# Abschlusspräsentation

## Semesterarbeit Racketsport Manager

Raphael Marques

Züricher Hochschule für Angewandte Wissenschaften

29. September 2015

# Inhaltsverzeichnis

# Ziele

- Benutzergruppen- und Anforderungsanalyse
- Technologien und Konzepte für Implementation
- Implementation API
- Implementation Web-Applikation
- Implementation Android App

# Benutzergruppen

- Clubmitglieder
  - Fixe Trainings
  - Flexible Liga-Spiele
  - Flexible Non-Liga-Spiele
- Regelmässige Spieler
  - Vereinbarte Spielzeiten mit einem oder wenigen Partnern
  - Unregelmässige Spiele mit vielen anderen Partnern
- Gelegenheitsspieler
  - Flexible und unregelmässige Spielzeiten mit vielen Partnern
  - Neigt zur Spielpausen ohne Mitspieler

# Anwendungsfälle

- Identifikation von Partnern
- Vereinbarung eines Spiels
- Protokollierung eines Spiels
- Management einer Liga
- Regelmässige Herausforderungen

# Anforderungen

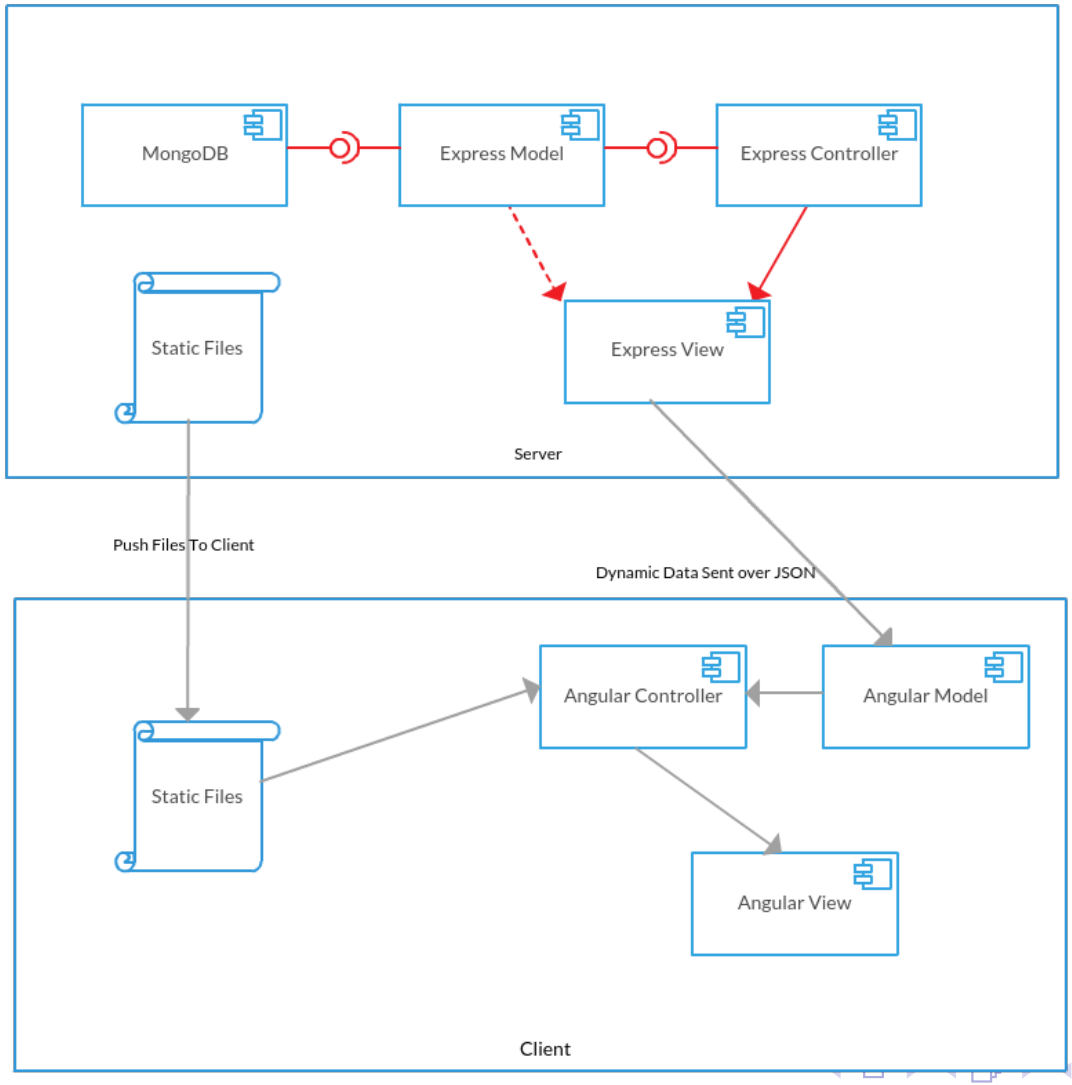
- Funktionale Anforderungen:
  - Essential: Implementation der Anwendungsfälle
  - Optional: Verschiedene Convenience Features
- Nicht-funktionale Anforderungen:
  - Sicherheit
  - Zuverlässigkeit
  - Benutzbarkeit
  - Effizienz
  - Wartbarkeit

