



**HSR**

HOCHSCHULE FÜR TECHNIK  
RAPPERSWIL

FHO Fachhochschule Ostschweiz

Bachelorarbeit

# Architekturkonzepte moderner Web-Applikationen

Hochschule für Technik Rapperswil  
Frühjahressemester 2013

Erstellt: 8. März 2013, 07:45

**Autoren** Manuel Alabor  
Alexandre Joly  
Michael Weibel

**Betreuer** Prof. Hans Rudin  
**Experte** Daniel Hildebrand  
**Gegenleser** tbd.

Our fancy abstract. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

# Inhaltsverzeichnis

1. Einleitung . . . . .	5
1.1. Involvierte Personen . . . . .	5
1.1.1. Team . . . . .	5
1.1.2. Betreuung & Bewertung . . . . .	5
2. Projektplanung . . . . .	7
2.1. Infrastruktur . . . . .	7
2.1.1. Projektverwaltung . . . . .	7
2.1.2. Entwicklungsumgebung . . . . .	7
2.1.3. Git Repositories . . . . .	7
2.1.4. Continuous Integration . . . . .	7
2.2. Meetings . . . . .	8
2.2.1. Regelmässiges Statusmeeting . . . . .	8
2.3. Phasenplanung . . . . .	8
2.4. Meilensteine . . . . .	9
2.5. Artefakte . . . . .	9
3. Technologie Evaluation . . . . .	10
3.1. Node.js . . . . .	10
4. Anforderungsanalyse . . . . .	11
4.1. Funktionale Anforderungen . . . . .	11
4.2. Nicht-Funktionale Anforderungen . . . . .	12
4.3. Use Cases . . . . .	13
5. Domainmodel . . . . .	14
6. Technische Architektur . . . . .	15
A. Abbildungen, Tabellen & Quellcodes . . . . .	16
B. Literatur . . . . .	17
C. Glossar . . . . .	18

---

D. Meetingprotokolle . . . . .	19
E. Aufgabenstellung . . . . .	26

## Kapitel 1 **Einleitung**

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetur id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

### 1.1. Involvierte Personen

#### 1.1.1. Team

Manuel Alabor

Lorem Ipsum

Alexandre Joly

Lorem Ipsum

Michael Weibel

Lorem Ipsum

#### 1.1.2. Betreuung & Bewertung

Prof. Hans Rudin

Lorem Ipsum

Kevin Gaunt

Lorem Ipsum

Daniel Hildebrand

Lorem Ipsum

## Kapitel 2 Projektplanung

### 2.1. Infrastruktur

<i>Ressource</i>	<i>URL</i>
<i>Projektverwaltung</i>	<a href="http://redmine.alabor.me/projects/ba2013">http://redmine.alabor.me/projects/ba2013</a>
<i>Code: Git Repository</i>	<a href="https://github.com/mweibel/ba">https://github.com/mweibel/ba</a>
<i>Code: CI</i>	<a href="https://travis-ci.org/mweibel/BA">https://travis-ci.org/mweibel/BA</a>
<i>Thesis: Git Repository</i>	<a href="https://github.com/mweibel/BA-Dokumentation">https://github.com/mweibel/BA-Dokumentation</a>
<i>Thesis: PDF</i>	<a href="http://mweibel.github.com/BA-Dokumentation/thesis.pdf">http://mweibel.github.com/BA-Dokumentation/thesis.pdf</a>
<i>Thesis: CI</i>	<a href="https://travis-ci.org/mweibel/BA-Dokumentation">https://travis-ci.org/mweibel/BA-Dokumentation</a>
<i>Meeting Protokollierung</i>	<a href="https://github.com/mweibel/BA-Dokumentation/wiki/Meetings">https://github.com/mweibel/BA-Dokumentation/wiki/Meetings</a>

Tabelle 2.1.: Projektrelevante URL's

#### 2.1.1. Projektverwaltung

Für die komplette Projektplanung, die Zeitrapportierung sowie das Issue-Management wird Redmine eingesetzt.

#### 2.1.2. Entwicklungsumgebung

Zur Entwicklung von Quellcode-Artefakten steht eine mit Vagrant[Has] provisionierte Virtual Machine bereit. Sie enthält alle notwendigen Abhängigkeiten und Einstellungen.

Eine IDE im klassischen Sinne wird nicht vorgeschrieben.

#### 2.1.3. Git Repositories

Sowohl Quellcodeartefakte als auch die in LaTeX formulierte Thesis (dieses Dokument) wird in auf GitHub abgelegten Git Repositories versioniert bzw. zentral gespeichert.

#### 2.1.4. Continous Integration

Dieses Projekt verwendet Travis CI als Continous Integration Lösung.

Beide Git Repositories (Code & Thesis) verfügen über einen Push-Hook welcher automatisch einen Build im CI-System auslöst.

