



**HSR**

HOCHSCHULE FÜR TECHNIK  
RAPPERSWIL

FHO Fachhochschule Ostschweiz



# Outdoor Object Tracing

## **Semesterarbeit**

Abteilung Informatik, Hochschule für Technik Rapperswil

Florian Ziltener

Mike Osmand Loydd

23. Dezember 2011/ Herbstsemester 2011

Themensteller Prof. Dr. Peter Heinzmann

Betreuer Prof. Dr. Peter Heinzmann

Projektpartner cnlab Software, 8640 Rapperswil



## Outdoor Object Tracing

### Erklärung

Ich erkläre hiermit,

- dass ich die vorliegende Arbeit selber und ohne fremde Hilfe durchgeführt habe, ausser derjenigen, welche explizit in der Aufgabenstellung erwähnt ist oder mit dem Betreuer schriftlich vereinbart wurde,
- dass ich sämtliche verwendeten Quellen erwähnt und gemäss gängigen wissenschaftlichen Zitierregeln korrekt angegeben habe.

Rapperswil, den 23.12.11

F. Hiltner  
.....  
Die Studentin/der Student

Rapperswil, den 23.12.2011

St. C...  
.....  
Die Studentin/der Student



Die Multiplattform-Anwendung *CoachAssist* hilft Fussballtrainern bei der Vorbereitung und Protokollierung von Fussballspielen. Mit *CoachAssist* können Coaches und Trainer Mannschaftsaufstellungen festlegen, Spielereignisse erfassen, Notizen zum Spielverlauf erstellen und die erfassten Daten auf die Web-Plattform des cnlab Outdoor-Player-Tracking-Systems übermitteln. Die *CoachAssist*-Daten ergänzen die GPS-Tracking-Daten zu den Spielern, womit noch aussagekräftigere Analysen und Statistiken möglich sind.

*CoachAssist* wurde mit den Web-Technologien JQuery und JQuery Mobile entwickelt und mit Phonegap zu einer nativ App umgewandelt. Damit kann *CoachAssist* auf iPhone, Android, WindowsPhone und Browser Plattformen genutzt werden. Für den "Mobile App Look" sorgt der Einsatz des JQuery Mobile Frameworks. Die Kommunikation mit dem cnlab Server erfolgt mit Jason.

Zur Authentisierung beim Server wird auf dem *CoachAssist*-Rechner ein Token hinterlegt. So können bereits erfasste Mannschaftsdaten vom cnlab-Server übernommen werden. Über die intuitiv geschaltete grafische Benutzerschnittstelle legt der Trainer vor dem Spiel die Mannschaftsaufstellung fest. Dies beinhaltet die Startaufstellung, die Ersatzbank und Informationen zu Gegner, Spielort und Anspielzeit. Während dem Spiel können über Ereignistasten (Eventbuttons) wichtige Spielsituationen "notiert" werden. Zu jedem Ereignis wird automatisch ein Zeitstempel erstellt und das Ereignis wird im Liveticker angezeigt. Spielstand und Spielzeit bzw. eine Spieluhr werden automatisch nachgeführt und angezeigt. Nach dem Spiel können die Daten lokal weiter verarbeitet oder zum cnlab Server hochgeladen werden.



---

# Inhaltsverzeichnis

<b>1</b>	<b>Aufgabenstellung</b>	<b>11</b>
<b>2</b>	<b>Management Summary</b>	<b>15</b>
2.1	Ausgangslage . . . . .	15
2.2	Vorgehen, Technologie . . . . .	15
2.3	Ergebnisse . . . . .	16
2.4	Ausblick . . . . .	16
<b>I</b>	<b>CouchAssist Übersicht</b>	<b>17</b>
<b>3</b>	<b>Einleitung</b>	<b>19</b>
3.1	Fussballtrackingsystem . . . . .	19
3.2	Trainernotizbuch . . . . .	20
3.3	Statistikanalyse . . . . .	20
<b>4</b>	<b>Trainernotizbuch</b>	<b>21</b>
4.1	Beschreibung . . . . .	21
4.2	Ausblick . . . . .	22
<b>5</b>	<b>Mitbewerberanalyse</b>	<b>23</b>
5.1	Apple Store . . . . .	23
5.1.1	Übersicht . . . . .	23
5.1.2	iCoachSoccer . . . . .	24
5.1.3	Coach it Soccer . . . . .	24

5.1.4	Football Plan . . . . .	24
5.1.5	Laola . . . . .	24
5.2	Android Markt . . . . .	25
5.2.1	Übersicht . . . . .	25
5.2.2	Mobile Soccer Coach und Lite Version . . . . .	25
5.2.3	Soccer Pad und Lite Version . . . . .	25
5.3	Crossplatform . . . . .	26
5.4	Fazit . . . . .	26
<b>6</b>	<b>Technologien</b>	<b>27</b>
6.1	JQuery Mobile . . . . .	27
6.2	PhoneGap . . . . .	28
6.2.1	Grund für die Nutzung . . . . .	28
6.2.2	PhoneGap vs. Appcelerator Titanium Mobile . . . . .	28
<b>II</b>	<b>Softwaredokumentation</b>	<b>29</b>
<b>7</b>	<b>Requirements</b>	<b>31</b>
7.1	Allgemeine Beschreibung . . . . .	31
7.1.1	Produkt Perspektive . . . . .	31
7.1.2	Produkt Funktion . . . . .	31
7.1.3	Einschränkungen . . . . .	32
7.1.4	Benutzer Charakteristik . . . . .	33
7.2	Funktionale Anforderungen . . . . .	33
7.2.1	UseCase Übersicht . . . . .	34
7.2.2	Mannschaft erstellen . . . . .	34
7.2.3	Spieler hinzufügen . . . . .	34
7.2.4	Daten herunterladen . . . . .	34
7.2.5	Daten hochladen . . . . .	35
7.2.6	Spiel aufzeichnen . . . . .	35
7.2.7	Event hinzufügen . . . . .	35
7.2.8	Spiel abspielen . . . . .	35
7.2.9	Aufstellung generieren . . . . .	35
7.3	Nicht funktionale Anforderungen . . . . .	35
7.3.1	Zuverlässigkeit . . . . .	35
7.3.2	Benutzbarkeit . . . . .	36
7.3.3	Effizienz . . . . .	36



7.3.4	Änderbarkeit . . . . .	36
7.3.5	Übertragbarkeit . . . . .	36
<b>8</b>	<b>Analyse</b>	<b>37</b>
8.1	Aufbau . . . . .	37
8.1.1	index.html . . . . .	38
8.1.2	info.html . . . . .	38
8.1.3	game.html . . . . .	38
8.1.4	note.html . . . . .	38
8.1.5	playerDialog.html . . . . .	38
8.1.6	formation.html . . . . .	38
8.1.7	optionSystem.html . . . . .	38
8.2	Systemsequenzdiagramme . . . . .	39
8.2.1	Match . . . . .	39
8.2.2	Formation - Drag and Drop . . . . .	40
<b>9</b>	<b>Architektural Design</b>	<b>41</b>
9.1	Logische Architektur . . . . .	41
9.1.1	Package Diagramm . . . . .	41
9.1.2	Verwendete Patters . . . . .	43
9.2	Systemoperationen . . . . .	43
9.2.1	Drag and Drop a Player . . . . .	43
9.2.2	Daten senden und empfangen . . . . .	44
9.3	Kommunikation Client und Server . . . . .	45
9.3.1	JSON . . . . .	45
9.3.2	Server-API . . . . .	45
9.4	Datenmodel . . . . .	47
9.4.1	Login . . . . .	47
9.4.2	Team . . . . .	47
9.4.3	League . . . . .	47
9.4.4	Person . . . . .	48
9.4.5	Player . . . . .	48
9.4.6	Position . . . . .	48
9.4.7	Game . . . . .	48
9.4.8	GameEvent . . . . .	48
9.4.9	LineupType . . . . .	49
9.4.10	LineupTypePosition . . . . .	49

<b>10 Qualitätsmanagement</b>	<b>51</b>
10.1 Codeguidelines . . . . .	51
10.1.1 HTML . . . . .	51
10.1.2 JavaScript . . . . .	52
10.2 Designguidelines . . . . .	54
<b>11 Testing</b>	<b>55</b>
11.1 Systemtests . . . . .	55
11.2 Usability Tests . . . . .	56
11.2.1 PaperPrototyp . . . . .	56
11.2.2 GUI Betatest . . . . .	56
11.3 Unit Tests . . . . .	57
<b>12 Schlussfolgerung</b>	<b>59</b>
12.1 Allgemein . . . . .	59
12.1.1 Ausblick . . . . .	59
12.2 Persönlicher Bericht Florian Ziltener . . . . .	60
12.2.1 Projektverlauf . . . . .	60
12.2.2 Was habe ich gelernt? . . . . .	60
12.2.3 Fazit . . . . .	60
12.3 Persönlicher Bericht Mike Osmand Loydd . . . . .	61
12.3.1 Projektverlauf . . . . .	61
12.3.2 Was habe ich gelernt? . . . . .	61
12.3.3 Fazit . . . . .	62
<b>Literaturverzeichnis</b>	<b>64</b>
<b>Glossar</b>	<b>65</b>
<b>Tabellenverzeichnis</b>	<b>67</b>
<b>Abbildungsverzeichnis</b>	<b>70</b>
<b>III Anhang</b>	<b>71</b>
<b>A Projekt-Management</b>	<b>73</b>
A.1 Zeitplan . . . . .	73
A.2 Meilensteine . . . . .	73
A.3 Risikoanalyse . . . . .	74

A.3.1	Risiken . . . . .	74
A.3.2	Fazit . . . . .	75
<b>B</b>	<b>User Interface</b>	<b>77</b>
B.1	PaperPrototyp . . . . .	77
<b>C</b>	<b>Statistikanalyse</b>	<b>79</b>
C.1	Neue Statistiken . . . . .	79
C.1.1	Sprint . . . . .	79
C.1.2	Mannschaftsheatmap . . . . .	80
C.1.3	Spielerbewegung . . . . .	80
C.1.4	Auswertung eines Spielers . . . . .	80
<b>D</b>	<b>Benutzerdokumentation</b>	<b>83</b>
D.1	Voraussetzung . . . . .	83
D.2	Installation . . . . .	83
D.3	Nutzung . . . . .	86



*Tabelle 1.1:  
Aufgabenstellung - Die  
wichtigsten Angaben*

<b>Abteilung:</b>	Informatik
<b>Name der Studierenden:</b>	Mike Hoskin OSMAND LOYDD, Florian ZILTENER
<b>Studienjahr:</b>	Herbstsemester 2011
<b>Titel der Studienarbeit:</b>	Fussballspieler-Tracking
<b>Examinator:</b>	Prof. Dr. P. Heinzmann
<b>Experte / Koreferent:</b>	-
<b>Themengebiet:</b>	Internet-Technologien und -Anwendungen
<b>Projektpartner:</b>	cnlab AG Rapperswil, Th. Bruhin, P. Eberli

Die Bedeutung von Internet Technologien und Anwendungen im Sport ist in den letzten Jahren immer grösser geworden. cnlab hat mit den TourLive-Systemen vor allem im Radsport Fuss gefasst. Das Outdoor Sportler Tracking System kommt im Fussball zum Einsatz. Im Rahmen dieser Studienarbeit soll das Fussballspieler-Tracking System mit einer Spielprotokollierungsanwendung ergänzt werden.

Das Fussballspieler-Tracking System erfasst die Bewegungen von Fussballspielern, welche kleine GPS-Empfänger auf sich tragen <http://www.cnlab.ch/fussball/>. Es ist eine Anwendung zur Visualisierung und Analyse von Spiel- und Spielerdaten auf verschiedenen Plattformen basierend auf jQuery Mobile zu realisieren.

Unter anderem sind folgende Aufgaben zu bearbeiten:

- Analyse von Spielerdaten
  - Distanz
  - Geschwindigkeit
  - Sprint
  - Pulsbereiche
- Analyse von Spieldaten
  - Spielergruppen-Breite und ?Tiefe
  - Spielergruppen-Fläche
- Tablet-Anwendung zum Aufzeichnen von Fussballspielen
  - Aufzeichnen von Spielereignissen
  - Erstellen von Notizeinträgen während dem Spiel ermöglichen
  - Festlegung der Mannschaftsaufstellung

### **Teilaufgaben**

#### **Analyse**

- Aufnahme von Spielen mit dem aktuellen System
  - Kontakt zu Verein
  - Einsatz bei zwei Spielen
  - Auswertung
- Studium ähnlicher Anwendungen
  - Inmotio
  - Publikationen
- Zusammenstellung und Vergleich verschiedener Auswerteverfahren
  - Heatmap
  - Relative Positions
  - Sprints
  - Squares (length-width squares)
  - Attack-/Defense Situations

## **Design**

- Systemarchitektur (DB)
- Plattform: Tablets (Ipad, Android Tablet)

## **Implementierung**

- Tablet-Anwendung mittels JQueryMobile und PhoneGap

## **Testing, Umsetzung**

- Analysen basierend auf bereits aufgenommenen Daten
- Aufnahmen und Auswertungen zu mehreren Spielen durchführen (Frühjahr 2012)

## **Administratives**

- <http://www.cnlab.ch/kurse/sada/>
- Redmine
- Dropbox





---

## 2.1 Ausgangslage

Diese Arbeit baut auf dem Trackingsystem der Firma “cnlab“ auf. Zurzeit werden Spielereignisse beim Aufzeichnen von Fussballspielern noch mit Stift und Papier getätigt. Die Daten werden anschliessend manuell von Hand in das System eingegeben.

Mit dieser Arbeit wurde ein Programm realisiert, welches es ermöglicht, Daten mittels eines Tablets zu erfassen. Diese Daten werden nachher direkt dem Server zugeschickt. Damit entfällt das umständliche Notieren.

---

## 2.2 Vorgehen, Technologie

In der ersten Phase der Arbeit wurden Fussballinformationen gesammelt. Zusätzlich wurden zwei Fussballspiele mit dem bestehenden Trackingsystem aufgezeichnet. Dabei wurden auch verschiedene Beobachtungen gemacht, welche Aspekte es bei einem Spiel zu berücksichtigen gilt. Nebenbei wurde nach Erweiterungen und Ideen der statistischen Erfassung gesucht, die das Online-System “[www.cnlab.ch/fussball](http://www.cnlab.ch/fussball)“ erweitern.

In der zweiten Phase wurden verschiedene Analysen durchgeführt. Zum einen wurde eine Mitbewerberanalyse erstellt, um ähnliche Programme auf ihre Funktionalitäten zu prüfen. Dazu wurden auch die Requirements definiert. Zusätzlich wurde schon einmal ein Paper-Prototyp erstellt, um das Grunddesign zu festigen.

Ein weiterer wichtiger Punkt in dieser Phase war es, sich auf eine Technologie zu einigen, mit der sich die Tablet-Anwendung realisieren lässt. Der Entscheid fiel zu Gunsten von JQuery Mobile aus. Hauptgrund dafür war, dass die Software damit auf mehreren Plattformen betrieben werden kann. Um schlussendlich das Ganze dann in eine native App umzuwandeln, wurde es mit dem PhoneGap Framework vereint.

Die dritte Phase startete mit der Analyse des Programmaufbaus. Zusätzlich wurde mit der Codeerzeugung von Grundelementen begonnen. Des Weiteren wurde der Paper-Prototyp getestet und die Änderungen in das Design übernommen.

In der vierten Phase wurde die Architektur festgelegt und der Code dementsprechend erweitert.

In der letzten Phase wurde die Software diversen Tests unterzogen und daraufhin ein paar wenige Bugs beseitigt.

---

## 2.3 Ergebnisse

Das entwickelte Programm ermöglicht es nun Spiele aufzuzeichnen. Dafür wurden diverse Events zur Verfügung gestellt, mit denen der Benutzer die Geschehnisse sehr einfach protokollieren kann.

Spieler- und Mannschaftsdaten werden von einem Server heruntergeladen und können dann vom Programm verwendet werden. Wenn ein Match aufgezeichnet wurde, kann der Benutzer diese Daten einfach auf den Server hochladen. Dabei werden alle erfassten Ereignisse mitgesendet.

Des Weiteren hat der Benutzer die Möglichkeit eine Aufstellung zu erstellen und diese für das aktuelle Spiel zu verwenden. Schlussendlich darf festgestellt werden, dass die wichtigsten Funktionen alle eingebaut wurden und verwendet werden können. Zusätzlich besteht die Möglichkeit diese auch zu erweitern.

---

## 2.4 Ausblick

Die Grundstruktur der Applikation steht und kann beliebig ergänzt werden. Sei es mit Daten oder zusätzlichen Funktionen.

Die Erfassung der Spielerdaten ist über das Serverinterface möglich.

Weitere mögliche Funktionen sind in den Requirements ersichtlich. Die Software kann man mit diesen Funktionen sehr einfach erweitert werden. Die wichtigsten Erweiterungen wären: auf anderen Plattformen testen und die Applikation so zu erweitern, dass kein Server benötigt wird. Des Weiteren wären die Funktionen Löschen und Editieren von Events ebenfalls wichtig, da der Benutzer Fehler beim Erfassen machen wird.

# **Teil I**

## **CouchAssist Übersicht**



Dieser Bericht soll einen Einblick in die Semesterarbeit geben. Zusätzlich soll sie auch eine Stütze für diejenigen Personen sein, welche nachher weiter dieser Software arbeiten werden. Die Arbeit baut auf der bestehenden Software des Fussballtrackingsystems auf. Das Auswertungssystem ist im Internet zu finden. Der Link lautet <http://www.cnlab.ch/fussball>.