# Technologsiestack

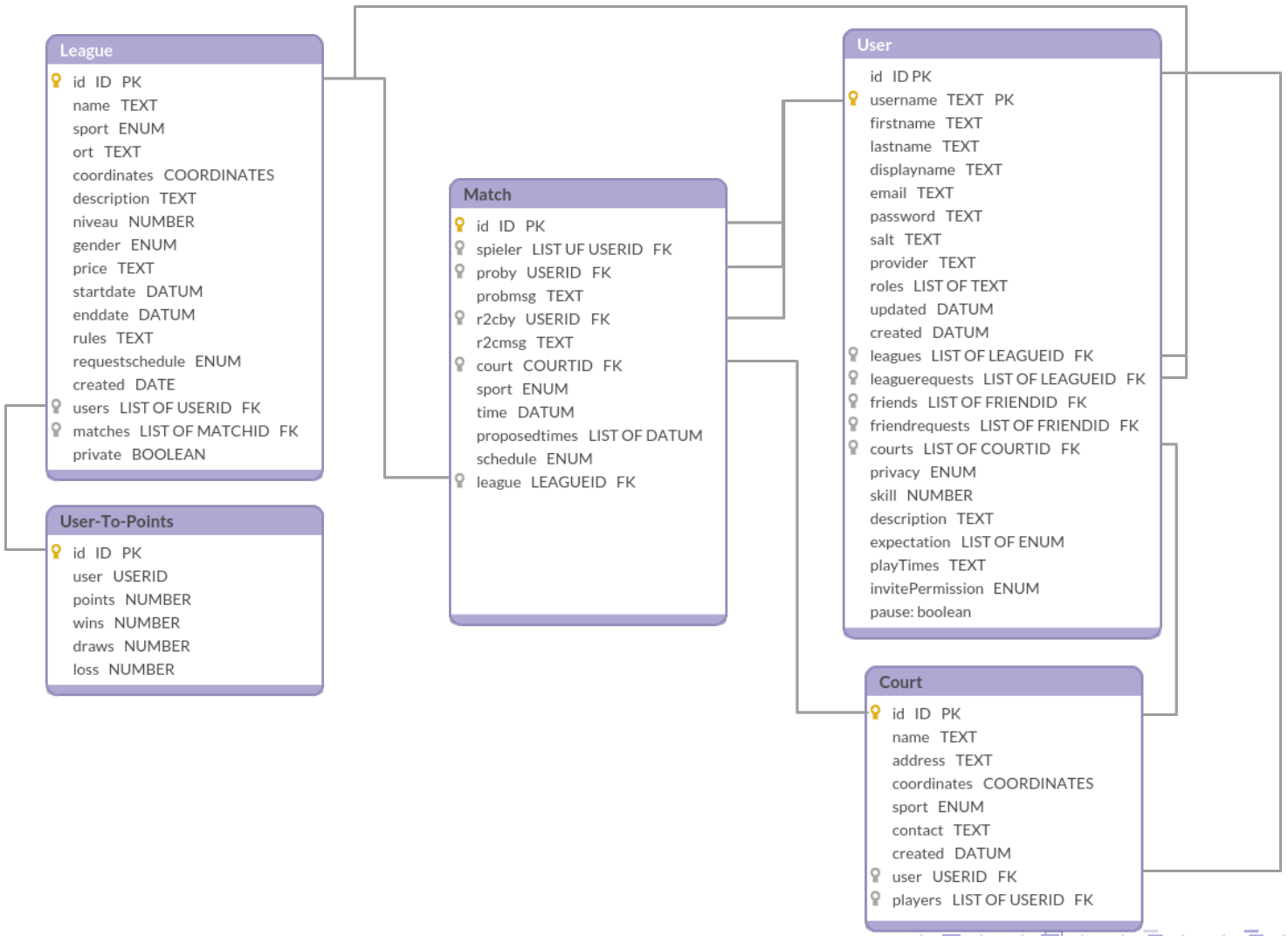
- Mean Stack:
  - M - MongoDB
  - E - ExpressJS
  - A - AngularJS
  - N - NodeJS
- JHipster
  - Spring
  - AngularJS