## 2.2. Meetings

### 2.2.1. Regelmässiges Statusmeeting

Während der gesamten Projektdauer findet jeweils am Mittwoch um 10 Uhr ein wöchentliches Statusmeeting statt. Die Sitzung wird abwechselungsweise jeweils von einer Person aus dem Projektteam geführt sowie von einer anderen protokolliert.

Das Projektteam stellt die Agenda der aktuellen Sitzung bis spätestens am vorangehenden Dienstag Abend bereit.

## 2.3. Phasenplanung

Die Phasenplanung orientiert sich grob am RUP und ist unterteilt in eine *Inception*-, *Elaboration*-, fünf *Construction*- sowie jeweils eine *Transition*- und *Abschlussphase*.

!!ADD PHASES IMAGE HERE!!



## 2.4. Meilensteine

<i>ID</i>	<i>Meilenstein</i>	<i>Termin</i>	<i>Beschreibung</i>
<i>M1</i>	Abschluss Inception	10.03.2013	Die Aufgabenstellung wurde gem. Auftrag klar definiert und die Projektinfrastruktur ist aufgesetzt. Eine initiale Projektplanung besteht.
<i>M2</i>	Abschluss Elaboration	17.03.2013	Neben der Auswahl einer konkreten Technologie sind nun auch Guidelines definiert. Anforderungsdokumente sind erstellt und abgenommen. Zudem besteht ein initiales Software Architektur Dokument mit dazugehörigem Architekturprototypen.
<i>M3</i>	Abschluss Construction 1	31.03.2013	Das Fundament der Applikation wurde implementiert. Weiter wurden die ersten Use Cases der Priorität <i>Hoch</i> umgesetzt.
<i>M4</i>	Abschluss Construction 2	14.04.2013	Alle Use Cases der Priorität <i>Hoch</i> sind umgesetzt.
<i>M5</i>	Abschluss Construction 3	28.04.2013	
<i>M6</i>	Abschluss Construction 4	12.05.2013	
<i>M7</i>	Abschluss Construction 5	26.05.2013	
<i>M8</i>	Abschluss Transition	02.06.2013	
<i>M9</i>	Abgabe HSR Artefakte	07.06.2013	Das A0-Poster sowie die Kurzfassung der Bachelorarbeit sind dem Betreuer zugestellt.
<i>M10</i>	Abgabe Bachelorarbeit	14.06.2013	Alle abzugebenden Artefakte sind dem Betreuer zugestellt worden.

Tabelle 2.2.: Meilensteine

## 2.5. Artefakte

Dieser Abschnitt beschreibt alle Arbeitsprodukte (Artefakte), welche zwingend erstellt und abgegeben werden müssen.

!!!!!! TODO ADD ARTEFACTS !!!!!!!!!!!!!

<i>ID</i>	<i>Meilenstein</i>	<i>Artefakt</i>	<i>Beschreibung</i>
<i>A10</i>	<i>M1</i>	Projektplanung	...
<i>A10</i>	<i>M1</i>	Projektplanung	...
<i>A10</i>	<i>M1</i>	Projektplanung	...
<i>A10</i>	<i>M1</i>	Projektplanung	...

Tabelle 2.3.: Abzugebende Artefakte

## Kapitel 3 Technologie Evaluation

### 3.1. Node.js

Node.js ..TODO

Web-Frameworks

Mögliche Auswahl

Name	Beschreibung	Pro/Kontra
<i>Express.js</i> [Exp, Express.js]	Minimales web application framework für single und multi-page sowie hybrid applikationen	<ul style="list-style-type: none"><li>+ Leichtgewichtige Abstraktion über Connect [Sen]</li><li>+ Sehr einfach zu sehen was passiert</li><li>+ Für anbindung an REST-Schnittstelle im Backend quasi optimal</li><li>- Kein ORM</li></ul>
<i>Geddy</i> [Ged]	Simple, strukturiertes web framework	<ul style="list-style-type: none"><li>+ MVC-Framework ähnlich wie Ruby on Rails [Han]</li><li>+ ORM mit Adapter u.a. für Postgres, in-memory, MongoDB und Riak</li></ul>
<i>Sails [bal]</i>	Realtime MVC Framework for Node.js	<ul style="list-style-type: none"><li>+ MVC-Framework ähnlich wie Ruby on Rails [Han]</li><li>+ Optimiert für den data-oriented style der web applikations-entwicklung</li></ul>

