



HTW Chur
Hochschule für Technik und Wirtschaft

Fachhochschule Ostschweiz
University of Applied Sciences

Anforderungsdokumentation

Projekt: Spionat

Version 1.1

Reto Guadagnini

Änderungsgeschichte

<u>Datum</u>	<u>Version</u>	<u>Autor</u>	<u>Beschreibung</u>
2008-05-01	1.0	R. Guadagnini	Dokument erstellt
2008-05-28	1.1	R. Guadagnini	Dokument anhand des Feedbacks überarbeitet.

Inhaltsverzeichnis

1	EINLEITUNG	6
1.1	Zweck	6
1.2	Grundlage	6
1.3	Umfang	6
1.4	Definitionen, Akronyme und Abkürzungen	6
1.5	Referenzen	6
2	BESCHREIBUNG DER ANFORDERUNGEN	7
2.1	Übersicht	7
2.2	Produkt-Funktionen	7
2.3	Software	7
2.4	Hardware	7
2.5	Schnittstellen	8
2.6	Installationsanforderungen	9
2.7	Einschränkungen	10
2.8	Annahmen und Abhängigkeiten	10
2.9	Unterteilung der Anforderungen	10
3	PRODUKTFUNKTIONEN (FUNKTIONALE ANFORDERUNGEN)	11
3.1	Actors	11
4	SYSTEMATTRIBUTE	12
4.1	Zuverlässigkeit und Verfügbarkeit	12
4.2	Sicherheit	12
4.3	Wartbarkeit	12
5	USE CASES	13
5.1	Use Case Steuerung mit der Tastatur	13
5.2	Use Case Steuerung mit der Maus	14
5.3	Use Case Verbindung herstellen	15

5.4	Use Case Akkus laden.....	16
5.5	Use Case Installation.....	17
5.6	Use Case Deinstallation.....	18
A	REFERENZEN	19

Abbildungsverzeichnis

Abbildung 1: Übersicht	7
Abbildung 2: GUI Steuerung mit der Maus	9
Abbildung 3: Use Case Diagramm	11

Tabellenverzeichnis

Tabelle 1: Steuerbefehle auf der Tastatur	8
---	---

Kapitel 1 – Einleitung

1 Einleitung

1.1 Zweck

Dieses Dokument stellt das Resultat der Anforderungsanalyse im Projekt Spionat dar. Es dokumentiert sowohl die System- als auch die Benutzeranforderungen und ist damit Pflichten- und Lastenheft zugleich. Für das weitere Vorgehen dient es als Grundlage, insbesondere macht es die Resultate des Projekts messbar.

1.2 Grundlage

Dem vorliegenden Dokument liegt kein Standard oder ein weiteres Dokument zu grunde.

1.3 Umfang

Das vorliegende Dokument beschreibt die Anforderungen in allgemeiner Form und geht auf die konkreten Use Cases ein.

1.4 Definitionen, Akronyme und Abkürzungen

COTS: Commercial Off The Shelf = kommerziell verfügbare Komponenten

Entity Klassen: Klassen, welche die Kernfunktionalität der Software darstellen (kein GUI oder ähnliches)

Fahrzeugsoftware: Software, die auf dem Fahrzeug läuft

GUI: Graphical User Interface

Operator: Person, die den Spionat fernsteuert

Steuersoftware: Software auf dem Computer, mit welcher das Fahrzeug ferngesteuert wird

UGV: Unmanned Ground Vehicle

1.5 Referenzen

Keine.

Kapitel 2 – Beschreibung der Anforderungen

2 Beschreibung der Anforderungen

2.1 Übersicht

Im Rahmen des Projekts Spionat ist der Prototyp eines unbemannten landgestützten Fahrzeugs (kurz UGV) zu entwickeln, der von einem Operator vom Computer aus drahtlos ferngesteuert werden kann und mit einer Kamera ausgerüstet ist. Das Bild der Kamera wird ebenfalls drahtlos auf den steuernden Computer übertragen und dem Operator dort angezeigt.

Da das Projekt Spionat im Rahmen des Faches Softwareengineering durchgeführt wird, soll der Schwerpunkt auf der Entwicklung der Software und der damit verbundenen Werkzeuge und Methoden liegen. Die Software auf dem Computer zur Fernsteuerung, die Software des Fahrzeugs und das Fahrzeug selbst sind im Rahmen des Projektes zu entwickeln und zu bauen.