---

## 3.1 Fussballtrackingsystem

Das Fussballtrackingsystem zeigt die Positionen der Spieler an. Die Positionsdaten werden mittels GPS-Logger im Sekunden- oder 0.2-Sekundentakt aufgezeichnet. Die GPS-Logger tragen die Spieler am Arm. Der 29g schwere und 10cm x 10cm x 10cm grosse GPS-Logger ist in einer Armbinde befestigt. Nach dem Spiel werden die Positionsdaten über den USB-Anschluss des GPS-Loggers auf den PC übertragen. Mit der Download-Software werden pro GPS-Logger insgesamt fünf Dateien erstellt. Mit der .html-Datei können die Laufwege des Fussballspielers auf Googlemap betrachten.

Die .vcs-Datei enthält für jedes GPS-Signal, welches empfangen wurde, einen Eintrag. Dieser definiert den auf einen Meter exakten Punkt, in welchem sich der Spieler befindet. Diese Datei wird zum Auswertungsserver übertragen, in welchem die absoluten Positionsdaten aufbereitet werden, dass der Spielverlauf, von Anpfiff bis zum Ende, ersichtlich ist. Das digitale Fussballspiel ist dann im Internet unter <http://www.cnlab.ch/fussball> zu finden. So können Trainer, Spieler, Schiedsrichter und andere Interessierte den Spielverlauf analysieren. Trainer können überprüfen, ob die Spieler ihre Position eingehalten haben. Oder, ob sie die Aufgaben ausgeführt haben, welche ihnen aufgetragen wurde. Die Software bietet jedoch nicht nur hilfreiche Informationen für den Trainer, sondern auch für den Fussballspieler. Er kann z. B. nachverfolgen wie viele Kilometer er während des Spieles gerannt ist.

---

## 3.2 Trainernotizbuch

Dieser Teil ist das Trainernotizbuch. Ziel dieser Software ist es, dem Trainer während eines Fussballspiels eine Stütze bereitzustellen. Mit dieser Software ist es möglich, bestimmte Ereignisse während eines Spiels aufzuzeichnen. Das Trainernotizbuch soll auf iPad, Android oder Notebook-Plattformen genutzt werden können. Die Software wird daher mit JQuery Mobile realisiert.

---

## 3.3 Statistikanalyse

Der zweite Teil umfasst unter anderem folgende Punkte:

- Analyse von Spielerdaten
  - Distanz
  - Geschwindigkeit
  - Definition und Auswertung charakteristischer Parameter (z. B. Sprint, Bewegungsrichtung)
  - Individualisierung der Auswertung, Anpassung an individuelle Fähigkeiten (Pulsbereiche, Spitzengeschwindigkeit)
- Analyse von Spieldaten
  - Spielergruppen-Breite und -Tiefe
  - Spielergruppen-Fläche

Durch das Entwickeln neuer Auswertungsmöglichkeiten soll es für einen Fussballtrainer einfacher werden, sein Team zu beurteilen und zu bewerten. Des Weiteren wird sich im Laufe der Arbeit zeigen, wie sich aus den neuen Statistikverfahren neue Daten gewinnen lassen. Ein Beispiel dafür ist, dass mittels der Werte des Spielers bestimmte Positionen zugeordnet werden können. Eine genauere Ausführung ist im Anhang unter dem Kapitel Statistik zu finden.

Das Trainernotizbuch *CoachAssist* soll den Trainern helfen, sich auf ein Spiel vorzubereiten, indem der Trainer die Formation, Mannschaftsaufstellung und die Ersatzbank für ein kommendes Spiel fixiert. Des Weiteren soll es den Trainern dienen einen Liveticker vom Spiel, inklusive Notizen, zwecks späteren Analysen zu erstellen.

#### 4.1 Beschreibung

Das Trainernotizbuch holt nach der Eingabe des zugewiesenen cnlab Tokens die erfassten Mannschaftsdaten vom Server [www.cnlab.ch/fussballapi/](http://www.cnlab.ch/fussballapi/) und speichert diese in der Datenbank ab. Nach der Eingabe des Tokens wird die Drop-Down Liste mit den Mannschaften aktiviert. Nach der Auswahl der Mannschaft werden die Spieler in der Formation Seite aufgelistet. In der Formation Seite kann der Trainer die Formation für das aufzuzeichnende Spiel definieren. Dazu kann er die Startformation sowie die Ersatzbank festlegen. Bevor die Spielaufzeichnung erfolgen kann, muss der Trainer die Spieldaten eingeben. Erst nach der vollständigen Erfassung kann er zur Spielaufzeichnung navigieren. Die Aufzeichnung des Spiels wird durch Drücken der "SSpielbeginn"Buttons gestartet. Dabei fängt die Spielzeit an zulaufen und sämtliche Ereignis Buttons der gegnerischen sowie der eigenen Mannschaft werden aktiviert. Ausserdem wird der Spielanfang als ein Ereignis Message im Liveticker erfasst mit dem Zeitpunkt der Aufzeichnung erfasst. Für jede Mannschaft können folgende Ereignisse erfasst werden:

- Auswechslung (Substitution)
- Tor geschossen (Goal shot)
- Rote Karte (Red card)
- Gelbe Karte (Yellow card)
- Elfmeter (Penalty)

- Freistoss (Freekick)
- Abseits (Offside)
- Einwurf (Throw-in)
- Eckball (Corner)

Beim Anwählen eines Ereignisses wird ein Dialogfenster, mit einer oder zwei Spielerlisten geöffnet. In diesem Fenster kann der Trainer die beteiligten Spieler des Ereignisses auswählen. Danach wird ein entsprechender Eintrag mit Ereignisname, Spielernummern und -namen, Mannschaftsname und Zeitstempel im Liveticker erstellt. Ein weiteres Feature, welche dem Trainer während der Spielaufzeichnung zur Verfügung steht, ist die Erstellung eines Notizeintrages. Beim Abspeichern dieses Eintrages wird im Liveticker ebenfalls ein Eintrag erstellt. Ein weiteres Feature ist die Möglichkeit zur Änderung der Formation während des Spiels und das senden an den Server dieser Daten.

---

## 4.2 Ausblick

Es gäbe eine Palette von Erweiterungsmöglichkeiten für *CoachAssist*. Obwohl ein heutiges Tablet eine gute Auflösung und gute Eingabefunktionen bieten, ziehen immer noch viele Leute das Arbeiten am Desktop vor. Deshalb sollten nur die am Ort nötigsten Feature implementiert werden.

Hier eine Auflistung von den wichtigsten Erweiterungen:

- Events mittels Player erfassen
- Eigene Formation definieren
- Applikation auf mehrere Sprachen erweitern
- Applikation auf anderen Plattformen testen und evtl. Anpassungen vornehmen
- Löschen und Editieren von Events
- Events mit Icons erweitern
- Applikation auch ohne Serveranbindung zur Verfügung stellen
- Eingabe des Notizeintrages per Voice
- Erfassen von temporären Spielern (Aushilfe vom Nachwuchses)
- Anzeige der erhaltenen gelben Karte der bisherigen Ligaspiele dieser Saison
- Mitsenden von Spieleraufstellungen beim Spielstart



Es wurden verschiedene Apps untersucht und in die Recherchen mit einbezogen. Diese wurden in drei Hauptkategorien unterteilt: nur androidfähig, nur iOS-fähig und Applikationen die in beiden Märkten vertreten sind

Neben den folgenden Tabellen befindet sich eine gesamte Übersicht der Mitbewerberapplikationen bei den Zusatzdokumenten

Für die Android Produkte wurden Informationen aus dem Android Market[6] bezogen. Die iOS Informationen wurden aus dem AppStore[5] genommen.

---

## 5.1 Apple Store

Im Shop von Apple wurden vier relevante Applikationen gefunden, welche die gleiche Grundfunktionalität aufweisen, wie das in der Arbeit beschriebenen App. Zuerst werden diese bereits bestehenden Apps in einer Gesamtübersicht aufgeführt. Im Anschluss gehen wir näher auf einzelnen Applikationen ein.

### 5.1.1 Übersicht

Die folgende Tabelle zeigt, welche Eigenschaften und Funktionen die Applikationen besitzen.

	iCoach Soccer	Coach it Soccer	Football Plan	Laola
Event-Erfassung	Nein	Ja	Nein	Ja
Notiz-Funktion	Nein	Nein	Ja	Nein
Statistik Funktion	Nein	Ja	Nein	Nein
Abgleich mit GPS Daten	Nein	Nein	Nein	Nein
Export Data	Nein	Nein	Nein	Nein

Tabelle 5.1: Apple Store -  
Mitbewerberanalyse

	iCoach Soccer	Coach it Soccer	Football Plan	Laola
Preis	5.00 Fr	10.00 Fr	2.00 Fr	gratis
Sprachen	Eng	Eng	Eng	De, Eng
Funktionen				
Spiel aufzeichnen	Nein	Ja	Nein	Ja
Mannschaft erfassen	Ja	Ja	Ja	Ja
Spieler erfassen	Ja	Ja	Ja	Ja
Taktik Darstellung	Ja	Nein	Ja	Ja
Drill-Funktion	Nein	Nein	Ja	Nein

### 5.1.2 iCoachSoccer

Diese Applikation ist speziell darauf ausgerichtet, dass der Trainer eine Aufstellung seiner Mannschaft auflisten kann. Somit ist es ihm möglich, Taktiken auszutüfteln, sowie seinen Spielern aufzeigen, wie sie im Spiel selbst vorzugehen habend. Diese zwei Funktionen werden in der entwickelten Version auch vorkommen.

### 5.1.3 Coach it Soccer

Diese Applikation beinhaltet einen grösseren Umfang. Es ist möglich Spiele aufzuzeichnen und Events (wie Goal oder Eckball) zu erfassen. Diese Ereignisse werden mittels den Spielern selbst erfassen. Zusätzlich wird auch eine Übersichtsstatistik angeboten.

### 5.1.4 Football Plan

Die Funktionalitäten sind ähnlich wie bei iCoachSoccer mit der Taktik Darstellung. Hier wird zusätzlich eine Drill-Funktion zur Verfügung gestellt, bei welcher der Trainer auch Übungen vorbereiten kann. Diese Applikation ist die einzige, die eine solche Drill-Funktion anbietet.

### 5.1.5 Laola

Laola vereint die Funktionen von iCoachSoccer und FootballPlan. Es ist möglich Spiele mittels Events aufzuzeichnen und der Trainer kann eine Taktikaufstellung vorbereiten. Das Einzige welches es von den anderen Apps nicht bietet ist eine Drill-Funktion.

## 5.2 Android Markt

Im AndroidMarket wurden vier bzw. zwei relevante Applikationen gefunden. Besonders fällt auf, dass es zu beiden gefundenen Programmen eine Liteversion gibt.

### 5.2.1 Übersicht

Die folgende Tabelle zeigt, welche Eigenschaften und Funktionen die Applikationen haben, die die gleichen Kunden ansprechen.

Tabelle 5.2: Android Market -  
Mitbewerberanalyse

	<b>Mobile Soccer Coach 2</b>	<b>Mobile Soccer Coach 2 Lite</b>	<b>Soccer Pad</b>	<b>Soccer Pad Free</b>
<b>Preis</b>	9.00 Fr	1.00 Fr	2.50 Fr	gratis
<b>Sprachen</b>	Eng	Eng	Eng	Eng
<b>Bewertung</b>	4.1	5.0	-	4.0
<b>Verbreitung</b>	100- 500	100- 500	10-50	1000- 5000
Funktionen				
<b>Spiel aufzeichnen</b>	Ja	Ja	Nein	Nein
<b>Mannschaft erfassen</b>	Ja	Ja	Nein	Nein
<b>Spieler erfassen</b>	Ja	Ja	Nein	Nein
<b>Taktik Darstellung</b>	Ja	Ja	Ja	Ja
<b>Drill-Funktion</b>	Nein	Nein	Nein	Nein
<b>Event-Erfassung</b>	Ja	Ja	Nein	Nein
<b>Notiz-Funktion</b>	Nein	Nein	Nein	Nein
<b>Statistik Funktion</b>	Ja	Ja	Nein	Nein
<b>Abgleich mit GPS Daten</b>	Nein	Nein	Nein	Nein
<b>Export Data</b>	Ja	Nein	Ja (JPEG)	Nein

### 5.2.2 Mobile Soccer Coach und Lite Version

Diese Applikation bietet alles, was ein Trainer sich wünscht: Spielaufzeichnung mittels Events, sowie Taktikaufzeichnung. Beides zwei Kernfunktionen, die eine Applikation für einen Trainer haben sollte. In der Hauptversion besitzt sie zudem die Möglichkeit die Dateien zu exportieren. Die Lite-Version ist im Umfang ein wenig geringer. Von den Funktionen bietet sie mit Ausnahme des Exports, jedoch die selben an.

### 5.2.3 Soccer Pad und Lite Version

Wie einige der bereits untersuchten Ipad Apps ist dieses Tool lediglich ein Taktikdarstellungstool. Als Ergänzung bietet das Hauptapp die Exportfunktion in JPEG-Dateien an. Deshalb erstaunt es, dass die Lite-Version am Verbreitetsten ist. Grund dafür dürfte der Preis sein.

---

## 5.3 Crossplatform

Das einzige Programm, welches auf beiden Märkten zur Verfügung stand, war Mourinho Tactical Board. Dieses ist auf Android-Tablets sowie iPads verfügbar. Ein grosser Unterschied zwischen der IOS- und Android-Version ist der Preis. Beim Android-Tablet beläuft er sich auf CHF 22.15. Beim iPad auf CHF 48.-.

Diese Software wird jedoch in einer Vielfalt von Sprachen angeboten. Begonnen mit Englisch und Deutsch, bis hin zu Koreanisch und Russisch.

Für diesen Preis wäre zu erwarten, dass es alle Funktionen oder wenigstens die meisten realisieren würde. Dies ist jedoch nicht der Fall. Es ist wie viele andere bereits untersuchten lediglich Programm zur Taktikaufstellung und Mannschafts-/Spielerfassung.

---

## 5.4 Fazit

Im Grossen und Ganzen darf festgelegt werden, dass die zwei Hauptfunktionen, die das Programm realisieren soll zu den bestehenden Programmen passt. Dies wären die Funktionen "Spiel aufzeichnen mittels Events" und "Aufstellung um das Spiel vorzubereiten".

Jedoch fehlen Funktionen, die in einer besseren Version eingebaut werden sollten. Zum einen die Notiz-Funktion, welche für den Trainer nützlich ist, zum andern eine Statistik-Funktion, in welcher der Trainer auswerten kann, wer wie viele Tore geschossen hat oder auch wie viele Einwürfe ein bestimmter Spieler ausgeführt hat.

Von der Drill-Funktion wird jedoch abgesehen, da das Ziel der zu erstellenden Applikation im eigentlich Spiel liegt und nicht direkt im Training.

Was das neue Programm speziell auszeichnen wird und uns von den anderen Applikationen abhebt ist die Möglichkeit später die Standortdaten der Spieler mit den GPS Daten abzugleichen.

Im Wesentlichen werden folgenden zwei Technologien für das Projekt *CoachAssist* verwendet. Es handelt sich hierbei um Web-Technologien, unterstützt durch das Framework JQuery Mobile für das Mobile Look and Feel. Und um PhoneGap, welches die Erstellung einer nativ Version ermöglicht. Dazu später mehr.

Für die Entwicklung von JQuery Mobile[7] [14] und PhoneGap[4] [16] wurden Informationen von den jeweiligen Webseiten und dem jQuery Mobile Buch[11] bezogen.

---

## 6.1 JQuery Mobile

Durch das Aufkommen der Smartphones und Tablets wollte das Framework JQuery Projektteam ein Touch optimiertes JavaScript Framework für Webseiten entwickeln. Das Endprodukt war JQuery Mobile. JQuery Mobile baut auf JQuery auf und unterstützt die selben Funktionen wie JQuery. JQuery ist eine mächtige, oft verwendete JavaScript Bibliothek, welche die DOM-Manipulation und -Navigation vereinfacht.

Die Ideologie “write less, do more“ von JQuery wurde auch für JQuery Mobile angewendet. Aus diesem Grund, und weil es auf JQuery basiert, fiel die Wahl auf JQuery Mobile. Obwohl es andere JavaScript Framework für Mobile Webanwendungen vorhanden sind wie JQTouch, LungoJs oder Sencha Touch. Dank JQuery Mobile können Webseiten Smartphones und Tablets freundlich dargestellt werden.

Dazu verleiht es den Webseiten den bekannten Mobile Look and Feel, wie das Slide in Effekt bei Seitenwechsel oder Zurück-Buttons im Header. Obwohl die erste Stable Release von JQuery Mobile erst ein paar Wochen her ist, hat es jetzt schon eine grosse Community. Der erste Stable Release war am 16. November 2011.

---

## 6.2 PhoneGap

PhoneGap ist ein “opensource mobile development framework“, das den Entwicklern ermöglicht plattformübergreifende, sogenannte “hybrid application“ (pseudo nativ Software) für Mobile Geräte zu erstellen, die mit Webtechnologien (JavaScript, HTML5, CSS3) entwickelt wurden. Dabei gewährt PhoneGap den Entwicklern Zugriff auf die Geräte API des entsprechenden Betriebssystems, d.h. die Applikation erhält native Dienste wie zum Beispiel Datenbank, Kompass oder Kamera.

### 6.2.1 Grund für die Nutzung

PhoneGap kommt bei Coach Assist zum Einsatz, da zum jetzigem Zeitpunkt nicht alle Benutzer Internetzugriff auf ihren Tablet haben. Deshalb muss Coach Assist in der Lage sein, lokal auf den Tablets zu laufen.