---

Tabelle 3.1.: Web-Frameworks in Node.js

## Kapitel 4 **Anforderungsanalyse**

### 4.1. Funktionale Anforderungen

<i><b>ID</b></i>	<i><b>Name</b></i>	<i><b>Beschreibung</b></i>	<i><b>Priorität</b></i>
F1	WG erstellen	Die Applikation erlaubt es eine WG zu erstellen.	hoch
F2	Einladung	Die Applikation erlaubt es, einen Benutzer in eine WG einzuladen.	hoch
F3	Aufgabe erstellen	Die Applikation erlaubt es, eine Aufgabe zu erstellen.	hoch
F4	Aufgabe erledigen	Die Applikation erlaubt es, eine Aufgabe zu erledigen.	hoch
F5	Verlassen	Die Applikation erlaubt es, eine WG zu verlassen.	mittel
F6	Aufgabe bearbeiten	Die Applikation erlaubt es, eine Aufgabe zu bearbeiten.	mittel
F7	Rangliste anzeigen	Die Applikation erlaubt es, eine Rangliste für die Bewohner einer WG anzuzeigen.	mittel
F8	Erfolge vergeben	Die Applikation erlaubt es, Erfolge aufgrund von Regeln zu vergeben.	mittel
F9	WG auflösen	Die Applikation erlaubt es, eine WG aufzulösen.	niedrig
F10	Bewohnerverwaltung	Die Applikation erlaubt es, die Bewohner einer WG zu verwalten.	niedrig
F11	Inhalte teilen	Die Applikation erlaubt es, Inhalte auf Social Media Kanälen zu teilen.	niedrig

Tabelle 4.1.: Funktionale Anforderungen

## 4.2. Nicht-Funktionale Anforderungen

<i>ID</i>	<i>Name</i>	<i>Beschreibung</i>
NF1	Antwortzeit	Die Applikation antwortet bei normalen Anfragen innerhalb von 0.2s.
NF2	Desktop Browserkompatibilität	Die Applikation unterstützt Internet Explorer 8 und höher, Chrome 25 und höher, Firefox 19 und höher sowie Safari 6 und höher.
NF3	Mobile Browserkompatibilität	Die Applikation unterstützt Safari 6.0 und Android Browser 4.0.
NF4	Sicherheit	Die Applikation kontrolliert den Zugriff auf geschützte Ressourcen.
NF5	ROCA Prinzipien	Die Applikation entspricht den ROCA [ROC] Prinzipien.