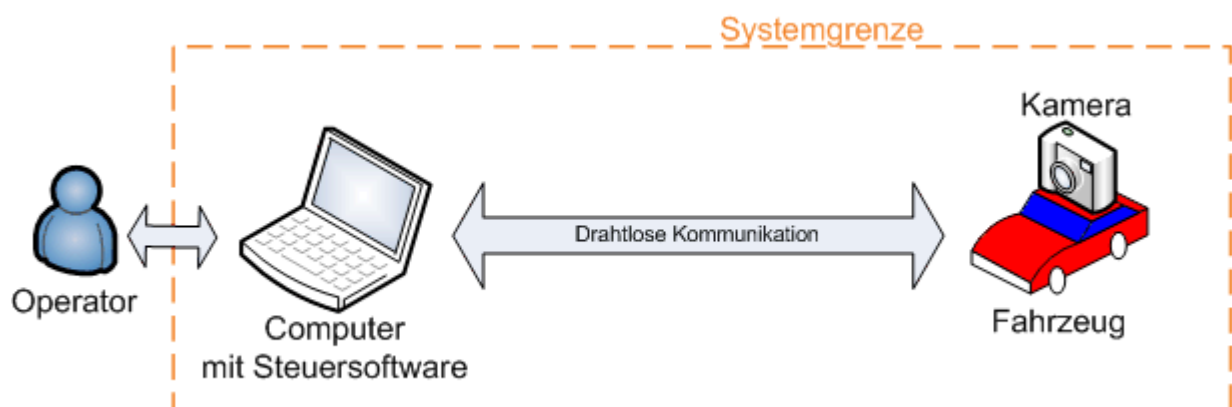


Abbildung 1: Übersicht

2.2 Produkt-Funktionen

(Alle Funktionen aus der Operatorsicht)

- Observieren/Erkunden von Räumen und Objekten mittels der eingebauten Kamera
- Auf-/Abbau der Verbindung zwischen dem steuernden Computer und dem Fahrzeug
- Fernsteuerung vom PC aus (siehe 2.5.1 Benutzerschnittstellen)
- Fahren (siehe 2.4 Hardware)

2.3 Software

Die Steuersoftware ist in Java zu erstellen. Für die Fahrzeugsoftware ist C als Programmiersprache zu verwenden. Der Quellcode ist mit Kommentaren zu versehen.

UML Diagramme sollen für wichtige Programmkomponenten (wie Entity Klassen) eingesetzt werden. Es muss mindestens ein Klassendiagramm erstellt werden.

2.4 Hardware

Für den Bau des Fahrzeugs sind soweit möglich handelsübliche COTS zu verwenden; ebenso für die Kamera und die Einrichtungen zur Kommunikation zwischen Fahrzeug und Computer. Dabei ist auf den Punkt „Kosten und Ressourcenverteilung“ im Projektplan zu achten.

Das Fahrzeug soll sich auf ebenem Gelände (Betonboden, Asphalt) fortbewegen können.

Kapitel 2 – Beschreibung der Anforderungen

2.5 Schnittstellen

2.5.1 Benutzerschnittstellen

Die Steuerungssoftware soll eine grafische Benutzeroberfläche haben, welche die Fernsteuerung des Fahrzeugs per Tastatur und Maus erlaubt und dem Operator das Livebild der Kamera anzeigt. Die Steuerung per Tastatur soll über die vier Pfeiltasten erfolgen. Zur Steuerung per Maus sind sieben Schaltflächen zu erstellen. Das Fahrzeug hat wie folgt auf die Steuerbefehle zu reagieren:

Tastenkombination	Auswirkung
Pfeiltaste nach oben	Fahrzeug fährt in Blickrichtung der Kamera (nach vorne)
Pfeiltaste nach unten	Fahrzeug fährt in die zur Blickrichtung der Kamera entgegengesetzte Richtung (nach hinten)
Pfeiltaste links	Fahrzeug dreht sich aus Sicht der Kamera um die eigene Achse nach links
Pfeiltaste rechts	Fahrzeug dreht sich aus Sicht der Kamera um die eigene Achse nach rechts
Pfeiltaste links und Pfeiltaste nach oben	Fahrzeug fährt nach vorne links
Pfeiltaste rechts und Pfeiltaste nach oben	Fahrzeug fährt nach vorne rechts
Pfeiltaste links und Pfeiltaste nach unten	Fahrzeug fährt nach hinten links
Pfeiltaste rechts und Pfeiltaste nach unten	Fahrzeug fährt nach hinten rechts

Tabelle 1: Steuerbefehle auf der Tastatur

Kapitel 2 – Beschreibung der Anforderungen

2.5.2 Maussteuerung

Abbildung 2 zeigt die zu realisierenden Schaltflächen auf dem GUI zur Steuerung per Maus. Die Beschriftung gibt an, welcher Tastenkombination ein Klicken und Halten auf dieser Schaltfläche entspricht. Die Beschriftung der Schaltfläche ist durch aussagekräftige Symbole zu ersetzen.

