

# Internet of Things Smart Home

## Bachelorarbeit

HSR - Hochschule für Technik Rapperswil  
Institute for networked Solutions

Dokumentation

Autoren: Marco Leutenegger, Dominik Freier

Betreuer: Prof. Hansjörg Huser

Gegenleser: <td>Prof. TODO

# Abstract

<td>

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

# Management Summary

<td>

## Ausgangslage

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea.

## Vorgehen / Technologien

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea.

## Ergebnisse

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea.

## Ausblick

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea.

## Eigenständigkeitserklärung

### Erklärung

Wir erklären hiermit,

- dass wir die vorliegende Arbeit selber und ohne fremde Hilfe durchgeführt haben, ausser derjenigen, welche explizit in der Aufgabenstellung erwähnt sind oder mit dem Betreuer schriftlich vereinbart wurde,
- dass wir sämtliche verwendeten Quellen erwähnt und gemäss gängigen wissenschaftlichen Zitierregeln korrekt angegeben haben.
- dass wir keine durch Copyright geschützten Materialien (z.B. Bilder) in dieser Arbeit in unerlaubter Weise genutzt haben.

Ort, Datum:

Rapperswil, <TBD>



Marco Leutenegger



Dominik Freier

# Danksagung

<td>

Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

# Inhaltsverzeichnis

<b>Abstract</b>	<b>2</b>
<b>Management Summary</b>	<b>3</b>
<b>Danksagung</b>	<b>5</b>
<b>Technischer Bericht</b>	<b>7</b>
1. Ausgangslage . . . . .	7
2. Problembeschreibung . . . . .	8
2.1. Motivation . . . . .	8
2.2. Funktionale Anforderungen . . . . .	8
3. Lösungskonzept . . . . .	11
3.1. Allgemeine Systemsicht . . . . .	11
4. Umsetzung . . . . .	12
4.1. Technologie und Plattform . . . . .	12
4.2. openHAB . . . . .	12
4.3. Deploymentübersicht . . . . .	15
<b>Glossar</b>	<b>17</b>
<b>A. Projektplan</b>	<b>18</b>
<b>B. Sitzungsprotokolle</b>	<b>32</b>
<b>C. Persönliche Reflektion</b>	<b>36</b>

# Technischer Bericht

## 1. Ausgangslage

Diese Bachelorarbeit befasst sich mit einem Teilgebiet des «Internet of Things», nämlich der lokalen Vernetzung von Sensoren und Aktoren.

Gemäss der Aufgabenstellung soll eine Smart-Home Beispielapplikation erstellt werden, welche wesentliche Aspekte einer IoT-Anwendung demonstriert. Das beinhaltet das Steuern von Devices, Lesen von Sensoren, Event-Verarbeitung, Überwachung und intelligente Abläufe steuern.

Das System soll auf einer tragbaren, erweiterbaren Architektur aufgebaut werden und Microsoft Azure als Cloud Plattform benutze

Die Aufgabenstellung sieht kein Anwendungsszenario vor und kann somit, in einem definierten Rahmen, selbst definiert werden.

## **2. Problembeschreibung**

### **2.1. Motivation**

Als Smart-Home Beispielszenario soll ein System aufgesetzt werden, das einen grossen Bezug zur Realität hat. Es soll für aussenstehende Personen attraktiv und nachvollziehbar sein und einen Mehrwert mit sich bringen.

### **2.2. Funktionale Anforderungen**

#### **2.2.1. Basisszenario**

Während der Analyse für das Basisszenario hat sich der Einsatz des openHAB Smart-Home Framework ergeben. Dieses Framework bietet eine Basis für das versenden von Events, definieren von Regeln, und anschliessen (über Bindings) von Aktoren bzw. Sensoren.

##### **F01: Sensoren Status**

Der Status eines Sensors kann abgefragt werden und aufgrund dessen sollen bestimmte Aktionen ausgeführt werden.

##### **F02: Steuern von Aktoren**

Von einem Client wird ein Command gesendet, der an einen Aktor weitergeleitet wird. Beim Empfang eines Commands ändert der Aktor seinen Status. Der Status wird im openHAB aktualisiert.

##### **F03: Persistieren der Events in Cloud**

Alle Events/Commands, die von Aktoren, Sensoren und Client gesendet werden, werden in der MS Azure Cloud persistiert. Sie werden in einer optimierten Form abgelegt, damit später Statistiken erstellt werden können (nicht Teil der Arbeit).

##### **F04: Regeln**

Aufgrund von Statusänderungen sollen vordefinierte Aktionen ausgelöst werden. Das betrifft einerseits das Ändern eines Zustandes eines Aktors, andererseits das propagieren von Notifikationen an einen Client.



### 2.2.2. Lösungsteil (Demo-System)

#### L01: Sicherheits-Status abfragen

Der Client soll den Status des Einbrecherschutzes abgefragt werden können.

Status «OK»:

- Fenster ist geschlossen
- Türe ist geschlossen
- keine Bewegung detektiert

Status «NOK»:

- Fenster ist offen
- Türe ist offen
- Bewegung detektiert

#### L02: Überwachungskamera

Der Client kann die Überwachungskamera ein- bzw. ausschalten und Livebilder anfordern.

