

# ZHAW Seminararbeit - Handheld



## Feuerwehr Off-Spick

Michael Ott, i10b

19. Februar 2013

# Inhaltsverzeichnis

<b>1. Einleitung</b>	<b>3</b>
1.1. Aufgabenstellung gemäss Eingabe EBS (ZHAW) . . . . .	3
1.1.1. Ausgangslage . . . . .	3
1.1.2. Ziel der Arbeit . . . . .	3
1.1.3. Aufgabenstellung . . . . .	3
1.1.4. Erwartetes Resultat . . . . .	3
1.2. Teaser . . . . .	4
1.3. Motivation . . . . .	4
<b>2. Randbedingungen / Kontext</b>	<b>5</b>
2.1. Konventionen . . . . .	5
2.2. Technische Randbedingungen . . . . .	5
2.3. Stakeholder . . . . .	5
2.4. Termine . . . . .	6
2.5. Aufwandschätzung . . . . .	6
<b>3. Umsetzung</b>	<b>7</b>
3.1. Grundidee . . . . .	7
3.2. modularisierte Darstellung . . . . .	7
3.3. Module . . . . .	8
3.3.1. Backend - Datenbank (generell) . . . . .	8
3.3.2. Backend - Authentifizierung . . . . .	8
3.3.3. Backend - Telefonverzeichnis . . . . .	9
3.3.4. Backend - Webservice . . . . .	10
3.3.5. Client - Authentifizierung . . . . .	11
3.3.6. Client - Darstellung . . . . .	12
<b>4. Projekt-Tools</b>	<b>13</b>
<b>5. Fazit / Ideen für Erweiterungen</b>	<b>14</b>
<b>A. Glossar</b>	<b>15</b>
<b>B. Literaturverzeichnis</b>	<b>16</b>
<b>C. Abbildungsverzeichnis</b>	<b>17</b>
<b>D. Quellcode</b>	<b>18</b>

# 1. Einleitung

## 1.1. Aufgabenstellung gemäss Eingabe EBS (ZHAW)

### 1.1.1. Ausgangslage

Einsatzleiter/Offiziere der Feuerwehr stehen im Einsatz immer wieder vor speziellen Situationen welche nach speziellen Lösungen verlangen. Oft ist dazu ein schneller und unkomplizierter Zugriff auf verschiedene Telefonnummern (Werkmeister, Jäger, Baugeschäft, spezielle Pikett-Nummern, usw.) nötig. Bis zum heutigen Zeitpunkt führte jeder Offizier ein kleines laminiertes Büchlein mit, welches die nötigen Telefonnummer enthält. Dieses muss bei Änderungen jeweils aufwändig nachgeführt werden.

### 1.1.2. Ziel der Arbeit

Jeder Offizier unserer Feuerwehr ist im Besitz eines Smartphones (iOS, Android). Es soll möglich sein, mittels App direkt auf die zentral verwalteten Telefonnummern zuzugreifen.

### 1.1.3. Aufgabenstellung

Einarbeitung in das Thema Handheld-Programmierung, erstellen und umsetzen eines Konzeptes in eine Handheld-App.

### 1.1.4. Erwartetes Resultat

Eine ausführliche Projekt-Dokumentation und eine Handheld-App mit folgenden Eigenschaften:

- App muss für verschiedene Systeme verfügbar sein (min. IOS, Android)
- Telefonnummern müssen schnell und übersichtlich in Kategorien geordnet abrufbar sein
- Die Verwaltung der Nummern muss zentral auf einem Webserver erfolgen. Der Zugriff erfolgt via http(s) Webservice
- Die Telefonnummern müssen von Fremdzugriff geschützt werden (Zugriff innerhalb der App erfolgt mit hinterlegter Benutzer/Passwort/Datenquelle)
- Zugriff auf die Nummern muss auch ohne Internet möglich sein
- Die Aktualisierung der Nummern muss automatisch erfolgen (z.B. bei App-Start und vorhandenem Internetzugriff)

## 1.2. Teaser

Unter dem Link <http://www.feuerwehr-wiesendangen.ch/offapp/teaser> befindet sich eine kurze Projektübersicht mit eingeschränktem Prototyp der angestrebten Lösung.

