



HTW Chur
Hochschule für Technik und Wirtschaft

Fachhochschule Ostschweiz
University of Applied Sciences

Projektplan

Projekt: *Spionat*

Version 1.1

Isabell Sutter

Änderungsgeschichte

Datum	Version	Autorin	Beschreibung
2008-03-21	1.0	Isabell Sutter	Dokument erstellt
2008-05-28	1.1	Isabell Sutter	Dokument überarbeitet nach Feedback

Inhaltsverzeichnis

1	EINLEITUNG	6
1.1	Zweck	6
1.2	Projektüberblick	6
1.3	Lieferumfang	6
1.4	Abnahme: Vorgehensweise	6
1.5	Definitionen, Akronyme und Abkürzungen	7
1.6	Referenzen	7
2	PROJEKTORGANISATION	8
2.1	Vorgehensmodell	8
2.2	Organisationsstruktur	8
2.3	Organisatorische Abgrenzung und Schnittstellen	9
2.4	Projektzuständigkeiten	9
3	MANAGEMENT-PROZESS	10
3.1	Managementziele und Prioritäten	10
3.2	Annahmen und Einschränkungen	10
3.3	Risikomanagement	10
3.4	Berichtswesen	10
3.5	Personalplanung	10
3.6	Dokumentationsplan	11
4	RICHTLINIEN FÜR DIE ENTWICKLUNG	12
4.1	Konfigurationsmanagement	12
4.2	Entwicklungsrichtlinien	12
4.3	Prüfungen (Review und Tests)	12
4.4	Einsatz von Werkzeugen	12
5	ENTWICKLUNGSPLAN	13
5.1	Projektstrukturplan (Arbeitsgliederung)	13

5.2	Abhängigkeiten.....	13
5.3	Ressourcen Anforderungen.....	13
5.4	Kosten- und Ressourcenverteilung	13
5.5	Terminplan	13
A	REFERENZEN	14

Abbildungsverzeichnis

Abbildung 1: Das Wasserfallmodell als Vorgehensform	8
Abbildung 2: Organisationsstruktur	8

Tabellenverzeichnis

Tabelle 1: Deliverables	6
Tabelle 2: Projektzuständigkeiten	9
Tabelle 3: Personalplanung	10
Tabelle 4: Werkzeugliste	12
Tabelle 5: Ressourcen.....	13

1 Einleitung

1.1 Zweck

Dieser Projektplan ist das zentrale Dokument für das Projekt-Management im Projekt *Spionat*.

1.2 Projektüberblick

Die Studienarbeit im 4. Semester des Studiengangs Telekommunikation/Elektrotechnik an der HTW Chur wird parallel zum Modul Software Engineering durchgeführt. In Teamarbeit soll ein reales Softwareprojekt nach den gängigen Regeln der Praxis durchgeführt werden.

Ziel unserer Arbeit ist es, den Prototypen eines unbemannten Fahrzeuges zu bauen, welches kabellos mit dem Laptop kommunizieren kann und mit einer Webcam ausgestattet ist. Mithilfe dieser Webcam soll der Spionat, so der Name unseres Prototypen, die Umgebung auskundschaften können.

Im Rahmen der Projektarbeit werden wir den Prototypen herstellen und mit der notwendigen Hardware bestücken. Um die Kommunikation zwischen Hardware und Laptop zu gewährleisten, werden zudem geeignete Codes entwickelt.

1.3 Lieferumfang

Tabelle 1: Deliverables

Lieferung	Abgabetermin	Abgabeform	An Kunden auszuliefern
Projektplan	2008-03-27	Papier und PDF-File auf Claroline	Ja
Anforderungsdokumentation	2008-05-01	Papier und PDF-File auf Claroline	Ja
Entwurfsdokumentation	2008-05-16	Papier und PDF-File auf Claroline	Ja
Systemtestvorschrift	2008-05-29	Papier und PDF-File auf Claroline	Ja
Lauffähiges Produkt	2008-06-06	Codes elektronisch auf Claroline, Hardware-Abgabe im Sekretariat	Ja
Benutzeranleitung	2008-06-06	Papier und PDF-File auf Claroline	Ja
Alle Dokumente zusammengestellt und vollständig überarbeitet.	2008-06-06	ZIP-File auf Claroline	Ja

1.4 Abnahme: Vorgehensweise

Die einzelnen Deliverables werden am definierten Abgabetermin Herrn Bonderer abgegeben, welcher als fiktiver Kunde gilt. Mithilfe seines Feedbacks können bis zum endgültigen Abgabetermin noch vorhandene Mängel ausgebessert werden. Am 6. Juni 2008 werden alle schriftlichen Dokumente auf das Claroline gestellt und somit Herrn Bonderer übergeben. Die Hardware wird im Sekretariat abgegeben. Das Projekt endet mit der Schlusspräsentation in der Blockwoche.

