

Entwicklung der Open-Street-Applikation für Windows Phone 7

Florian Rapp

Universität Ulm, Abt. DBIS
florian.rapp@uni-ulm.de

Zusammenfassung Diese Arbeit beschäftigt sich mit der Entwicklung einer Routing- und Navigationsapplikation, namentlich der *Open Street App*, für das mobile Betriebssystem Windows Phone 7 (WP7). Es wird aufgezeigt welche Schritte von der Idee dieser Applikation bis zur Fertigstellung umgesetzt, und welche Probleme und Herausforderungen überwunden werden mussten. Die Arbeit soll einen grundlegenden Eindruck der praktischen Entwicklung für WP7 vermitteln.

1 Einleitung

1.1 Rahmen der Arbeit

Im Herbst 2010 veröffentlichte die Firma Microsoft ihr eigenes Betriebssystem für Smartphones, Windows Phone 7 (WP7). Basierend auf WinCE 7, allerdings mit einem komplett neuen und modernen User Interface, sollte ein flexibles, innovatives System auf den Markt gebracht werden. Um mit den Branchenführern *Apple iOS* und *Android* konkurrieren zu können, wurde insbesondere die einfache und schnelle Entwicklung von Applikationen für das neue System von Microsoft angepriesen. Im Rahmen des Seminars *Entwicklung für Windows Phone 7* der Abteilung *Datenbank- und Informationssysteme* der Universität Ulm, war es unser Ziel zu prüfen wie effektiv sich Programme für WP7 zum aktuellen Stand entwickeln lassen.

1.2 Idee der Open Street App

Im Blickpunkt unserer Entwicklung stand nicht nur das reine Erstellen einer Applikation, sondern gerade auch die Verwendung der von Microsoft bereitgestellten Controls und Bibliotheken. Wir wollten herausfinden ob diese sich effektiv und schnell integrieren lassen. Zum Zeitpunkt des Seminarbeginns war für WP7 keine zufriedenstellendes Third-Party-Routing verfügbar. WP7 liefert von sich aus eine Karte/Routing-Applikation. Unsere Idee war es nun eine eigene Software zu schreiben, welche den Anforderungen der mitgelieferten *Karten-App* genügt und falls möglich übertrifft, unter Verwendung der erwähnten Controls.

1.3 Aufbau dieser Arbeit

Im zweiten Kapitel wird erklärt mit welchen Quellen und Umsetzungsmitteln gearbeitet wurde um Kartendaten und zugehörige Informationen zu erlangen und darzustellen. Das folgende Kapitel beschäftigt sich mit der Gestaltung der Applikation im WP7 typischen UI-Stil. Hierbei wurde besonderes Augenmerk gelegt auf ein konsistentes Look-and-Feel im Vergleich zu den native Applikationen für WP7. Kapitel vier berichtet exemplarisch von den Herausforderungen und Problemen die bei der Entwicklung aufgekommen sind. Kapitel fünf erklärt wie einmal entwickelte Applikationen ihren Weg auf den Markt finden. Hier wird auf den Marketplace und die Zukunft der Open Street App eingegangen. Im letzten Kapitel wollen wir einen Ausblick über die möglichen Entwicklungen von WP7 geben sowie ein Fazit zu der Entwicklung in diesem frühen Stadium ziehen.[1] [6] [4] [8] [9] [2] [3] [5]

2 Open Street Map und Konsorten

2.1 Umsetzung des Map-Controls

Der Kernaspekt der Open Street App ist die Kartenansicht. Die Karten bilden sowohl die Grundlage für alle Operationen im Routing- und Navigationsbereich, als auch den Dateninput. Es war also unserer erste Aufgabe passende Karten auf dem Gerät darstellen zu können. Dies wird von Microsoft durch das MapControl und durch Verwendung einer MultiTileSource ermöglicht. Die MultiTileSource ist eine Komponente die eine Quelle für Tiles erhält. Ein Tile bezeichnet einen bestimmten Kartenausschnitt, definiert durch Zoomstufe sowie horizontale und vertikale Start- bzw. Endkoordinaten. Die MultiTileSource setzt bei uns dann immer aus vier Tiles unsere aktuelle Karte zusammen die dann im MapControl angezeigt wird. Beim Scrollen oder Zoomen der Karte werden die neuen Tiles vom Quell-Server angefragt und das MapControl entsprechend aktualisiert.