## 1.3. Motivation

Mobile Geräte begleiten uns immer häufiger in allen Lebenslagen. Der Funktionsumfang, die Leistung und damit die Möglichkeiten im Bereich Handheld steigen mit jeder neuen Geräte-Generation.

Diese Seminararbeit erlaubt einen stufengerechten Einstieg in diese Technologie, um zukünftig in weiteren Projekten die weiteren Möglichkeiten eines Handheld nutzen zu können.

## 2. Randbedingungen / Kontext

Die Applikation soll möglichst ohne Zusatzkosten auf dem bestehenden Webserver des FZWB (Feuerwehr-Zweckverband Wiesendangen-Bertschikon) betrieben werden.

### 2.1. Konventionen

Bereich	Bedingung
Projektdokumentation	12-15 Seiten (gem. Vorgabe ZHAW)
Versionsmanagement	Quellcode muss durch ein geeignetes System verwaltet sein

### 2.2. Technische Randbedingungen

*Backend*

Funktion	System/Details	Version	Bemerkung
Webserver	Apache	2.2.3	Aktualisierung jederzeit durch Hoster möglich
Datenbank	MySQL	5.0.77	Aktualisierung jederzeit durch Hoster möglich
Zugriff	via http(s)-Webservice		nur mittels gültiger Login-Daten
Kontakte	Verwaltung		manuell via existierendes phpMyAdmin

*Clients (muss HTML5 LocalStorage unterstützen)*

System	Name	Version	Bemerkung
Apple	iOS	ab 4.0	
Google	Android	ab 3.0	
Microsoft	Windows Mobile	ab 8.0	optional

Die Clients greifen via mobilem Internet auf den Server resp. das Backend zu

### 2.3. Stakeholder

Person	Funktion	E-Mail
Michael Ott	Student (i10b) / Offizier FZWB	ottmic02@students.zhaw.ch
Christian Vils	Experte	xvil@zhaw.ch
Marco Scheuring	Kommandant FZWB	marco.scheuring@ feuerwehr-wiesendangen.ch

## 2.4. Termine

Datum	Schritt	Inhalt
19.09.2012	Projekt-Start	grobe Zieldefinition
03.10.2012	Abgabe Projektdefinition	Eruieren eines Projektes Ausformulierung und Projekt in EBS einrichten
05.12.2012	Abgabe Teaser	Teaser als HTML für Moodle erstellen
20.02.2013	Abgabe Projekt	Dokumentation inkl. App
27.02.2013	Präsentation	Projekt-Präsentation

## 2.5. Aufwandschätzung

Schritt	Geschätzt (h)	Effektiv (h)	Abweichung (h)
Konzept, Projekteingabe	4	4	0
IDE / Tools einrichten	2	2	0
Datenbank einrichten	1	1	0
Webservice BackEnd	10	12	+2
Authentifizierung BackEnd	10	11	+1
Webserver-Anbindung Client	5	4	+1
Frontend	15	25	+10
Authentifizierung Client	10	8	-2
Offline-Modus Client	10	13	+3
Teaser erstellen	5	4	-1
Dokumentation	25	30	+5
Testing	5	4	-1
Präsentation	4	pendent	
<b>Total</b>	<b>106</b>	<b>(118)</b>	<b>(+13)</b>

Der geschätzte Arbeitsaufwand ist höher als in der Aufgabenstellung definiert, da mir eine Umsetzung sehr wichtig war. Durch die fehlenden Kenntnisse musste daher in der Aufwandschätzung auch die Einarbeitungszeit für die nötigen Sprachen und Konzepte einberechnet werden.

## 3. Umsetzung

### 3.1. Grundidee

Die App [3] soll zur einfacheren Erweiterung modular aufgebaut werden. Basis stellt die Authentifizierung und die Kommunikation zum Server dar. Als erstes Modul wird gemäss der Aufgabenstellung ein zentral verwaltetes Telefonverzeichnis eingefügt. Weitere Funktionen können entsprechend ergänzt werden.