#### L03: Event Kontaktsensor

Der Kontaktsensor hat permanent einen Status. Der Status ist entweder «offen» oder «geschlossen».

#### L04: Event Bewegungsmelder

Sobald der Bewegungsmelder eine Bewegung registriert, sendet dieser einen Event. Dieser wird nach interner Logik verarbeitet.

#### L05: Aktor: Philips Hue

Der Aktor wird via Command angesteuert. Das Licht kann durch eine Regel (Zeit-Mechanismus zur Prävention) oder durch eine Aktion des Clients ein bzw. abgeschaltet werden.

#### L06: Aktor: NFC Sticker

Die NFC Stickers können sehr vielfältig eingesetzt werden. Generell wird durch Auflegen eines NFC-fähigen Smartphones eine Aktion ausgeführt. Was diese Aktion genau beinhaltet ist offen. Beispielsweise könnte das Sicherheitssystem «scharfgestellt» werden.

**L07: Aktor: Funksteckdose**

Die Funksteckdose kann ebenfalls vielseitig eingesetzt werden. Etwas abstrahiert betrachtet, kann jedes Gerät per Remote ein- bzw. ausgeschaltet werden. An dieser Funksteckdose kann zum Beispiel eine Musikanlage oder ein Fernsehgerät eingeschaltet werden.

### 3. Lösungskonzept

Dieses Kapitel beinhaltet die Beschreibung der Architektur und wichtiger Komponenten. Die eingesetzten Technologien und genauen Implementationsdetails stehen im Hintergrund und werden im Kapitel 4 aufgegriffen. Neben der Architektur beinhaltet das Lösungskonzept auch eine allgemeine Schnittstellenbeschreibung. Informationen zur technischen Anbindung an die Schnittstellen können im Abschnitt 7 ~~<td>~~im Anhang entnommen werden.

#### 3.1. Allgemeine Systemsicht

Anhand der Problembeschreibung wurde ein Plan erarbeitet, der das ganze System verständlich beschreibt. Abbildung 1 stellt die wichtigsten Komponenten und Entitäten aus der Problemdomäne in gegenseitiger Beziehung dar.

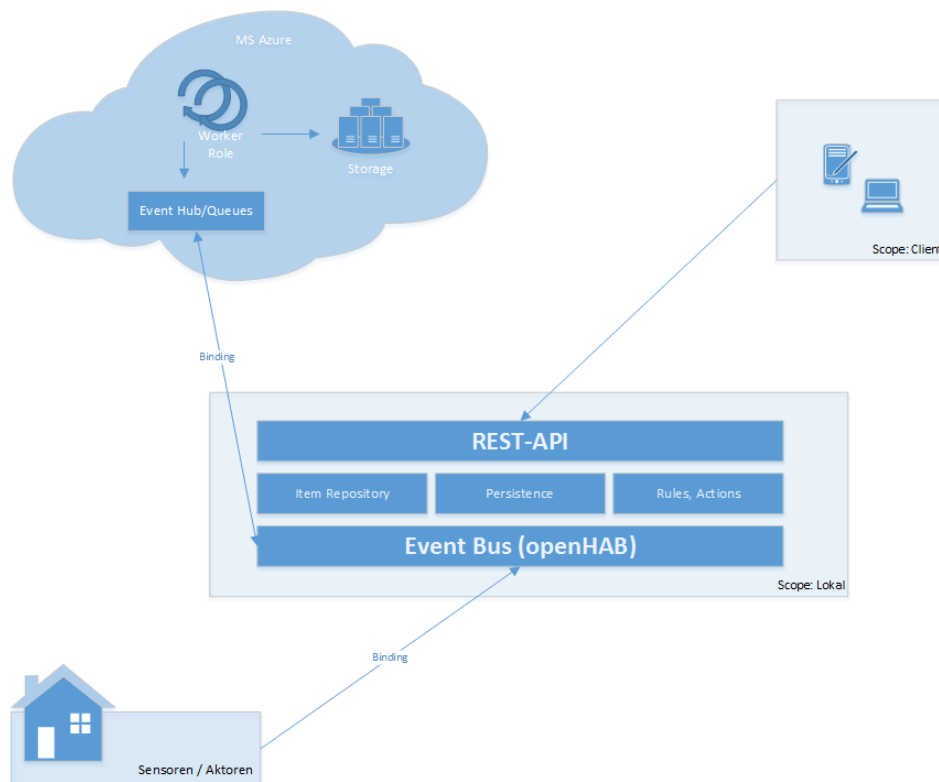


Abbildung 1.: Systemübersicht

## 4. Umsetzung

### 4.1. Technologie und Plattform

Für unser Projekt werden verschiedene Technologien bzw. Plattformen eingesetzt. Auf der Clientseite, im SmartHome, wird openHAB mit verschiedenen Bindings eingesetzt. Cloudseitig wird MS Azure Cloud zur Persistierung von Events verwendet.

### 4.2. openHAB

Das System openHAB wird eingesetzt, um verschiedene Home-Automatisierungssysteme unter einen Hut zu bringen. Um dies zu realisieren bietet openHAB eine grosse Anzahl von Bindings mit, mit denen die verschiedenen Systeme angesprochen werden können.