Ein weiterer Grund für das Einsetzen von PhoneGap ist der Zugriff auf die Datenbank. Durch diesen Service können die Mannschaftsdaten sowie die erfassten Spieldaten und Formation persistent abgelegt werden.

### 6.2.2 PhoneGap vs. Appcelerator Titanium Mobile

Appcelerator Titanium Mobile bietet die gleiche Funktionalitäten wie PhoneGap an. Es dient zu Erstellung einer pseudo nativen Mobile Applikation mit Webtechnologien. Der grundlegender Unterschied besteht darin, dass Titanium Mobile eine eigene umfangreiche API besitzt, das sich gänzlich von den üblichen Web Frameworks unterscheidet. Titanium Mobile kompiliert im Gegensatz zu PhoneGap die Webanwendung in die native Programmiersprache. Das bedeutet, wenn eine Webapplikation auf einem IOS Gerät laufen soll, kompiliert Appcelerator Titanium Mobile die ganze Anwendung in Objective C. Dadurch wird die Performance gesteigert, aber der Nachteil ist das bekannte Problem “Memory Leak“.

Durch die begrenzte Zeitspanne für das Erlernen der Titanium API und durch das Memory Leak Problem, wurde gegen Appcelerator Titanium Mobile und für PhoneGap entschieden.

## **Teil II**

# **Softwaredokumentation**





In diesem Abschnitt wird die Software auf deren Anforderungen analysiert. Dazu wird zuerst eine relevante Persona beschrieben und deren Eigenschaften erläutert. Anschliessend werden alle funktionalen sowie nicht-funktionalen Anforderungen detailliert aufgelistet und mögliche Einschränkungen abgeleitet.

---

## 7.1 Allgemeine Beschreibung

In diesem Abschnitt wird die Software in Bezug auf deren Anforderungen analysiert. Dazu wird zuerst kurz auf das Programm selbst und auf den Endbenutzer eingegangen. Nachher werden die funktionalen und nicht-funktionalen Anforderungen detailliert aufgelistet.

### 7.1.1 Produkt Perspektive

Dieses Produkt dient zur Unterstützung des jetzigen Produktes, nämlich mittels GPS-Sender die Fussballspieler zu erfassen. Es soll für Trainer zugänglich gemacht werden, die nur ein Spieler protokollieren möchten. Dies ist jedoch zweitrangig.

Deshalb ist die Erfassung von Events wie Torschuss oder Einwurf das wesentliche Merkmal der Applikation. Dazu kommt, dass die Standorte der Spieler erfasst werden können, welches für spätere Auswertungen mit den GPS-Daten interessant sein wird.

### 7.1.2 Produkt Funktion

Die folgenden Funktionen werden in unserer Applikation *CoachAssist* realisiert. Hierbei ist zu beachten, dass sie nach Priorität eingestuft sind und wir bei der Realisierung zuerst die wichtigeren Funktionen implementieren werden. Priorität "A" eingestufte Funktionen sind Kernfunktionen. Priorität "B" zusätzliche Funktionen die für einen Trainer von

zentraler Bedeutung sind. Priorität “C“ sind Funktionen die nützlich wären, jedoch vernachlässigbar sind und Priorität “D“ sind für das Design zu beachtenden Funktionen.

- Priorität A
  - Spielaufzeichnung
  - Ereigniserfassung mittels Events
  - Aufstellung mittels vorgegebenen Formationen
  - Mannschafts- und Spielererfassung mittels Server
- Priorität B
  - Notizen allgemein
  - Spielunterbrechung
  - Verbindung zum Server (Daten hoch- und herunterladen)
  - Aufstellung selbst definieren
  - Mannschafts- und Spielererfassung auf dem Tablet inkl. anschliessendem Hochladen neuer Spieler
  - Lineup während dem Spiel ändern
  - Löschen und Editieren von erfassten Events
- Priorität C
  - Notizen mittels Ton
  - Statistikanzeige
  - Einbinden der Onlineplattform ([www.cnlab.ch/fussball](http://www.cnlab.ch/fussball))
  - Matchaufzeichnung abspielen
- Priorität D
  - Trikotfarbe anpassen
  - Optimieren für links-/rechtshändige Benutzer

### **7.1.3 Einschränkungen**

Die wichtigste Einschränkung besteht dahingehend, dass die Daten zum Server hochgeladen werden. Dort werden sie vorerst nicht weiterverarbeitet. Es werden lediglich Daten aufgezeichnet und dann weiter versandt.

Eine weitere Einschränkung besteht darin, dass die Funktionen nach Prioritäten in das Programm entwickelt werden und, je nachdem wird aus Zeitgründen, auf Funktionen einer niedrigeren Stufe verzichtet.

#### 7.1.4 Benutzer Charakteristik

Die Software *CoachAssist* wird zukünftig von Trainern und Mitarbeitenden von der Firma “cnlab“ verwendet. Dafür wurde folgende fiktive Persona eines Trainers erstellt.

- Carlos Coach
- 48 Jahre
- Fussballtrainer des FC Sargans
- Softwareentwickler bei K-Soft
- Verheiratet, Vater von zwei Kindern
- Technikfan

Carlos ist Trainer des Fussballvereins einer Fünft-Ligamannschaft namens FC Sargans. Er macht dies schon fast 10 Jahre lang. Er ist 48 Jahre alt. Fussballtrainer ist er nur nebenberuflich. Sein Hauptberuf ist Softwareentwickler. In diesem Job arbeitet er zu 70 Prozent.

Das Fussball Training findet meistens Abends statt. Jedoch muss er auch manches Wochenende opfern um an Turnierspielen teilzunehmen. Während des Spiels erstellt er öfters Notizen, um dann im Training auf das Spiel einzugehen.

Softwareentwickler ist er in einer kleinen Firma namens K-Soft. Dort entwickelt er unter anderem Webseiten, aber auch mobile Applikationen.

Neben diesen zwei Berufen ist Carlos aber auch ein guter Familienvater. Er hat zwei Kinder, ein Mädchen und einen Jungen. Er nimmt seine zwei Kinder auch öfters zu den Fussballspielen mit.

Sein Sohn spielt selbst sehr gerne Fussball und möchte später in der Mannschaft seines Vaters spielen.

Weil Carlos Informatiker ist und “kleine Spielsachen“ liebt, hat er vom PC bis zum iPhone jedes Gerät. Unter anderem hat er auch ein Galaxy Pad, welches er fast immer dabei hat. Er verwendet es für seinen Informatikberuf wie auch in der Freizeit. So nutzt er es auch um mit dem Standard Notiz App Notizen zu den Fussballspielen zu machen.

---

## 7.2 Funktionale Anforderungen

Die funktionalen Anforderungen beschränken sich auf das Aufzeichnen eines Fussballspieles. Dabei soll es möglich sein, verschiedene Events (z. B. Tor oder Eckball) erfassen zu können. Dies soll für die eigene sowie auch für die gegnerische Mannschaften möglich sein. Dafür müssen natürlich die Mannschaften vorhanden sein. Zusätzlich soll der Trainer Notizen erstellen können und es soll möglich sein, eine Aufstellung für die eigene Mannschaft zu erfassen.

Mehr zu den funktionalen Anforderungen sind aus den folgenden Use-Cases ersichtlich.

### 7.2.1 UseCase Übersicht

Das folgende Diagramm zeigt die UseCases:

Abbildung 7.1: UseCase Diagramm



Nun folgt eine Auflistung der UseCases in der “Brief“-Form.

### 7.2.2 Mannschaft erstellen

Der Benutzer möchte seine Mannschaft im System erfassen. Dafür trägt er die nötigen Informationen im System ein und erstellt die Mannschaft.

### 7.2.3 Spieler hinzufügen

Der Benutzer hat seine Mannschaft erfolgreich erfasst und möchte nun die einzelnen Spieler der Mannschaft erfassen. Dafür füllt er die Pflichtfelder Name, Trikotnummer, Passnummer und sonstige Informationen zur Person. Nachher klickt er auf erstellen und der Spieler wird der Mannschaft hinzugefügt.

### 7.2.4 Daten herunterladen

Es werden Informationen vom Server benötigt, wie zum Beispiel eine Mannschaft mit den dazugehörigen Spieler. Dazu geht er ins Download-Menü und gibt da die Zugangsdaten zum Server ein. Sobald die Daten im System sind muss der Benutzer nur noch Download klicken und die Daten werden vom Server in das System übernommen.

#### **7.2.5 Daten hochladen**

Der Benutzer hat soeben ein Spiel aufgezeichnet. Deshalb schliesst er sein Tablet ans Internet an und lädt das von ihm gewünschte Spiel hoch.

#### **7.2.6 Spiel aufzeichnen**

Der Trainer möchte ein Spiel der eigenen Mannschaft aufzeichnen um möglichst viele Informationen zu erfassen. Dafür füllt er im Game-Menü die Machtinformationen ab und klickt auf “Weiter“. Daraufhin öffnet sich das Spielfenster und der Benutzer klickt auf Spiel starten. Daraufhin werden die Event-Buttons aktiviert und der Benutzer kann das Spiel nun aufzeichnen.

#### **7.2.7 Event hinzufügen**

Das Spiel ist in vollem Gange. Der Benutzer erfasst nach und nach die Events, wie sie eintreffen. Dies geschieht durch klicken auf den Ereignis Button.

#### **7.2.8 Spiel abspielen**

Der Trainer möchte ein Tag nach dem Spiel nochmal anschauen, wer wann welches Tor gemacht hat. Dafür öffnet er das Menü “Aufgezeichnete Spiele“ und wählt das gestrige Spiel aus. Danach kann er die Events in einer Liste betrachten.

#### **7.2.9 Aufstellung generieren**

Der Trainer möchte für das morgige Spiel eine Aufstellung aufbauen. Falls er noch keine Mannschaft und Spieler Daten auf dem System hat, muss er zuerst die Daten herunterladen. Danach geht er in das Formations-Menü und kann da zwischen definierten Aufstellungen auswählen. Nach der Auswahl verteilt er die Spieler auf dem Feld und der Ersatzbank.

---

### **7.3 Nicht funktionale Anforderungen**

Die nicht funktionalen Anforderungen wurden nach sechs Hauptkriterien des ISO 9126 Schema aufgebaut und dokumentiert.

#### **7.3.1 Zuverlässigkeit**

Die Zuverlässigkeit ist wichtig, da die Daten live aufgezeichnet werden und es dabei keine Datenverluste geben darf. Der Benutzer muss sich 100-prozentig darauf verlassen können, dass die Eingaben, die er tätigt auch gespeichert werden.

### **7.3.2 Benutzbarkeit**

Die Software soll auch von Laienawendern benutzt werden können. Deshalb ist hier die Benutzbarkeit bzw. die Bedienbarkeit wichtig. Der Benutzer soll sich möglichst einfach zu recht finden.

Dadurch, dass das App mit PhoneGap in ein natives App umgewandelt werden kann, ist die Bereitstellung ebenfalls simpel. Dafür werden die Stores verwendet, die bei den Geräten bereits vorhanden sind. Somit kann es jeder Benutzer egal welcher Tabletfraktion er angehört die Software problemlos anwenden.

### **7.3.3 Effizienz**

Die Leistung ist dahingehend wichtig, dass der Benutzer nicht lange warten muss, bis er Daten hoch- oder herunter geladen hat. Zweitens sollte die Bedienung ohne Probleme und, ohne grossen Zeitverlust durch Codeablauf im Hintergrund möglich sein.

Des Weiteren sollte auch beim Abbruch eines Spieles schnellstmöglich wieder ein neues Spiel erstellt werden können.

### **7.3.4 Änderbarkeit**

Hier liegt speziell der Punkt "Wartbarkeit" im Zentrum. Der Code soll möglichst einfach erweitert werden können. Deshalb ist es notwendig, dass der bestehende Code verständlich und es somit möglich ist, ihn beliebig zu erweitern.

### **7.3.5 Übertragbarkeit**

Die Übertragbarkeit ist in zwei Punkten von zentraler Bedeutung. Zum einen, dass sich die Software auf möglichst viele Tablettypen übertragen lässt. Dies sollte mit der gewählten Technologie JQuery Mobile und PhoneGap möglich sein.

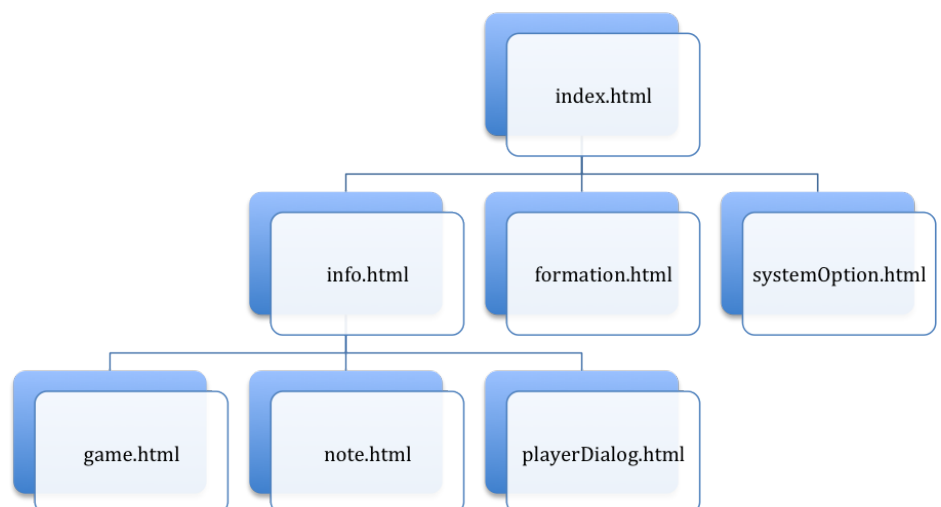
Zum andern sollen die Daten vom Server zum Tablet und wieder zurück möglichst einfach übertragen werden.

In diesem Kapitel wird die Software in Bezug auf deren Domainlogik analysiert. Der Fokus liegt dabei auf dem Datenmodell und den komplizierten Vorgängen, die in Form von System Sequenz Diagrammen beschrieben werden.

## 8.1 Aufbau

Der Aufbau spiegelt in diesem Projekt wieder, was bei andern Projekte eine Domain Analyse ist. Auf diese wurde jedoch verzichtet, da keine richtigen Klassen vorhanden sind. Dies lässt sich damit erklären, da die Software mittels Webtechnologien erstellt wurde. Jedoch beschreibt dieser Abschnitt wie die Applikation bzw. die Webseite aufgebaut ist. Das folgende Diagramm zeigt den Seitenaufbau.

Abbildung 8.1:  
Programmstruktur



### **8.1.1 index.html**

Dieser Teil besteht aus dem Startmenü bzw. dem Eingangspunkt der Applikation. Von hier aus besteht die Möglichkeit für den User, wie im Diagramm ersichtlich ist, zu “info.html“, “formation.html“ und zu “optionSystem.html“ zu wechseln.

### **8.1.2 info.html**

Die Seite info.html ist der Zugang zur Spielerfassung. Hier müssen zuerst alle Spiel relevanten Informationen eingegeben werden, dass sie dann in der Datenbank gespeichert werden können. Sobald dies geschehen ist, werden die drei Fenster game, note und formation in der Navigationsleiste freigeschaltet. Zusätzlich wird automatisch auf “game“ gewechselt.

### **8.1.3 game.html**

Dieses Seite ist das Kernstück der Spielerfassung. Sobald das Spiel gestartet wird, beginnt die Uhr zu laufen. Der Benutzer hat von hier die Möglichkeit die verschiedenen Ereignisse zu erfassen. Die Events werden schlussendlich in einer Liste dargestellt. Die Liste funktioniert genau so wie ein Liveticker.

### **8.1.4 note.html**

Diese Seite stellt lediglich ein zusätzliches Event dar. Dieser Event macht zur Spielzeit einen Eintrag in der Eventliste mit der vom Benutzer eingegebenen Notiz.

### **8.1.5 playerDialog.html**

Dieser Dialog erscheint, falls der Benutzer auf ein Event klickt. Falls Spieler in der Datenbank vorhanden sind, werden diese zur Auswahl bereitgestellt. So hat der Benutzer die Möglichkeit Events auf die Spieler zu beziehen.

### **8.1.6 formation.html**

Diese Seite ermöglicht dem Benutzer zwischen verschiedenen Formationen zu wählen. Die Spieler, welche sich in der Datenbank befinden, können dann auf die Positionen gezogen werden. Somit kann der Benutzer eine taktische Aufstellung vollziehen.

### **8.1.7 optionSystem.html**

Auf dieser Seite werden die Einstellungen gespeichert. Zusätzlich hat es hier einen Menüpunkt der das Senden und Empfangen der Daten vom Server regelt. Der Benutzer muss dafür lediglich ein Authentifizierungstoken eingeben und kann die Spieler und Teams vom Server herunterladen. Diese Daten hat er zuvor auf dem Server erfasst.