Tabelle 4.2.: Nicht-Funktionale Anforderungen

## 4.3. Use Cases

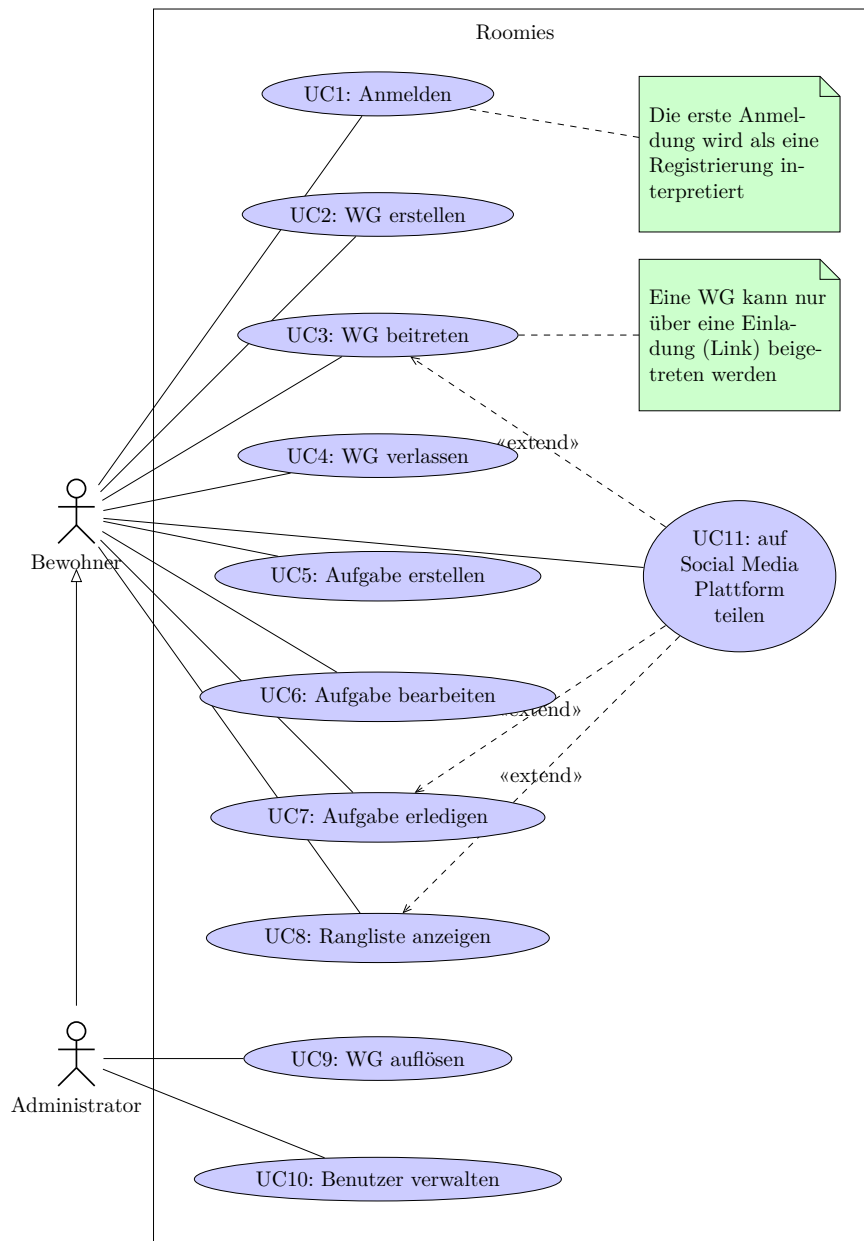


Abbildung 4.1.: Use Case Diagramm

## Kapitel 5 Domainmodell

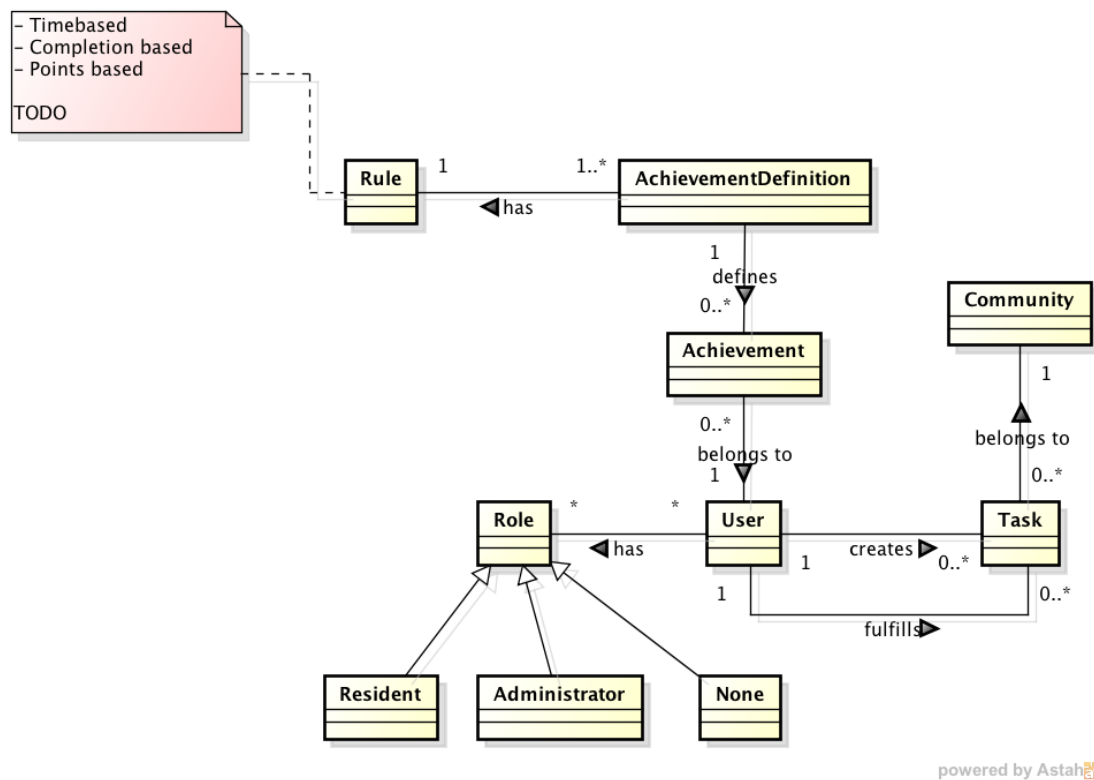


Abbildung 5.1.: Domain Modell

## Kapitel 6 **Technische Architektur**

## Anhang A **Abbildungen, Tabellen & Quellcodes**

### Abbildungsverzeichnis

4.1. Use Case Diagramm . . . . .	13
5.1. Domain Modell . . . . .	14

### Tabellenverzeichnis

2.1. Projektrelevante URL's . . . . .	7
2.2. Meilensteine . . . . .	9
2.3. Abzugebende Artefakte . . . . .	9
3.1. Web-Frameworks in Node.js . . . . .	10
4.1. Funktionale Anforderungen . . . . .	11
4.2. Nicht-Funktionale Anforderungen . . . . .	12