| Anforderung | MEAN | JHipster |
|---|---|---|
| Qualitätsmerkmal: Funktionalität | | |
| NREQ0.01 — Sicherheit | 9 | 10 |
| Qualitätsmerkmal: Zuverlässigkeit | | |
| NREQ1.01 — Fehlertoleranz | 10 | 9 |
| NREQ1.02 — Wiederherstellbarkeit | 10 | 9 |
| Qualitätsmerkmal: Effizienz | | |
| NREQ3.01 — Effizient in Programmierung | 10 | 6 |
| NREQ3.02 — Effizient in Installation | 10 | 7 |
| Qualitätsmerkmal: Wartbarkeit | | |
| NREQ4.01 — Einfach erweiterbar/änderbar | 10 | 7 |
| NREQ4.02 — Stabilität | 8 | 10 |
| NREQ4.03 — Testbarkeit | 10 | 10 |
| NREQ4.04 — Analysierbarkeit | 9 | 9 |
| Total | 86 | 77 |

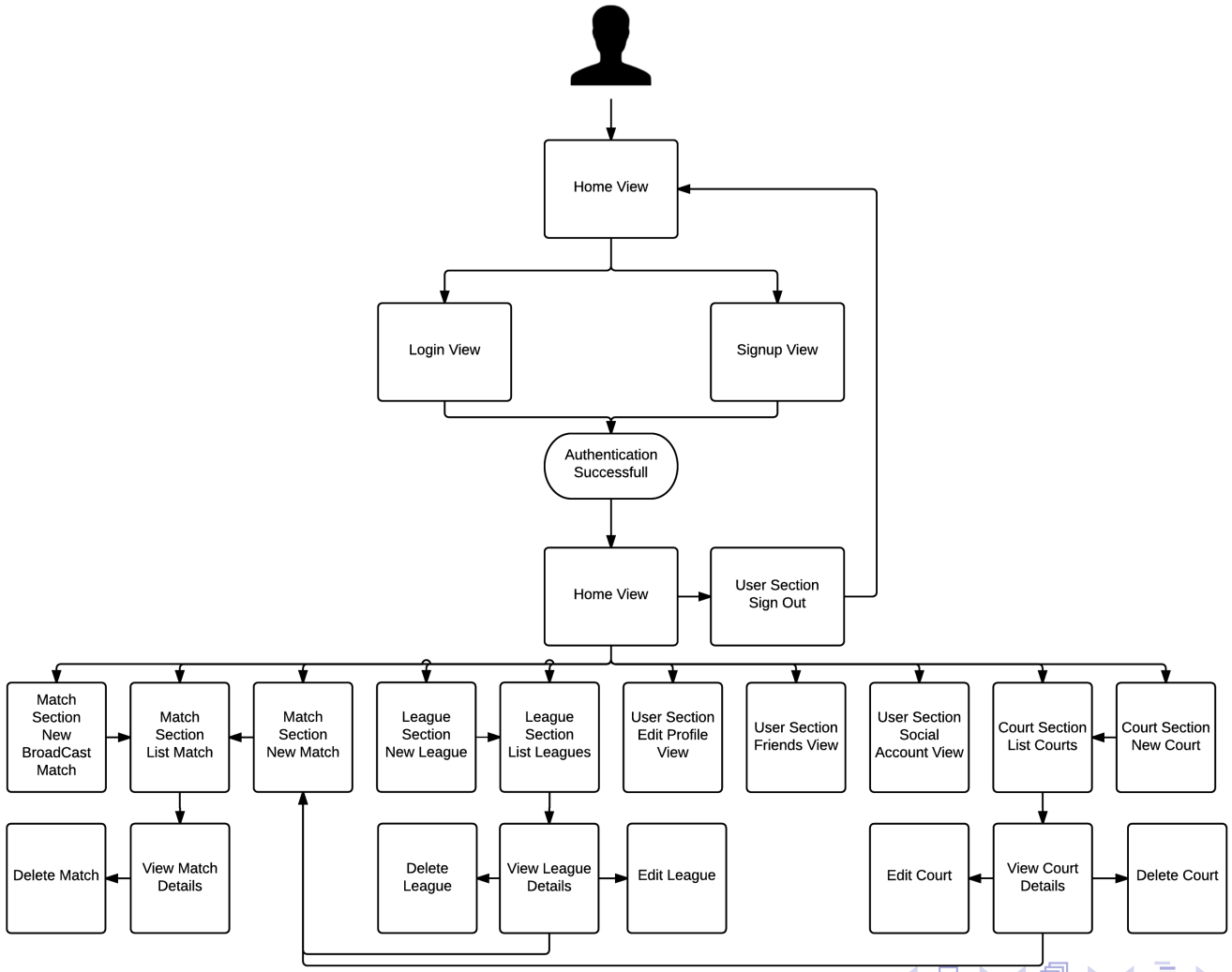
# Kontext


MongoDB
Express Model
Express Controller
Static Files
Express View
Server
Push Files To Client
Dynamic Data Sent over JSON
Angular Controller
Angular Model
Static Files
Angular View
Client

