

Projektplan

Inhaltsverzeichnis

1	EINFÜHRUNG	3
1.1	ZWECK DES DOKUMENTS	3
1.2	REFERENZEN	3
2	PROJEKTÜBERSICHT	4
3	PROJEKTORGANISATION	5
3.1	ORGANISATIONSSTRUKTUR	5
3.1.1	STUDIERENDE	5
3.1.2	BETREUER HSR	5
3.1.3	ANSPRECHPARTNER AUFTRAGGEBER	5
4	MANAGEMENT ABLÄUFE	6
4.1	ZEITLICHE PLANUNG	6
4.2	VORGEHEN	6
4.3	MEILENSTEINE	6
4.4	BESPRECHUNGEN	8
4.4.1	PROTOKOLLFÜHRUNG	8
5	RISIKOMANAGEMENT	9
5.1	RISIKEN	9
5.2	UMGANG MIT RISIKEN	9
6	ARBEITSPAKETE	10
7	INFRASTRUKTUR	11
8	QUALITÄTSMASSNAHMEN	12
8.1	ENTWICKLUNGS-WORKFLOW	12
8.2	CODE STYLE	12
8.3	TESTEN	12

1 Einführung

1.1 Zweck des Dokuments

Ziel des Dokuments ist es, dem Leser einen Überblick über die Studienarbeit zu geben und so einen schnellen Einstieg in den Projektablauf zu gewähren. Dabei wird der Projektablauf definiert, die möglichen Risiken analysiert und ein Überblick über die Arbeitsweise und Infrastruktur gegeben.

1.2 Referenzen

Dieses Dokument dient als Ergänzung zum technischen Bericht der Studienarbeit. Alle projektspezifischen Informationen, die im Bericht keinen Platz finden, sind hier aufgeführt.

2 Projektübersicht

Motivation, Zweck und Ziel, Lieferumfang, Annahmen und Einschränkungen werden im technischen Bericht bereits detailliert ausgeführt.

3 Projektorganisation

Diese Arbeit wird als Bachelorarbeit an der HSR Hochschule für Technik Rapperswil im Frühjahrssemester 2018 durchgeführt.

3.1 Organisationsstruktur

Diese Bachelorarbeit findet im Auftrag der Firma *HSi Elektronik AG* statt.

3.1.1 Studierende

- Esteban Luchsinger, esteban.luchsinger@hsr.ch
- Patrick Scherler, patrick.scherler@hsr.ch

3.1.2 Betreuer HSR

- Prof. Dr. Farhad Mehta, *Institut für Software*, farhad.mehta@hsr.ch

3.1.3 Ansprechpartner Auftraggeber

- Tobias Hofer, *HSi Elektronik AG*, tobias.hofer@hsi-ag.ch

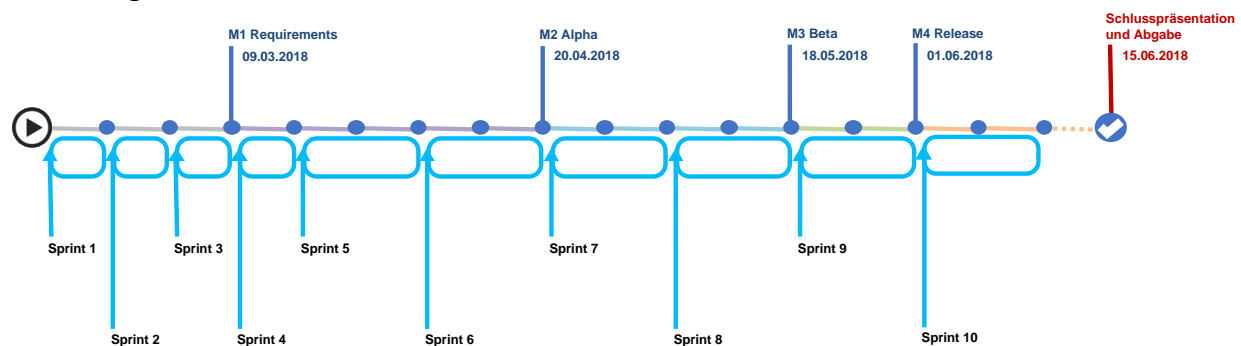
4 Management Abläufe

4.1 Zeitliche Planung

Die zeitliche Planung ist grösstenteils durch die Organisation als Bachelorarbeit gegeben. Folgende Termine sind definiert:

Termin	Beschreibung	Betrifft
19.02.18	Beginn der Bachelorarbeit	Studierende
23.02.18	Kickoff Meeting Auftraggeber	Stud. & Auft.
26.02.18	Kickoff Meeting Betreuer	Stud. & Betreuer
08.06.18	Abgabe von Abstract und Poster an Betreuer	Studierende
13.06.18	Der Betreuer gibt das Dokument mit dem korrekten und vollständigen Abstract zur Weiterverarbeitung an das Studiengangsekretariat frei.	Betreuer
15.06.18	Präsentation und Ausstellung der Bachelorarbeiten, 16 bis 20 Uhr	Alle
15.06.18	Abgabe des Berichts an den Betreuer bis 12:00 Uhr	Studierende
Ab 15.06.18	Mündliche BA-Prüfung	Studierende

4.2 Vorgehen



4.3 Meilensteine

M1 Requirements

Die Anforderungen sind definiert und mit allen Beteiligten abgesprochen.