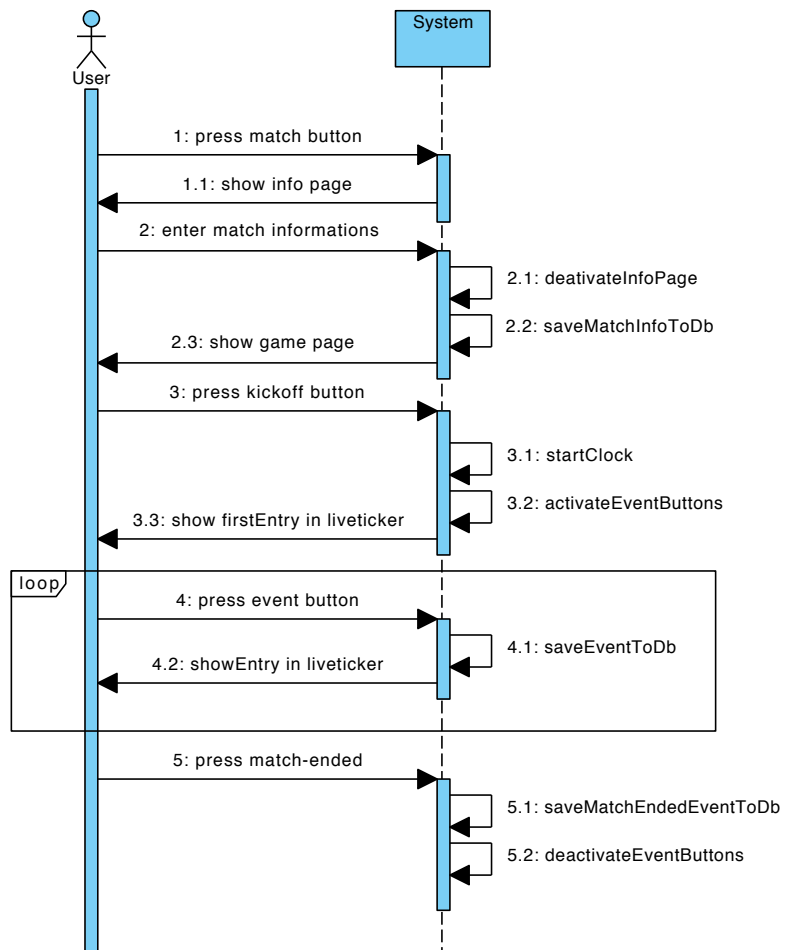
## 8.2 Systemsequenzdiagramme

Die folgenden Diagramme zeigen die wichtigsten Abläufe des Systems.

### 8.2.1 Match

Dieses SSD zeigt wie ein Match gestartet wird und was während eines Matches abläuft bzw. was geschieht wenn man ein Event erfasst.

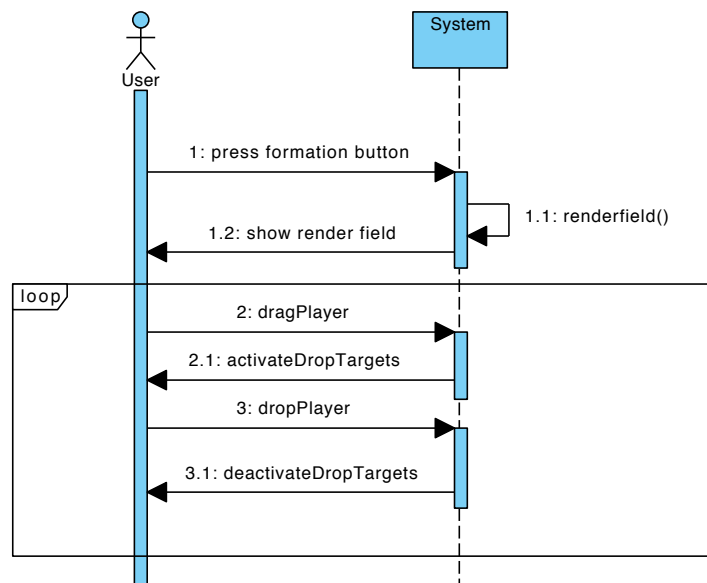
Abbildung 8.2: SSD - Start  
eines Matches



### 8.2.2 Formation - Drag and Drop

Das folgende Diagramm zeigt, was geschieht, wenn ein Player-Target angeklickt wird. Zusätzlich zeigt das Diagramm auch, durch welche Regelung das Player-Target auf dem Bildschirm positioniert werden kann.

Abbildung 8.3: SSD -  
Formation Drag and Drop



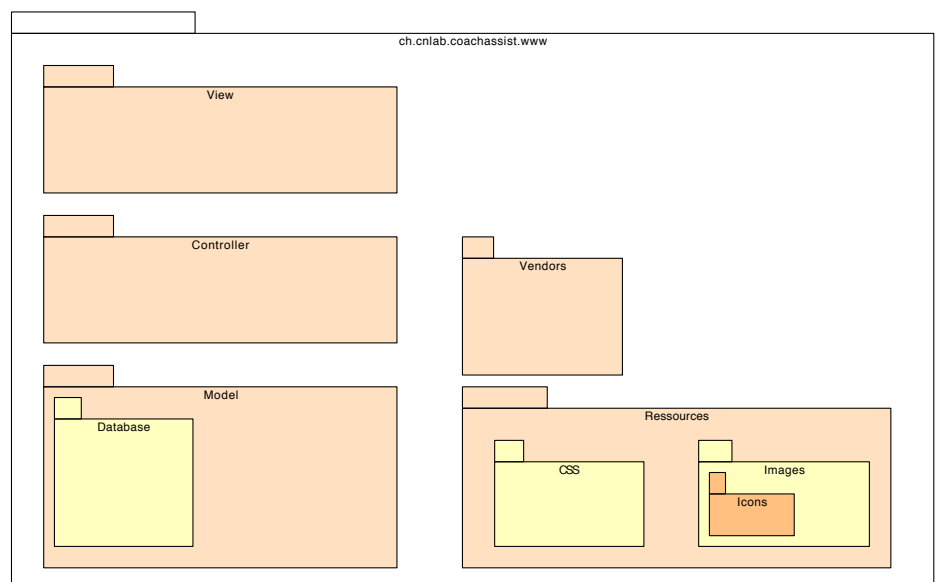
## 9.1 Logische Architektur

### 9.1.1 Package Diagramm

Die Package Struktur ist einfach gehalten, da es in den Grundzügen eine Web Applikation ist. Als erstes werden die Packages der reinen Applikation dargestellt. Das zweite Package-Diagramm zeigt, wie die Applikation mit dem PhoneGap Framework zusammen aufgebaut ist. Darauf wird im weiteren nochmals konzentriert eingegangen.

Auf dem Diagramm ist die Aufteilung vom Package “View“ und dem

Abbildung 9.1: Package Diagramm WWW



Package “Ressources“ ersichtlich. Zuerst zum Hauptpackage “www“. In diesem Package befindet sich die ganze Applikation. Dabei handelt es sich um eine lauffähige Webapplikation, die problemlos auch als normale Webseite benutzt werden könnte.

Das Package “View“ enthält die Elemente, die der Endnutzer schlussendlich zu sehen kriegt. Dabei handelt es sich um .html-Files. Speziell zu erwähnen ist, dass sich in den html-Dateien ausschliesslich html-Code verbirgt. Jegliche Ajax/Javascripts wurden in die Packages “Controller“ und “Model“ ausgelagert. Die gesamten “Styles“ wurden ebenfalls ausgelagert, sie befinden sich im Package “Ressources“.

Das Package “Ressources“ enthält neben den Styles Bilder und Icons. Mit Styles sind die css-Files gemeint. In diesen Files sind die Darstellung der Ansichtselemente definiert, dabei handelt es sich um Grössen, Farben und weitere Eigenschaften. Das Unterpackage “Images“ enthält ergänzend ein Package “Icons“. Für Icons wurde deshalb ein spezielle Untergruppe erstellt, da viele in der Applikation vorkommen.

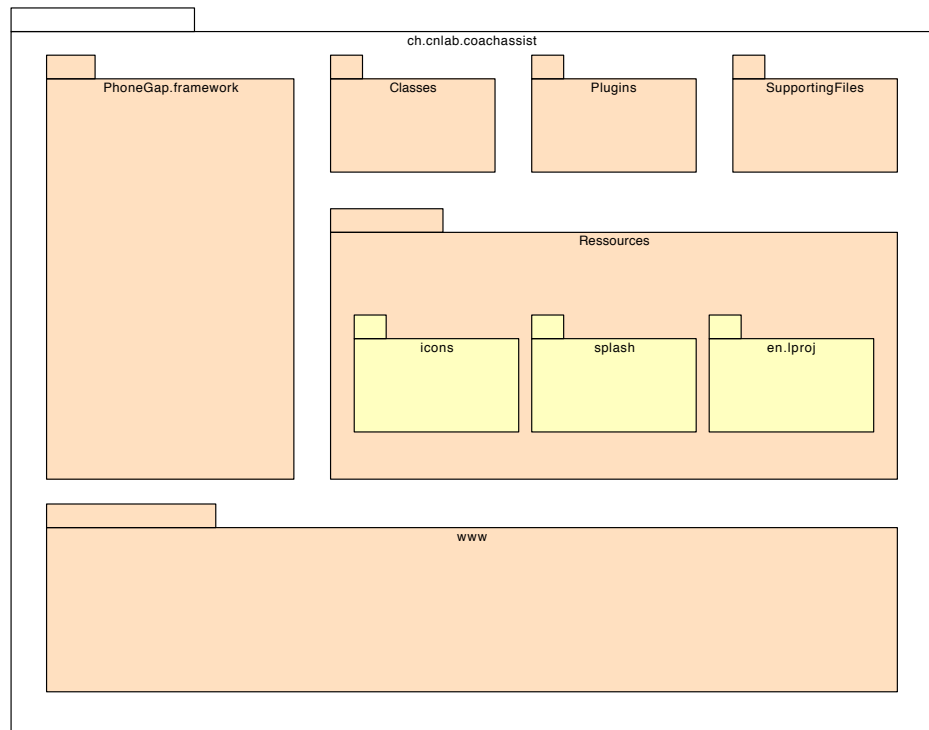
Im Package “Controller“ sind die Javascript-Files die von der View benachrichtigt werden. Diese holen dann die benötigten Informationen und teilen diese wieder der View mit.

Das Package “Model“ enthält bei der vorliegenden Applikation hauptsächlich die Datenbank, da im Hintergrund wenig Logik vorhanden ist. Die Daten werden hier bereitgestellt und dem Controller übergeben.

Das letzte Package namens “Vendors“ beinhaltet 3rd Party Software (“Libraries“), die vom Controller und Model verwendet werden.

Das zweite Packagediagramm zeigt den Aufbau mit dem PhoneGap Framework. Es ist erkennbar das die Webseite ein geschlossener Bereich ist, der von Framework aufgerufen und verarbeitet wird.

Abbildung 9.2: Package Diagramm PhoneGap



### 9.1.2 Verwendete Patters

In dem Programm wurde auf ein Pattern besonders geachtet. Es lautet MVC (Model View Controller). Dies ist aus dem ersten Package-diagramm deutlich zu erkennen. Die Views sind die html Dateien. Die Javascript-Dateien sind je nach Funktion dem Controller oder Model teil zuzuordnen.

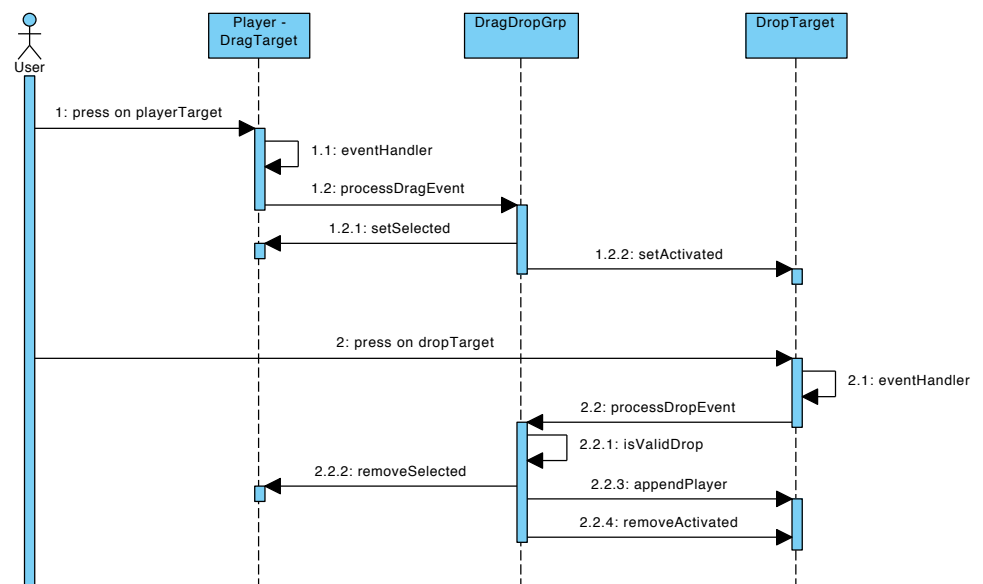
## 9.2 Systemoperationen

Die wichtigsten Operationen wurden in Systemdiagrammen dargestellt und werden in diesem Abschnitt aufgeführt.

### 9.2.1 Drag and Drop a Player

Im Analyse Kapitel wurde bereits auf den ganzen DragAndDrop-Vorgang eingegangen. Das folgende Diagramm zeigt, wie DragAndDrop im System funktioniert.

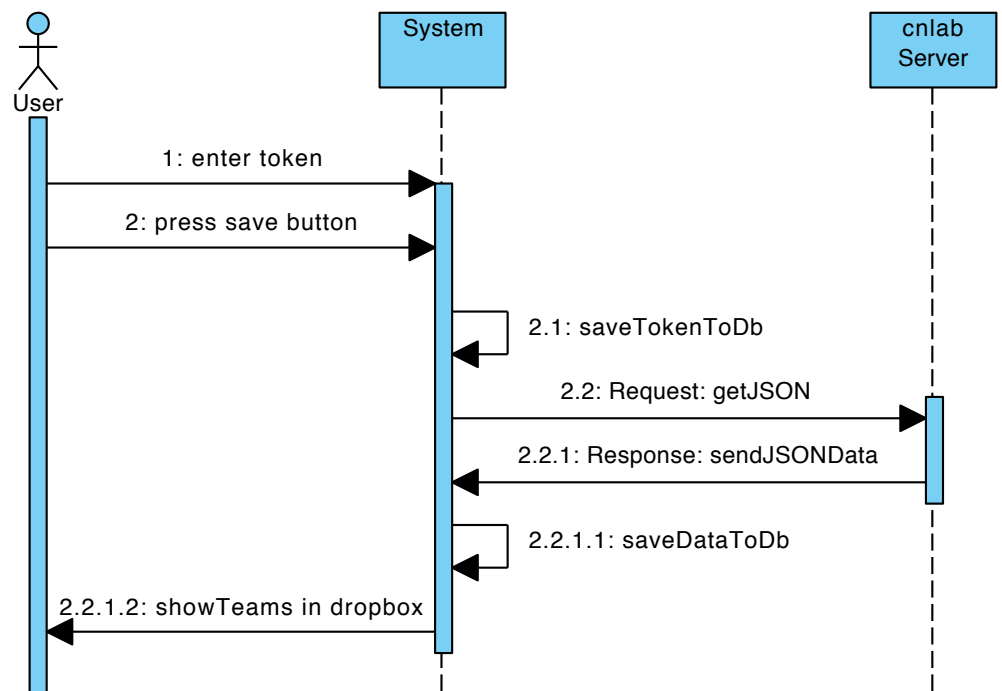
Abbildung 9.3: SSD -  
Formation Drag and Drop



## 9.2.2 Daten senden und empfangen

Der nächste Ablauf zeigt das Downloaden von Daten.

Abbildung 9.4: JSON -  
Download von Daten



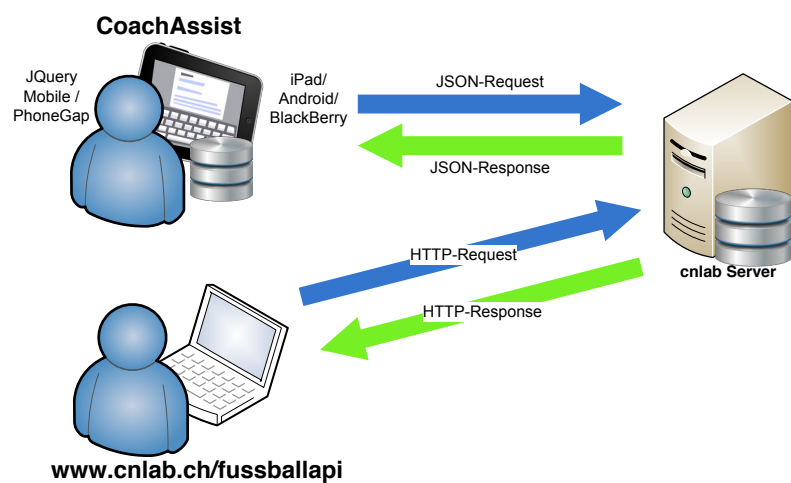
Daten senden erfolgt genau gleich, ausser das ein Post anstatt eines Get-Request gesendet wird.

## 9.3 Kommunikation Client und Server

Die Kommunikation funktioniert grundsätzlich von Tablet zum Server über das Internet. Natürlich sind noch diverse andere Komponenten im Spiel, wie zum Beispiel eine Firewall. Aus Gründen der Übersichtlichkeit wird jedoch darauf verzichtet. Zusätzlich sieht man die Verbindung vom Computer zum cnlab-Server. Die Spieler und Mannschaften werden auf dem Computer via Browser erfasst und auf dem Server abgelegt. Von da an, stehen die Daten auf dem Tablet zur Verfügung, wenn er sich mit seinem Benutzer einloggt.

### 9.3.1 JSON

Abbildung 9.5: JSON - Request/Response-Diagramm



Das Diagramm zeigt, wie das Tablet mit dem Server kommuniziert. Das Tablet sendet einen JSON Request an den Server und bekommt daraufhin eine Antwort. Je nach Anfrage handelt es sich dabei um Daten die dann von der Tablet-Anwendung verwendet werden. Die Request-Befehle sind im nächsten Abschnitt "Server-API" definiert. Bei der Applikation dreht es sich um eine reine Request/Response Beziehung handelt.

