

Think Mobile – Effiziente Entwicklungsstrategien im GIS

Kai Hoffmann, BTC AG



Mobile Endgeräte – Was ist zu beachten?



In dieser Betrachtung

- **Smartphone**
- **Tablet**

Unterschiedliche Hardwareausstattung

- **Bildschirmgröße**
- **Bildschirmauflösung**
- **Ressourcen (z.B. CPU, GPU, Speicher)**
- **Technische Ausstattung, z.B. Bluetooth, NFC, Wifi, GPS**

Mobile Endgeräte – Was ist zu beachten?



In dieser Betrachtung

- Smartphone
- Tablet

Unterschiedliche Betriebssysteme

- iOS
- Android
- Windows Phone
- WebOS
- u.v.m.

Mobile Endgeräte – Was ist zu beachten?



Für Web Apps besteht eine weitere Abhängigkeit...

Der verwendete Browser:

- **Chrome (iOS, Android)**
- **Safari (iOS)**
- **Android Browser (Android)**
- **Explorer (Windows Phone)**
- **Firefox (Android, iOS -> beta)**
- **Opera Mini (iOS, Android, Windows Phone)**

Software als Web App (HTML5-Webseite)

- **Vorteile**
 - **ist im Browser aufrufbar**
 - **braucht nicht installiert werden**
 - **unabhängig vom Betriebssystem und (fast) von der Hardware**
 - **Verwendung von „Responsive Webdesign“, d.h. es kann das Layout der Web App dynamisch an die Auflösung angepasst oder eine Desktop Version der Web App zur Verfügung gestellt werden**
- **Nachteile**
 - **Berücksichtigung unterschiedlicher Browser**
 - **kann nicht im App-Store angeboten werden**
 - **nur online verfügbar**
 - **eingeschränkter Hardwarezugriff**
 - **kaum für rechenintensive Anwendungen nutzbar**

Mobile Endgeräte – Was ist zu beachten?



Software als App

- **Vorteile**
 - an das Betriebssystem optimal angepasst
 - kann über den App-Store installiert werden
 - direkter Zugriff auf die Hardware
 - unterstützte Auflösungen können definiert werden
 - bestmögliche Ausnutzung vorhandener Ressourcen (CPU, GPU, Speicher usw.)
- **Nachteile**
 - für jedes Betriebssystem ein eigener Entwicklungszweig
 - gemeinsame Nutzung von Quellcode kaum möglich, dadurch hoher Entwicklungs- und Pflegeaufwand
 - eine Desktop-Version müsste ebenfalls separat/gepflegt entwickelt werden

Welcher Weg ist der richtige?



Diese Kriterien und die Beantwortung der Fragen...

- **Was soll die Software leisten?**
- **Wer ist die Zielgruppe?**

geben einen klaren Hinweis darauf, welcher Weg der Entwicklung eingeschlagen werden sollte.

Aber...

Welcher Weg ist der richtige?



Mittlerweile gibt es eine Reihe moderner Technologien, die Entwicklung effizienter zu gestalten, d.h. für mehrere Betriebssysteme gleichzeitig zu entwickeln, und dabei die bisherigen Nachteile, wie z.B.

- getrennte Entwicklung der gleichen Software
- keine gemeinsame Nutzung des Quellcodes
- eingeschränkte Nutzung der Hardware in Web Apps
- u.v.m.

zu umgehen.

Allerdings müssen auch hier Kompromisse eingegangen werden -
in Abhängigkeit der bekannten Kriterien.

Entwicklungsunterstützende Technologien.



Hierbei handelt es sich um folgende Technologien

- QML
- XAMARIN
- PhoneGAP

Was ist QML

QML ist eine deklarative Programmiersprache, deren Syntax mit JSON ähnlich ist. Die Oberfläche und die Ausführung des Codes basiert auf der Qt-Klassenbibliothek (C++).

- **Eigenschaften**
 - deklarative Beschreibung der Oberfläche
 - Entwicklung der Business-Logik mittels Javascript und C++, die Ausführung erfolgt als C++-Code bzw. optimierter Javascript-Code
 - -Implementierung des ECMAScript Standards 5 (mit Restriktionen)
 - Zugriff auf Hardware- und Software-Schnittstellen
 - UI Look and Feel fast dem Betriebssystem entsprechend
 - performante Ausführung
 - kann im App-Store angeboten werden
 - **kostenlos**

Was ist XAMARIN

Software kann mit der Programmiersprache C# für iOS-, Android- und WindowsPhone-Apps entwickelt werden. Der Quellcode wird für alle Plattformen gemeinsam genutzt. Als Basis dient die .NET-Technologie.

- **Eigenschaften**
 - Verwendung einer Programmiersprache und eines einheitlichen Basis Frameworks (XAMARIN – Mono-Derivat)
 - Zugriff auf native C, C++, Objective-C, und Java Bibliotheken
 - Zugriff auf Hardware- und Software-Schnittstellen
 - UI Look and Feel dem Betriebssystem entsprechend
 - performante Ausführung
 - Portierung bestehender .NET- Bibliotheken
 - kann im App-Store angeboten werden
 - kostenpflichtig

Was ist PhoneGAP

PhoneGap ist ein Framework zur Erstellung hybrider Applikationen für mobile Endgeräte. Mit HTML5, CSS und Javascript erzeugte Webseiten werden in einem Container ausgeführt. Für den Anwender sieht es so aus, als würde eine App ausgeführt.

- **Eigenschaften**
 - **Verwendung von HTML5, CSS und Javascript**
 - **bereits existierende Web Apps können in eine PhoneGAP Anwendung (manuell) konvertiert werden**
 - **Zugriff auf Hardware- und Software-Schnittstellen**
 - **UI Look and Feel dem Betriebssystem entsprechend**
 - **Performance kann problematisch sein (UI, Berechnungen)**
 - **kann im App-Store angeboten werden**
 - **kostenlos**

Fazit.



Die Entscheidung, wie eine Software entwickelt werden sollte, hängt von vielen Faktoren ab.

Soll die Software auf unterschiedlichen Betriebssystemen ausführbar sein, steigt die Komplexität.

Web Apps können – je nach Anforderung – einen guten Kompromiss darstellen.

Moderne Entwicklungstechnologien können die Komplexität mindern.

Der Entwickler hat die Qual der Wahl. Die richtige Entscheidung beeinflusst, wie schnell und gut eine bestehende Software gepflegt und weiterentwickelt werden kann.

Letztendlich muss immer ein Kompromiss eingegangen werden.

Ein falsche Entscheidung am Anfang einer Entwicklung kann zu Unwirtschaftlichkeit und/oder einer recht kurzlebigen Software führen.

**Vielen Dank für Ihre
Aufmerksamkeit.**

Fragen?