1.5 Definitionen, Akronyme und Abkürzungen

Spionat	Name des Prototypen
Deliverables	Dokumente und Prototyp
SW	Software
HW	Hardware

1.6 Referenzen

Siehe Anhang A auf Seite 14 dieses Dokuments.

2 Projektorganisation

2.1 Vorgehensmodell

Als Grundlage für das weitere Vorgehen haben wir das Wasserfallmodell¹ ausgewählt.

Normalerweise werden die Systemanforderungen direkt vom Kunden in Form eines Lastenheftes aufgegeben. In unserem Projekt werden diese Anforderungen selber definiert. Die nachfolgende Analyse ist dennoch wichtig, um sicher zu gehen, dass die Anforderungen realistisch sind, und um abzuklären, wie sie umgesetzt werden können. Eventuell müssen die Systemanforderungen angepasst werden. Nach der Analyse kann mit der Codierung und dem Bau der Hardware begonnen werden. Die Testphase dient dazu, weitere Mängel aufzuspüren und sie zu beheben, bevor die endgültige Inbetriebnahme erfolgt.

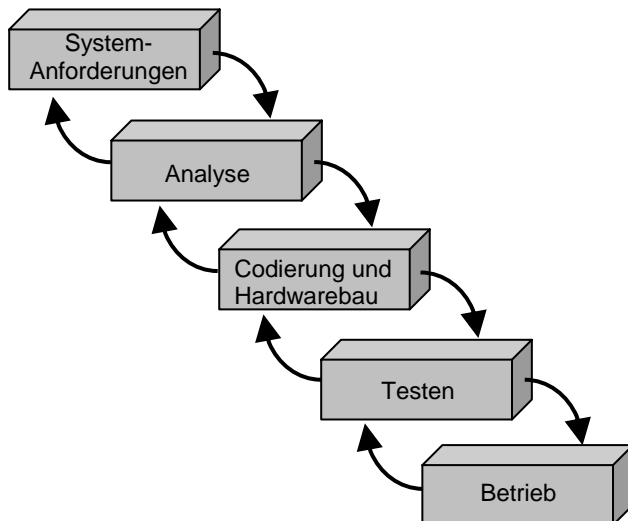


Abbildung 1: Das Wasserfallmodell als Vorgehensform

In Tabelle 1 auf Seite 6 sind die Eckdaten mit den entsprechenden Deadlines aufgelistet.

2.2 Organisationsstruktur

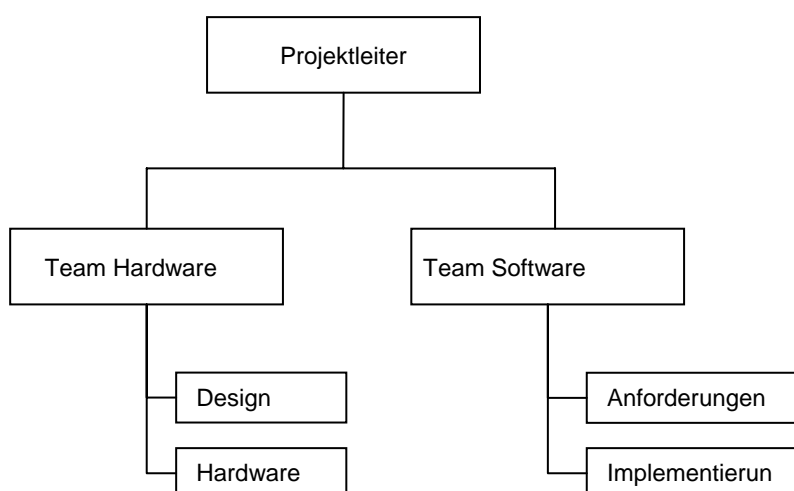


Abbildung 2: Organisationsstruktur

¹ Vgl. Anhang A [2]

2.3 Organisatorische Abgrenzung und Schnittstellen

Interne Schnittstelle: Projektteam – Projektmentor (Herr Bonderer)

Intern ist Herr Bonderer unser Ansprechpartner bei Problemen. Er überwacht das gesamte Projekt und wird es am Ende bewerten. Es steht allen Teammitgliedern offen, mit der Projektbegleitung zu kommunizieren. Um Redundanzen zu vermeiden, sollte allerdings zuerst versucht werden, das Problem innerhalb der Gruppe zu lösen.