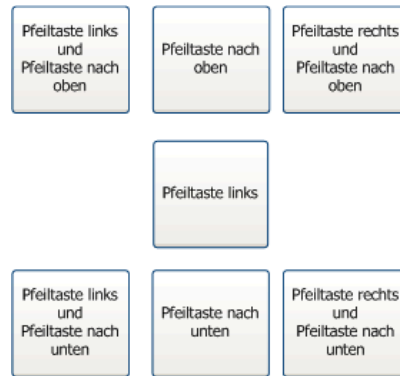


Abbildung 2: GUI Steuerung mit der Maus

2.5.3 Software-Schnittstellen

Für die Kommunikation der Steuersoftware mit der Fahrzeugsoftware ist ein Protokoll zu definieren. Das Protokoll muss es ermöglichen, dass die über die Benutzerschnittstelle erteilten Anweisungen an das Fahrzeug übermittelt werden können, und dass ein Unterbruch der Verbindung erkannt wird.

2.5.4 Kommunikationsschnittstellen

Die Kommunikation des Computers mit dem Fahrzeug hat drahtlos zu erfolgen. Für die Realisierung der Kommunikationsschnittstelle ist soweit möglich auf COTS zurückzugreifen. Es sind Komponenten zu verwenden, welche mit dem in Kapitel 2.6 beschriebenen Computer kompatibel sind. Die Kommunikation des Computers mit dem Fahrzeug soll bei einer Sichtverbindung über eine Entfernung von mindestens 25 Metern möglich sein.

2.6 Installationsanforderungen

An den Computer stellen sich für die Steuersoftware folgende Anforderungen:

- Betriebssystem: Windows XP
- Java Runtime Environment 1.6 von Sun Microsystems
- Wireless LAN, IEEE 802.11G kompatibel
- Bildschirm, Tastatur und Maus

2.6.1 Anforderungen an den installierenden Benutzer

Siehe 3.1 Actors.

Kapitel 2 – Beschreibung der Anforderungen

2.7 Einschränkungen

Der im Rahmen des Projekts zu erstellende Spionat ist ein Prototyp. Daher muss die Zuverlässigkeit der Hardware nicht der eines industriell angefertigten Produkts entsprechen. Ziel ist, dass das Fahrzeug unter Laborbedingungen (Aula der HTW) funktioniert. Für die Software gilt diese Einschränkung nicht.

2.8 Annahmen und Abhängigkeiten

Keine.

2.9 Unterteilung der Anforderungen

Die Anforderungen werden nicht unterteilt.

Kapitel 3 – Produktfunktionen (Funktionale Anforderungen)

3 Produktfunktionen (Funktionale Anforderungen)

Das zu erstellende System wurde für die Erfassung der Use Cases als Ganzes modelliert. Der Spionat besteht aus der Steuerungssoftware, der Hardware und der Fahrzeugsoftware. Die einzelnen Use Cases sind in Kapitel 5 beschrieben.