### Quellcodeverzeichnis



## Anhang B **Literatur**

- [bal] balderashy. *Sails / The future of API development*. URL: <http://sails.org> (besucht am 01.03.2013).
- [Exp] Express.js. *Express - node.js web application framework*. URL: <http://expressjs.com/> (besucht am 27.02.2013).
- [Fie00] Roy Fielding. "Chapter 5, Representational State Transfer". dissertation. University of California, Irvine, 2000. URL: [http://www.ics.uci.edu/~fielding/pubs/dissertation/rest\\_arch\\_style.htm](http://www.ics.uci.edu/~fielding/pubs/dissertation/rest_arch_style.htm).
- [Ged] Geddy. *Geddy / The original MVC Web framework for Node - a simple, structured way to create full stack javascript applications*. URL: <http://geddyjs.org/> (besucht am 27.02.2013).
- [Han] David Heinemeier Hansson. *Ruby on Rails*. URL: <http://rubyonrails.org>.
- [Has] HashiCorp. *Vagrant*. URL: <http://www.vagrantup.com/> (besucht am 06.03.2013).
- [ROC] ROCA. *Resource-oriented Client Architecture*. URL: <http://roca-style.org> (besucht am 06.03.2013).
- [Sen] Senchalabs. *Connect - High-quality middleware for node.js*. URL: <http://www.senchalabs.com/connect/> (besucht am 01.03.2013).

## Anhang C **Glossar**

### *Benutzer*

Ein Benutzer .... TOOODOOO. 11

### *Bewohner*

Ein Bewohner .... TOOODOOO. 11

### *CI*

Continous Integration. 7

### *IDE*

Integrated Development Environment. 7

### *Node.js*

Node.js ist ein Framework um Javascript auf dem Server laufen zu lassen. Es ist auf der V8 Engine von Google Chrome aufgebaut. 10

### *Push-Hook*

In *git* bezeichnet ein Push-Hook ein Script, welches nach jedem Commit ausgeführt wird.. 8

### *REST*

Representational State Transfer, definiert von Roy Fielding in seiner Dissertation [Fie00]. 10

### *RUP*

Rational Unified Process; Iteratives Projektvorgehen. 8

### *WG*

Eine WG .... TOOODOOO. 11

## Anhang D **Meetingprotokolle**

# **Bachelorarbeit Vorbesprechung 14. Februar 2013**

## **Teilnehmer**

- Hans Rudin, HRU (HSR)
- Kevin Gaunt, KGA (HSR)
- Daniel Hildebrand, DHI (Crealogix)
- Manuel Alabor, MAL (Team)
- Alexandre Joly, AJO (Team)
- Michael Weibel, MWE (Team, Protokoll)

## **Traktanden**

- StoryboardBuilder - Was ist der aktuelle Stand?
- Oder was habt ihr für Ideen?

## **Meeting**

### **StoryboardBuilder**

---

#### **Einführung DHI**

- Ende Januar Entscheid: Nicht weiterentwickeln
- Allerdings nicht weil Produkt/Markt nicht interessant wäre
- Ziel war: damit die Crealogix UX-Services zu unterstützen
- Crealogix wird keine UX-Services gegen aussen mehr anbieten
  - mehr interne Projekte betreuen
- Zwei Schwerpunkte: Education & Financial Services

- StoryboardBuilder gehört nicht in einen solchen Schwerpunkt
- Dies obwohl gutes Potential gesehen wird für das Produkt
- Anforderungsspezifikation verfeinert bis Ende Januar
- Technischer Prototyp gestartet, aber wieder gestoppt aufgrund der Neuorientierung
- Idee wäre: Storyboardbuilder an externe Firma weitergeben
- Bestehende Mitbewerber bauen ihre Angebote aus

## Diskussion BA

---

- Frage an die Runde: ist es interessant für euch, den Storyboardbuilder in der BA weiter zuentwickeln?
  - MAL: Wie würde das aussehen?
    - DHI: Beispiel aufgrund gewählter Technologie zu entwickeln
    - DHI: Auf Basis der fachlichen Spezifikation der Crealogix
    - HRU: BA sollte nicht zu einer Fleissarbeit werden
    - HRU: Was wären denn die Herausforderungen wenn das weiterentwickelt werden würde in BA?
    - HRU: Die momentane Ausgangslage ist anders, da kein wirklicher Kunde existiert
    - DHI: Es sind sicher einige Ideen da, die technisch Herausfordernd sind
    - DHI: Grafische Repräsentation auf Screen, Objektmodell umsetzen
    - DHI: Wie gesagt, hat auch nichts dagegen, wenn eine neue Arbeit gemacht werden würde
    - DHI: Laut Kenntnisstand von DHI sollte auch von seiten Industriepartner weniger Betreuung beinhalten, mehr von Team
- KGA: Was ist nun der Anspruch an das Meeting? Müssen wir am Ende des Meetings schon wissen was gemacht werden soll?
  - DHI: Ist offen, hat keinen Anspruch auf Entscheid jetzt - sollte aber bald geschehen, da Bachelorarbeit bald startet
  - DHI:

- DHI: Hat div. Alternativen die man anschauen könnte
- MAL: für ihn ist das Durchführen der BA mit dem Storyboardbuilder nicht mehr besonders interessant, aufgrund Änderung seitens Crealogix
  - DHI: Ja das stimmt, BA würde nicht mehr zu einem realen Produkt führen
- MWE: gehts ähnlich wie MAL
- DHI: Thema 1:
  - **Kino reservations system**
  - Kennt Kinobesitzer in rapperswil
  - Buchungssystem
  - Mit anbindung an Kassensystem
  - nicht nur einzelne Plätze
  - sondern auch Firmenanlässe, Frauenkino, Catering etc.
  - Konzeptionell entwerfen
  - soweit wie möglich implementieren
  - könnte sehr interessant sein, DHI's Meinung nach
- DHI: Thema 2
  - **Personal Finance Management**
  - Zusatz zu Ebanking
  - Eigene Zahlungen analysieren
  - Verschiedene Banken
  - Soviel für Versicherung, Einkauf, etc.
  - Basiert auf einer isländisch-schwedischen Firma
  - Integrationsarbeit (in ein Bankensystem einbauen)
  - MAL: Geht es darum, das zu integrieren und anschaulich darzustellen, und Data-Mining ist schon gemacht?
  - DHI: Ja
    - Schwierigkeit Sicherheit
    - Zertifizierung, Authentifizierung
    - nur Teilaspekt möglich zum lösen
    - Versch. Komponenten
    - Aspekt wichtig auf welcher sich die BA konzentrieren soll
  - DHI: Müsste das genauer anschauen wie das machbar wäre
    - Da es ein sehr grosses System wäre
    - Müsste verifizieren ob das gehen soll?

- HRU: Thema 2 wohl eher interessant (Team bejaht)
  - HRU: Wäre das so kurzfristig machbar?
    - DHI: müsste angeschaut werden
    - DHI: Verträge bestehen
    - DHI: anderes ist Ebanking in Java mit Oracle
      - DHI: Verfügbar machen möglich
  - HRU: Zwei Komponenten, was ist Webfrontend?
    - DHI: Netty server von airlock für authentifizierung
    - DHI: möglichst HTML das übers web geht
    - DHI: J2EE - JSP vorne
  - HRU: Isländische Software
    - DHI: .NET basiert
    - DHI: hat aber ein WCF/REST/AJAX schnittstelle welche relativ gut ins Frontend integrierbar wäre
    - DHI: von einem System ins andere System transferieren (Oracle zu MSSQL DB)
    - DHI: Statistisch aufwerten, und wieder anzeigen
    - DHI: machbarkeit unklar, muss verifiziert werden
- MAL: Was hat HRU für Projekte
  - MAL: Realtime sachen wären interessant (Mobile, Chat, Messaging?)
  - HRU: hat keine Projekte
- DHI: Hat evtl. noch andere Projekte, müsste das aber noch anschauen
- MAL: Bis Testumgebung steht würden wohl wochen vergehen
  - DHI: stimmt wohl
- MWE: Persönlich interessiert vorallem WebRTC
- HRU: was ist mit web realtimecommunication (WebRTC) gemeint?
- MWE: Near-realtime communication (Daten, Video, Audio) zw. Browsern
- MWE: Was wäre denn für Crealogix interessant - zentrale Frage?
  - DHI: Crealogix muss nicht unbedingt dabei sein, wenn nicht nötig
- DHI: update maus-scanner
  - MAL: wäre denn mobile auch ein Thema?
  - DHI: Mobile ist sehr zentral

- HRU: gibt es fraktionen (web/mobile) im Team?
  - MWE: Web-Mensch, aber Mobile wäre auch sehr interessant
- DHI: Fragt bei Crealogix CEO/entwicklungsleiter nach bzgl. Mobile
- MAL: Elearning wäre auch interessant bzgl. Mobile
  - DHI: könnte nachfragen obs da auch was geben würde?
- AJO: Auch vorallem Mobile interessant
  - arbeitet auch vorallem in Mobile
  - HRU: Auch sie MAL ;)
- DHI: müsste nachfragen, aber wäre sicher interessant
- MAL: Wäre sicher interessant auf DHI's Themen zu warten, andererseits müsste auch Teamintern bzw. mit KGA/HRU geschaut werden
- DHI: wann beginnt die Arbeit? - Montag
- DHI: 3 Bereiche Education
  - Campusmanagement
  - Time to learn
    - 30'000 die mit TTL arbeiten
    - Mit Center for young professional zusammenarbeit (evtl. da was interssantes)
    - In Bubikon
- DHI: was machen wenn nichts herauskommt?
  - DHI: würden gerne mit euch zusammenarbeiten, wenn möglich
- HRU: Wäre schon der Weg zum gehen
  - parallel müssten überlegungen angestellt werden obs was anderes geben würde
  - gibt den 3 Studierenden möglichst freie Hand
- KGA: Termin bis wann die Entscheidung möglich sein
  - HRU: allerspätistens erste Woche
  - DHI: wird noch heute mit den 3 Crealogix leuten schauen
  - DHI: Bis morgen, 15.02. Antwort wenn möglich
  - MAL: Mittwoch zu spät oder zu früh?
  - HRU: Wann sind sie @HSR?
    - MAL: MO/DI/MI