### 9.3.2 Server-API

Um mit dem Server kommunizieren zu können sind spezielle Befehle nötig. Es sind folgende Request-Befehle definiert:

- GET \fussballapi\teams(.:format)
- POST \fussballapi\teams(.:format)
- GET \fussballapi\teams\new(.:format)
- GET \fussballapi\teams\.:id\edit(.:format)

- GET \fussballapi\teams\id(.:format)
- PUT \fussballapi\teams\id(.:format)
- DELETE \fussballapi\teams\id(.:format)

Der erste GET Befehl für ein Team gibt folgende Response zurück:

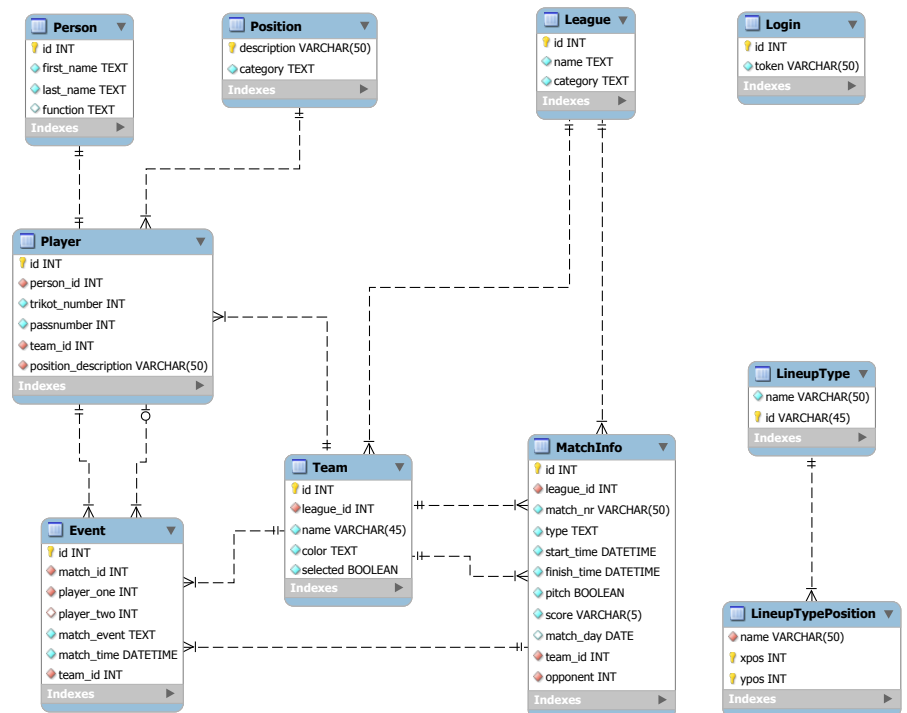
```
1 {  
2   "created_at": "2011-12-15T11:43:54Z",  
3   "default_color": "Green",  
4   "id": 3,  
5   "league_id": 1,  
6   "name": "FC St.Gallen",  
7   "sponsor_url": "",  
8   "team_url": "",  
9   "updated_at": "2011-12-15T11:43:54Z",  
10  "memberships": [ ],  
11  "matches": [ ]  
12 }
```

Die Response ist mittels JSON formatiert. Dies ermöglicht eine einfache Verarbeitung der empfangenen Daten. Natürlich erleichtert es auch das Senden von Daten. Dies war lediglich ein Ausschnitt der Api. Die gesamte Api befindet sich bei den Zusatzdokumenten.



## 9.4 Datenmodell

Das folgende Diagramm zeigt die Datenbank auf dem Tablet. Es handelt sich dabei um eine SQLite Datenbank, die mittels Javascript realisiert wurde.



### 9.4.1 Login

Die Tabelle “Login“ dient ausschliesslich zur Serverauthentifizierung. Der Benutzer der Applikation kann später den Token via GUI eingeben und erhält dann den Zugriff auf die Serverdaten.

### 9.4.2 Team

Die Tabelle “Team“ dient zur Gruppierung in Teams. Sie enthält ebenfalls eine Zugehörigkeit zu einer Liga sowie zu einem Verein (Club). Das Flag “selected“ dient zur Kennzeichnung, welches im Moment das Team des Benutzers ist.

### 9.4.3 League

Die Tabelle “League“ dient der Zuordnung eines Teams und auch eines Games. Es kann Spiele geben, bei denen Mannschaften aus verschiedenen Ligen gegeneinander antreten, jedoch das Game ganz einer anderen Liga angehört.

#### 9.4.4 Person

In der Tabelle “Person“ sind Personen mit Namen und Funktion abgelegt.

#### 9.4.5 Player

In der Tabelle “Player“ werden die Spieler der Mannschaft aufgelistet. Jeder Spieler gehört zu einer Person. Sie verfügt über ein Team sowie auch einer Spielposition. Zusätzlich hat sie noch Attribute, die ihren Typ genauer beschreiben, wie die Trikot-Nummer.

#### 9.4.6 Position

Die Tabelle “Position“ dient der Zuordnung eines Spieler zu seiner Position, die er im Spiel einnimmt.

#### 9.4.7 Game

Die Tabelle “Game“ beschreibt die verschiedenen Matches, die aufgezeichnet wurden. Jedes offizielle Spiel hat eine Spielnummer, welche dann im Attribut “match\_nr“ eingetragen werden kann. Zusätzlich enthält sie die Anfangs- und Endzeiten eines Spieles. Die restlichen Attributfelder beschreiben lediglich Eigenschaften eines Matches.

#### 9.4.8 GameEvent

In der Tabelle “GameEvent“ werden die verschiedenen Events eines Spiels aufgezeichnet. Da es sich hierbei um eine grosse Vielfalt von Events handelt, wird lediglich kurz auf diese eingegangen sowie auch welche Datenfelder dafür benötigt werden.

Die einfachsten Events sind Spielbeginn, Spielpause etc., diese benötigen ausschliesslich einen Timestamp und das dazugehörige Game. Diese Attribute werden von allen andern Events auch benötigt. Die Events “red- and yellow card“, “throw-in“, “offside“, “corner“, “penalty“ und “freekick“ sind Standartevents, welche jeweils einen Spieler benötigen, der den Event verursacht hat. Die zwei Events “substitution“ und “goal shot“ benötigen jeweils zwei Spieler. Bei Auswechsel-Events werden der das Spiel verlassende Spieler und der Spieler der das Feld betritt, aufgezeichnet. Beim Torschuss-Event werden der Torschütze und der Passgeber eingetragen. Der Passgeber ist optional.

Ein spezielles Event ist das “note“-Event. Der Benutzer hat die Möglichkeit eine Notiz zum Spielgeschehen hinzuzufügen. Dafür wird das Attribut “text“ benötigt.

Das zweite spezielle Event lautet “lineup change“. Dieser tritt ein, wenn während dem Spiel eine Aufstellungsänderung stattfindet. Dabei wird das geänderte Lineup gespeichert. Dies wird zudem zu Beginn des Spieles gespeichert. Es handelt sich dabei um die Startaufstellung.

Abschliessend ist das Attributfeld “kind“ zu benennen, welcher definiert, um welches der zuvor genannten Events es sich handelt.

#### **9.4.9 LineupType**

Die Tabelle “LineupType“ beschreibt die verschiedenen Formationsmöglichkeiten.

#### **9.4.10 LineupTypePosition**

Die Tabelle “LineupTypePosition“ definiert die genauen Positionen auf einer 11x11 Matrix für einen bestimmten LineupTyp. Dies wird in x- und y- Koordinaten aufgeteilt.



---

## 10.1 Codeguidelines

Zuerst einmal ist zu erwähnen, dass bei der Programmierung an die folgenden Code Guidelines orientiert wurde:

- HTML:  
<http://www.w3.org/TR/WCAG10-HTML-TECHS/>
- JavaScript:  
<http://google-styleguide.googlecode.com/svn/trunk/javascriptguide.xml>
- JQuery:  
[http://docs.jquery.com/JQuery\\_Core\\_Style\\_Guidelines](http://docs.jquery.com/JQuery_Core_Style_Guidelines)
- JQuery Widget:  
<http://wiki.jqueryui.com/w/page/12138135/Widget%20factory>

### 10.1.1 HTML

Coach Assist wurde mit dem neusten HTML Norm dem HTML 5 entwickelt. Bei der Entwicklung von *CoachAssist* wurde darauf Wert gelegt, dass die im neuen Standard eingeführten Tags und Verbesserungen voll umfänglich benutzt werden können.

Erwähnenswert ist die Tatsache, dass JQuery Mobile ein Fehlerpotenzial in der Bedienung hat. Es lädt die externen Seiten (separates HTML File) bei den Aufrufen dynamisch und bindet diese im bestehenden DOM ein. Deshalb benötigt nur der index.html bzw. die erste HTML Seite den HEAD und BODY Tag. Auf den weiteren Seiten ignoriert JQuery Mobile die HEAD und BODY Tags. Aufgrund dieser Tatsache

müssen die externen JavaScript und CSS Files bereits im index.html festgelegt werden. Die restlichen HTML Dateien können mit dem Section Tag beginnen.

Erste HTML Seite:

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <meta charset="UTF-8" />
5     <link rel="stylesheet" href="css/jquery.mobile
      -1.0.css" />
6     <script src="js/jquery-1.7.js"></script>
7     <script src="js/jquery.mobile-1.0.js"></script>
8   </head>
9   <body>
10    <section id="indexPage" data-role="page">
11      <header data-role="header"></header>
12      <div data-role="content" class="content"></div>
13      <footer data-role="footer">
14        <h3>&copy; 2011 by cnlab</h3>
15      </footer>
16    </section>
17  </body>
18 </html>
```

Weitere HTML Seiten:

```
1 <section data-role="page">
2   <header data-role="header">
3   </header>
4   <div data-role="content" class="content"></div>
5   <footer data-role="footer">
6     <p>
7       &copy; 2011 by cnlab
8     </p>
9   </footer>
10 </section>
```

### 10.1.2 JavaScript

Wie bereits erwähnt, werden wegen JQuery Mobile die separaten HTML Files dynamisch bei ihren Aufrufen im bestehenden DOM eingebunden. Deshalb müssen die JavaScript Files, welche erst beim Aufruf einer bestimmten externen HTML File ausgeführt werden, in einer “pageinit“ Event Funktion, welche beim Initialisieren der Seite ausgelöst wird, gepackt werden.

```
1 $(document).delegate("#indexPage", "pageinit",
2   function() {
3   });
```

Beim laden bzw. erstellen jedes HTML Files wurde eine Third-Party Funktion ResizePageContentHeight() ausgeführt. Aufruf der Funktion beim erstellen einer externen Seite:

```

1 $(document).delegate('#optionPage', 'pagecreate',
    function() {
2     ResizePageContentHeight($(this));
3 });

```

Die ResizePageContentHeight() Funktion aus scrollview.js

```

1 function ResizePageContentHeight(page) {;
2     var $page = $(page),
3         $content = $page.children( ".ui-content" ),
4         hh = $page.children( ".ui-header" ).outerHeight()
5             || 0,
6         fh = $page.children( ".ui-footer" ).outerHeight()
7             || 0,
8         pt = parseFloat($content.css( "padding-top" )),
9         pb = parseFloat($content.css( "padding-bottom" )),
10        wh = window.innerHeight;
11    $content.height(wh - (hh + fh) - (pt + pb));

```

Es wurden einige Verbesserungen von JQuery Mobile implementiert.  
Verflüssigung der Transition: <http://jquerymobile.com/test/docs/pages/touchoverflow.html> :

```

1 $.mobile.touchOverflowEnabled = true;

```

Fixierung des Headers und Footers: <http://jquerymobile.com/test/docs/toolbars/bars-fixed.html> :

