umsetzung
4. Ergebnis
5. Zusammenfassung

Motivation

- Fortführung der Methoden und Ideen aus der Bachelor-Arbeit¹
- Neue Debatte der Einführung der Infrastrukturabgabe²
- Entwicklung des Tools zur automatischen Berechnung der Mauttarife und -routen mit Hilfe von OSM- und Behörden-Daten
- Entwicklung einer Android-Applikation (offline) zur Routenberechnung der LKW-Maut in Deutschland



<http://www.giga.de/wp-content/uploads/2014/04/android-offline-navigation-rcm992x0.jpg>

¹ Fossgis 2015 = Bachelor-Präsentation in Münster (<https://youtu.be/Gcr3sRPFTfQ>)

² Infrastrukturabgabe = Maut von Nutzfahrzeugen ab 7,5 Tonnen

Rückblick

- seit 2005 Mauterfassung durch die Firma Toll Collect GmbH
auf Autobahnen für KFZ ab 12 Tonnen
- seit 2012 Mauterfassung auf ausgewählten Bundesstraßen³
- seit 2015 Mauterfassung ab 7,5 Tonnen⁴
- gepl. Juli 2018 Mauterfassung ab 7,5 Tonnen auf allen
Bundesstraßen
- Zukunftsidee Mauterfassung ab 3,5 Tonnen und Busse



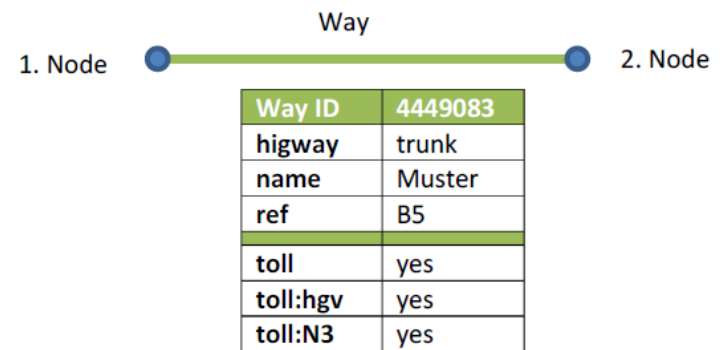
Satellitentechnologie Deutschland
(Toll Collect 2017)

³ ausgewählten Bundesstraßen = Bundesstraßen, die den Mautkriterien entsprechen

⁴ 7,5 Tonnen = ein zulässiges Gesamtgewicht, das mindestens 7,5t beträgt (bsp. Busse sind ausgenommen)

Rückblick

- Kartierungsvorschrift als „Tagging-Schema“
 - beschreibt die OSM-Attribute für Node- oder Way-Instanzen
 - dient als Richtlinie in OSM
 - Geodaten lassen sich nach dem Schema auswerten oder analysieren



abstrakte Darstellung mit den Instanzen vom Node und Way und des Maut-Tagging-Schemas

toll:hgv = heavy goods vehicle

toll:N3 = Fahrzeugklasse nach der Wirtschaftskommission für Europa der Vereinten Nationen (UNECE)

Die drei Grundprobleme

- I. keine freie Routing-API, die schnell anpassbar und individuell nach LKW-Profilen einstellbar ist
 - Auswählen nach LKW-Profilen⁵
- II. Nutzen von OSM-Daten
 - Besseres Aufbereiten der Verkehrsdaten
 - Maut-Strecken nach Konsistenz prüfen
- III. keine LKW-Maut-Routing-API auf der Grundlage von OSM-Daten
 - Anzeigen der Mautpflicht in den Straßendaten
 - Berechnung der Maut anhand der Mautparameter
 - Routen nach der kostengünstigsten Maut-Strecke
 - Auf mobilen Geräten -> offline -Modus

⁵ Gewichtung der Route nach Maut-Kosten in den vordefinierten Profilen von GraphHopper-Routing Engine

Zielsetzung

- Erstellung eines mautbezogenen Berechnungsalgorithmus der jederzeit einstellbar ist (Mautparameter und Routingprofile)
- (offline) ausführbar auf mobilen Endgeräten
- Berechnung der kostengünstigsten und schnellsten Route
- Aktualisierungsmöglichkeiten über die Applikation