### 3.2. modularisierte Darstellung

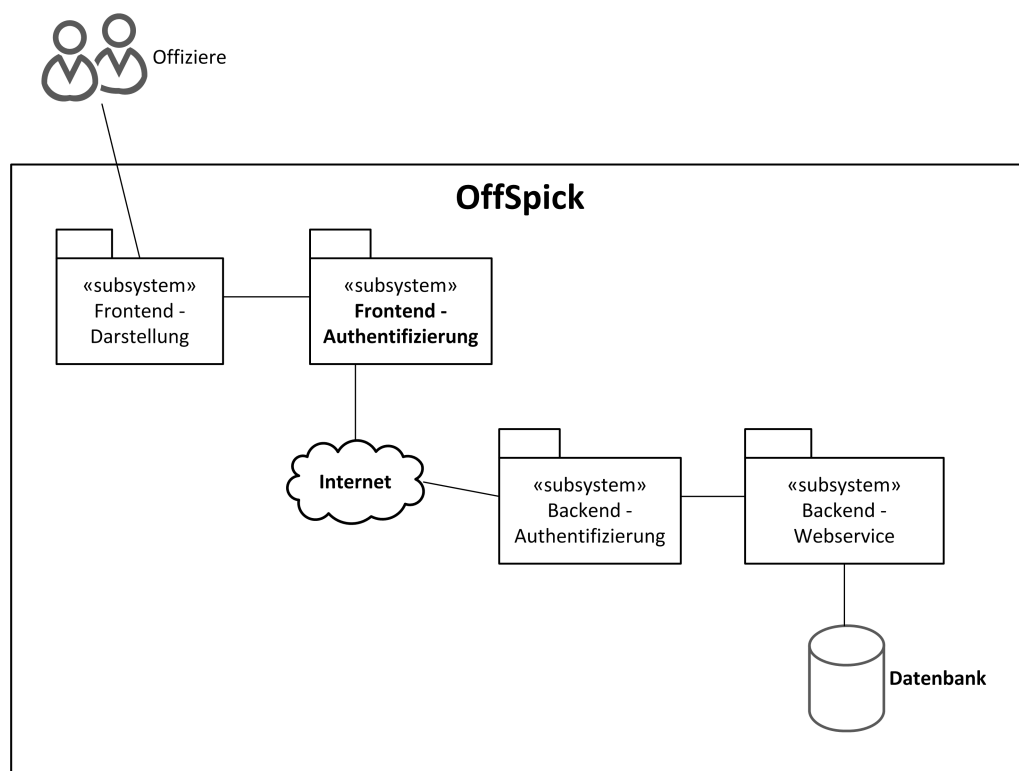


Abbildung 3.1.: modularisierte Darstellung

## 3.3. Module

In den folgenden Abschnitten wird von Backend und Frontend gesprochen. Dies bezieht sich auf die folgenden Verzeichnisse und die darin enthaltenen Dateien:

Begriff	Funktion	Pfad
Backend	Webservice auf bestehendem Webserver	../offspick/
Frontend	touch-optimierte Webseite für Darstellung auf Endgerät	../webapp/

### 3.3.1. Backend - Datenbank (generell)

Für die Ablage wird eine MySQL-Datenbank auf dem bestehenden Webserver verwendet. Die Anbindung erfolgt dynamisch bei Zugriff auf den Webservice und ist mit der PHP-MySQLi-Erweiterung umgesetzt. Die Verbindungsparameter sind unter configuration.php abgelegt.

### 3.3.2. Backend - Authentifizierung

Um das Telefonverzeichnis vor unbefugtem Zugriff zu schützen, muss die Abfrage entsprechend gesichert werden. Dazu erfasst der Administrator via der OffSpick-Admin-Homepage für jeden Benutzer einen Username, Passwort und den Gültigkeitsbereich (um neue Benutzer vorgängig zu erfassen resp. zu sperren). Diese Angaben muss der Benutzer einmalig auf seinem Client eintragen.