#### Module

OpenHAB ist durch OSGi-Bundles modular aufgebaut und beinhaltet folgende Komponenten:

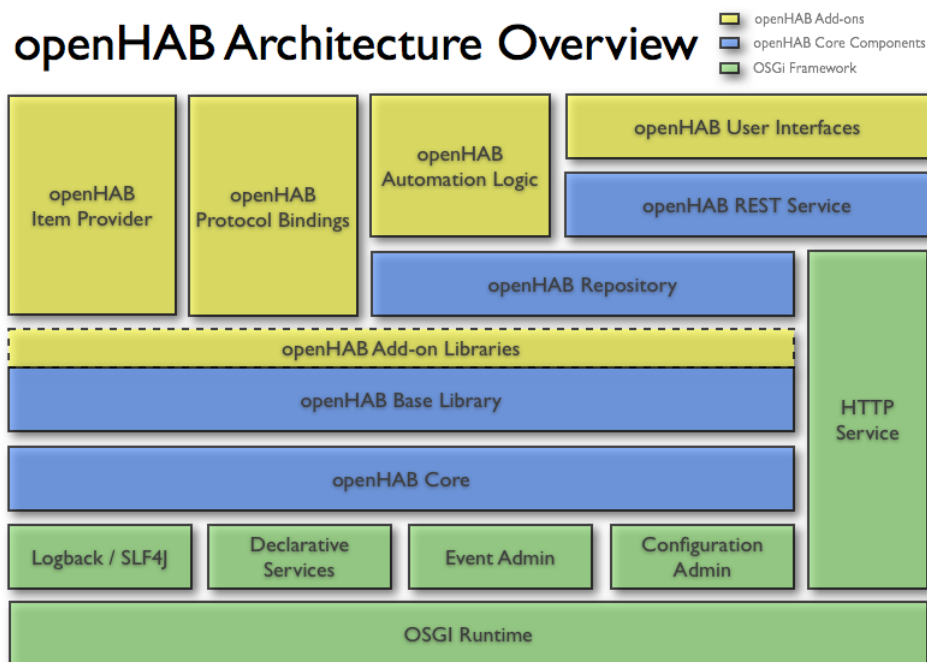


Abbildung 2.: openHAB Architektur

## Kommunikation

Der Basisservice von openHAB stellt der Event Bus. Über diesen Bus werden Events zwischen den verschiedenen Bundles gesendet. Die Events sind entweder Commands, welche eine Aktion ausführen, oder Status-Updates, welche Statusänderungen der Devices beinhaltet.

Durch den Einsatz dieses EventBus wird die Kopplung reduziert und können somit einfach ausgetauscht werden.

Für die Verwaltung der verschiedenen Status ist das Item Repository zuständig, welches permanent den Event Bus auf Status-Updates abhört und die Änderungen ins Repository schreibt. Falls auf einem GUI ein Status eines Devices angezeigt werden soll, kann dazu das Item Repository abgefragt werden.

Das Repository persistiert die Status und ist somit auch nach einem Neustart verfügbar.