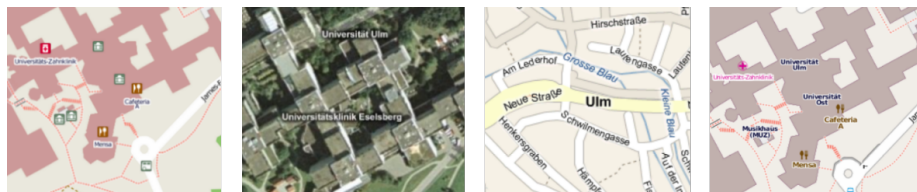


Abbildung 1. Verschiedene Tiles: Cludemade, VE Areal, VE Maps, Open-StreetMap

2.2 Verwendung von Services

Im Vergleich zu Applikationen die bei Start bereits vollständig auf dem Gerät vorhanden sind, zeichnet sich eine Routing-App vor allem durch Informationen die jederzeit abgefragt werden können aus. Dem Benutzer müssen zur jeder Zeit und an jedem Ort an dem eine Mobilfunkverbindung aufgebaut werden kann, alle Dienste die man zum Navigieren auf Grundlage einer Karte braucht, bereitgestellt werden. Fast alle diese Daten werden über bestimmte Services angefragt und dann entsprechend auf dem Handy präsentiert. Open Street App verwendet mehrere dieser Services. Grundlegend lassen sich die Anfragen in verschiedene Gruppen einteilen:

Geocoding Der Begriff Geocoding oder auch Geotagging bezeichnet die Anreicherung eines (Address)Datensatzes um raumbezogene Information. In unserem Fall verstehen wir darunter die Anreicherung einer Adresse oder eines bestimmten Ortes um eine Koordinate. Dem Benutzer wird über eine Suchmaske die Möglichkeit geboten einen Suchbegriff einzugeben. Dieser Suchbegriff wird URL-Encoded und dann Teil einer Anfrage an einen REST orientierten Webservice, wie er von Firmen wie Yahoo, Google und Microsoft bereitgestellt wird. Open Street App arbeitet in seiner ersten Version mit mehreren Verschiedenen Webservices um die Suchergebnisse zu optimieren.

Reverse Geocoding Unter Reverse Geocoding versteht man die Rücktransformierung einer Koordinate in eine lesbare Adresse. Dies ist eine Grundfunktionalität die von allen Applikationen die zur Orientierung dienen verwendet wird. Über ein GPS Signal liegen der App die aktuellen Koordianten des Benutzers vor. Abbildung 2 Zeigt einen beispielhaften Aufruf des Yahoo Placefinder Service. Es werden die Koordinaten des Gebäudes O28 der Universität Ulm übergeben.

Point of Interest

Routing

FLORIAN RAPP

```
1 Uri adress = new Uri("http://where.yahooapis.com/geocode?q=" +  
    48.421997 + "," + 9.956117 + "&gflags=R&locale=" + System.  
    Globalization.CultureInfo.CurrentCulture.Name + "&appid  
    =12345");  
2  
3 System.Net.WebClient wc = new System.Net.WebClient();  
4 wc.DownloadStringAsync(adress);
```

Abbildung 2. Reverse Geodocing Anfrage für die Universität Ulm

Das Ergebnis eines solchen Aufrufes kann in verschiedenen Formaten angefordert werden. Üblich sind das JSON (JavaScript Object Notation) und das XML Format. Abbildung 3 zeigt einen Ausschnitt des responses zu der Anfrage aus Abbildung 2 im XML Format. Die Koordinaten wurden in eine Adresse transformiert. Je nach Anforderung können diese responses, über entsprechende Optionen der Anfrage, mehr oder weniger Details enthalten.

```
1 <Result>  
2 <latitude>48.421997</latitude>  
3 <longitude>9.956117</longitude>  
4 <line1>James-Franck-Ring</line1>  
5 <line2>89081 Ulm</line2>  
6 <line4>Germany</line4>  
7 <statecode>BW</statecode>  
8 </Result>
```

Abbildung 3. Response für die Geolocation Anfrage(Ausschnitt)

3 Umsetzen der WP7 Designkonzepte

3.1 WP7 Look and Feel

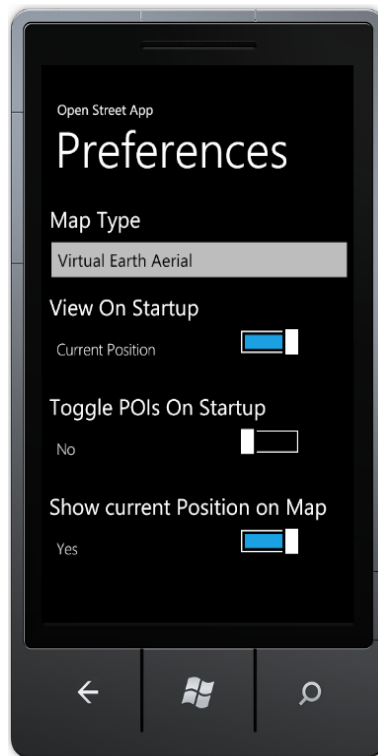


Abbildung 4. WP7 typische UI controls

4 Herausforderungen

4.1 Probleme der gegebenen Controls

Die von WP7 gegebenen Controls erlauben es sehr schnell zu Ergebnissen zu kommen.