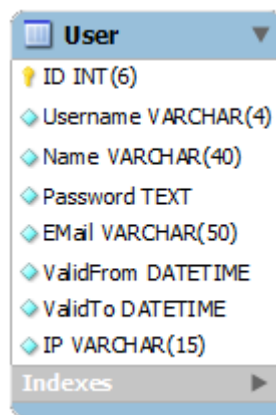


Abbildung 3.2.: USER-Tabelle

Damit die Passwörter nicht im Klartext übermittelt werden, wird lediglich ein per SHA256 erstellter Hashcode des Passwortes in der USER-Tabelle abgelegt, resp. übertragen. Um die Entschlüsselung zu erschweren, wird das Passwort künstlich mit einem zusätzlich angefügten und via configuration.php wählbarem String (SECURITYSALT) verlängert. Die folgende Codezeile ist für den Hash-Code zuständig:

```
$password = hash('sha256', $param['pw'].SECURITYSALT)
```



Wichtig: Da anstelle der Passwörter ein Hash-Code des Passwortes in der USER-Tabelle verschlüsselt abgelegt wird, müssen bei einer Anpassung des SECURITYSALT-String auch gleichzeitig alle Hash-Codes der Passwörter neu abgelegt werden!

Die serverseitige Authentifizierung zur Erstellung einer Session erfolgt nach folgendem Ablauf:

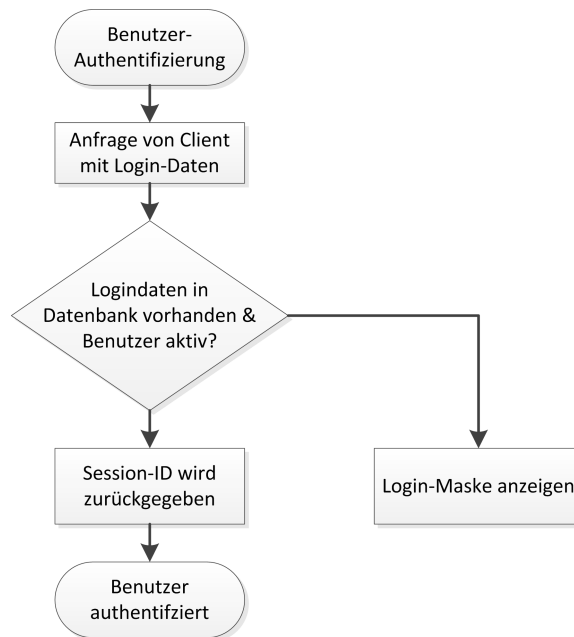


Abbildung 3.3.: Benutzerauthentifizierung Backend

Für die Validierung der User-Daten steht im Backend die Funktion `OffSpickCheckLogin()` zur Verfügung. Diese Funktion prüft mittels SQL-Befehl, ob in der USER-Tabelle ein Datensatz mit diesen Login-Daten vorhanden ist. Falls dem so ist, wird die Session-Variable `$_SESSION['logged.in'] = true` gesetzt und der Client erhält als Antwort eine Session-ID.

### 3.3.3. Backend - Telefonverzeichnis

Für das Telefonverzeichnis gelten folgende Anforderungen:

- Jeder Kontakt kann in mehreren Kategorien vorhanden sein
- Jeder Kontakt hat eine Gültigkeit (um spez. Pikettpläne abzudecken)
- Jede Kategorie kann ein oder mehrere Kontakte enthalten

Dieses Anforderungen werden mit den folgenden 4 Tabellen realisiert:

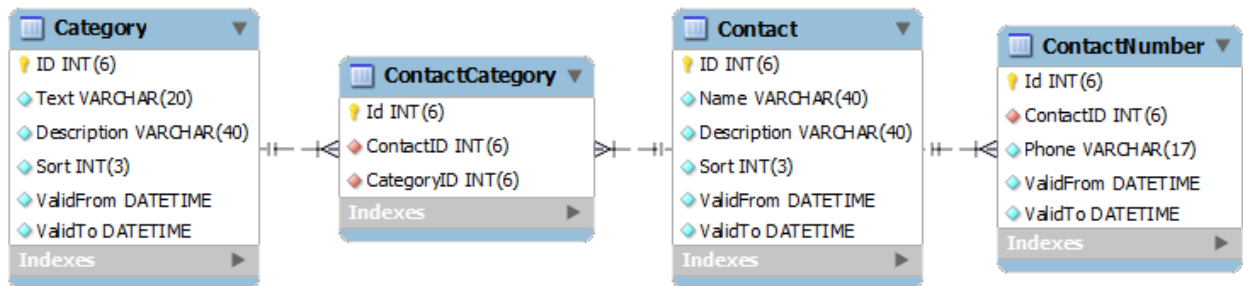


Abbildung 3.4.: DB-Struktur für Telefonverzeichnis

Tabelle *CATEGORY* (entspricht den Kategorien in der obersten Ebene)

Tabelle *CONTACTCATEGORY* (Zuteilung Kontakte an Kategorien)

Tabelle *CONTACT* (Kontakteinträge: Bezeichnung des Kontaktes)

Tabelle *CONTACTNUMBER* (Telefonnummer des Kontaktes)

### 3.3.4. Backend - Webservice

Der Zugriff der Daten soll aus dem Client via http-Webservice erfolgen. Für das Austauschformat der Daten kommen grundsätzlich zwei Technologien in Frage:

Format	Overhead	Typisiert	Vorteil
XML [1]	mimal-mittel	nein	viele Definitionsmöglichkeiten (flexibler)
JSON [2]	minimal	ja	i.d.R. sind JSON-Daten valides Javascript

Da die Datenmenge wegen der Übermittlung möglichst klein sein sollte und die Struktur sehr simpel gehalten ist, fällt der Entscheid zu Gunsten von JSON aus. Der Inhalt des JSON-Objekt soll bewusst in einer nicht bis auf die letzte Stufe normalisierte Struktur aufgebaut werden (Jeder Datensatz entspricht beim Telefonverzeichnis einem Kontakt, enthält aber ebenfalls die Bezeichnung der dazugehörigen Kategorie). Die Reihenfolge ist entsprechend der definierten Sortierung nach Kategorie resp. Kontakt.

Da in PHP mit den Befehlen `json_encode()` resp. `json_decode()` eine einfache JSON-Integration möglich ist, wird für diesen Softwareteil PHP verwendet.

Die Anfrage an den Webservice im Backend erfolgt nach folgendem Schema:

`http://www.domain.com/offspick/?func=Parameter`

In der damit aufgerufenen index.php wird dann mittels folgendem Switch geprüft, welche Funktion explizit angefordert wurde:

```
switch($func){  
  case "offSpickList": offSpickList($_REQUEST); break;  
  case "offSpickCheckLogin": offSpickCheckLogin($_REQUEST); break;  
  default: doerror("hmmm* - diese Funktion wurde noch nicht definiert");  
}
```

Für das Telefonverzeichnis wird der folgende Aufruf verwendet:

```
http://www.domain.com/offspick/?func=offSpickList
```

Innerhalb der Funktion offSpickList werden dann mittels dem folgenden SQL-Befehl sämtliche nötigen Angaben für die Anzeige der Kontakte aus der MySQL-DB ausgelesen.

### 3.3.5. Client - Authentifizierung

Damit die Anmeldedaten nicht bei jedem Aufruf eingegeben werden müssen, speichert die App diese Angaben lokal im Browser (im sog. LocalStorage [3]).