```

1 $.mobile.fixedToolbars.setTouchToggleEnabled(false);

```

Wegen Problemen mit diversen 3rd Party Libraries (Vendors) wurden in diesen kleine Änderungen vorgenommen, die es zu beachten gilt.

- jquery.mobiledraganddrop.js
  - Touchevents wurden geändert, dass sie mit den Tablets reibungslos funktionieren
  - “active“-Klassen werden am Schluss jedes Drop-Events komplett gelöscht
- jquery.mobile.scrollview.js
  - Default Ereignisbehandlung (e.preventDefault()) wurde auskommentiert, wegen Problemen mit “mobiledraganddrop“
  - Complete-Content-Scrolling wurde entfernt, zur Vermeidung von komplettes scrollen auf dem Tablet

---

## 10.2 Designguidelines

Beim Design wurde das Framework JQuery Mobile für den default Look and Feel Eindruck verwendet. Es ist den Mobile Anwendungsnormen entsprechend entworfen. Alle Seiten haben einen Header und einen Footer für die Navigation, wie es für ein Tabletanwendung üblich ist. Bei jedem Header hat man die Möglichkeit, zum Hauptmenü zurück zu kehren.

Die Strukturierung der Eingabefelder sind durchgehend konsistent gehalten sowie die Darstellung der Seiten. Auch die Farbwahl ist auf ein Minimum beschränkt um eine einfach Erscheinung der Anwendung zu erzeugen. Alle Buttons und Eingabefelder müssen grösser dargestellt, damit sie besser und einfach getroffen werden.



In diesem Kapitel sind die Tests aufgeführt, die durchgeführt wurden. Hierbei ist anzumerken, dass die Tests ausschliesslich mit dem iPad 1 und der iPad SDK mit der iOS Version 5 statt gefunden haben.

## 11.1 Systemtests

Die folgende Tabelle zeigt auf, welche Use Cases erfolgreich umgesetzt wurden und Beschreibt sie.

*Tabelle 11.1: Systemtests -  
Auswertung Use Cases*

Use Case	Resultat	Beschreibung
Mannschaft erfassen	Ja	Die Mannschaft wird auf dem Server erfasst
Spieler hinzufügen	Ja	Die Spieler werden ebenfalls auf dem Server erfasst
Daten herunterladen	Ja	Die Mannschafts- und Spielerdaten werden in das System heruntergeladen
Daten hochladen	Ja	Erfolgreich Aufgezeichnete Spiele werden zum Server hochgeladen
Spiel aufzeichnen	Ja	Erfassen eines Spieles ist möglich
Event erfassen	Ja	Verschiedene Events wurden realisiert
Spiel anschauen	Nein	Dieser UseCase wurde aus Zeitgründen nicht umgesetzt
Aufstellung generieren	Ja	Aufstellungen werden mit vorgegebenen Formationen generiert

---

## 11.2 Usability Tests

Wir haben zwei Arten an Usability Tests durchgeführt. Der erste Test am Anfang des Projektes mit einem Paper Prototyp und der zweite Test mit der GUI der Applikation.

### 11.2.1 PaperPrototyp

Während dem Projekt haben wir drei verschiedene Varianten eines Paper Prototyps erstellt. Der erste war ein provisorische Entwurf, der in einer Sitzung mit dem Kunden überarbeitet wurde. Daraufhin entstand eine zweite Version. Dieser enthielt wesentlich mehr Informationen. Nach einer weiteren Sitzung gab es ein paar weitere kleine Änderungen. Schlussendlich entstand die dritte Version. Diese Version wurde diversen Testpersonen vorgelegt, die sich durch die “Applikation“ durchgeklickt haben. Bei diesen Tests sind uns speziell folgende GUI-Fehler aufgefallen:

- Die Fläche der Ereignisbuttons sind nach ihrer Wichtigkeit angepasst und sortiert.
- Scrollen wenn möglich vermeiden
- Touch-Targets gross genug machen (Fitt’s Law)
- Anzahl Touches auf ein Minimum reduzieren. Event-Buttons jeweils für beide Mannschaft darstellen.

Das waren die grösseren Änderungen die wir dann schlussendlich auf die End-GUI übernommen haben.

Die dritte PaperPrototyp-Version ist im Anhang unter UserInterfaces zu finden.

### 11.2.2 GUI Betatest

Der GUI Test hat zu weiteren kleinen Änderungen geführt. Zum einen wurde der Home-Button um einiges vergrössert. Zusätzlich wurden die Buttons in der Header-Bar unterschiedlich gross dargestellt. Spielunterbrechen wurde bewusst passiver dargestellt als der Spielbeginn/Spielende Button.

---

## 11.3 Unit Tests

Für das Testen von JQuery bzw. JavaScript existiert QUnit ein Unit Tester von JQuery Team. Es ist ein extrem mächtiges und einfach erlernbarer Unit Tester, das in der Lage ist auch serverseitigen JavaScript Code zu testen. Aus Zeitgründen und des späten Entdecken von QUnit wurde auf die Unit Test verzichtet und auf Usability und System Tests begrenzt. Für das weiter Entwickeln der Software wird empfohlen die JavaScripts mit QUnit zu testen.



---

### 12.1 Allgemein

Leider haben wir nicht alle Grundfunktionen so implementiert wie wir es wollten. Grund dafür war mangelnde Erfahrung mit der Technologie. Im Groben wurden alle Funktionen realisiert, jedoch nicht vollständig. Eine Funktion die wir gerne implementiert hätten, ist: ein Event mit dem dazugehörigen Spieler zu erfassen. Jedoch gab es Probleme mit dem JQuery Mobile Pop-up. Beim Beenden des Pop-ups wurde das Spiel aus ungeklärten Gründen immer beendet. Dieses Problem kostete uns sehr viel Zeit, weshalb wir uns entschieden haben, die Events im Moment nur über die Mannschaft zu erfassen. Sonst wurden alle Grundfunktionen wie gewollt implementiert.

Leider sind wir nicht mehr dazu gekommen Zusatzfeatures wie eine eigene Formation zu erstellen oder Mehrsprachigkeit umzusetzen. Ein wichtiges Features, welches leider auch noch fehlt, ist das löschen und editieren von Events.

#### 12.1.1 Ausblick

Mit den erreichten Funktionen ist es möglich, ein Fussballspiel zu protokollieren. Leider könnten Fussballspiele nicht mit allen Details erfasst werden. Dies wäre sicher ein Ansatzpunkt, bei dem man die Arbeit fortsetzen könnte. Die Zusatzfeatures wurden bereits im Kapitel "Trainernotizbuch" aufgeführt.

---

## 12.2 Persönlicher Bericht Florian Ziltener

### 12.2.1 Projektverlauf

Das Projekt hat mich von Anfang an interessiert und begeistert. Zu Beginn des Projektes durften wir Spiele mit dem Trackingsystem aufzeichnen. Besonders zugesagt hat mir daran, dass wir den Einsatz von Technologie selbst in einer Alltagssituation testen konnten. Selbstverständlich sammelten wir nebst den Fussballspielen auch noch weitere Informationen, wie zum Beispiel Informationen aus Zeitungsartikeln und auch Informationen aus dem Internet.

Im zweite Teil haben mein Kollege und ich uns dann schon mit Statistikauswertungen und der Tablet-Anwendung auseinandergesetzt. Zuerst haben wir einen Paper Prototyp erstellt, den wir im späteren Verlauf des Projektes gefühlte 100x angepasst haben. Obwohl das sehr Aufwändig war, sehe ich trotzdem den Nutzen dahinter. Nebenbei haben wir auch noch Konkurrenzprodukte unter die Lupe genommen, um damit unserer Software die besten Funktionalitäten mitzugeben.

Als wir die nötigen Informationen gesammelt und die Requirements definiert hatten, begannen wir den Aufbau unserer Software zu planen. Danach folgte die Entwicklung der Software. Zwar gab es einige Hürden zu bewältigen, da wir Probleme mit der neuen Technologie hatten. Schlussendlich wurde diese Hürde jedoch von uns bewältigt auch wenn wir bei den Feature-Implementation einstriche machen mussten. Dies führte Version 1.0 unserer Software.

### 12.2.2 Was habe ich gelernt?

Das Wichtigste welches ich persönlich aus dieser Arbeit mitnehme, ist das Wissen über die neue Technologie JQuery Mobile. Ich lernte sehr viel über die Javascript Programmierung und konnte mich zusätzlich mit Tablet Anwendungen auseinandersetzen. Da wir die Software auf einem iPad getestet haben, durfte ich das PhoneGap Framework kennen lernen.

Ausser den neuen Technologien habe ich auch viel über den grossen Zeitaufwand einer Dokumentation gelernt. Bei meiner nächsten Arbeit (Bachelorarbeit) weiss ich nun bereits, dass ich fortwährend an der Dokumentation arbeiten muss, damit die Arbeit für mich einfach wird.

### 12.2.3 Fazit

Die Software haben wir grösstenteils realisiert, jedoch konnten wir nicht alle gewünschten Features einbauen, was ich persönlich schade finde. Besonders das wir keine Player auswählen können ärgert mich am meisten.

Die Endversion sieht meiner Meinung nach viel versprechend aus. Besonders gefällt mir die Aufstellungsmöglichkeit der Spieler und das Erfassen eines Fussballspieles. Mit der jetzigen Software ist es erheblich

leichter ein Spiel zu Erfassen. Wir haben selbst festgestellt wie mühsam es sein kann, wenn man jeden Event der bei einem Spiel eintrifft, mitschreibt. Dies ist meiner Meinung nach der grosse Plus unserer Software.

---

## 12.3 Persönlicher Bericht Mike Osmand Loydd

### 12.3.1 Projektverlauf

Von Anfang an war ich über die Semesterarbeit sehr begeistert. Nicht nur weil es etwas mit meinem Lieblingssportart Fussball zu tun hat, sondern weil wir uns auch selbst mit dem Fussball Tracking System auseinandersetzen durften. Besonders erstaunt hat mich die Erkenntnisse, welche GPS Daten auswerten werden können. Es hat mir sehr viel Spass bereitet eigene Auswertungsmöglichkeiten auszutüfteln und die Programmierung der eigenen Auswerteideen mit Flot zu entwickeln. Natürlich liessen wir uns nicht die Chance entgehen das Fussball GPS Tracking System selbst in ein bis zwei Fussballspiele zu testen und live mitzerleben.

In der Mitte der Semesterarbeit einigten wir uns darauf, ein Trainernotizbuch als Tabletanwendung zu realisieren. Zuerst entwarfen wir ein Paper Prototyp für das GUI, den wir mit einem Human Computer Interaction Experten Herrn Markus Stolze ein Usability Test durchführten. Wir erhielten sehr nützliche Informationen und Verbesserungsvorschläge. Während dessen analysierten wir die zu implementierenden Features, welches zu intensiven und interessanten Diskussionen führte. Nach der definitiven Festlegung der Features und des Designs, setzten wir uns mit der Technologie-Möglichkeiten zur Umsetzung auseinander. Der Entscheid fiel auf JQuery Mobile und Phonegap, da es erstens, cross-platform ist und zweitens, auf Webtechnologien basiert. Leider mussten wir bei der Entwicklung der Tabletanwendung *CoachAssist* mit diversen unerwarteten Problemen kämpfen, welche durch die third-party JQuery Plugins verursacht wurden. Was uns dazu bewegte einige Features, die wir gerne implementiert hätten, auslassen mussten.

### 12.3.2 Was habe ich gelernt?

Durch das Trainernotizbuch *CoachAssist* Projekt konnte ich meine Fähigkeiten im Bereich Webtechnologien vertiefen und erweitern. Dazu konnte ich neue Technologien wie JQuery Mobile und PhoneGap kennenlernen und anwenden. Einer der wichtigsten Erfahrung war der Prozess zur Festlegung des Design und der Features.