4.2 Routing

Text zu Routing

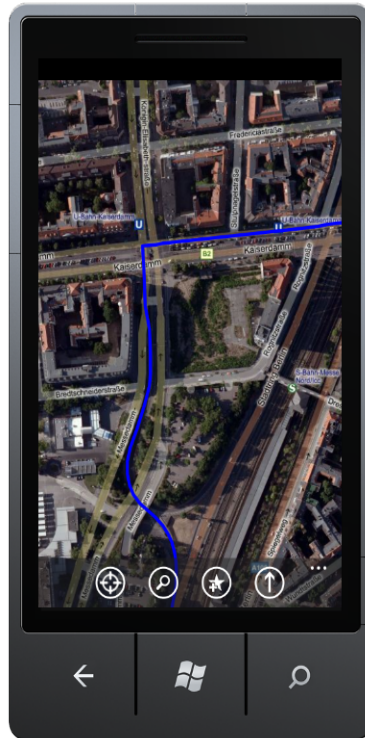


Abbildung 5. Vereinfachte Route

5 Marketplace

5.1 Struktur

5.2 Entwicklung

5.3 Zukunft der Open Street App

Die Open Street App wurde im Sommer 2011 fertiggestellt und steht nun vor der Veröffentlichung im Marketplace. Grundsätzlich bestand für uns die Auswahl aus drei Konzepten für unserer Applikation:

- kostenpflichtige App für die im Vorraus bezahlt werden muss
- kostenfreie App mit (eventuell drittanbieter-) Werbung
- werbe- und kostenfreie Gratisapp

Da wir die Ergebnisse unserer Arbeit allen WP7 Nutzern zur Verfügung stellen wollten war die Variante der kostenpflichtigen Applikation nicht relevant für unsere Veröffentlichung. Die Möglichkeit unser Programm durch Werbung anzureichern war lange Zeit das Konzept unserer Wahl. Applikationen die vom Kaufsystem auf eine Werbefinanzierung umgestiegen sind haben in den letzten Monaten einen großen Anteilszuwachs zu verzeichnen. Gerade der Spielebereich hat sich durch den hohen Wiederspielwert als profitabel für Werbeeinblendungen gezeigt. [7] Schlussendlich haben wir uns dafür entschieden die App frei von Kosten und Werbung in den Store zu bringen. Eine dauerhafte Werbeeinblendung ist bei der Bedienung einer Karte störend und würde die Usability und damit die Zufriedenheit der Benutzer einschränken.

6 Ausblick und Fazit

6.1 Weiterentwicklung der Plattform

6.2 Chancen für die Zukunft

6.3 Fazit

Entwickeln toll. Schnelle Ergebnisse.

Literatur

1. CloudMade. Geocoding and geosearch. <http://developers.cloudmade.com/wiki/geocoding-http-api/Documentation>. Zugriffen am 05.04.2011.
2. OpenStreetApp Community. Osm api v0.6. http://wiki.openstreetmap.org/wiki/API_v0.6. Zugriffen am 05.04.2011.
3. Yahoo! Inc. Yahoo! placefinder guide. <http://developer.yahoo.com/geo/placefinder/guide/>. Zugriffen am 05.04.2011.
4. Jan Kluczniok. Wp7: Alle smartphones mit microsofts neuem handy-os im Überblick. <http://www.netzwelt.de/news/84335-wp7-smartphones-microsofts-neuem-handy-os-ueberblick.html>. Zugriffen am 05.04.2011.
5. Microsoft. Bing maps api. <http://msdn.microsoft.com/en-us/library/dd877180.aspx>. Zugriffen am 05.04.2011.
6. Microsoft. *Programming Windows Phone 7*. Microsoft Press, 2010.
7. Elbert Perez. Ads and wp7 games. <http://www.occasionalgamer.com/2011/02/09/ads-and-wp7-games/>. Zugriffen am 09.02.2011.
8. Puja Pramudya. *Silverlight for Windows Phone - learn & practice*. Microsoft Innovation Center, 2010.
9. Jacob Schulman. Windows phone 7 marketplace hits 10,000 apps, microsoft wp7 updates still way outnumbered. <http://www.engadget.com/2011/03/11/windows-phone-7-marketplace-hits-10-000-apps-wp7-updates-still/>. Zugriffen am 11.03.2011.