Externe Schnittstelle: Projektteam – Auftraggeber, Kunde (fiktiv)

Nach aussen gibt es keine wirkliche Schnittstelle. Der Kunde existiert (noch) nicht, trotzdem ist es wichtig, diese Schnittstelle explizit zu erwähnen, da sie verdeutlicht, wie wichtig es ist, auch eine externe Sichtweise einfließen zu lassen, um den Kundenbedürfnissen gerecht zu werden. Zuständig für die externe Schnittstelle ist der Projektleiter.

2.4 Projektzuständigkeiten

Tabelle 2: Projektzuständigkeiten

Rolle	Zuständigkeiten
Projektleiter	Projektplan, Zeitmanagement, Anforderungsdokumentation, Benutzeranleitung, Präsentation
Team Hardware: erstellt den Prototypen und liefert die benötigte Hardware, erstellt den Code für den Microcontroller	
Design	Systemtestvorschrift, Kostenkalkulation und Effizienzbetrachtungen
Hardware	Erstellung der lauffähigen HW, Koordination der Kommunikation zwischen HW und SW, Inbetriebnahme
Team Software: stellt die Kommunikation des Prototypen mit dem Laptop sicher, erstellt den Java-Code	
Anforderungen	Anforderungsdokumentation, Codeentwicklung
Implementierung	Entwurfedokumentation, Codeentwicklung, Implementierung

3 Management-Prozess

3.1 Managementziele und Prioritäten

Ziel dieser Arbeit ist es, einen leicht steuerbaren, fahrtüchtigen Prototypen zu erstellen. Der Fokus liegt dabei besonders auf der Funktionalität. Das Gefährt soll sich frei im Raum bewegen können und die Umgebung filmen. Der gesamte Aufwand berechnet sich aus den geleisteten Arbeitsstunden der Teammitglieder und den Kosten für die Hardware. Alle Deliverables sind termingerecht fertig zu stellen.

3.2 Annahmen und Einschränkungen

Zeitliche Einschränkungen entstehen durch die Tatsache, dass für jedes Teammitglied ein Aufwand von 60 Stunden zu leisten ist. Insgesamt stehen somit 300 Stunden für die Verwirklichung zur Verfügung. Die Terminfristen, welche zwingend einzuhalten sind, sind in Tabelle 1 auf Seite 6 ersichtlich.

Weiter wird die Qualität der Hardware durch die Budgetierung begrenzt. Es ist deshalb darauf zu achten, Kosten einzusparen, indem wieder verwendbare private Hardware benutzt wird.

Softwareseitig wird Open Source Software verwendet und die von Herrn Bonderer zur Verfügung gestellten lizenzpflichtigen Produkte.

3.3 Risikomanagement

In der Analysephase werden in einer separaten Liste alle möglichen Risiken festgehalten. Darin wird auch beschrieben, was zu unternehmen ist, um die entsprechenden Risiken zu minimieren.

3.4 Berichtswesen

3.4.1 Teamsitzungen

Teamsitzungen werden bei Bedarf angeordnet; jedoch findet mindestens alle zwei Wochen eine Sitzung aller Teammitglieder statt. Diese Sitzung dauert ca. eine Stunde und findet üblicherweise am Mittwochnachmittag statt. Es können gruppenintern weitere Sitzungen eingeplant werden, an welchen nicht alle Mitglieder teilnehmen müssen.

3.4.2 Budgetplanung

Es werden – mit Ausnahme von privaten – nur Ausgaben getätigt, welche sowohl vom Projektmentor (Herr Bonderer) als auch allen Teammitgliedern gutgeheissen wurden.

3.4.3 Kommunikation

Als Kommunikationsmittel innerhalb der Gruppe wird der tägliche mündliche Austausch sowie die Teambesprechung bevorzugt. Schriftliche Dokumente, die per E-Mail versandt werden, sind grundsätzlich an alle Teammitglieder zu senden.

3.5 Personalplanung

Tabelle 3: Personalplanung

Rolle	Verantwortlich
Projektleiter	Isabell Sutter
Team Hardware	
Design	Maik Nager
Hardware	Daniel Knöpfel
Team Software	
Anforderungen	Reto Guadagnini
Implementierung	Giorgio Crameri

3.6 Dokumentationsplan

Das Projektteam verfasst während der Durchführung verschiedene Dokumente. Nachfolgende Dokumente müssen zwingend abgegeben werden. Die Abgabefristen sind in Tabelle 1 auf Seite 6 ersichtlich.

- Projektplan
- Anforderungsdokumentation
- Entwurfsdokumentation
- Systemtestvorschrift
- Lauffähiges Produkt
- Benutzerhandbuch
- Präsentation in Powerpoint

Weiter entstehen je nach Projektphase zusätzliche Dokumente, welche aber nicht abgegeben werden müssen.