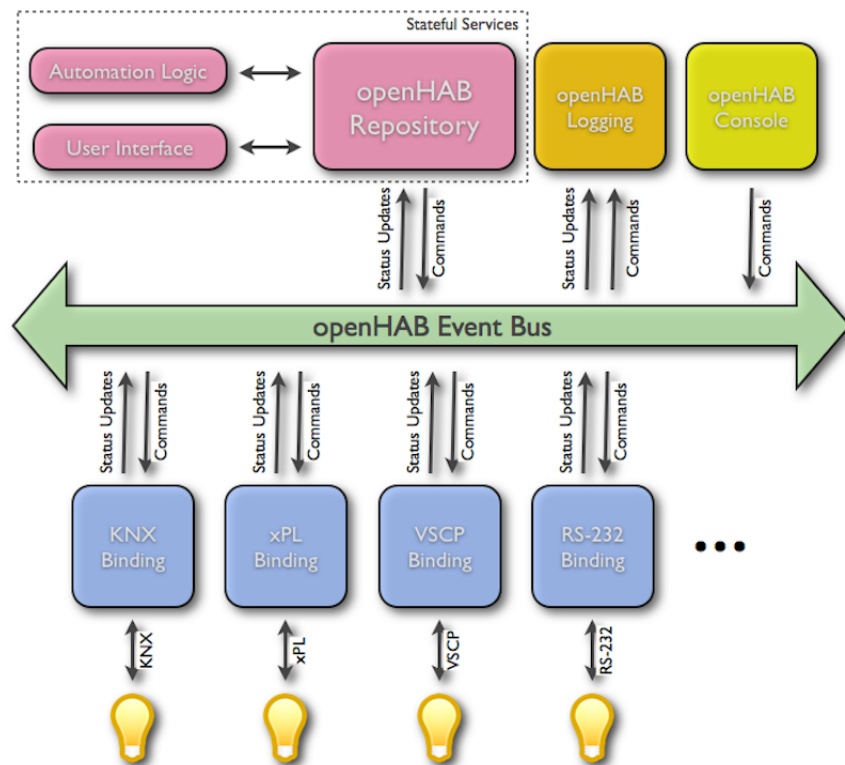


Abbildung 3.: Kommunikation openHAB

## **Bindings**

Ein Binding ist eine Verbindung zwischen dem Event Bus und den externen Systemen. Diese Verbindungen sind aufgrund der verschiedenen Technologien verschieden. Dadurch muss für jede Technologie ein eigenes Binding geschrieben werden. Für einige Systeme sind Bindings vorhanden, die einzeln heruntergeladen und als «Add-on» installiert werden können. Die Bindings stellen nur sicher, dass Events zwischen Event Bus und den jeweiligen Devices ausgetauscht werden können. Sie müssen sich also nicht um Statusänderungen oder ähnliches kümmern, da dies durch das Item Repository übernommen wird.

Alle momentan verfügbare Bindings sind unter folgendem Link zu finden: <https://github.com/openhab/openhab/wiki/Bindings>

## 4.3. Deploymentübersicht

### Binding Azure

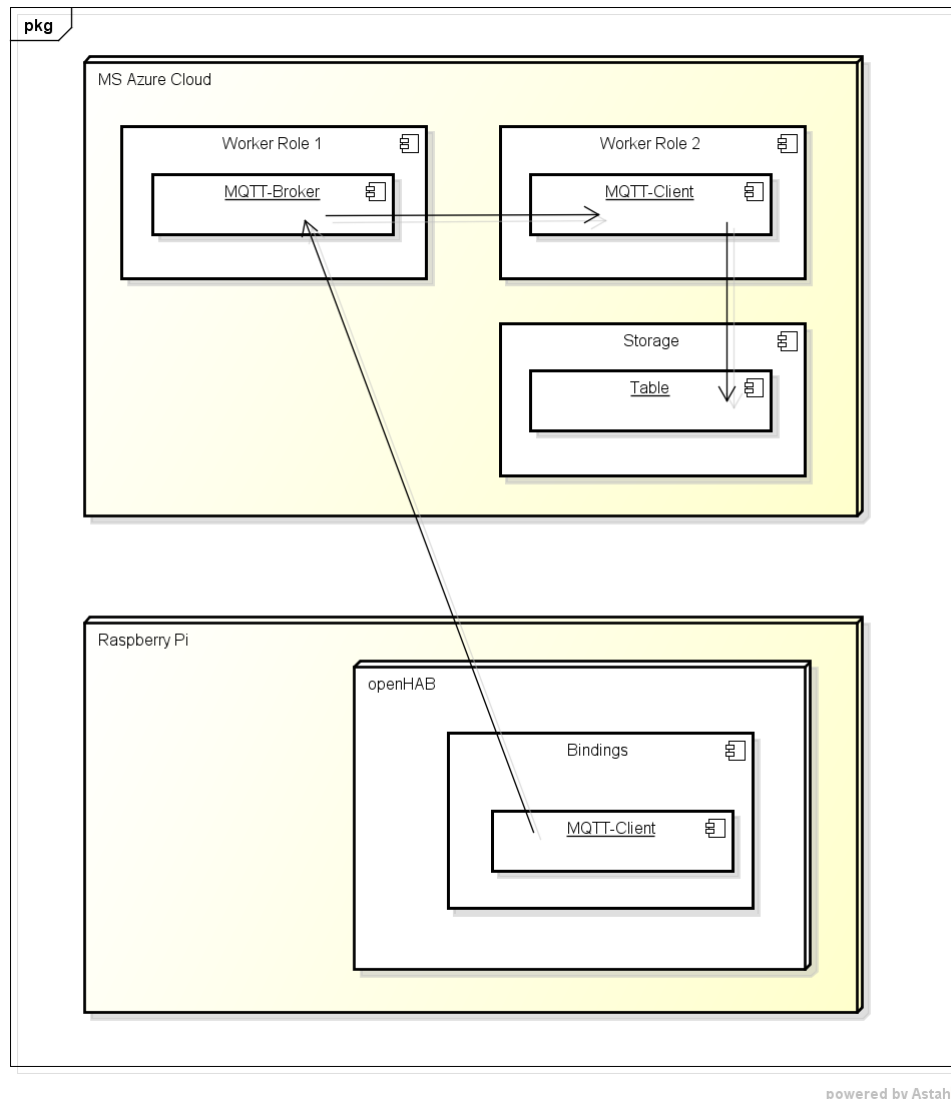


Abbildung 4.: Binding Azure Cloud

# Literaturverzeichnis

- [1] E. Clayberg, *Eclipse plug-ins*, 3rd ed., ser. The eclipse series. Upper Saddle River, NJ: Addison-Wesley, 2009.
- [2] ISO C++, “Standard for programming language c++,” Tech. Rep., Oct. 2013.
- [3] H. IFS, “CDTTesting git repository,” Oct. 2014. [Online]. Available: <http://cevelop.com/cdt-test-plugins/development/>



# Glossar

**API** Application Programming Interface

**IoT** Internet of Things

**MS** Microsoft

## A. Projektplan



<b>Thema</b>	Aufbau einer Smart-Home Beispielapplikation
<b>Studenten</b>	Dominik Freier, Marco Leutenegger
<b>Betreuer</b>	Prof. Hansjörg Huser

## Änderungsgeschichte

Datum	Version	Änderung	Autor
25.02.2015	0.0.1	Dokument erstellen	M. Leutenegger
27.02.2015	0.0.2	Meilensteine erfasst	D. Freier, M. Leutenegger
04.03.2015	0.1.0	Risikomanagement angepasst	D. Freier, M. Leutenegger

## Einführung

### Zweck

Dieses Dokument dient als Projektplan für die Bachelorarbeit von Dominik Freier und Marco Leutenegger und definiert alle organisatorischen Rahmenbedingungen.