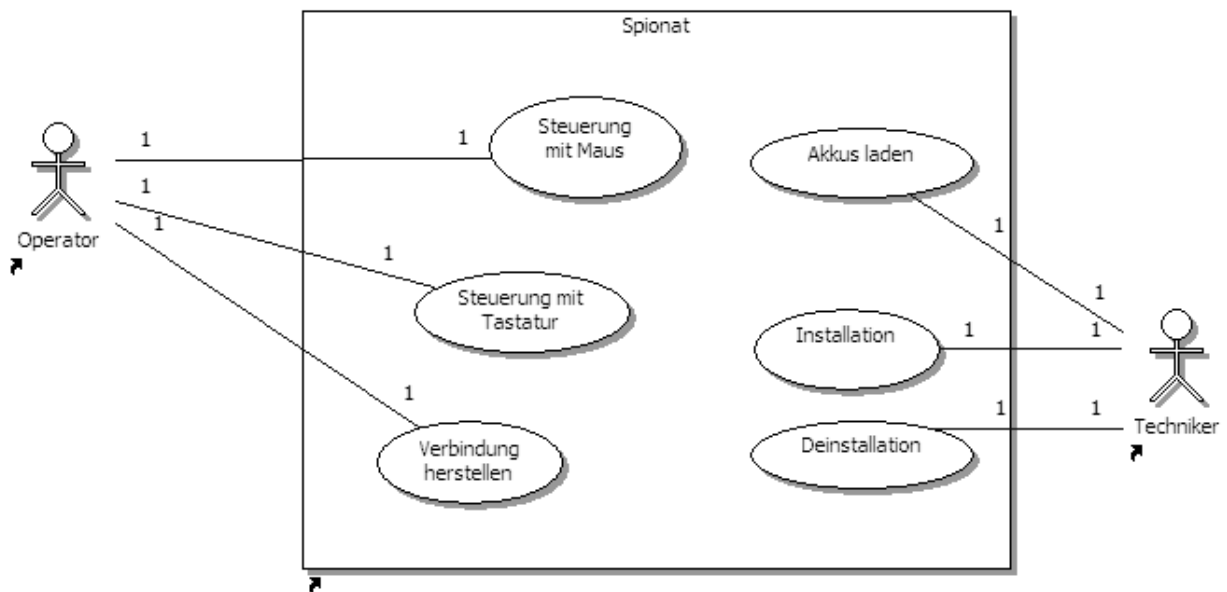


Abbildung 3: Use Case Diagramm

3.1 Actors

Als Actors treten der Operator, der den Spionat im normalen Betrieb bedient, und der Techniker auf, welcher die Installations-, Deinstallations- und Wartungsarbeiten (Akkus aufladen) durchführt.

Der Operator muss über gute Anwenderkenntnisse von Windowsprogrammen verfügen, um den eingerichteten Spionat zu bedienen.

Da es sich um einen Prototypen handelt, muss der Techniker über Kenntnisse verfügen, wie sie ein gelernter Elektroniker/Informatiker mitbringt.

Kapitel 4 – Systemattribute

4 Systemattribute

4.1 *Zuverlässigkeit und Verfügbarkeit*

Für Angaben zur Zuverlässigkeit und Verfügbarkeit siehe Kapitel 2.7.

4.2 *Sicherheit*

Falls die Kommunikation zwischen dem Fahrzeug und der Steuersoftware unterbrochen wird, muss das Fahrzeug innert einer Sekunde anhalten, um allfällige Schäden am Fahrzeug und der Umgebung zu verhindern.

4.3 *Wartbarkeit*

Da es sich um einen Prototypen handelt wird, kein spezieller Wert auf die Wartbarkeit der Hardware gelegt.

Für die Wartbarkeit der Software siehe unter 2.7.

Kapitel 5 – Use Cases

5 Use Cases

5.1 Use Case Steuerung mit der Tastatur

5.1.1 Vorbedingungen

Die Verbindung der Steuersoftware auf dem Computer mit der Fahrzeugsoftware ist hergestellt.

5.1.2 Nachbedingungen

Das Fahrzeug steht still.

5.1.3 Nicht-funktionale Anforderungen

Die Steuerung per Tastatur ist der häufigste Use Case.

5.1.4 Hauptszenario

Der Operator betätigt eine oder zwei Pfeiltasten auf der Tastatur, um das Fahrzeug fernzusteuern. Die Steuersoftware sendet daraufhin die entsprechenden Steuerbefehle an das Fahrzeug. Die Fahrzeugsoftware wertet die Befehle aus und setzt sie um.

Für genauere Angaben zur Steuerung per Tastatur siehe Kapitel 2.5.1 Tabelle 1.

5.1.5 Unterszenarien

Für die Funktion der einzelnen Tastenkombinationen siehe 2.5.1.

5.1.6 Fehlerszenarien

[E-1]: Der Operator drückt mehr als zwei Tasten gleichzeitig oder verwendet eine ungültige Tastenkombination. Auf ungültige Tastenkombinationen soll keine Reaktion erfolgen.

5.1.7 Regeln

Keine.

5.1.8 Anmerkungen

Keine.

5.1.9 Beispiele

Der Operator will, dass das Fahrzeug nach vorne fährt. Er betätigt also die Pfeiltaste nach oben und das Fahrzeug fährt in die gewünschte Richtung.

Kapitel 5 – Use Cases

5.2 Use Case Steuerung mit der Maus

5.2.1 Vorbedingungen

Die Verbindung der Steuerungssoftware auf dem Computer mit der Fahrzeugsoftware ist hergestellt.

5.2.2 Nachbedingungen

Das Fahrzeug steht still.

5.2.3 Nicht-funktionale Anforderungen

Keine.

5.2.4 Hauptszenario

Der Operator klickt mit der Maus auf eine der Schaltflächen im GUI und hält die Maustaste gedrückt um das Fahrzeug fernzusteuern. Die Steuerungssoftware sendet daraufhin die entsprechenden Steuerbefehle an das Fahrzeug. Die Fahrzeugsoftware wertet die Befehle aus und setzt sie um.