- Sitzungsprotokolle
- Entwurfskizzen
- Budgetplan
- Risikoliste
- Ergänzung des Quellcodes mit Kommentaren

4 Richtlinien für die Entwicklung

4.1 Konfigurationsmanagement

Bezüglich des Konfigurationsmanagements halten wir uns an die im Theorieunterricht vermittelten Grundlagen über Software Configuration Management[3].

Gemäss unserem Vorgehensmodell in Abbildung 1 auf Seite 8 werden die einzelnen Deliverables laufend angepasst, so dass mehrere Versionen entstehen. Als Verwaltungstool verwenden wir dafür das Concurrent Versions System (CVS) [4]. Es ist jedoch darauf zu achten, die Anzahl der verschiedenen Versionen gering zu halten, um das Zusammenfügen nicht unnötig zu erschweren.

4.2 Entwicklungsrichtlinien

Es werden für die Codierung keine normierten Richtlinien verwendet.

4.3 Prüfungen (Review und Tests)

Getestet wird auf zwei verschiedenen Stufen. Es wird ein Integrationstest und ein Systemtest durchgeführt. Ein Unittest wurde nicht explizit durchgeführt, da sich die verwendeten Klassen nicht gut dafür eignen.

4.4 Einsatz von Werkzeugen

Für die Entwicklung und den Betrieb des Prototyps werden folgende Werkzeuge verwendet:

Tabelle 4: Werkzeugliste

Werkzeug	Anwendung
Windows XP und Vista	Betriebssystem
MS Office Programme	Dokumentationsgestaltung, Graphiken, Präsentation
Eclipse 3.4	Entwicklungsumgebung
Borland Together 2007 Service Pack 1	UML Gestaltungselement, Java-Code-Generierung
Linksys Wireless Internet-Videokamera Modell WVC54GC	Hardware zur Umgebungswahrnehmung
Source Boost DIE 5.9.7	Software für die Codeerstellung in C
Java Development Kit 6.03	Java Library
Open Source Modul RXTX	Serielle Schnittstelle mit Java Umgebung

5 Entwicklungsplan

5.1 Projektstrukturplan (Arbeitsgliederung)

Siehe externes Dokument Projektdiagramme[5].

5.2 Abhängigkeiten

Siehe externes Dokument Projektdiagramme[5].

5.3 Ressourcen Anforderungen

Für die erfolgreiche Durchführung des Projekts werden die in der untenstehenden Tabelle aufgeführten Ressourcen benötigt.

Tabelle 5: Ressourcen

Ressource	Menge
Werkzeuge gemäss Werkzeugliste in Tabelle 4	
Projektteam mit 5 Personen	je 60 Std. pro Person = 300 Std.
Projektbegleitung während des Semesters	1 (Herr Bonderer)
Budget für die Wirelesskamera	CHF 140.-
Linksys Wireless Internet-Videokamera Modell WVC54GC	Hardware zur Umgebungswahrnehmung
Microcontroller: PIC16F688	1 Stk.
Nickel Metallhydrid (NiMH) Akku	1 Stk.

5.4 Kosten- und Ressourcenverteilung

Die Kosten im Rahmen des genehmigten Budgets gehen zu Lasten der HTW Chur. Alle weiteren finanziellen Aufwände werden von den einzelnen Gruppenmitgliedern privat aufgebracht. Diese werden innerhalb der Gruppe nicht aufgeteilt, da die Ressourcen nach Beendigung des Projekts wieder privat gebraucht werden. Der HTW Chur steht nach Abschluss des Projektes eine Kaufoption zum Erwerb des Prototypen offen.

Für die Ressourcenverteilung der Arbeitsstunden siehe externes Dokument Projektdiagramme.

5.5 Terminplan

Grundlage für den Terminplan bildet der Semesterplan des Moduls Software-Engineering im Sommersemester 2008.

Die wichtigsten Termine und Eckdaten sind in Tabelle 1 auf Seite 6 und im externen Dokument Projektdiagramme ersichtlich.

A Referenzen

- [1] IEEE Standard for Software Project Management Plans. IEEE Std 1058-1998.
- [2] Frühauf, K., Ludewig, J., Sandmayr, H.: Software- Projektmanagement und –Qualitätssicherung. vdf Hochschulverlag, Zürich 2002, ISBN 3 7281 2822 8.
- [3] Software Configuration Management, Vers. 1.4 von Reto Bonderer, Modul TeSWE 27.03.08.
- [4] Introduction to CVS. CVS - Concurrent Versions System, 27.03.08
<http://www.nongnu.org/cvs/>
- [5] Projektdiagramme, Vers. 1.1 von Isabell Sutter, Studienarbeit 4, TeSWE 28.05.08.