### Gültigkeitsbereich

Die Gültigkeit des Projektplans beschränkt sich auf die Bachelorarbeit von Dominik Freier und Marco Leutenegger im Frühjahrssemester 2015.

### Referenzen

Bezeichnung	Referenz
Risikomanagement	Siehe separates Dokument
Security Infos	<a href="https://github.com/openhab/openhab/wiki/Security">https://github.com/openhab/openhab/wiki/Security</a>

# Projekt und Übersicht

## Zweck und Ziel

Diese Bachelorarbeit hat als Ziel, eine Smart-Home Beispielapplikation aufzubauen, welche wesentliche Aspekte einer Internet-of-Things-Anwendung demonstriert, wie Steuern von Devices, Lesen von Sensoren, Event-Verarbeitung, Überwachung und intelligente Abläufe steuern, Streaming von Sensordaten und Online-Analyse der Daten usw.

Das System soll auf einer tragfähigen und erweiterbaren Architektur aufgebaut werden und Microsoft Azure als Cloud Plattform benutzen.

## Lieferumfang

Die abzuliefernden Dokumente und Software-Artefakte des Projekts richten sich im Wesentlichen nach den Vorgaben aus den Dokumentationsanleitungen der HSR. Eine davon abweichender Lieferumfang wurde mit dem Betreuer besprochen und genehmigt.

## Referenzen

Nr.	Art	Bezeichnung	Form	Empfänger
1	Publikation	Poster	PDF	H.Huser
2	Publikation	Kurzfassung	PDF	H.Huser
3	Dokument	Bericht	PDF/Ausdruck	H.Huser
4	Dokument	Projektplan	PDF/Ausdruck	H.Huser
5	Dokument	Sitzungsprotokolle	PDF/Ausdruck	H.Huser
6	Dokument	Eigenständigkeitserklärung	PDF/Ausdruck	H.Huser
7	Dokument	Erfahrungsbericht D.Freier	PDF/Ausdruck	H.Huser
8	Dokument	Erfahrungsbericht M.Leutenegger	PDF/Ausdruck	H.Huser
9	Source	Code-Abgabe	ZIP	H.Huser
10	Archiv	2x Deliverables 1-9	DVD	H.Huser

## Projektorganisation

Die Dokumentation des Projekts gliedert sich in diesen Projektplan und einen Bericht. Im Projektplan werden alle organisatorischen Aspekte festgehalten, wie etwa die Planung der Meilensteine, Aufgaben der Teammitglieder oder Abmachungen zum Dokumentenmanagement. Im Bericht werden technische Beschreibungen der Ausgangslage, Diskussionen für Lösungsansätze, Requirements und Details zur Umsetzung dokumentiert.

Damit die Teammitglieder möglichst parallel und effizient arbeiten können, werden alle Dokumente mit LaTeX geschrieben und auf einem Git-Repository verwaltet. Daruch wird das Risiko von Versionskonflikten reduziert und der Zugriff insbesondere für den Betreuer vereinfacht.

Die Verwaltung der Aufgaben und agilen Vorgänge erfolgt durch Jira. Wir erhielten zu diesem Zweck eine Classroom Lizenz vom Hersteller Atlassian. Jira wurde auf einem virtuellen Server der HSR installiert.

## Organisationsstruktur

Verantwortung	Teammitglied
Verwaltung und Bereinigung de Dokumente	D. Freier, M. Leutenegger
Pflege virtueller Server, Jira inkl. Backups	D. Freier, M. Leutenegger
Sitzungsprotokolle verfassen	D. Freier, M. Leutenegger
Iterationsplanung	D. Freier, M. Leutenegger
Risikomanagement	D. Freier, M. Leutenegger
Architekturdesign	D. Freier, M. Leutenegger

## Externe Schnittstellen

Betreuer der Bachelorarbeit ist Prof. Hansjörg Huser. Experte ist Herr Stefan Zettel. Gegenleser ist <td>.

## Management Abläufe

### Zeitliche Planung

Das Projekt wird während des Frühjahrssemester 2015 durchgeführt. Der Start der Arbeit war am Montag, den 16. Februar 2015. Die Abgabe der Vollständigen Dokumentation an den Betreuer erfolgt am Freitag, den 12. Juni 2015. Als Zeitbudget sollen in den 17 Wochen insgesamt 720 Stunden, bzw. rund 21 Stunden pro Woche und Student eingeplant werden.

