

# Protokoll BA Meeting 22.09.14

<b>Datum, Zeit</b>	<b>Mo 2.09.14., 13:00 - 14:10</b>
<b>Ort</b>	Rapperswil, HSR, 6.112
<b>Teilnehmende</b>	Olaf Zimmermann, Tobias Blaser, Laurin Murer
<b>Protokoll</b>	Tobias Blaser, Laurin Murer

## Traktanden

1. Weitere Meetings planen
2. Nutzungsvereinbarung
3. Aufgabenstellung
4. GitHub Repo / Zugriff
5. Meilensteine / Planung / Status
6. Zitieren/Literaturhinweise

## Meetings

Wöchentlich am Montag, 13:00.

## Nutzungsvereinbarung

Wurde angepasst, wird von den Studenten gedruckt, unterschrieben und ZIO zum Unterschreiben ins Fach gelegt.

## Aufgabenstellung

Wurde ZIO zum Review gesendet während dem Meeting.

## GitHub Repo

ZIO schickt den Studenten seinen Benutzernamen.

## Milestones

- Milestone Beschreibung von "Architektur Konzept" mit "Analyse" ergänzen
- Milestone Beschreibungen als Ziel definieren und nicht als Aktivität (z.B. "Infrastructure ready", "First draft available")

## Zitieren

Zitatform ist o.k. so, allgemein: Autor, Titel, wo veröffentlicht (z.B. URL mit Timestamp) z.B. wie <http://dblp.uni-trier.de/rec/bibtex/conf/wicsa/MiksovicZ11>

## Protokoll

- Neuen Punkt "Beschlüsse" einfügen.
- Keine Antwort == Protokoll angenommen

## Wunsch vom Betreuer

Links jeweils nicht als Text, sondern direkt als URL (z.B. bei Aufforderung für Review per Mail den Link gleich mitschicken zum draufklicken)

## Input vom Betreuer

### Domain

statischer Baum: Inhaltsverzeichnis über den Inhalt. dynamischer Baum: Auflistung, was von was abhängt (Entscheidungsflow)

Eine Option kann sein, dass wir das GUI der CDAR "ignorieren" und lediglich das API verwenden. Und evtl. gar ein eigenes API schreiben oder das bestehende erweitern.

Starke Präferenz für Java von ZIO

ZIO empfiehlt uns [Developing in the Cloud](#) aus dem IEEE Software March/April 2014 Seite 41. Darin hat es eine Reihe von Issue-Tracking Software. Idee: einige davon anschauen und schauen, ob es mit unserem zu entwickelnden Tool funktioniert.

### Requirements:

- Desto weniger Abhängigkeiten, desto besser (Software Base)
- Lizenzen: für jede einzelne Komponenten nachweisen
- Technologiepark: Für Benutzer möglichst Benutzer klein halten -> weniger ist mehr, möglichst auf bestehende und weit verbreitete Technologien setzen

## CDAR-BA

- Sie hatten ein Code-Review erhalten von ZIO.

## TODOs

Studenten:

- Aufgabenstellung an ZIO schicken zum Review
- Nutzungsvereinbarung ausdrucken, unterschreiben, ins Fach legen

ZIO:

- GitHub Benutzernamen an Studenten schicken -> Studenten laden zum Repo ein.

- Aufgabenstellung reviewen

## Beschlüsse

- Protokoll kommt spätestens 48h nach Sitzung an ZIO
- Das EEPPI ermöglicht das Einpflegen der zu treffenden Entscheidungen und Optionen (also der EE-Teil in EEPPI) **ausschliesslich** über (die API) des CDAR