Der Authentifizierungsprozess läuft wie folgt ab:

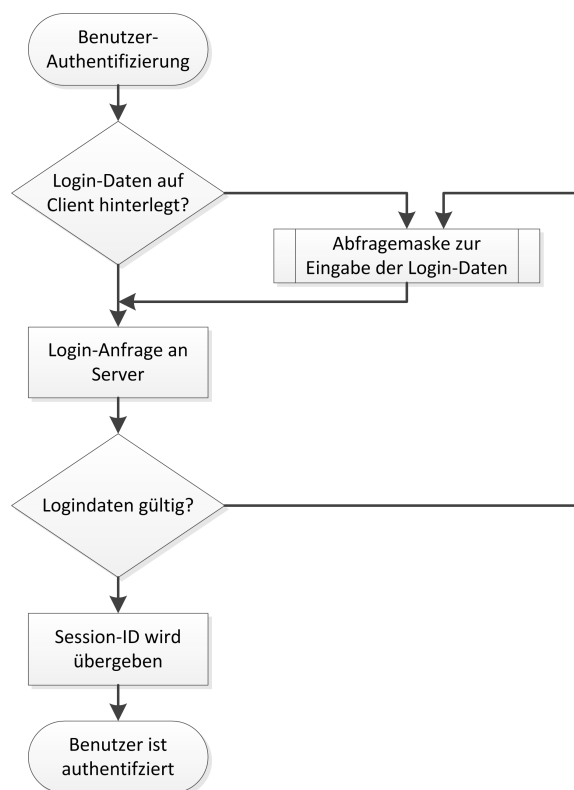


Abbildung 3.5.: Benutzerauthentifizierung Client

### 3.3.6. Client - Darstellung

Die Darstellung der Telefonnummern soll einfach und übersichtlich sein. Mit HTML könnte eine solche Liste erstellt werden, was jedoch durch die dynamische Grösse nicht ganz einfach ist. Alternativ bietet jQueryMobile [4] [5] bereits sehr komfortable Listen an. Für die Darstellung wird daher die Nested-List von jQueryMobile verwendet

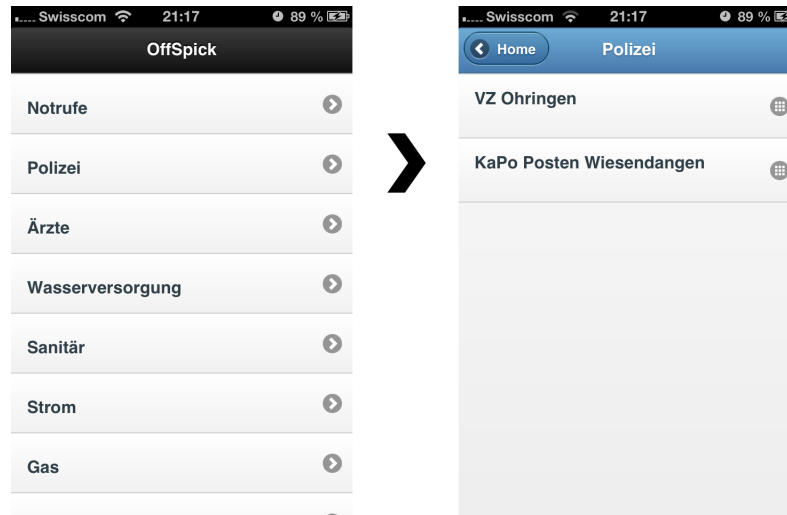


Abbildung 3.6.: Kategorie-Übersicht und Sub-Kategorie 'Polizei'



Abbildung 3.7.: Nummer von 'KaPo Posten Wiesendangen' wählen

## 4. Projekt-Tools

Für dieses Projekt wurden folgende Tools eingesetzt:

Produkt/Dienst	Beschreibung	Link
Dropbox	Dateiablage für sonstige Projekt-Dokumente	Auf Anfrage
Github	Quellcode-Verwaltung	<a href="https://github.com/M0tt/OffSpick">github.com/M0tt/OffSpick</a>
LaTeX	Dokumentation	<a href="http://www.texworks.org">www.texworks.org</a>
HTML5	Sprache für Webseiten	<a href="http://www.php.net">www.php.net</a>
jQueryMobile V1.20	Touch-Optimiertes Web Framework für Smartphones + Tablets	<a href="http://www.jquerymobile.com">www.jquerymobile.com</a>
mySQL V5.0.77	Datenbank	<a href="http://www.mysql.com">www.mysql.com</a>
PHP	serverseitig interpretierte Skriptsprache	<a href="http://www.php.net">www.php.net</a>