### Vorgehensmodell

Als Vorgehensmodell wurde der Rational Unified Process ausgewählt, da das Projektteam mit diesem Modell aus früheren Arbeiten (inkl. Semesterarbeit) vertraut ist und damit gute Erfahrungen gemacht hat. Die Phasen wurden nach dem Schema «eins, drei, drei, eins» in insgesamt acht Iterationen à zwei Wochen aufgeteilt.

### Meilensteine

MS	Iter.	Beschreibung	Datum
MS1	I1	Der Projektauftrag wurde zusammen mit dem Betreuer besprochen und ist akzeptiert. Den Teammitgliedern ist klar, welches die Ziele des Projekts sind und haben eine gemeinsame Vision. Die organisatorischen Aspekte wurden so weit wie möglich abgeklärt und die benötigte Infrastruktur steht allen Beteiligten zur Verfügung.	04.03.2015
MS2	E1	Die Analyse der funktionalen und nicht-funktionalen Anforderungen ist abgeschlossen und die Use Cases definiert. Die technische Umsetzung der Use Cases wurde analysiert und mit Umsetzung kann begonnen werden. Die Hardware wurde bestellt und für das Mobile-App wurden erste Mockups gezeichnet.	18.03.2015
MS3	E2	Ein Architekturprototyp (Installation und Konfiguration openHAB) existiert. Ein Prototyp für die Use Cases mit existierenden Bindings wurde entwickelt.	01.04.2015

MS4	E3	Prototyp mit eigenen Bindings wurde entwickelt, parallel dazu wird die Cloud mit den benötigten Komponenten aufgesetzt.	15.04.2015
MS5	C1	Die Use Cases mit den eigenen Bindings sind fertig implementiert.	29.04.2015
MS6	C2	Das Android-App ist gemäss den, in der Analyse (E1) gezeichneten, Mockups entwickelt und die geplanten Funktionen sind implementiert.	13.05.2015
MS7	C3	Der geschriebene Code wurde überarbeitet und optimiert. Die nötigen Komponenten sind gemäss FR und NFR getestet.	27.05.2015
MS8	T1	Die Dokumentation wurde nachgeführt, und finalisiert. Die Deliverables werden am darauf folgenden Freitag den entsprechenden Personen übergeben. Dieser Meilenstein definiert den Abschluss des Projektes.	10.06.2015

## Iterationsplanung

It.	Arbeitspakete	Ziele	SW
I1	1. Besprechung Projektauftrag 2. Einarbeitung Thematik 3. Aufsetzen LaTeX-Dokument	<input type="checkbox"/> MS1: Projektauftrag erhalten <input type="checkbox"/> Gemeinsame Vision des Projekts	1-2
E1	4. Definition der Use Cases 5. Aufbau/Setup/Anordnung 6. Hardware Evaluation 7. Abklären technische Machbarkeit 8. Android Mock-Up 9. Meilensteine und Iterationsplan	<input type="checkbox"/> MS2: Review Projektplan <input type="checkbox"/> Hardware bestellt <input type="checkbox"/> Mockups für App gezeichnet	3-4



E2	10. Installation openHAB auf Raspberry Pi 11. Einrichten WLAN und Router 12. Use Cases mit DSL umsetzen 13. Integration HomeMatic 14. Integration Philips Hue 15. Integration Webcam 16. Dokumentation nachführen	<input type="checkbox"/> MS3: Erster Prototyp existiert <input type="checkbox"/> Erste Version der Architektur-Dokumentation	5-6
E3	17. Aufsetzen und Anpassen der Azure Cloud 18. Programmierung Azure Binding 19. Integration Azure Cloud	<input type="checkbox"/> MS4: Prototyp mit Bindings fertig <input type="checkbox"/> Cloud aufgesetzt	7-8
C1	20. Alle Komponenten vollständig integrieren 21. Vernetzung der Hardware 22. Dokumentation nachführen	<input type="checkbox"/> MS5: Binding für Cloud erstellt	9-10
C2	23. Android App Model portieren (HABDroid) 24. Anbindung Android an Cloud 25. User Interface	<input type="checkbox"/> MS6: Android-App entwickelt	11-12
C3	26. Refactoring und Unit-Testing 27. Systemtests 28. Überprüfung NFR und FR	<input type="checkbox"/> MS7: Refactoring und Testing durchgeführt	13-14
T1	29. Dokumentation abschliessen 30. Poster erstellen 31. Dokumentation drucken & binden 32. CD erstellen 33. BA abgeben	<input type="checkbox"/> MS8: Abschluss des Projektes <input type="checkbox"/> Dokumentation abgeschlossen <input type="checkbox"/> Deliverables übergeben	15-16

## Besprechungen

Wöchentliche Besprechungen:

Bezeichnung	Ziel	Wochentag	Uhrzeit	Ort
Teambesprechung	Projektarbeiten im Plenum erledigen	Donnerstag	08:10-08:40	HSR (Labor)
Fortschrittsbesprechung	Fortschritte bzw. Probleme besprechen	Mittwoch	10:10-10:50	HSR (6.010)

## **Risikomanagement**

### **Risiken**

Nachstehend wird auf die projektbezogenen Risiken eingegangen. Eine Übersicht in Form einer Tabelle ist auf der nächsten Seite zu finden. Die Tabelle wird während des ganzen Projektes angepasst und aktualisiert, falls notwendig.

