

DOMAIN SPECIFIC SOFTWARE ENGINEERING: INTERNET OF THINGS

Seminar Domain Specific Software Engineering

Version 0.1

Zürcher Hochschule für Angewandte Wissenschaften

Daniel Brun

xx. Juni 2015

Eigenständigkeitserklärung

Hiermit bestätige ich, dass vorliegende Seminararbeit zum Thema „Evaluation einer Mini ERP Lösung für einen Verein“ gemäss freigegebener Aufgabenstellung ohne jede fremde Hilfe und unter Benutzung der angegebenen Quellen im Rahmen der gültigen Reglemente selbständig verfasst wurde.

Thalwil, 11. Februar 2015

Daniel Brun

Inhaltsverzeichnis

1	Einleitung	1
1.1	Hintergrund	1
1.2	Aufgabenstellung	1
1.3	Abgrenzung	1
1.4	Motivation	1
1.5	Struktur	2
2	Ausgangslage	3
2.1	Internet of Things	3
2.2	Domain Specific Software Engineering	3
2.3	Domain Specific Languages	3
3	Die Domäne „Internet of Things“	5
3.1	Einführung	5
3.2	Anforderungen an Software	5
3.3	Anforderungen an die Softwarearchitektur	5
3.4	Anforderungen an die Programmiersprache	5
3.5	Anforderungen / Unterschiede zur „Standard-Domäne“	5
3.5.1	Software Requirements	5
3.5.2	Software Design	5
3.5.3	Software Construction	6
3.5.4	Software Testing	6
3.5.5	Software Maintenance	6
3.5.6	Software Configuration Management	6
3.5.7	Software Engineering Management	6
3.5.8	Software Engineering Process	6
3.5.9	Software Engineering Tools and Methods	6
3.5.10	Software Quality	6
4	Software Engineering in der Domäne „Internet of Things“	7
4.1	Programmiersprachen	7
4.1.1	Assembler	7
4.1.2	C / C++	7

4.1.3	Java / .NET	7
4.2	Protokolle und Standards	7
4.2.1	Thread	7
4.2.2	AllJoyn	7
4.3	Frameworks	8
4.3.1	Kommunikation zwischen Internet of Things (IOT)-Geräten	8
5	Case-Study „iotivity“	9
	Anhang	15
	Liste der noch zu erledigenden Punkte	15

KAPITEL 1

Einleitung

1.1 Hintergrund

Im Rahmen meines Bachelor-Studiums in Informatik an der Zürcher Hochschule für Angewandte Wissenschaften (ZHAW) muss im 6. Semester eine Seminararbeit zu einem vorgegebenen Themenbereich erarbeitet werden. Ich habe mich für den Themenbereich „Domain Specific Software Engineering“ entschieden.

Aus einem Themenkatalog konnte ein spezifisches Thema im Bereich „Domain Specific Software Engineering“ ausgewählt werden. Ich habe mich für das Thema „Internet of Things“ entschieden.

Für die Arbeit sollen circa 50 Arbeitsstunden aufgewendet werden. Dies entspricht etwa einem Umfang von 15 bis 20 Seiten. Zusätzlich gelten die Rahmenbedingungen gemäss dem Reglement zur Verfassung einer Seminararbeit ([**ZHAW:2012:Seminararbeit:Reglemente**])

1.2 Aufgabenstellung

Es soll ein Dokument zum Thema Domain Specific Software Engineering im Bereich Internet of Things erstellt werden. Das Papier soll die Schwierigkeiten der Software-Entwicklung in diesem Bereich aufzeigen und einen groben Überblick über das Thema eben.

1.3 Abgrenzung

Abgrenzung

1.4 Motivation

motivation

1.5 Struktur

Diese Arbeit gliedert sich in folgende Hauptteile:

-

Struktur

Im ersten Kapitel werden die Details zur Ausgangslage und die Hintergründe der Arbeit aufgezeigt. Im zweiten Kapitel wird mit Hilfe einer Umfrage innerhalb des Turnvereins eine Analyse erstellt. Aus dieser Analyse gehen die Randbedingungen, Ziele und Anforderungen an das Mini System hervor. Diese Randbedingungen, Ziele und Anforderungen werden im Kapitel 'Evaluation' als Kriterien für die Vorselektion, Selektion und anschliessenden die Evaluation der Produkte verwendet. Im letzten Kapitel wird ein Fazit gezogen, eine Empfehlung an den abgegeben und über die gesamte Arbeit reflektiert.