## 5. Fazit / Ideen für Erweiterungen

Diese Technologie wird erstmalig im FZWB eingesetzt - und führt bereits jetzt zu weiteren Wünschen. Durch diese technische Unterstützung sind viele weitere Möglichkeiten wie Zugriff auf Wasserpläne, Kartenmaterial abhängig vom aktuellen Ort, Fotos mit gleichzeitigem Protokoll-Eintrag (Datum/Zeit, Koordinaten, Kamerawinkel), ... denkbar.

Aus diesen Gründen wird im Moment noch auf eine Umwandlung mittels PhoneGap in eine iOS resp. Android-App verzichtet. Die Funktionalität ist daher als WebApp höchstens bei fehlender Internetverbindung eingeschränkt.

Mit dem in diesem Projekt erarbeiteten Wissen lassen sich aber bereits weitere Wünsche umsetzen.

## A. Glossar

Begriff	Beschreibung	Link
EBS	Elektronisches Bewertungssystem - System der ZHAW	<a href="https://ebs.zhaw.ch">https://ebs.zhaw.ch</a>
FZWB	Feuerwehr Zweckverband Wiesendangen-Bertschikon	<a href="http://www.feuerwehr-wiesendangen.ch">www.feuerwehr-wiesendangen.ch</a>
Handheld	Mobiltelefone mit erweiterten Funktionen	
HTTP	Hypertext Transfer Protocol - Protokoll zur Datenübertragung	
jQM	jQueryMobile - TouchOptimiertes Web Framework für Smartphones und Tablets	<a href="http://www.jquerymobile.com">www.jquerymobile.com</a>
JSON	JavaScript Object Notation - kompaktes Datenformat	<a href="http://www.json.org">www.json.org</a>
MySQL	relationalen OpenSource Datenbankverwaltungssystem	<a href="http://www.mysql.org">www.mysql.org</a>
OffSpick	Offizier-Spick - Telefonverzeichnis für weitere Einsatzelemente, Firmen oder Fachpersonen	
PhoneGap	Framework für Mobile Geräte	<a href="http://www.phonegap.com">www.phonegap.com</a>
WebApp	Web-Applikation, lauffähig auf gängigem Handheld	
ZHAW	Zürcher Hochschule für angewandte Wissenschaften	<a href="http://www.zhaw.ch">www.zhaw.ch</a>

## B. Literaturverzeichnis

- [1] <http://www.w3schools.com/xml/>. *XML Tutorial*. w3schools.com, Stand: 27.12.2012.
- [2] <http://www.w3schools.com/json/>. *JSON Tutorial*. w3schools.com, Stand: 27.12.2012.
- [3] Johannes Ippen Florian Franke. *Apps mit HTML5 und CSS3*. Galileo Press, 28.05.2012.
- [4] John Reid. *jQuery Mobile*. O'Reilly, 28.11.2011.
- [5] <http://jquerymobile.com/>. *jQuery Mobile Doku*. jQuery, Stand: 27.12.2012.



## C. Abbildungsverzeichnis

3.1. OffSpick - modularisierte Darstellung . . . . .	7
3.2. Backend Authentifizierung - USER-Tabelle . . . . .	8
3.3. Backend Authentifizierung - Prozess . . . . .	9
3.4. Datenbank-Struktur für Telefonverzeichnis . . . . .	10
3.5. Frontend Authentifizierung - Prozess . . . . .	11
3.6. OffSpick Beispiel - Übersicht . . . . .	12
3.7. OffSpick Beispiel - Telefonnummer wählen . . . . .	12

(Quelle Grafiken: eigene Darstellung)

## D. Quellcode

Auf das Anfügen von Quellcode wurde aus Platzgründen verzichtet. Den kompletten Quellcode findet man im GIT-Repository (gemäss Angaben unter Kapitel 4)