- HRU: Mittwoch wäre nicht unbedingt zu spät
- MAL: Mittwoch wäre Zusammenkunft, um definitiv zu entscheiden.  
Aber mit Kommunikation bis dann
- MAL: DHI kann sicher schnell entscheiden, je nach dem wäre pers.  
Anwesenheit nicht nötig
- DHI: würde gerne persönlich dabei sein
- DHI: gibt morgen Feedback
- HRU: das ist gut - in der Runde Team/HSR diskutieren was gemacht werden kann
- KGA: Was wird mit UX passieren? (Expert Talks)
  - DHI: Prio 1 hat interne Aufträge und Prio 2 externe

## Diskussion im Team

---

- KGA: also ist keines der beiden Thema sehr interessant für Team?
- Team: Ja, insbesondere auch zu gross für die kurze Zeit (Thema 2)
- HRU: Reservationssystem vor allem Businessanalyse
- HRU: Sicher gut zu schauen was DHI einbringt
- HRU: Aber auch schauen was wir für Ideen haben
- KGA: was wäre ursprüngliche Idee für SA gewesen
- MWE: XMPP Server in node.js modular, flexibel bzgl.  
Datenbankanbindung
- MAL: und auch Skalierbarkeit von node.js interessant
- MAL: interessant wäre vielleicht frontend framework
- HRU: Alle miteinander auf dem Laufenden halten bzgl. Ideen

## Nächstes Meeting

---

- Mittwoch, 20. Februar 2013, 10:10 Uhr

## Anhang E **Aufgabenstellung**

Die folgenden drei Seiten enthalten die offizielle Aufgabenstellung dieser Bachelorarbeit.

---

## Aufgabenstellung Bachelorarbeit für Manuel Alabor, Alexandre Joly und Michael Weibel „Architekturkonzepte moderner Web-Applikationen“

---

### 1. Auftraggeber, Betreuer und Experte

Bei dieser Arbeit handelt es sich um eine HSR-interne Arbeit zur Unterstützung des Moduls Internettechnologien.

Auftraggeber/Betreuer:

- Prof. Hans Rudin, HSR, IFS [hрудin@hsr.ch](mailto:hрудin@hsr.ch) +41 55 222 49 36 (Verantw. Dozent, Betreuer)
- Kevin Gaunt, HSR, IFS [kgaunt@hsr.ch](mailto:kgaunt@hsr.ch) +41 55 222 4662 (Betreuer)

Experte:

- Daniel Hildebrand, Crealogix

### 2. Studierende

Diese Arbeit wird als Bachelorarbeit an der Abteilung Informatik durchgeführt von

- Manuel Alabor [malabor@hsr.ch](mailto:malabor@hsr.ch)
- Alexandre Joly [ajoly@hsr.ch](mailto:ajoly@hsr.ch)
- Michael Weibel [mweibel@hsr.ch](mailto:mweibel@hsr.ch)

### 3. Ausgangslage

Das Modul Internettechnologien ist stark Technologie-zentriert. Wünschbar ist eine Weiterentwicklung (Aktualisierung, Verbesserung) mit vermehrter Beachtung von konzeptionellen und Architektur-Fragen. In letzter Zeit haben sich Prinzipien und Konzepte herauskristallisiert, nach denen Web-Applikationen am besten aufgebaut werden. Siehe zum Beispiel [1] oder [2]. Um diese Prinzipien und Konzepte anschaulich zu vermitteln, braucht es neben Erläuterungen möglichst anschauliche Beispiele und Übungsaufgaben. Ziel dieser Arbeit ist es, die Weiterentwicklung des Moduls Internettechnologien entsprechend zu unterstützen.

### 4. Aufgabenstellung

In dieser Arbeit sollen die in [1], [2] und weiteren Quellen dargestellten Prinzipien und Konzepte analysiert werden. Gemeinsam mit dem Betreuer sollen daraus in das Modul Internettechnologien zu transferierende Prinzipien und Konzepte ausgewählt werden, und es soll überlegt werden, wie diese Inhalte anschaulich für den Unterricht aufbereitet werden können. In der Folge sollten entsprechende Resultate erarbeitet werden, welche das Unterrichten der ausgewählten Inhalte möglichst gut unterstützen. Eine wichtige Rolle dürfte dabei eine anschauliche Beispielapplikation bilden.