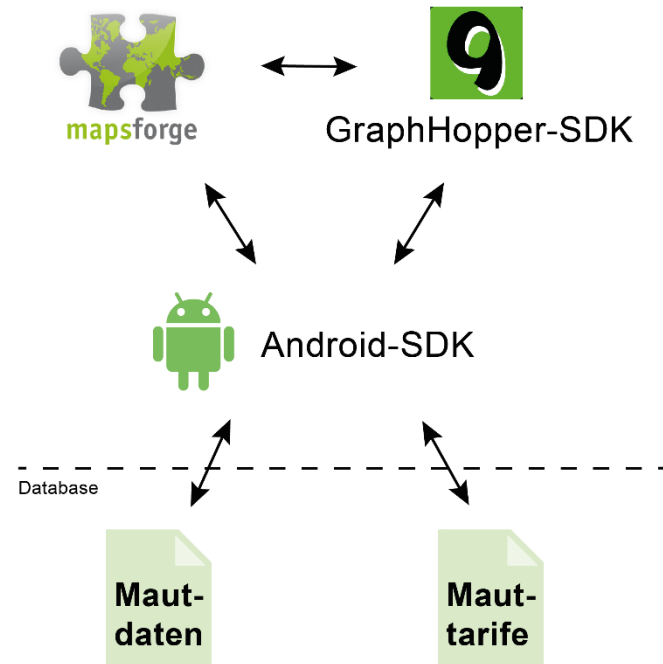


abstrakte Darstellung der verwendeten Quellen

Angewandte Technologien

- Kartendarstellung
 - Mapsforge⁶
- Routing-Framework
 - GraphHopper-SDK⁶
 - GraphHopper-Repository
- Programmiersprache
 - Java
- Datengrundlage
 - Aufbereitete OSM-Daten
 - Mauttarif-Daten

User Interface /
Application Client



⁶ In Anlehnung der GraphHopper Demo App im GraphHopper-Repository

Workflow



- Aufbereitung und Analyse der Verkehrsdaten
- Transformierung und Vollständigkeitsüberprüfung der Mautdaten und Mauttarife
- Anpassung und Erweiterung des GraphHopper-Repositorys
- Erstellung eines Routinggraphen und LKW-Routingprofils
- Anpassung und Erweiterung des GraphHopper-Android-Repositorys
- Funktionen: Offline-Routing, LKW-Maut-Berechnung, Mautdaten-Update
- Android-App auf dem Client installieren und testen

Mauttarife / -daten aufbereiten und analysieren

- Manuelle Überprüfung der Mauttarife
- Automatische Erstellung einer CSV-Datei aus den Mauttarifinformationen
- Automatische Aufbereitung, Überprüfung und Erstellung von OSM-Mautabschnitten
- Erstellung eines Parsers als Jar-File

Mautsätze pro Kilometer ab 1. Oktober 2015				
Kategorie	Mautsatz-Anteil (in Cent) Kosten für Luftverschmutzung	Achszahl**	Mautsatz-Anteil (in Cent) Kosten für Infrastruktur	Mautsatz (in Cent)
A	0	2	8,1	8,1
		3	11,3	11,3
		4	11,7	11,7
		ab 5	13,5	13,5
B	2,1	2	8,1	10,2
		3	11,3	13,4
		4	11,7	13,8
		ab 5	13,5	15,6