### **Umgang mit Risiken**

#### **Reserven/Rückstellungen**

Das grösste Risiko stellt R1 (ungeplante Machbarkeiten) dar. Aus diesem Grund werden in diesem Projekt Rückstellungen von 20 Stunden eingeplant.

### **Überprüfung von Risiken**

Weitere Risiken werden im Laufe des Entwicklungsprozesses erkennbar. Hierfür aktualisieren wir dieses Dokument, welches als zentrale Stelle dient, um Entscheidungen und Risiken zu Dokumentieren und auch eine zentrale Anlaufstelle bei Fragen darstellt. Des weiteren wird in der Beschreibung des betroffenen Vorgangs auf mögliche Risiken hingewiesen und dokumentiert.

Nr	Titel	Beschreibung	Scha- den[h]	Eintritts- wahrsch.	Gew. Schad.	Vorbeugung.	Verhalten beim Eintreten.
<b>R1</b>	Ungeplante Machbar- keit	Nicht alle Arbeits- pakete in Iteration oder Meilensteine ab- gedeckt.	20	40%	8	Laufende Kontrolle des Zeitplans	Überstunden in Kauf nehmen, um folgende Iteration nicht in Ge- fahr zu bringen.
<b>R2</b>	Absturz Jira-Server und Daten- verlust	Der virtuelle Server der HSR stürzt ab, und die Daten des Ji- ra gehen verloren.	2	10%	0.2	Backup pro Woche er- stellen.	Letztes Backup ein- spielen und die Diffe- renz von Hand erneut eintragen.
<b>R3</b>	Verlust von Code	Das persönliche Note- book stürzt ab und die Daten sind verlo- ren.	2	10%	0.2	Code wird ständig auf GitHub gepusht.	Lab-PC oder sonstige Computer verwenden und GIT Repository Klonen.
<b>R4</b>	Fabrikations- fehler Sensoren	Die Sensoren kommen mit einem Fabrikati- onsfehler an.	20	10%	2		Sensor zurücksenden und mit anderem wei- terarbeiten.
<b>R5</b>	Schnittstellen Sensoren	Schnittstellen zu ande- ren Systemen bereitet Probleme	16	5%	0.8	Dokumentation gut prüfen.	Community durchfors- ten, Workaround su- chen.

## Arbeitspakete

Die Arbeitspakete wurden im Projektmanagementtool Jira als Vorgänge definiert. Einige Vorgänge beinhalten weitere Untertätigkeiten, die wir ebenfalls als einzelne Arbeitspakete betrachten.

Eine Übersicht mit allen Arbeitspaketen und dem zeitlichen Ablauf nach Iterationen befindet sich unter: <http://sinv-56046.edu.hsr.ch:8080> > Agile > Zeige alle Boards > baIOTBoard > Plan

## Infrastruktur

### Software

Wie in jedem Projekt kommt verschiedene Software zum Einsatz.

Software	Version (Major)	Beschreibung/Einsatzbereich
GitHub	v3	Source Code Verwaltung inkl. Branchmanagement, Web Interface für Git-Verwaltung.
Atlassian: Jira	6.4	Projektmanagement
Windows Server	2012 R2 (64Bit)	Virtueller Server für Jira
<td>	<td>	<td>

## Qualitätsmassnahmen

Massnahme	Zeitraum	Ziel
Einsetzen eines Projekt-Management-Tools	ganzes Projekt	Alle auf dem aktuellsten Stand halten
Versionierungssystem (git)	Sicherung des Codes/Doku, ganzes Projekt	keine Blockaden
Koordinationsmeetings	ganzes Projekt	Ressourcen optimal zuteilen: Wer benötigt wo Hilfe, wer ist schon fertig?
Vier-Augen-Prinzip	ganzes Projekt	Dokumentation/Programmcodewird jeweils von beiden Partnern kontrolliert. Bei einem Ausfall einer Person, ist das andere Mitglied informiert.

## Dokumentation

### Ablage

Alle Dokumente können auf dem GitHub Repository gefunden werden. Die Vorgänge werden mit Jira auf einem virtuellen Server der HSR verwaltet.

- Dokumentation: [https://github.com/greekins/baIoT\\_TeX](https://github.com/greekins/baIoT_TeX)
- Vorgänge: <http://sinv-56046.edu.hsr.ch:8080>

Der Source-Code wird mit Git verwaltet: <tbd>

### Qualität

- Commits verlangen eine Beschreibung
- Benutzerfreundliche Commit-Übersicht dank Github

- Für die Qualität des Codes wird in jeder Iteration (ab Elaboration E2) Codereviews durchgeführt (siehe Managementabläufe)

## **Projektmanagement**

Es wird die von Atlassian zur Verfügung gestellte Umgebung eingesetzt:

`http://sinv-56046.edu.hsr.ch:8080`

**Gast Login:** hhuser

## **Entwicklung**

### **Code Reviews**

Die Commits sind für alle Projektmitglieder ersichtlich und werden in einem Activity Stream auf dem Repository unter «Graph» angezeigt. Diese werden sporadisch von den anderen Mitgliedern geöffnet und kurz überprüft.