Details werden im Verlauf der Arbeit zwischen Studierenden und Betreuer vereinbart.

## 5. Zur Durchführung

Mit dem Betreuer finden wöchentliche Besprechungen statt. Zusätzliche Besprechungen sind nach Bedarf durch die Studierenden zu veranlassen.

Alle Besprechungen sind von den Studierenden mit einer Traktandenliste vorzubereiten, die Besprechung ist durch die Studierenden zu leiten und die Ergebnisse sind in einem Protokoll festzuhalten, das den Betreuern und dem Auftraggeber per E-Mail zugestellt wird.

Für die Durchführung der Arbeit ist ein Projektplan zu erstellen. Dabei ist auf einen kontinuierlichen und sichtbaren Arbeitsfortschritt zu achten. An Meilensteinen gemäss Projektplan sind einzelne Arbeitsresultate in vorläufigen Versionen abzugeben. Über die abgegebenen Arbeitsresultate erhalten die Studierenden ein vorläufiges Feedback. Eine definitive Beurteilung erfolgt auf Grund der am Abgabetermin abgelieferten Dokumentation.

## 6. Dokumentation

Über diese Arbeit ist eine Dokumentation gemäss den Richtlinien der Abteilung Informatik zu verfassen (siehe <https://www.hsr.ch/Allgemeine-Infos-Diplom-Bach.4418.0.html>). Die zu erstellenden Dokumente sind im Projektplan festzuhalten. Alle Dokumente sind nachzuführen, d.h. sie sollten den Stand der Arbeit bei der Abgabe in konsistenter Form dokumentieren. Alle Resultate sind vollständig auf CD/DVD in 3 Exemplaren abzugeben. Der Bericht ist ausgedruckt in doppelter Ausführung abzugeben.

## 7. Referenzen

- [1] Stefan Tilkov  
Building large web-based systems: 10 Recommendations  
Präsentation an der OOP 2013, München  
PDF als Beilage
- [2] <http://roca-style.org>  
ROCA Resource-oriented Client Architecture - A collection of simple recommendations for decent Web application frontends

## 8. Termine

Siehe auch Terminplan auf <https://www.hsr.ch/Termine-Diplom-Bachelor-und.5142.0.html>.

<b>Montag, den 18. Februar 2013</b>	<b>Beginn der Bachelorarbeit, Ausgabe der Aufgabenstellung durch die Betreuer</b>
<b>7. Juni 2013</b>	Abgabe Kurzbeschreibung und A0-Poster. Vorlagen stehen unter den allgemeinen Infos Diplom-, Bachelor- und Studienarbeiten zur Verfügung.
<b>14. Juni 2013, 12:00</b>	<b>Abgabe</b> der Arbeit an den Betreuer bis 12.00 Uhr. Fertigstellung des A0-Posters bis 12.00 Uhr. Abgabe der Posters im Abteilungssekretariat 6.113.
<b>14. Juni 2012</b>	HSR-Forum, Vorträge und Präsentation der Bachelor- und Diplomarbeiten, 16 bis 20 Uhr
<b>5.8. - 23.08.2013</b>	Mündliche Prüfung zur Bachelorarbeit

## 9. Beurteilung

Eine erfolgreiche Bachelorarbeit zählt 12 ECTS-Punkte pro Studierenden. Für 1 ECTS Punkt ist eine Arbeitsleistung von 30 Stunden budgetiert (Siehe auch Modulbeschreibung der Bachelorarbeit [https://unterricht.hsr.ch/staticWeb/allModules/19419\\_M\\_BAI.html](https://unterricht.hsr.ch/staticWeb/allModules/19419_M_BAI.html)).

Für die Beurteilung ist der HSR-Betreuer verantwortlich.

Gesichtspunkt	Gewicht
1. Organisation, Durchführung	1/6
2. Berichte (Abstract, Mgmt Summary, technischer u. persönliche Berichte) sowie Gliederung, Darstellung, Sprache der gesamten Dokumentation	1/6
3. Inhalt*)	3/6
4. Mündliche Prüfung zur Bachelorarbeit	1/6

\*) Die Unterteilung und Gewichtung von 3. Inhalt wird im Laufe dieser Arbeit mit den Studierenden festgelegt.

Im Übrigen gelten die Bestimmungen der Abteilung Informatik für Bachelorarbeiten.

Rapperswil, den 20. Februar 2013



Prof. Hans Rudin  
 Institut für Software  
 Hochschule für Technik Rapperswil