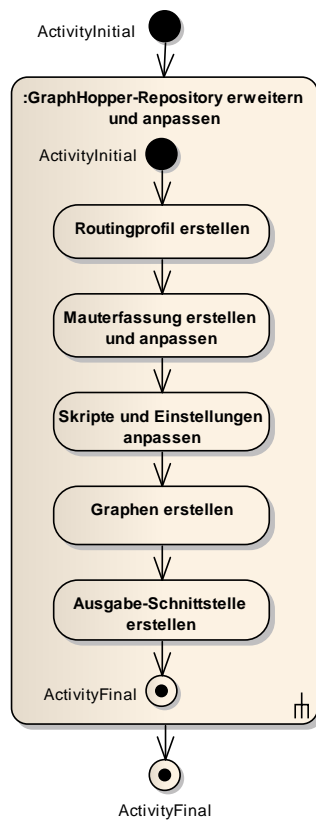
Tarifübersicht auf der Webseite von Toll Collect GmbH

```

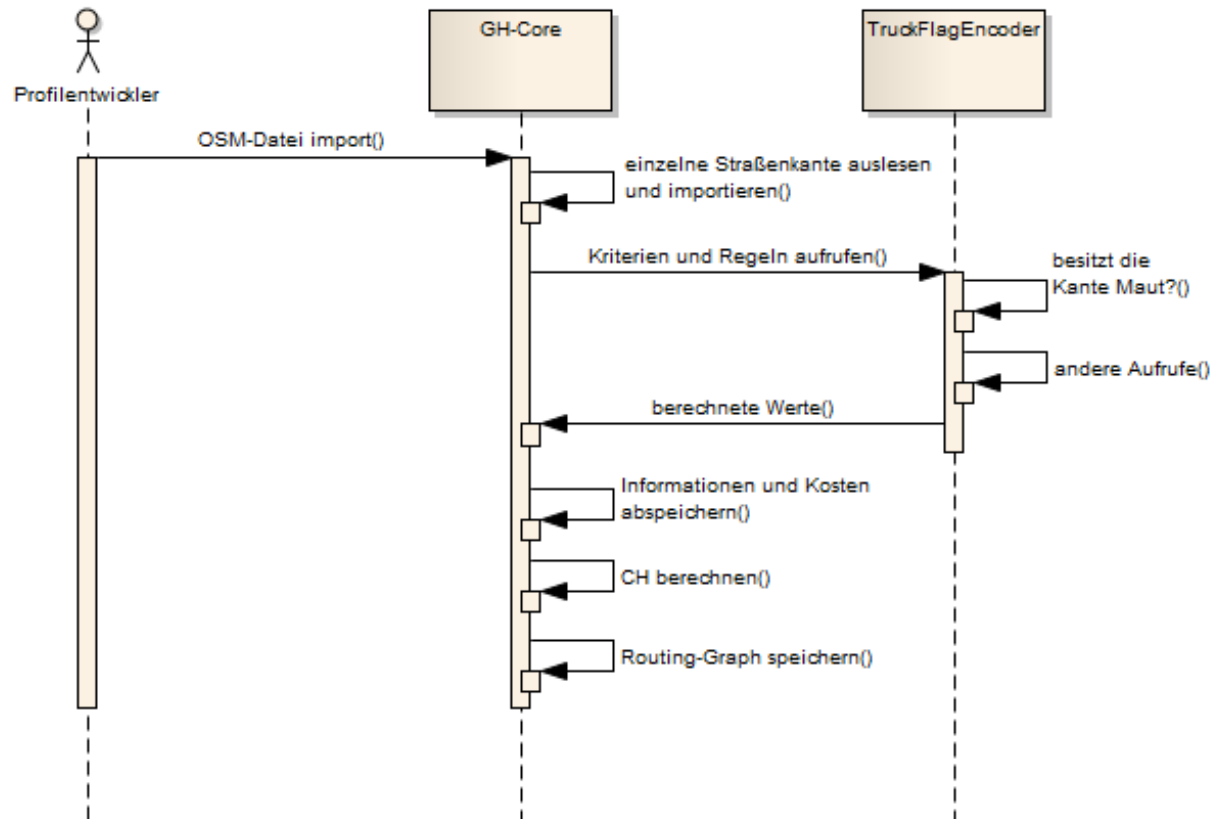
1 0.1
2 2016-8-22
3 Toll Collect
4 MautKategorie luftverschmutzung MautBetrag1 MautBetrag2 MautBetrag3 MautBetrag4 MautBetrag5
5 A 0 0 8.1 11.3 11.7 13.5
6 B 2.1 0 10.2 13.4 13.8 15.6
7 C 3.2 0 11.3 14.5 14.9 16.7
8 D 6.3 0 14.4 17.6 18.0 19.8
9 E 7.3 0 15.4 18.6 19.0 20.8
10 F 8.3 0 16.4 19.6 20.0 21.8
    
```

Transformierung der Tarife in einer CSV-Datei

GraphHopper-Repository erweitern und anpassen



Arbeitsschritte in der GraphHopper-Bibliothek



Abstraktes UML-Diagramm: Route mit GraphHopper berechnen

Android-Projekt erweitern und anpassen

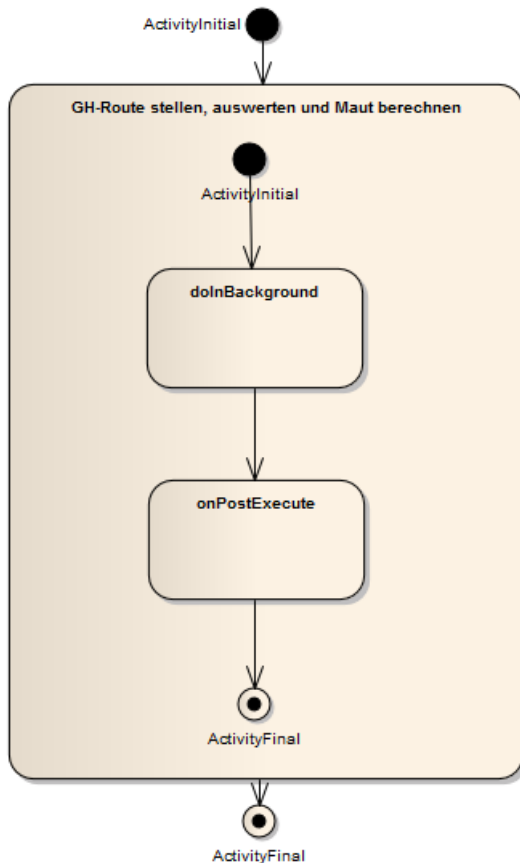
- Einbindung externer JAR-Bibliotheken⁷
- Transformierung und Auslesen der Maut-JSON-Datei
- Erstellung und Anpassen der Anwendungsfälle
 - Verkehrsdaten auswählen
 - Mautdaten oder -tarife aktualisieren
 - Mauttarifparameter eingeben
 - Mautroute , -tarif berechnen und anzeigen
- Erstellung eines Android-Package-Files

```
{
  "MautKategorieList": [
    {
      "mautKategorieET": "A",
      "mautSatzList": [
        {
          "betrag": 0,
          "achse": 1
        },
        {
          "betrag": 8.1,
          "achse": 2
        },
        {
          "betrag": 11.3,
          "achse": 3
        },
        {
          "betrag": 11.7,
          "achse": 4
        },
        {
          "betrag": 13.5,
          "achse": 5
        }
      ]
    },
    {
      "luftverschmutzung": 0
    }
  ],
  [
    ...
  ]
  "version": {
    "date": "2016-8-22",
    "id": 0.1,
    "operator": "Toll Collect"
  }
}
```