# Persistenz / Business View



# User Interface



# API

- Die REST API besteht aus fünf Endpunkten: /matches, /leagues, /courts, /users, /core
- Beispiel: /leagues/join

```
1  exports.join = function (req, res) {
2
3      var league = req.league;
4      var user = req.user;
5      var userToPoints = new UserToPoints();
6      userToPoints.user = req.user;
7      userToPoints.save();
8
9      league.users.push(userToPoints);
10     user.leagues.push(league)
11
12     console.log(league);
13     league.save(function (err) {
14         ....
```

# Web Applikation

- JavaScript MVC Framework
- JavaScript bezieht Models über API und stellt sie dar

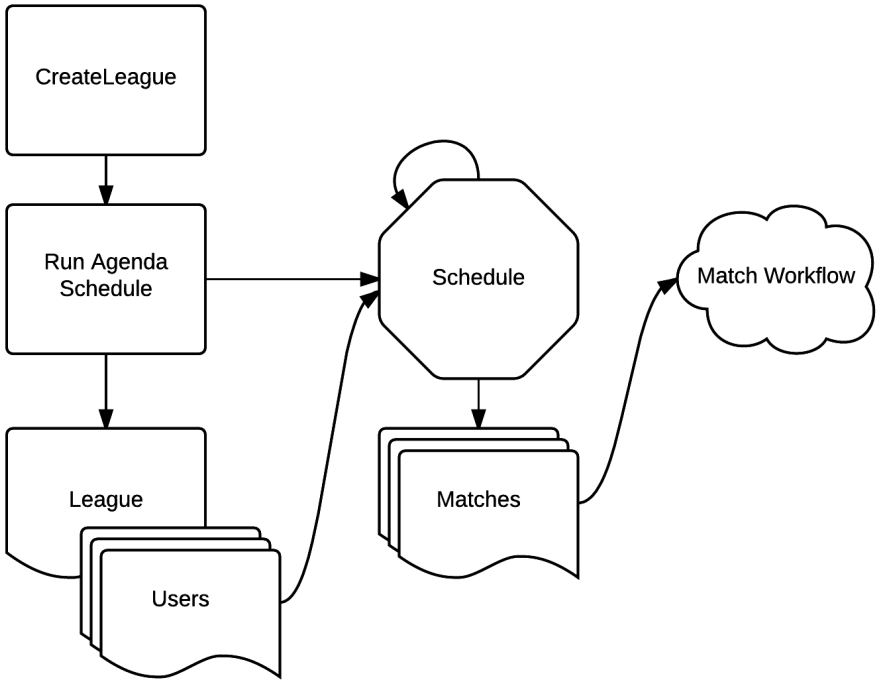
```
<section data-ng-controller="CourtsController" data-ng-init="find()">
    ....
        <table id="courtslist" class="table">
            <tr>
                <th>Name</th><th>Adresse</th><th>Verfügbare Sportarten</th>
            </tr>

            <tr data-ng-repeat="court in courts" id="{{court._id}}" ng-click="go(
    court)"  onMouseover="this.bgColor='#DDDDDD'" onMouseout="this.bgColor='#FFFFFF
    '">
                <td data-ng-bind="court.name"></td>
                <td data-ng-bind="court.address"></td>
                <td data-ng-bind="court.sports"></td>
            </tr>
        </table>
....
```

# Android Applikation

- Stellt über die WebView die Mobil-Seite dar
- Wurde generiert über GoNative.io

# Workflows

- Komplexe Workflows
- Einfaches Beispiel: automatische Herausforderung



# Test

- Jede funktionale Anforderung wurde mit Testfälle getestet
- Nicht-funktionale Anforderungen wurden nicht vollumfänglich getestet da PoC
- Unit Tests für Models und Controller wurden mit Karma(Client) und Mocha (Server) implementiert

# Reflexion

- Komplexere Workflows als gedacht
- Analyse und Dokumentation benötigten mehrere Iterationen
- Auswahl von Technologiestack war nicht im Vordergrund
- Zeitintensive Implementation, da neue Sprache und komplexe Algorithmen
- Zeitmanagement war nicht optimal, gegen Ende wurde Zeit knapp

# Fragen