Dank die Designentwurf für *CoachAssist* entdeckte ich meine Interesse für “uman computer interaction“. Die Herausforderung beim Designentwurf, welche mich am meisten fasziniert hat, war das eindenken in den Endbenutzer und in sein Benutzungsverhalten.

Die Implementationsphase war die härteste und lehrreichste Phase, da ich mich mit neuen Technologien anfreunden durfte. Durch die entstan-

denen Problemen konnte ich mit JSON, JQuery und JQuery Mobile intensiver auseinandersetzen. Jetzt nachdem ich die Technologien kenne, behaupte ich es war die richtige Entscheidung auf Phonegap und JQuery Mobile zu setzen. Diese Technologien besitzen extrem hohes Potential. Jedoch ist zu beachten, dass Query Mobile noch in Kinderschuhen stecken.

### **12.3.3 Fazit**

Für mich als Fussballfanatiker war es in erste Linie sehr faszinierend zu erfahren, welche Technologien im Fussball eingesetzt oder in welche Richtung geforscht werden, um die Spieler zu analysieren und sie zu Höchstleistungen zu bewegen.

Im grossen und ganzen hat mir unsere Semesterarbeit sehr viel Spass bereitet und auch meine Projektmanagementfähigkeiten massiv verbessert. Es war für mich eine sehr positive Erfahrung mit cnlab zu arbeiten, da ich mein Fachwissen sehr stark ausweiten und vertiefen konnte. Zum Schluss möchte ich Herrn Heinzmann und cnlab Software für die Zusammenarbeit danken.



---

# Literaturverzeichnis

- [1] Remy Sharp Bruce Lawson. *HTML 5*. Addison-Wesley, 2011.
- [2] cnlab. cnlab fussball. <http://www.cnlab.ch/fussball/>, November 2011.
- [3] Steve Fenton. jquery mobile drag and drop. <http://www.stevefenton.co.uk/Content/Jquery-Mobile-Drag-And-Drop>, Dezember 2011.
- [4] Adobe Systems Inc. Phone gap. <http://docs.phonegap.com/en/1.2.0/index.html>, Dezember 2011.
- [5] Apple Inc. Apple store. <http://itunes.apple.com/ch/genre/ios/id36?mt=8>, November 2011.
- [6] Google Inc. Android market. <https://market.android.com/?hl=de>, November 2011.
- [7] The jQuery Project. jquery mobile. <http://jquerymobile.com/>, Dezember 2011.
- [8] Uni Magdeburg. Latex vorlage. <http://wdok.cs.uni-magdeburg.de/studium-und-lehre/hinweise-und-vorlagen/wdok-diplom.tar.gz/view>, Oktober 2011.
- [9] Thomas Myer. *Beginning PhoneGap*. Wrox, 2011.
- [10] Pineheadtv. Introduction to jquery mobile. <http://www.youtube.com/watch?v=VY20Q60N4QA>, Dezember 2011.
- [11] Jon Reid. *jQuery Mobile*. O'Reilly Media, 2011.
- [12] Sega. Football manager. <http://www.sega.com/fm11/>, Oktober 2011.

- [13] Wikipedia. Fussball regelwerk. [http://de.wikibooks.org/wiki/Fussball:\\_Regelwerk](http://de.wikibooks.org/wiki/Fussball:_Regelwerk), Dezember 2011.
- [14] Wikipedia. jquery mobile. [http://en.wikipedia.org/wiki/JQuery\\_Mobile](http://en.wikipedia.org/wiki/JQuery_Mobile), Dezember 2011.
- [15] Wikipedia. Nachspielzeit. <http://de.wikipedia.org/wiki/Nachspielzeit>, Oktober 2011.
- [16] Wikipedia. Phonegap. <http://en.wikipedia.org/wiki/PhoneGap>, Dezember 2011.

**Api**

Abkürzung für Application Programming Interface. Eine Api ist eine Programmierschnittstelle, welche eine Anbindung an das System zur Verfügung stellt. 40, 41

**App**

App ist die Kurzform für Applikation. 11, 13, 21

**DOM**

Abkürzung für Document Object Model. DOM ist eine Spezifikation einer Schnittstelle für den Zugriff auf HTML-Dokumente. 45

**Fitt's Law**

Fitt's Law ist ein Gesetz, welches besagt, dass die benötigte Zeit eine Zielfläche zu erreichen eine Funktion der Distanz zu dieser Fläche und deren Größe ist. 50

**Framework**

Ein Framework ist ein Programmiergerüst. Es ist eine Rahmenstruktur das verwendet wird um ein Programm zu erstellen.. 13, 25

**GPS**

GPS ist kurz für Global Positioning System. Es dient zur zur Positionsbestimmung und Zeitmessung. 17

**JQuery**

JQuery ist eine JavaScript-Klassenbibliothek. 25

**JSON**

Abkürzung für JavaScript Object Notation. JSON ist ein Datenformat, welches für Mensch und Maschine einfach lesbarer ist.

Es ist ein Textform zum Zweck des Datenaustauschs zwischen Anwendungen. 40, 41

**Memory Leak**

Ein Memory Leak ist ein Speicherleck. Ein laufender Prozess belegt einen Speicherbereich, diesen jedoch im Zuge der Ausführung weder freigeben noch genutzt werden kann. 26

**Paper Prototyp**

Darstellung des Designs in Papierform. 49, 50

**Persona**

Eine Persona ist ein Beschrieb einer Person. 29, 31

**Pop-up**

Ein Pop-up ist ein visuelles Element einer Applikation, dass aufspringt und dabei andere Teile der Applikation verdeckt. 53

**Request**

Ein Request ist eine Anfrage eines Clients an einen Server. 40

**Response**

Eine Response ist die Antwort des Servers auf eine Anfrage eines Clients. 40, 41

**SQLite**

SQLite ist ein relationales Datenbanksystems. 41

**Token**

Ein Token ist eine Bitsequenz in einem bestimmten EDVNetzwerk. Hier dient er zur Authentifizierung des Benutzers.. 11

---

# Tabellenverzeichnis

1.1	Aufgabenstellung - Die wichtigsten Angaben . . . . .	11
5.1	Apple Store - Mitbewerberanalyse . . . . .	24
5.2	Android Market - Mitbewerberanalyse . . . . .	25
11.1	Systemtests - Auswertung Use Cases . . . . .	55



---

# Abbildungsverzeichnis

7.1	UseCase Diagramm . . . . .	34
8.1	Programmstruktur . . . . .	37
8.2	SSD - Start eines Matches . . . . .	39
8.3	SSD - Formation Drag and Drop . . . . .	40
9.1	Package Diagramm WWW . . . . .	41
9.2	Package Diagramm PhoneGap . . . . .	42
9.3	SSD - Formation Drag and Drop . . . . .	43
9.4	JSON - Download von Daten . . . . .	44
9.5	JSON - Request/Response-Diagramm . . . . .	45
B.1	PaperPrototyp - Menü . . . . .	77
B.2	PaperPrototyp - Matchaufzeichnung . . . . .	78
B.3	PaperPrototyp - Aufstellung ohne Spieler . . . . .	78
B.4	PaperPrototyp - Aufstellung mit Spieler . . . . .	78
D.1	Installation - Sidemenu . . . . .	84
D.2	Installation - Add . . . . .	84
D.3	Installation - Browser . . . . .	84
D.4	Installation - Application Übersicht . . . . .	85
D.5	Installation - iPad Übersicht . . . . .	85
D.6	Nutzung - Start Menü . . . . .	86
D.7	Nutzung - Option . . . . .	86
D.8	Nutzung - Option with Token . . . . .	87
D.9	Nutzung - Komplettes Menü . . . . .	87
D.10	Nutzung - Match Information . . . . .	88
D.11	Nutzung - Match . . . . .	88

D.12 Nutzung - End Match . . . . .	89
D.13 Nutzung - Lineup . . . . .	89
D.14 Nutzung - Set-Lineup . . . . .	90



## **Teil III**

## **Anhang**



---

## A.1 Zeitplan

Der Zeitplan befindet sich in der Zusatzdokumenten.

Hier ist zu sagen, dass die Inception Phase deshalb so lange gedauert hat, weil wir im ersten Teil der Arbeit uns mit Fussballstatistikauswertungen beschäftigt haben.

---

## A.2 Meilensteine

Dies ist eine detaillierte Erklärung der Meilensteine, welche ebenfalls im Zeitplan zu finden sind. Es wird aufgezeigt, welche Ziele damit verfolgt wurden.

### **MS 1: Abgabe der verfeinerten Aufgabenstellung an Dozent**

Ziel dieses Meilensteins ist es die Aufgabenstellung genau zu definieren. Dies geschieht in enger Zusammenarbeit mit dem Dozenten.

### **MS 2: Analyse von Fussballdaten und Auswertungsmöglichkeiten zur Review an Dozenten**

Es werden zur Informationsbeschaffung diverse Fussballspiele mit den GPS-Geräten aufgezeichnet. Die Daten werden im Vorfeld ausgewertet und schlussendlich mit dem Dozenten besprochen.

### **MS 3: Designentwurf zur Bestätigung dem Dozenten vorlegen**

Genehmigung des Designentwurfs in Form eines Paperprototyps. Das Design soll im Groben vorhanden sein. Änderungen sind natürlich während der Implementation noch vorbehalten.

#### **MS 4: Feedback zum jetzigen Stand abholen, Code-Review**

Dieser Meilenstein dient zu einer Einschätzung des Dozenten über die Arbeit. Zugleich soll er den Projektmitarbeitenden aufzeigen, wo Verbesserungspotential liegt.

#### **MS 5: Implementation abgeschlossen**

Dieser Meilenstein umschreibt das Ende der Implementation. Dies bezieht sich auf neue Features und/oder grosse Änderungen im Code. Vorbehalten ist hier: kleine Anpassungen und Bugfixes.

#### **MS 6: Testing abgeschlossen**

Wie der Titel schon sagt, geht es hier darum, dass die Tests abgeschlossen sind. Vorwiegend werden hier Usability-Test und Systemtests durchgeführt. Unit-Tests stehen wegen der Plattformwahl eher im Hintergrund.

#### **MS 7: Dokumentation abgeschlossen und dem Dozenten überreicht**

Dies ist der letzte Meilenstein der Arbeit. Das Ziel dieses Meilensteins ist die Beendigung sowie die Abgabe der Semesterarbeit.

---

## **A.3 Risikoanalyse**

### **A.3.1 Risiken**

Während der ganzen Arbeit wird mit folgenden Risiken gerechnet:

- Risiko 1: Krankheit / Ausfall eines Teammitglieds
  - Auswirkung: Mehraufwand für das andere Teammitglied/ Zeitverlust
  - Massnahme: Vorarbeiten oder Puffer im Zeitplan einfügen
- Risiko 2: Das Projekt wird nicht rechtzeitig fertig
  - Auswirkung: mehr Aufwand / Kosten
  - Massnahme: Zeitplan immer im Auge behalten
- Risiko 3: Datenverlust Dokumente
  - Auswirkung: Dokumente müssen neu geschrieben werden /mittlerer Zeitaufwand
  - Massnahme: Sicherung mit Backups (Dropbox, Redmine)
- Risiko 4: Datenverlust Code
  - Auswirkung: Code muss nochmals generiert werden / hoher Zeitaufwand

- Massnahme: Repository einrichten
- Risiko 5: Ausfall Arbeitsplatz
  - Auswirkung: Teammitglied kann einige Zeit nicht arbeiten
  - Massnahme: Notfall-Arbeitsplatz bereit halten / einrichten
- Risiko 6: Konflikte im Team
  - Auswirkung: Kommunikation ist geschädigt, führt zu fehlenden Absprachen und dann zu unnötiger Arbeit
  - Massnahme: Teambildende Massnahmen durchführen
- Risiko 7: Missverständnisse mit Kunde enlab
  - Auswirkung: Kommunikation ist geschädigt, führt zu fehlenden Absprachen und dann zu unnötigem Mehraufwand
  - Massnahme: Regelmässige Besprechungen
- Risiko 8: Vereinbarten Anforderungen werden nicht erfüllt
  - Auswirkung: Projektziel wird nicht erfüllt. Folge: Semesterarbeit nicht bestanden
  - Massnahme: Fortlaufend Feedback vom Betreuer/Dozent einholen
- Risiko 9: Schwerwiegender Fehler beim Testen entdecken
  - Auswirkung: Code muss nochmals überarbeitet werden.
  - Massnahme: Code fortlaufend testen
- Risiko 10: Nicht ausreichende Kenntnisse in der neuen Technologie (jQuery Mobile, PhoneGap, etc)
  - Auswirkung: Schreiben des Codes wird eher schwerfällig
  - Massnahme: Mehr Zeit in das Lernen / Üben investieren
- Risiko 11: Falsch programmierte Funktionen / Methoden
  - Auswirkung: Programm erfüllt nicht seinen Zweck, Funktionen müssen neu geschrieben werden