JSON-Struktur der Mauttarife

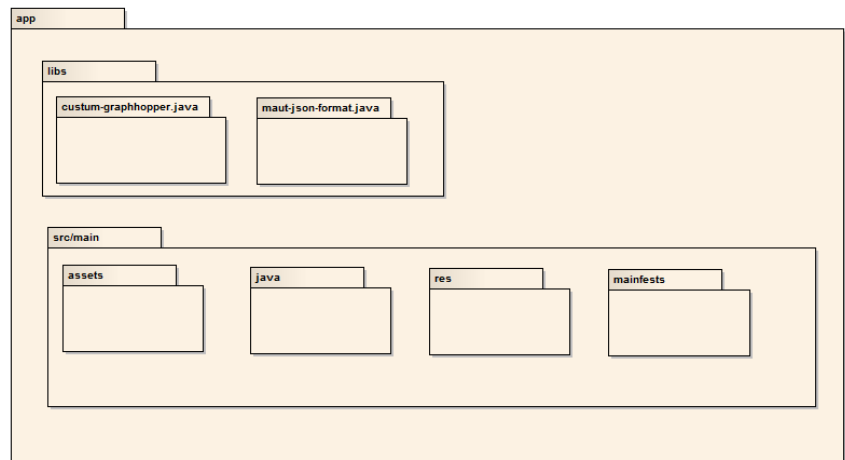
⁷ die angepasste GraphHopper-JAR- und eigenentwickelte JAR-Bibliothek

Android-Projekt erweitern und anpassen



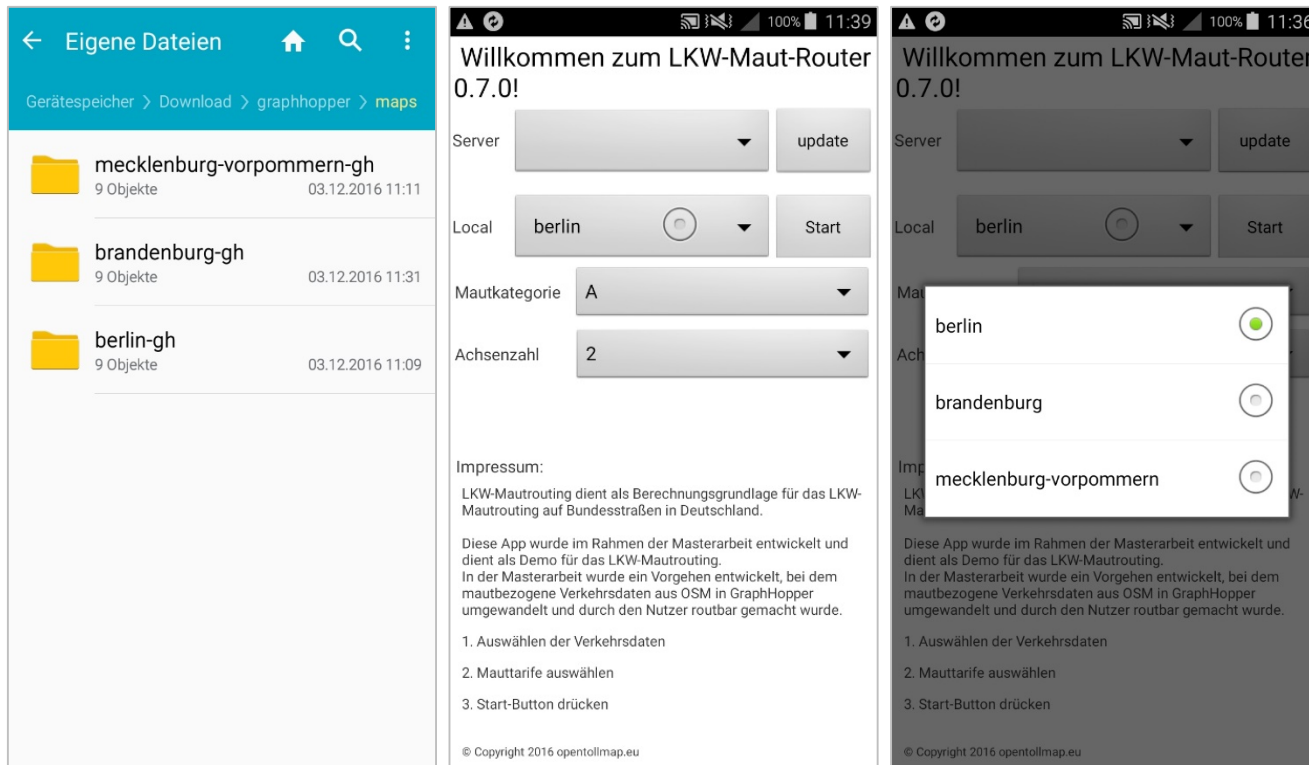
abstrakte Darstellung von GH-Route stellen,
auswerten und Maut berechnen