Work Products: Use-Case Diagramme, Mockups, Prototypen

M2 Alpha

Die Architektur des Systems steht fest und ist implementiert. Es gibt einen «Durchstich» durch alle Komponenten des Systems.

TBD

Work Products: Executable Architecture

M3 Beta

Dieser Meilenstein markiert den Übergang in die Stabilisationsphase des Projekts. Alle Features sind zu diesem Zeitpunkt implementiert. Ab hier liegt der Fokus auf Bugfixing und Analyse (Performance, UX, etc.).

TBD

Feature Freeze: Ab diesem Meilenstein werden keine grösseren Features mehr eingeführt.

Work Products: Executables für Server und Clients

M4 Release

Die Applikation kann an den Auftraggeber ausgeliefert werden.

Code Freeze: Ab diesem Meilenstein gibt es einen Code-Freeze. Es sollten keine offensichtlichen Fehler mehr vorhanden sein. TBD

Work Products: Executables für Server und Clients

MP Abgabe und Präsentation

Der letzte Meilenstein ist die Präsentation der Arbeit.

TBD

Work Products: Bericht, Poster, Präsentation und Demo

4.4 Besprechungen

<TBD>

4.4.1 Protokollführung

Alle Besprechungen werden mit einer Traktandenliste vorbereitet und den Teilnehmern wenn möglich 24 Stunden vorher zugestellt.

Zu allen Besprechungen werden Protokolle geführt und auf OneDrive abgelegt.

5 Risikomanagement

Die Risikobeurteilung führt alle erfassten projektspezifischen Risiken als Produkt aus der geschätzten Eintrittswahrscheinlichkeit und dem erwarteten Schadensausmass in Stunden auf.

5.1 Risiken

#	Titel	Beschreibung	Massnahme	Ausmass [h]	Wahrscheinlichkeit [%]	Risiko [h]
1	Geringe Leistung des Raspberry Pi	Die Hardware des Raspberry Pi Zero (RAM, CPU, Schnittstellen) reicht für das Projekt nicht aus.	- Mit Architektur-Prototyp testen - Erneute Evaluation der Hardware	16	40	6.4
2	Lux-Sensor Schnittstelle nicht vorhanden oder dokumentiert	Die Lux-Sensor Werte lassen sich nicht über eine dokumentierte Schnittstelle auslesen.	- Funktionstests zu Projektbeginn - Risiko mit Kunde besprechen	40	60	24
3	Pozyx entspricht nicht den Anforderungen	Stabilität, Genauigkeit oder Performance von Pozyx reichen nicht aus.	- Funktionstests zu Projektbeginn - Risiko mit Kunde besprechen	40	40	16
4	Ausfall / Verlust der Projekthardware	Die für die Umsetzung benötigte Hardware (Raspberry Pi, Pozyx, Lux-Sensor) steht nicht mehr zur Verfügung.	- Hardware-Lieferzeiten beachten - Hardware einschliessen - ESD-Massnahmen beachten	16	30	4.8
5	Probleme mit Projektinfrastruktur	Datenverlust oder fehlende Verfügbarkeit der Management Tools	- Self-hosting - automatisches Backup	16	20	3.2
6	Verzögerungen bei der Entwicklung der Benutzerschnittstelle	Verzögerungen durch fehlende Praxiserfahrung	- Zeit für eine detaillierte Einarbeitung in die Technologie einplanen	16	30	4.8

5.2 Umgang mit Risiken

Die grössten Risiken sollen bereits im ersten Drittel des Projekts minimiert werden. Dazu werden Funktionstests durchgeführt und Prototypen erstellt, um bei Problemen möglichst schnell reagieren zu können.

Mit dem Auftraggeber wurde ausserdem besprochen, dass der Fokus dieser Arbeit in der Software-Entwicklung und Visualisierung der Sensordaten liegt. Die Ansteuerung der Sensor-Hardware oder andere nicht Software-spezifische Aufgaben können notfalls dem Auftraggeber übergeben werden.

6 Arbeitspakete

Die Arbeitspakete, Meilensteine und Sprints werden auf Jira verwaltet.

<http://jira.flux-coordinator.com>

7 Infrastruktur

<TBD>

8 Qualitätsmassnahmen

8.1 Entwicklungs-Workflow

Es wird nach dem *Feature Branch Workflow* entwickelt:

- Separate Branches für die Entwicklung von Features
- Feature-Branches nach Abschluss zurück in den Master mergen

8.2 Code Style

<TBD>

8.3 Testen

<TBD>