  - Massnahme: Regelmässig auf ihre Funktionalität testen
- Risiko 12: Speicherung der Daten mit jQuery Mobile
  - Auswirkung: Aufbewahrung der Daten auf dem Gerät schlecht oder nicht möglich
  - Massnahme: Abklären, was mit der Technologie möglich ist

### **A.3.2 Fazit**

Da es sich bei den meisten Risiken um Standardprobleme handelt, kann man diesen mit geeigneten Mitteln leicht begegnen. Mit den getroffenen Massnahmen sollten keine schwerwiegenden Probleme auftauchen



---

## B.1 PaperPrototyp

Abbildung B.1:  
PaperPrototyp - Menü

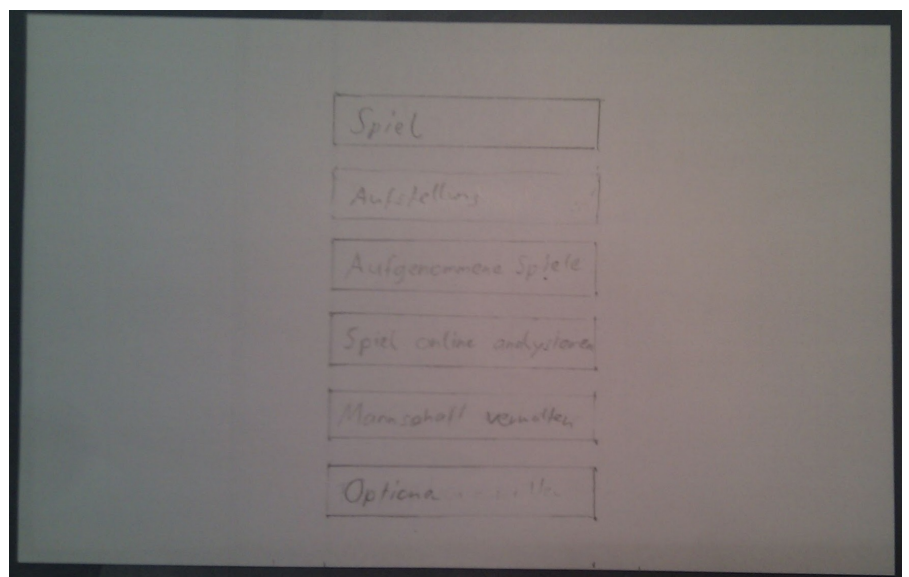


Abbildung B.2:  
PaperPrototyp -  
Matchaufzeichnung

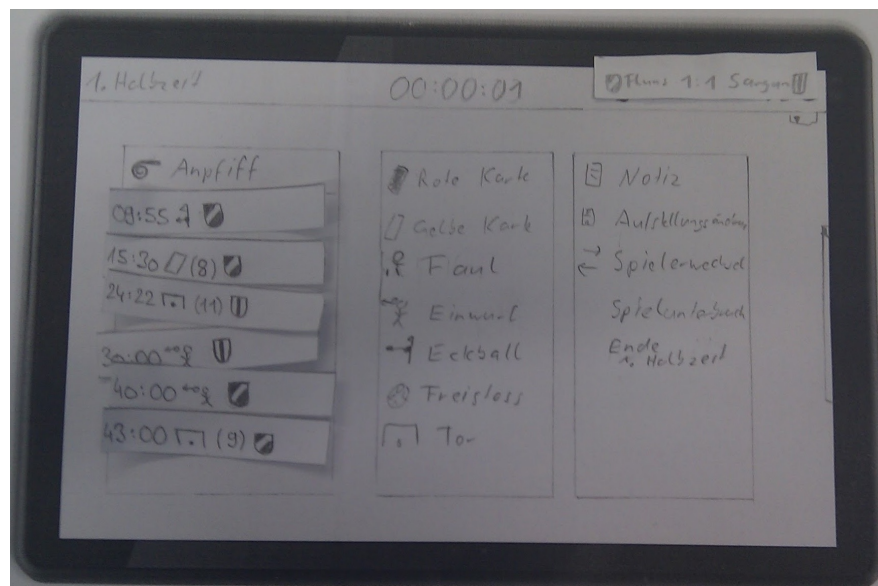


Abbildung B.3:  
PaperPrototyp - Aufstellung  
ohne Spieler

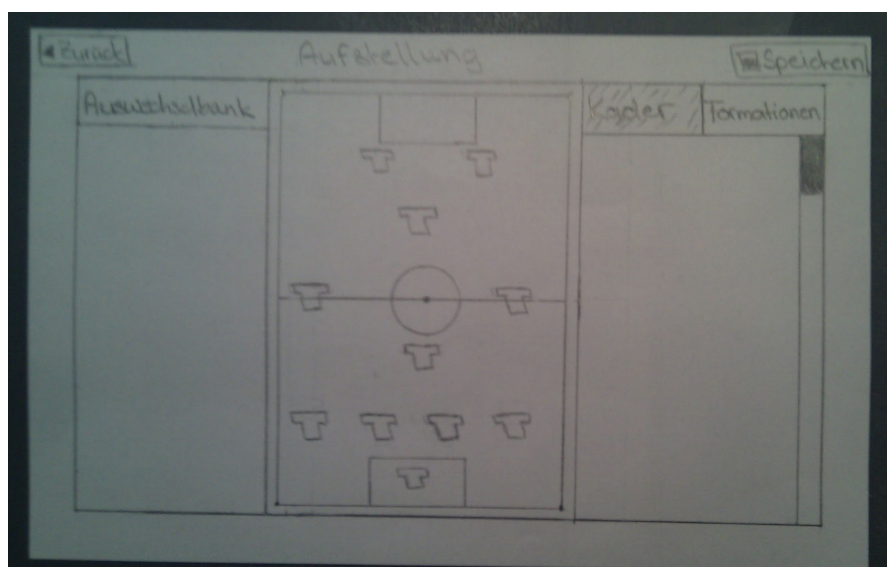
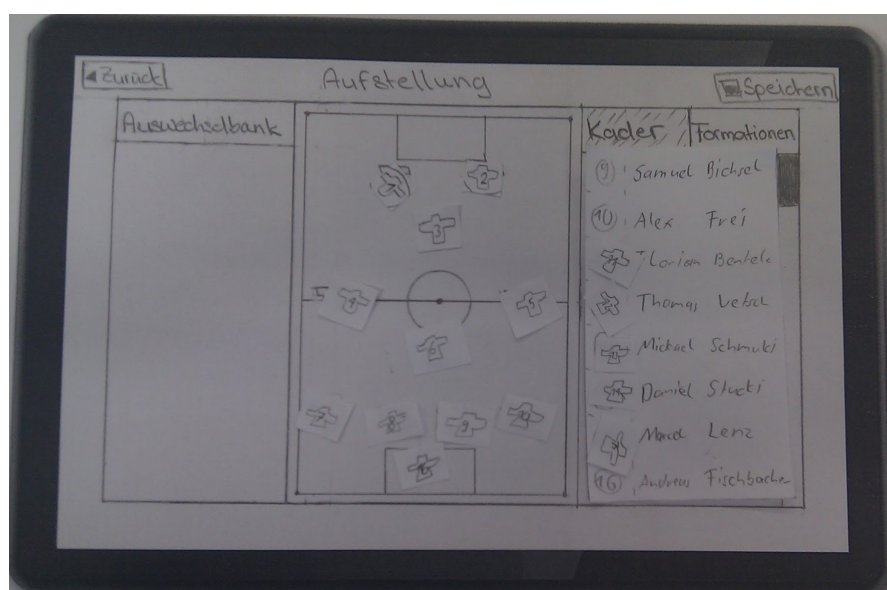


Abbildung B.4:  
PaperPrototyp - Aufstellung  
mit Spieler





In diesem Kapitel geht es um die erarbeiteten Statistikauswertungen, welche Ideen dahinter stecken und wie sie zustande gekommen sind. Begonnen wird mit den fertigen Statistiken und diejenigen, die weiteres Interesse haben, können die Hintergründe zu diesen Auswertungen einsehen.

---

## C.1 Neue Statistiken

Zu Beginn gehen wir auf die einzelnen Auswertungen ein. Die wohl wichtigste und langwierigste Auswertung war die, der Sprints. Andere Ideen waren unter anderem Richtung, Gruppenbildung auf dem Feld und Zeichendeutung. Jedoch waren auch banale Dinge ersichtlich, wie eine Auswertung eines Spielers auf das gesamte Spiel hinweg.

### C.1.1 Sprint

Die erste Idee, die verfolgt wurde, war eine Auswertung nach Sprints. Definiert wird der Sprint durch die Dauer, die Distanz und die Geschwindigkeit.

Die erste Frage, die sich gestellt hat, war: Ab wann beginnt ein Sprint? Hier haben wir die Annahme von  $\pm 5$  km/h getroffen.

Eine Auswertungsmöglichkeit ist sicher die Häufigkeit der Sprints während einem Spiel. Dabei muss die Zeit eines Sprints auch miteinbezogen werden. Aus der Häufigkeit und der Zeit eines Sprints wäre auszusagen, wie sich das Sprintverhalten während eines kompletten Spieles verhält. Wenn der Spieler gut konditioniert ist, sollten geringere Unterschiede vom Beginn bis zum Ende eines Spieles ersichtlich sein. Dafür ist nicht unbedingt die Dauer relevant. Eine weitere Möglichkeit wäre auch die Durchschnittsgeschwindigkeit eines Sprints.

Eine weitere Darstellungsmöglichkeit wäre hier ein Spinnennetzdiagramm. Dies zeigt auf wie viele Sprints in eine bestimmte Richtung

gemacht hat. Dabei muss die Anzahl Sprintstrecken summiert werden und als fix Punkte in der jeweiligen Richtung gesetzt werden. Wenn man dann die Punkte verbindet, könnte sogar seine Spielerposition bestimmt werden.

### **C.1.2 Mannschaftsheatmap**

Diese Auswertung soll ähnlich wie die Spielerheatmap sein. Jedoch anstatt einen Spieler auf einen Standort auf dem Spielfeld zu beziehen, sollten die Mannschaft als gesamtes angezeigt werden. Dies könnte man einfach realisieren, indem man die Spielerheatmaps der Mannschaft im Hintergrund aufeinander legt und dann darstellt.

### **C.1.3 Spielerbewegung**

Eine wichtige Auswertungsmöglichkeit ist das Herauslesen einer Spielerbewegung. Dadurch kann man einige neue Zahlen herauslesen, die schlussendlich in Statistiken umgewandelt werden können. Durch beobachten einiger Fussballspiele auf der Onlineplattform haben sich folgende Verhaltensweise heraus kristallisiert:

- Einwürfe sind dadurch ersichtlich, dass ein Spieler immer hinter die Seitenlinie rennt.
- Eckbälle sind ebenfalls durch Positionierung eines Spielers an der Ecke eines Fussballfeldes ersichtlich. Wie bei Einwurf verharren die meisten kurz an dieser Stelle bis sie den Einwurf/Eckball tätigen.
- Tore sind uns nur speziell bei höher liegenden Mannschaften aufgefallen. Jeder hat sicher schon bei einem Fussballspiel gesehen, wie die Mannschaft dann eine grosse Gruppe bildet. Da es jedoch nicht bei allen Mannschaften vorkommt, trifft dies nicht bei jedem Spiel zu und kann somit nicht sicher verwendet werden.
- Wechsel von Angriff und Verteidigung kann anhand der Mannschaftsbewegung herausgelesen werden. Hier wäre besonders interessant, wie schnell der Wechsel stattfindet. Die Mannschaft müsste als ganzes betrachtet werden, dies könnte man mit einem "length-width-squery" sicher erreichen.

### **C.1.4 Auswertung eines Spielers**

Des Weiteren entstand die Idee, dass es für einen Spieler interessant wäre, wenn er seine früheren Leistungen mit seiner jetzigen vergleichen könnte. Dafür wäre eine Auswertung pro Spieler nötig.

Um das zu ermöglichen müsste ein Spielerprofil erstellt werden. Dieses würde in der Datenbank dementsprechend verlinkt werden. So wäre es dann möglich ohne weitere Umstände eine Auswertung zu generieren.

Mögliche Vergleiche wären Distanz des zurückgelegten Weges, Sprintdauer sowie seine Geschwindigkeiten. Dies muss im Verhältnis zur aktuellen Spielzeit betrachtet werden.



---

## D.1 Voraussetzung

Um die Applikation Nutzen zu können sind folgende Dinge vorausgesetzt:

- Mac mit XCode Version 4
- Entwickleraccount

Um es mit nicht mit XCode öffnen zu müssen, muss es sich im App Store befinden.

---

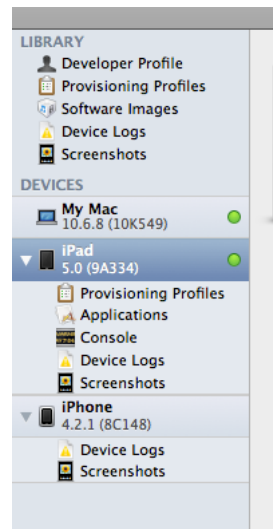
## D.2 Installation

Zuerst muss das Gerät bzw. das iPad dem Entwickleraccount hinzugefügt werden. Dafür befolgen Sie bitte die Anleitung unter dem folgenden Link: <http://mobile.tutsplus.com/tutorials/iphone/iphone-sdk-install-apps-on-iphone-devices-for-development>

Sobald der Tablet dem Account hinzugefügt wurde, fahren Sie bitte mit folgenden Schritten weiter:

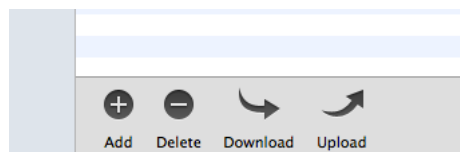
Zuerst müssen Sie XCode öffnen und zum “Window-Organizer“ wechseln. Dann wählen Sie, wie auf dem folgenden Bild, ihr iPad aus. Dort wechseln Sie zum Unterpunkt “Applications“.

Abbildung D.1: Installation -  
Sidemenu



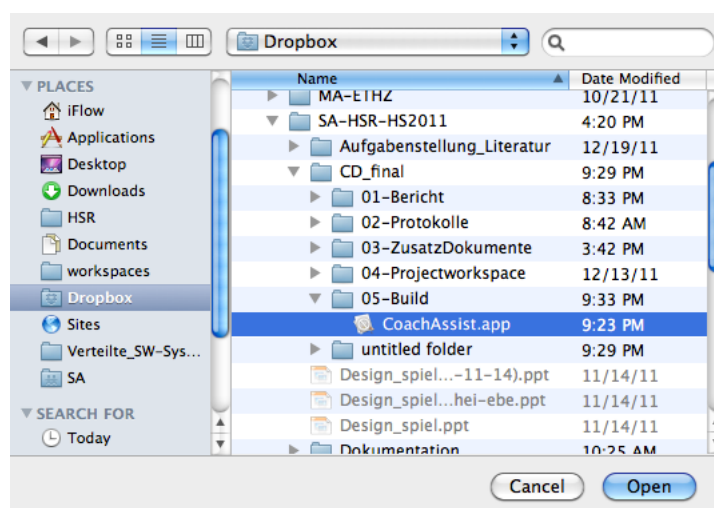
Nun da Sie ihr Gerät ausgewählt haben, können Sie, wie auf dem folgenden Bild, den Button “Add“ drücken. Damit fügen Sie die Applikation ihrem iPad hinzu.

Abbildung D.2: Installation -  
Add



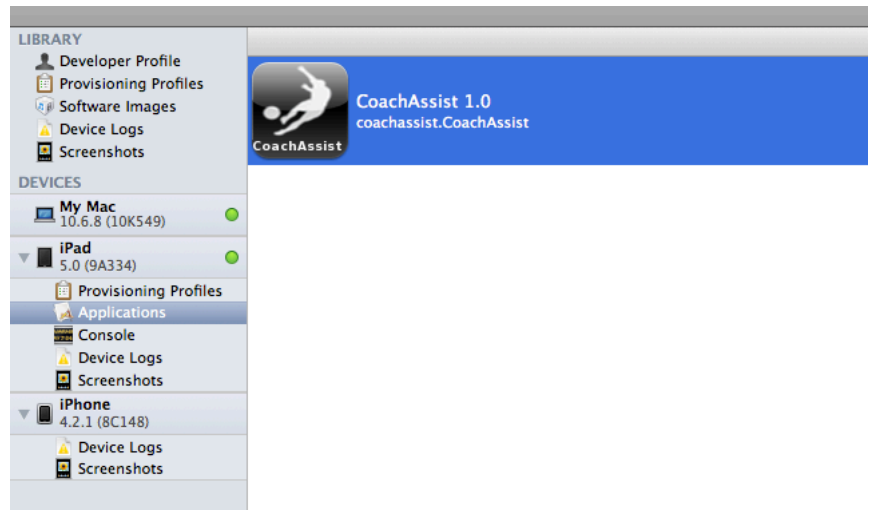
Wenn sich das folgende Fenster, wie auf dem nächsten Bild, öffnet, navigieren Sie im Finder zum Ort der Applikation. Die Datei heisst CoachAssist.app.

Abbildung D.3: Installation -  
Browser



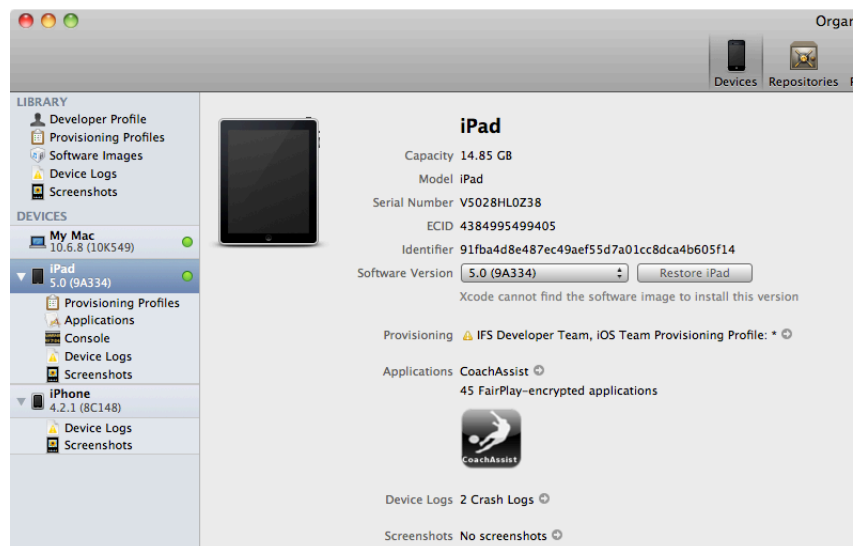
Das App sollte nun, wie auf folgendem Bild, ihren Applications hinzugefügt sein.

Abbildung D.4: Installation - Application Übersicht



Im iPad Hauptfenster sollte es jetzt wie auf dem nächsten Bild gezeigt aussehen.

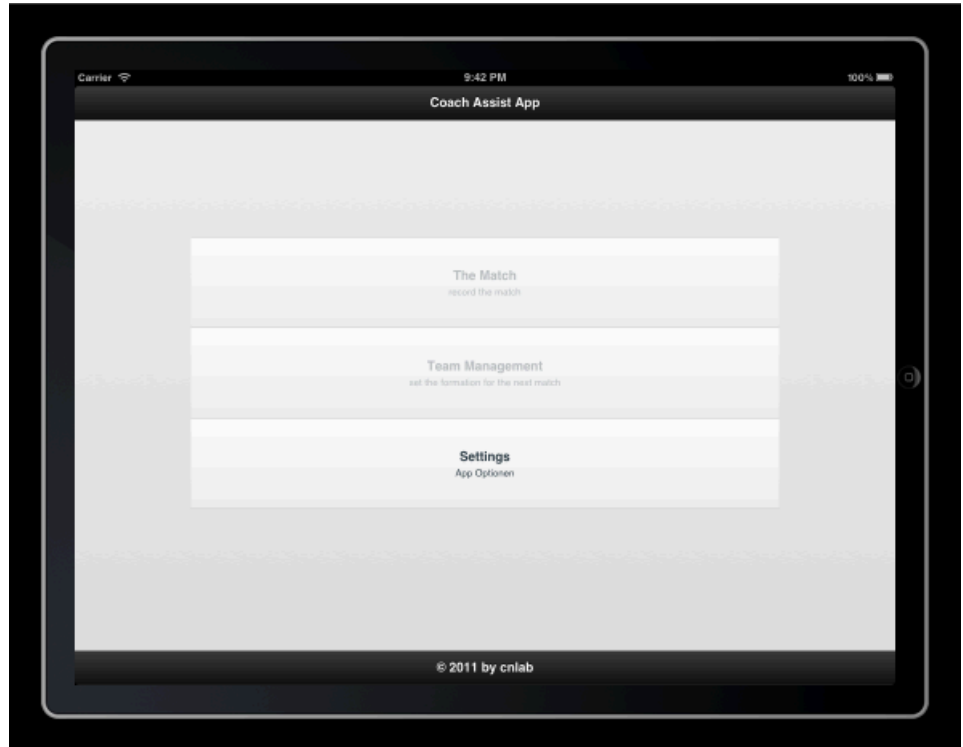
Abbildung D.5: Installation - iPad Übersicht



## D.3 Nutzung

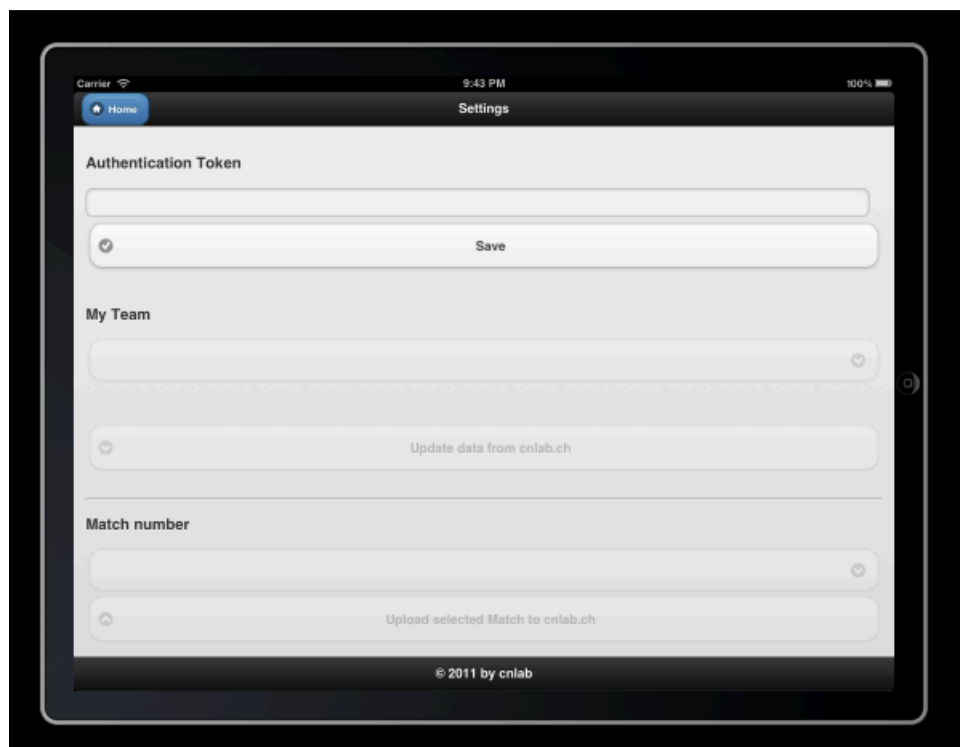
Beim ersten Start des Programmes sind zuerst alle Menüpunkte bis auf den untersten “Settings“ deaktiviert. Um die anderen Punkte Nutzen zu können, wählen Sie die “Settings“ aus.

Abbildung D.6: Nutzung -  
Start Menü



Sie gelangen nun auf folgende Ui:

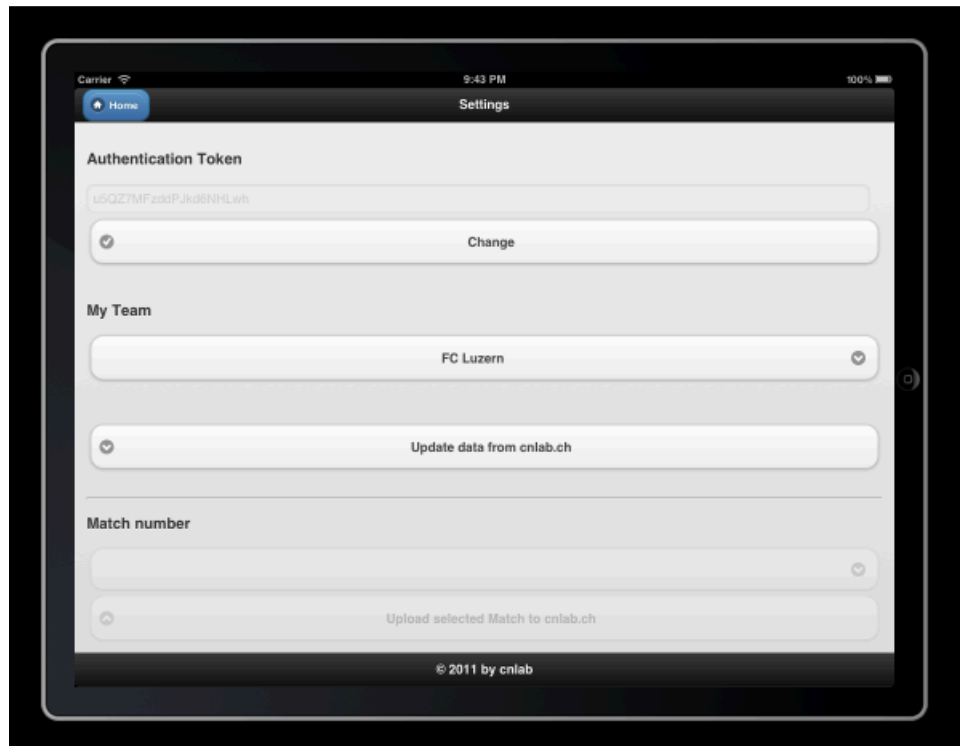
Abbildung D.7: Nutzung -  
Option





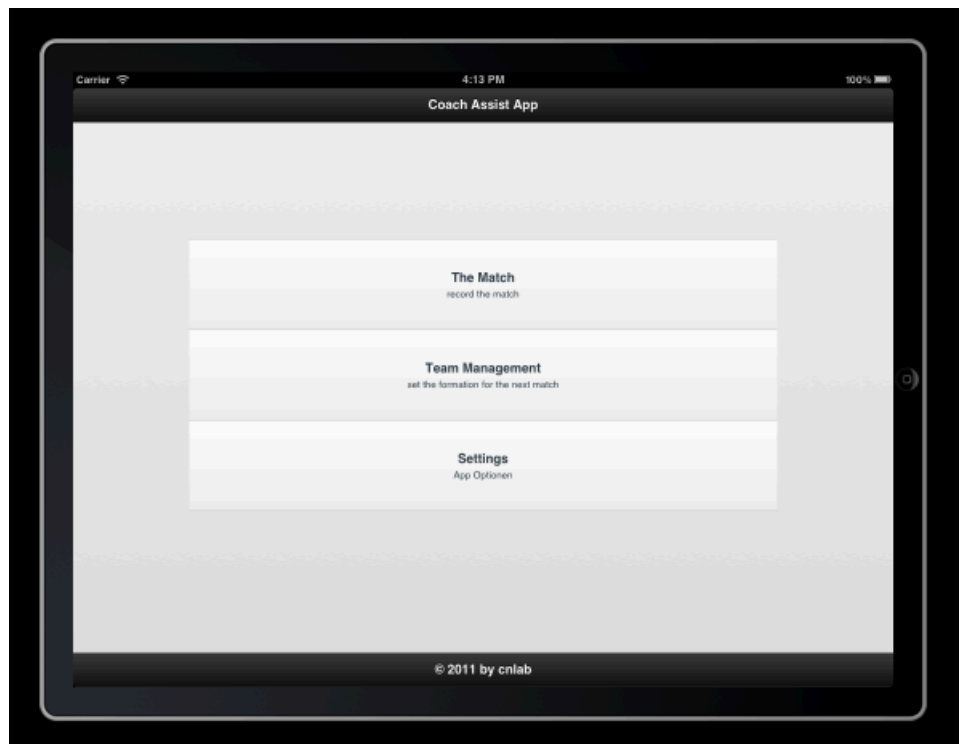
Geben Sie hier den Token ein, welchen Sie bei der Anmeldung auf [www.cnlab.ch/fussballapi](http://www.cnlab.ch/fussballapi) erhalten. Sie finden ihn nach der Anmeldung/Registrierung unten rechts. Drücken sie nachher auf den “Save“ Button um die Daten herunterzuladen.

Abbildung D.8: Nutzung -  
Option with Token



Über den Home-Button gelangen Sie wieder zurück zum Einstiegsme-  
nü. Wie nun zu erkennen ist, sind die anderen zwei Punkte aktiviert.

Abbildung D.9: Nutzung -  
Komplettes Menü



Wenn Sie nun das Menü “The Match“ auswählen, gelangen Sie zur Eingabe der Matchinformationen. Füllen Sie diese aus und klicken Sie auf “Save“ um die Daten zu speichern und anschliessend werden Sie automatisch zur Spielaufzeichnung weiter geleitet.

Abbildung D.10: Nutzung - Match Information

The screenshot shows a mobile application interface for entering match information. At the top, there's a status bar with 'Carrier', signal strength, '4:14 PM', and '100%' battery. Below is a navigation bar with 'Home', 'Match Info', and 'New match' buttons. The main area contains several input fields: 'Match number' with the value '44445555', 'Opponent team name' with a dropdown showing 'FC St.Gallen', 'Match type' with a dropdown showing 'Liga', and 'Pitch' with radio buttons for 'Home' (selected) and 'Away'. A 'Save' button is located at the bottom right of the form. At the very bottom, there's a navigation bar with three tabs: 'Match info' (active), 'Events record', and 'Note record'.

Sobald Sie auf den Button “Kick-off“ klicken, werden die Event-Buttons aktiviert. Sie können nun beliebige Events für die jeweiligen Teams erfassen.

Abbildung D.11: Nutzung - Match

The screenshot shows the 'Match' screen after a kick-off. The top navigation bar now includes 'Home', 'Playtime: 00:15', 'Finish first half', 'Kick-off extra time', 'Effective playtime: 00:15', and 'Spielunterbruch'. The main area is divided into two columns for 'FC St. Gallen' and 'FC Luzern'. Each column has a grid of event buttons: 'Substitution', 'Goal shot', 'Red card', 'Yellow card', 'Freekick', 'Penalty', 'Offside', 'Thro...', and 'Corner'. On the left, there's a 'Live-Ticker' section with 'Kick-off first half', 'Uhrzeit: 18:14:22 Uhr', and 'Halbzeit Spielstand: 0 : 0'. At the bottom, the navigation bar has 'Match info', 'Events record' (active), and 'Note record' tabs.

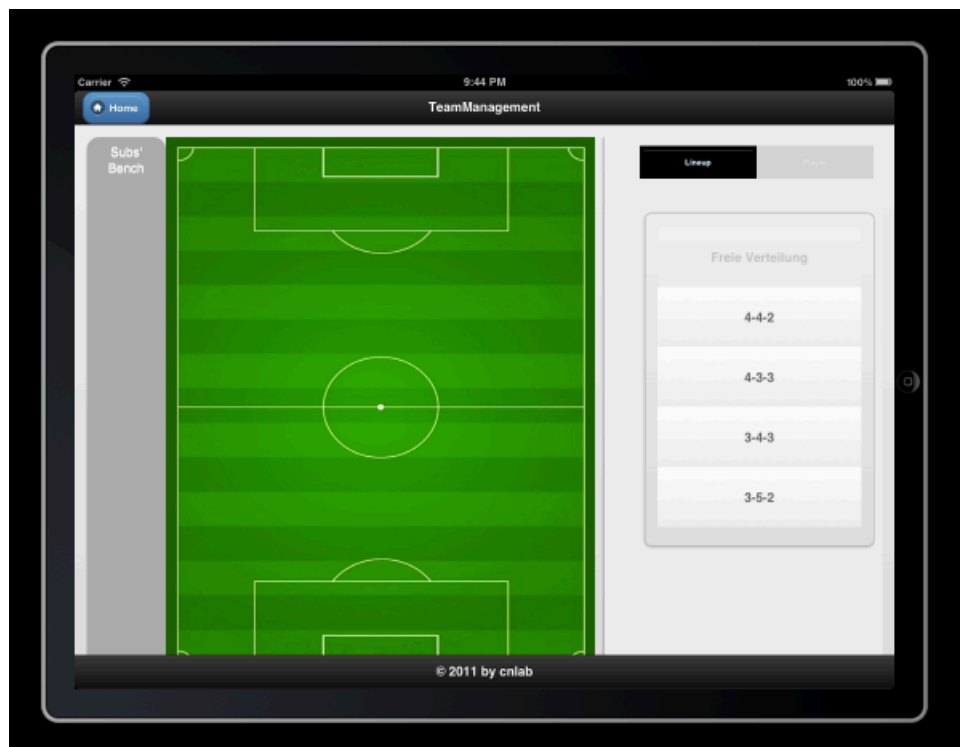
Wenn das Spiel beendet ist, klicken Sie auf den Button “End Match“. Somit ist das Spiel beendet. Sie gelangen mit dem Home Button, wieder zurück zur Übersicht.

Abbildung D.12: Nutzung - End Match



Wählen Sie bei Menü den Punkt “Team Management“ aus, damit gelangen Sie zur Auswahl der vorgegebenen Formationen. Wählen Sie eine beliebige Formation aus.

Abbildung D.13: Nutzung - Lineup



Nun können Sie die Spieler beliebig auf dem Spielfeld und auf der Ersatzbank verteilen.

Mit “swipe-left“ auf einen Spieler, wird der Spieler automatisch auf die Ersatzbank geschickt.

Mit “swipe-right“ auf einen Spieler, wird dieser wieder zurück an seinen Stammsplatz in der Auswahlliste geschickt.

Durch klicken auf den jeweiligen Spieler erhalten Sie eine Auswahl und können den Spieler dann auf die Flächen ablegen.

Abbildung D.14: Nutzung -  
Set-Lineup