- Anpassung der Routenanfrage GraphHopper (doInBackground)
- Route auswerten und Mautberechnen (onPostExecute)
- Ergebnisse anzeigen



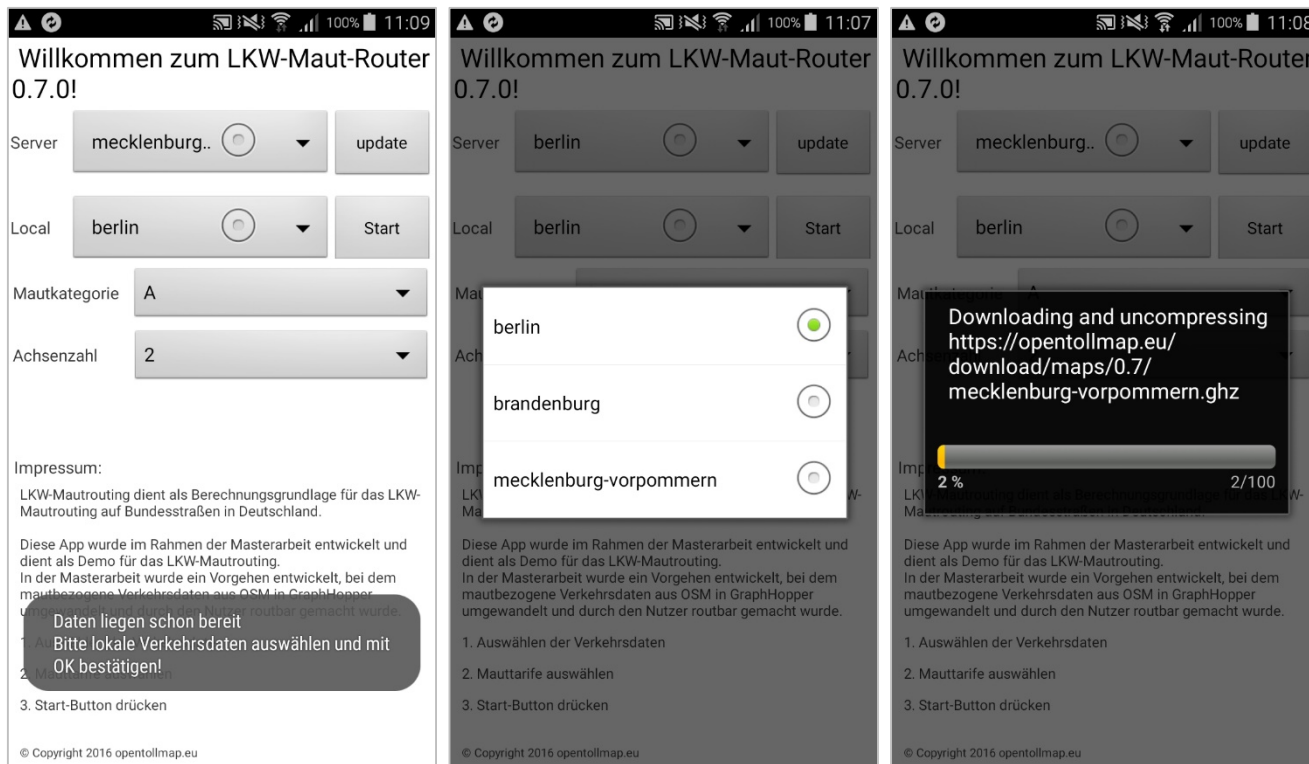
Benötigte Pakete der Android-Applikation