5.2.5 Unterszenarien

Für die Funktion der einzelnen Schaltflächen siehe 2.5.2.

5.2.6 Fehlerszenarien

Keine.

5.2.7 Regeln

Keine.

5.2.8 Anmerkungen

Keine.

5.2.9 Beispiele

Der Operator will, dass das Fahrzeug nach vorne fährt. Er klickt auf die Schaltfläche mit dem entsprechenden Symbol und hält die Maustaste gedrückt, solange er will, dass das Fahrzeug vorwärts fährt. Das Fahrzeug fährt in die gewünschte Richtung.

Kapitel 5 – Use Cases

5.3 Use Case Verbindung herstellen

5.3.1 Vorbedingungen

Die Steuersoftware ist gestartet und es besteht noch keine Verbindung zum Fahrzeug.

5.3.2 Nachbedingungen

Die Verbindung steht und das Fahrzeug ist bereit, seine Befehle zu empfangen.
Das Livebild der Kamera wird in der Steuersoftware angezeigt.

5.3.3 Nicht-funktionale Anforderungen

Keine.

5.3.4 Hauptszenario

Der Operator klickt mit der Maus auf die Schaltfläche „Verbinden“ des GUI. Die Steuersoftware stellt die Verbindung mit dem Fahrzeug her und meldet den erfolgreichen Verbindungsaufbau.

5.3.5 Unterszenarien

Keine.

5.3.6 Fehlerszenarien

[E-1]: Der Verbindungsaufbau ist fehlgeschlagen. Es wird eine entsprechende Fehlermeldung angegeben.

5.3.7 Regeln

Keine.

5.3.8 Anmerkungen

Keine.

5.3.9 Beispiele

Keine.

Kapitel 5 – Use Cases

5.4 Use Case Akkus laden

5.4.1 Vorbedingungen

Die Akkus sind ganz oder teilweise entladen.

5.4.2 Nachbedingungen

Die Akkus sind geladen.

5.4.3 Nicht-funktionale Anforderungen

Keine.

5.4.4 Hauptszenario

Der Techniker schliesst die Akkus an das Ladegerät an.

5.4.5 Unterszenarien

Keine.

5.4.6 Fehlerszenarien

[E-1]: Die Akkus sind defekt. Sie müssen durch neue ersetzt werden.

5.4.7 Regeln

Keine.

5.4.8 Anmerkungen

Bei diesem Use Case handelt es sich um keine Softwarefunktion.

5.4.9 Beispiele

Keine.

Kapitel 5 – Use Cases

5.5 Use Case Installation

5.5.1 Vorbedingungen

Der Computer erfüllt die unter 2.6 definierten Anforderungen.

5.5.2 Nachbedingungen

Der Spionat ist einsatzbereit.

5.5.3 Nicht-funktionale Anforderungen

Keine.

5.5.4 Hauptszenario

Der Spionat soll vom Techniker betriebsbereit gemacht werden. Dazu installiert er die Steuersoftware auf dem Computer, konfiguriert die drahtlose Kommunikation, lädt die Akkus und prüft, ob eine Verbindung zwischen der Steuersoftware und dem Fahrzeug hergestellt werden kann.

5.5.5 Unterszenarien

Keine.

5.5.6 Fehlerszenarien

[E-1]: Der Computer erfüllt die Installationsanforderungen nicht. In diesem Fall ist eine Installation nicht möglich.

5.5.7 Regeln

Keine.

5.5.8 Anmerkungen

Keine.

5.5.9 Beispiele

Keine.

Kapitel 5 – Use Cases

5.6 Use Case Deinstallation

5.6.1 Vorbedingungen

Der Spionat ist auf dem Computer eingerichtet.

5.6.2 Nachbedingungen

Auf dem Computer ist keine Steuersoftware und keine Kommunikationseinstellung mehr für den Spionat vorhanden.

5.6.3 Nicht-funktionale Anforderungen

Keine.

5.6.4 Hauptszenario

Die Steuerung soll nicht mehr von diesem Computer aus erfolgen. Der Techniker löscht die Steuersoftware auf dem Computer, entfernt die Einstellungen für die drahtlose Kommunikation und deinstalliert die Treiber.

5.6.5 Unterszenarien

Keine.

5.6.6 Fehlerszenarien

Keine.

5.6.7 Regeln

Keine.

5.6.8 Anmerkungen

Keine.

5.6.9 Beispiele

Keine.

Anhang A – Referenzen

A Referenzen

- [1] Anforderungsanalyse (Skript), Reto Bonderer, Modul TeSWE