Bei einem wöchentlichen Meeting werden getätigte Implementierungen im Plenum angeschaut und besprochen. Auch lautet unsere Regel, dass bei Unsicherheiten bei laufender Entwicklung Rat vom anderen Teammitglied eingeholt wird.

### **Code Style Guidelines**

Es wird sich an die gängigen Style Guidelines gehalten, die im Laufe des Studiums eingeführt wurden.

## B. Sitzungsprotokolle

### Sitzung 1 - Kick-Off

Datum: 17. Februar 2015

Teilnehmer: Prof. Hansjörg Huser, Dominik Freier, Marco Leutenegger

#### Projektdefinition

Aufbauen einer Demoanwendung für «Smart Home». Wie genau die aussehen wird, steht noch nicht fest. Ist Bestandteil der Analyse und Evaluation. Als Resultat der Arbeit soll ein Showcase entstehen mit ein paar Anwendungsfällen.

Mögliche Bestandteile:

- Sensoren, Raspberry-Pi
- Cloud (simple gehalten, Service Bus)
- UI (Mobile/Tablet)

#### Anstehende Arbeiten

- Evaluation HW-Plattform
- Evaluation Framework
- erste Version des Projektplans

#### Organisatorisches

- Virtueller Server beantragt
- Wöchentliche Besprechungen: Mittwoch, 10.10 Uhr



## **Sitzung 2**

Datum: 25. Februar 2015

Teilnehmer: Prof. Hansjörg Huser, Dominik Freier, Marco Leutenegger

### **Organisatorisches:**

- System-Architektur von Herrn Huser zur Kenntnis genommen und akzeptiert.
- Anwendungsszenarien sollen richtung Einbrecherschutz gehen (Türkontakte, Bewegungssensoren etc.

### **Anstehende Arbeiten:**

- Bestellliste mit Sensoren und Aktoren erstellen.
- Anwendungsszenarien Anpassen.
- Erste Version des Projektplans erstellen.

## **Sitzung 3**

Datum: 04. März 2015

Teilnehmer: Prof. Hansjörg Huser, Dominik Freier, Marco Leutenegger

### **Organisatorisches:**

- Wunderbar wird vernachlässigt, die Antwort abgewartet. Als Ersatz wird Tinkerforge gewählt.

### **Anstehende Arbeiten:**

- Bestellliste anpassen.
- Einige Änderungen am Projektplan.
- Risikoliste anpassen.
- Detailplanung erstellen.

## Sitzung 4

Datum: 11. März 2015

Teilnehmer: Prof. Hansjörg Huser, Dominik Freier, Marco Leutenegger

### Organisatorisches:

- Iterationsplanung wurde besprochen und gutgeheissen.
- Use Cases wurden besprochen und gutgeheissen.
- Projektplan wurde als Ganzes angenommen.
- Tinkerforge wird nach Absprache nicht weiter verfolgt. Falls genügend Zeit vorhanden ist, werden, aus der bestehenden Hardware, die Use Cases erweitert.
- **Meilenstein 1 erreicht und abgenommen!** - Phase E1 abgeschlossen.

### Anstehende Arbeiten:

- Beginn der Phase E2.
- Nach Erhalt der Hardware mit Prototyp beginnen.

## Sitzung 5

Datum: 17. März 2015

Teilnehmer: Prof. Hansjörg Huser, Dominik Freier, Marco Leutenegger

### Organisatorisches:

- **Beginn der Phase E2**
- Fassung für Glühbirne kaufen die Studenten in einem Elektro-Fachgeschäft. Die Kosten werden vom Institut übernommen.
- Router wird von Herrn Huser organisiert.

### Anstehende Arbeiten:

- Installation openHAB auf Raspberry Pi.
- Einrichten des Netzwerkes (Router).
- Use Cases umsetzen mit HomeMatic, Philips Hue und Webcam.
- Erste Version der Architekturdokumentation erstellen.

## **Sitzung 6**

Datum: 25. März 2015

Teilnehmer: Prof. Hansjörg Huser, Dominik Freier, Marco Leutenegger

### **Organisatorisches:**

- Prototyp wurde vorgeführt und für gut befunden.
- Man befindet sich im Zeitplan und beginnt mit den nächsten Arbeitspaketen.

### **Anstehende Arbeiten:**

- Termin für Zwischenpräsentation mit Prof. Dr. Rinkel in zwei Wochen (8. April 2015)
- Integration Webcam abschliessen.
- Mit Aufsetzen von Azure Cloud und Binding beginnen.

## **Sitzung 7**

Datum: 01. April 2015

Teilnehmer: Prof. Hansjörg Huser, Dominik Freier, Marco Leutenegger

### **Organisatorisches:**

### **Anstehende Arbeiten:**

## C. Persönliche Reflektion

### Marco Leutenegger

<td>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.

### Dominik Freier

<td>Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet. Lorem ipsum dolor sit amet, consetetur sadipscing elitr, sed diam nonumy eirmod tempor invidunt ut labore et dolore magna aliquyam erat, sed diam voluptua. At vero eos et accusam et justo duo dolores et ea rebum. Stet clita kasd gubergren, no sea takimata sanctus est Lorem ipsum dolor sit amet.