Anwendungstest



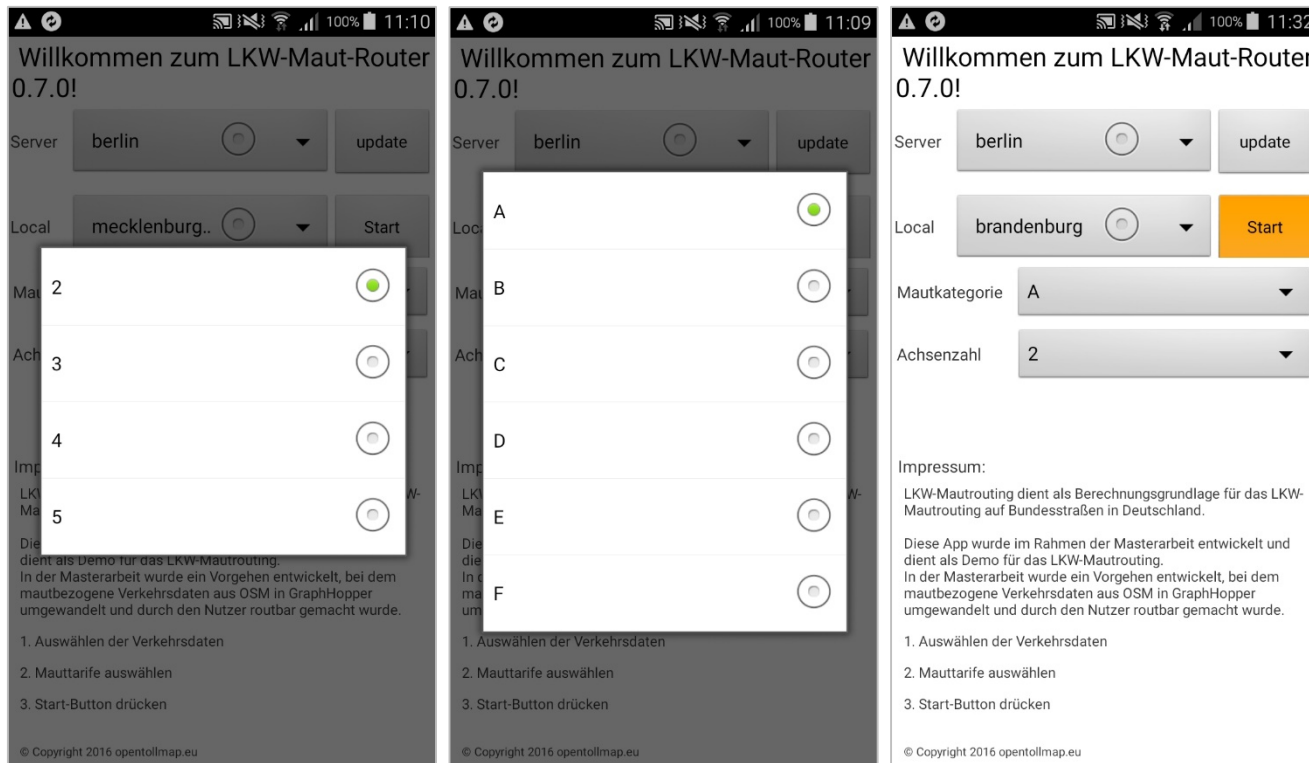
Lokale Daten auswählen

Anwendungstest



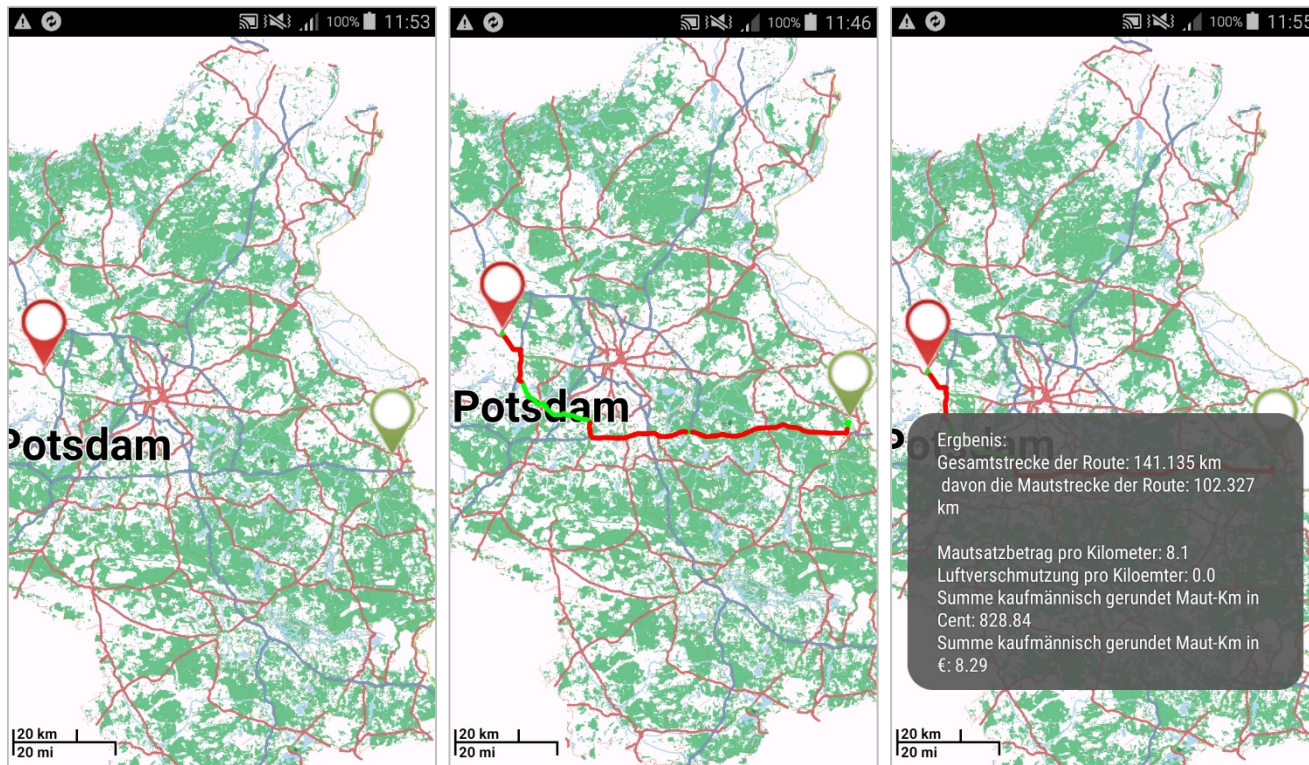
Server-Daten aktualisieren

Anwendungstest



Eingabe bestätigen und Routing starten

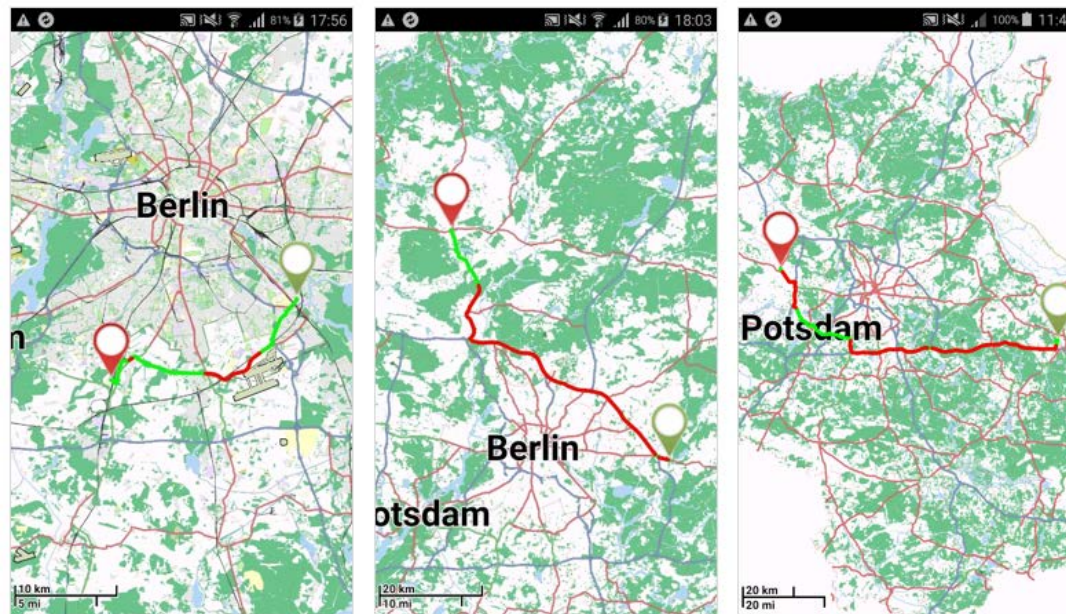
Anwendungstest



Routen-, Mautberechnung und Infoanzeige

Ergebnisbeschreibung

- Durchführung des Anwendungstests auf dem mobilen Client
- Erzeugen und Vergleichen von drei Beispielrouten in Berlin-Brandenburg
- Überprüfung der berechneten Tariflängen und der Kartierregeln in OSM