KAPITEL 2

Ausgangslage

Mit dem Fortschritt in der Computertechnik und der fortlaufenden Miniaturisierung und Mobilisierung kommt das IOT

Allgemeins Gadgets, IOT

2.1 Internet of Things

Das Internet of Things (IOT), beziehungsweise das „Internet der Dinge“

2.2 Domain Specific Software Engineering

2.3 Domain Specific Languages

Eine „Domain Specific Language (DSL)“ bezeichnet eine Programmier-, Modellierungs- oder Metasprache, welche für eine spezifische Domäne entworfen und entwickelt wurde. Eine DSL adressiert dabei spezifische Schwächen und Problemstellungen der angesprochenen Domäne.

KAPITEL 3

Die Domäne „Internet of Things“

3.1 Einführung

Beschreibung der Domäne

—

3.2 Anforderungen an Software

Stark von Einsatzgebiet abhängig (Kühlschrank vs. Low Energy ohne Stromversorgung)

3.3 Anforderungen an die Softwarearchitektur

3.4 Anforderungen an die Programmiersprache

—

3.5 Anforderungen / Unterschiede zur „Standard-Domäne“

In diesem Kapitel werden für die einzelnen Abschnitte des Software Engineering Prozesses die Anforderungen, beziehungsweise die Unterschiede, zum Software Engineering Prozess in der „Standard-Domäne“ aufgezeigt. Als „Standard-Domäne“

3.5.1 Software Requirements

eindeutig, Komplexität schwierig, nachhaltig (schwieriger Update)

3.5.2 Software Design

Schlank, Energieeffizient

3.5.3 Software Construction

Je nach Anwendungsbereich: Hardwarenahe, Optimierung Für die Entwicklung von Anwendungen für IOT-Geräte können grundsätzlich

3.5.4 Software Testing

Wenn bei Consumer: Update evtl. schwierig, Fehlfunktion schwerwiegend,

3.5.5 Software Maintenance

Schwierig,...

3.5.6 Software Configuration Management

3.5.7 Software Engineering Management

3.5.8 Software Engineering Process

3.5.9 Software Engineering Tools and Methods

3.5.10 Software Quality

Höhere Qualität notwendig

KAPITEL 4

Software Engineering in der Domäne „Internet of Things“

4.1 Programmiersprachen

4.1.1 Assembler

Eine Implementation in Assembler kann unter bestimmten Voraussetzungen die beste Lösung für ein bestimmtes Problem sein. In der Regel wird der Assembler-Ansatz gewählt, wenn das Programm so effizient und sparsam wie möglich ablaufen und mit so wenig Ressourcen als möglich auskommen soll.

4.1.2 C / C++

4.1.3 Java / .NET

Für die Implementation kann es durchaus sinnvoll sein eine Hochsprache, wie Java oder .NET C# einzusetzen.

4.2 Protokolle und Standards

4.2.1 Thread

Protokolle: Thread, Netzwerk-Protokoll, Fokus: Security, Low Energy, notwendiger Chip, schon in vielen Geräten vorhanden, gestützt auf 6LoWPAN, IPv6 over Low power Wireless Personal Area Network,

4.2.2 AllJoyn

Qualcomm entwickelt, anschliessend: Linux Foundation, AllSeen Alliance (Cisco, Microsoft, LG, HTC, ...) Verbindung, Wartung Geräte in WLAN-Netzwerk, Kontrolle, Benachrichtigungs Service,

MQTT CoAP XBEE XMPP DDS HTTP AMQP (IOT-Client-Seitig?)

4.3 Frameworks

In diesem Kapitel werden einige Frameworks vorgestellt, welche

4.3.1 Kommunikation zwischen IOT-Geräten

AllSeen IOTIVITY Z-Wave

KAPITEL 5

Case-Study „iotivity“

Abbildungsverzeichnis

Tabellenverzeichnis

Liste der noch zu erledigenden Punkte

Abgrenzung	1
motivation	1
Struktur	2