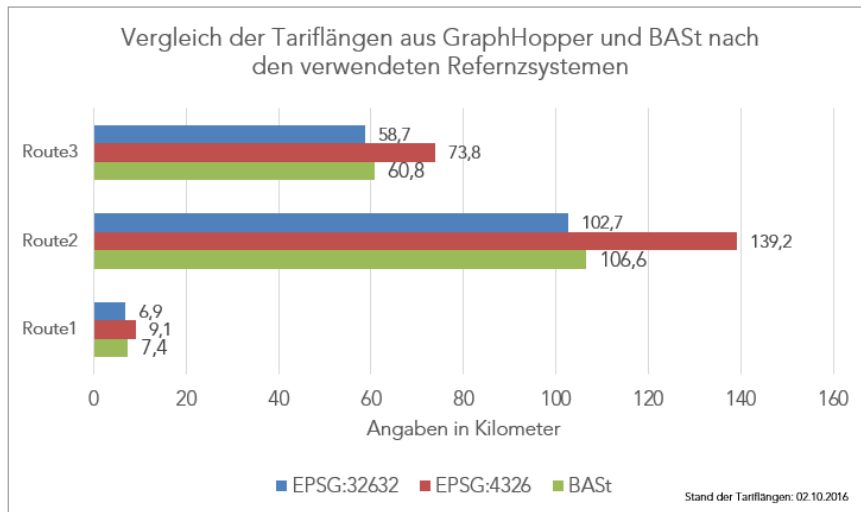
Drei Vergleichsrouten in Berlin und Brandenburg

Ergebnisbeschreibung

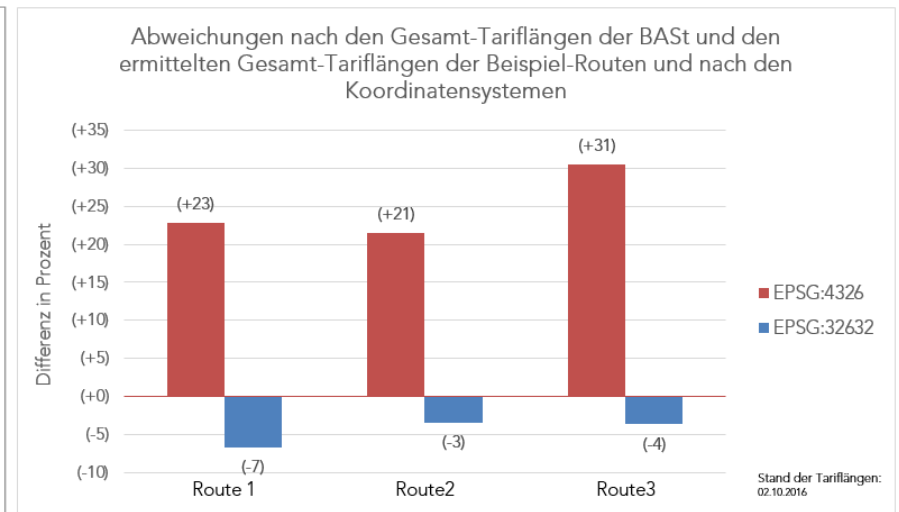
- Abweichungen der ermittelten Tariflängen
- falsche Verwendung des Zielkoordinatensystems
- unterschiedliche Verzerrungen durch die Verwendung von Koordinatensystemen

	EPSG:4326	EPSG:32632
Route1	+23 %	-7 %
Route2	+21 %	-3 %
Route3	+31 %	-4 %
Mittelwert	+25 %	-5 %

Übersicht der Abweichungen der Routen mit dem Mittelwert



Überblick der ermittelten Tariflängen



Übersicht der Abweichungen der ermittelten Tariflängen zu der BAST

Vorteile des Verfahrens & Änderungsvorschläge

- Überprüfung und Planung der Maut-Routen durch den Nutzer
 - hoher Qualitätsanspruch für die OSM-Nutzer
 - kostenfrei nutzbar und erweiterbar mit anderen Daten
 - auf andere Mautsysteme übertragbar
- Veröffentlichung der Mauttarife als CSV-Datei
 - Anpassung und Erweiterung der Routingprofile
 - Weiterführung der Idee der kostengünstigen Mautroutenberechnung
 - Weiterentwicklung und Optimierung der offlinefähigen App

Vielen Dank für Ihre Aufmerksamkeit!

