

## *Racer Empire GmbH*

Gr.-Nr.	Gruppenname	Teilnehmer mit Email-Adresse
6	Raceisten	Christian Eisenberger, eisenbec@hs-weingarten.de Denis Herdt, herdtd@hs-weingarten.de Felix Wiedemann, wiedemaf@hs-weingarten.de Michael Sommer, sommermi@hs-weingarten.de

Titel	Datum	Version
Fernsteuerung per Raspberry Pi	16.10.14	0.1

## Software Engineering Praktikum

# Pflichtenheft

## Inhaltsverzeichnis

1. Zielsetzung .....	3
1.1 Pflichtkriterien .....	3
1.2 Optionale Kriterien .....	3
2. Produkteinsatz .....	4
2.1 Einsatzbereich .....	4
2.2 Zielgruppen .....	4
2.3 Betriebsbedingungen .....	4
3. Produktumgebung .....	5
4. Produktdetails .....	6
4.1 Steuerung des Autos .....	7
4.2 Webcam .....	7
5. Produktleistung .....	8
6. Hardware .....	9
7. Qualitätszielbestimmung .....	10
8. Tests .....	10
9. Entwicklungsumgebung .....	11
10. Use Case Diagramm .....	11

## 1. Zielsetzung

### 1.1 Pflichtkriterien

- Der Benutzer
  - Startet das System
  - Steuert mit der Tastatur eines Laptops das Auto

Hinweis: In Use-Case genauer beschrieben

- Das Auto
  - Fährt in alle 4 Richtungen (Links, Rechts, Vorwärts, Hinten)
  - Reagiert zeitnah (innerhalb einer Sekunde) auf Steuerbefehle
  - Reagiert flüssig auf Steuerbefehle
  - Reagiert zuverlässig auf Steuerbefehle
  - Hält automatisch an, wenn es keine Befehle bekommt

### 1.2 Optionale Kriterien

- Der Benutzer steuert das Auto mit einen PS3 Controller
- Der Benutzer steuert das Auto über die Kippsensoren eines Smartphones
- Das Auto sendet ein Kamerabild zum Laptop, welches in einem Fenster dargestellt wird
- Autonomes Fahren über eine KI und Sensoren
- Geschwindigkeits- und Lenkverhalten ist stufenlos anpassbar

## 2. Produkteinsatz

### 2.1 Einsatzbereich

Das ferngesteuerte Auto soll die gleichen Funktionen wie ein handelsübliches ferngesteuertes Auto bieten.

Durch die Steuerung über den Raspberry Pi soll eine Plattform geboten werden, um ohne größeren Aufwand Erweiterungen wie zB. Die Übertragung eines Kamerabildes oder Ähnliches realisierbar zu machen.

### 2.2 Zielgruppen

Wir unterscheiden zwischen 2 Anwendern:

- Der erste Nutzer will das Auto mit den aktuellen Funktionen nutzen. Dafür müssen geringe Computerkenntnisse vorhanden sein, um die WLAN Verbindung herzustellen und das Programm zu starten.
- Der zweite Nutzer entscheidet sich bewusst für das Produkt, da er das Auto mit weiteren Funktionen erweitern will. Dafür müssen Grundkenntnisse in Programmieren und Elektrotechnik vorhanden sein, um neue Komponenten einzubauen, mit dem Raspberry Pi zu verbinden und die gewünschte Funktion zu implementieren.

### 2.3 Betriebsbedingungen

Das Auto funktioniert nur, wenn beide Akkus (Raspberry Pi und Motor/Servos) geladen sind. Des weiteren muss ein WLAN Netz vorhanden sein, welches genutzt werden kann.

### 3. Produktumgebung

#### Software:

- Raspbian (OS des Raspberry Pi, basierend auf Debian)
- Windows 8.1
- Samba Server
- Putty und WinSCP

#### Hardware

- Auto inc. Motor und Servos
- Raspberry Pi inc. Gehäuse
- 2 Akkus (Raspberry Pi und Motor)
- H-Brücke
- WLAN Stick
- Laptop
- Diverse Jumperkabel

optional werden noch folgende Komponenten benötigt:

- PS3 Controller
- Bluetooth Empfänger
- Webcam
- Smartphone
- Annäherungssensoren (Ultraschallsensor)

## 4. Produktdetails

Als Basis dient ein ferngesteuertes Auto mit Elektromotor als Antrieb. Dort wird die Platine entfernt und der Raspberry Pi samt Akku platziert. Motor und Servos der Lenkung werden an die GPIOs des Raspberry Pis angeschlossen und über diese angesteuert. Die Stromversorgung des Motors wird über extra Akkupacks gewährleistet, welche mit einer höheren Spannung arbeiten als der Akku des Raspberry Pis. Um das Rückwärtsfahren zu realisieren, wird eine H-Brücke verwendet.

Auf dem Raspberry Pi ist das OS Raspbian installiert. Dort ist die zur Kommunikation mit dem WLAN und die Steuerung des Autos bzw. Interpretation der Fahrbefehle des Laptops/PS3 Controller/Smartphone notwendige Software installiert. Es wird, soweit möglich, vorhandene Software genutzt. Die verwendete Software ist in Kapitel 3 aufgelistet.

Auf dem Laptop werden die eingehenden Signale verarbeitet:

- Weitergabe über Bluetooth an den PS3 Controller
- Ausgabe eines Bildes auf dem Bildschirm (bei Verwendung einer Webcam)

Des weiteren sendet der Laptop an den Raspberry folgende Signale:

- Weitergabe der Steuerungssignale des PS3 Controllers
- Fahrbefehle der Tastatur

Falls ein Smartphone zur Steuerung des Autos verwendet wird, ist dieses über WLAN mit dem Raspberry Pi verbunden. Es empfängt, falls eine Webcam verbaut ist, das Kamerabild und versendet die Fahrbefehle. Die Darstellung des Kamerabildes und die Steuerung werden über eine App realisiert.

## 4.1 Steuerung des Autos

Es gibt 3 Realisierungskonzepte für die Steuerung des Autos:

### Laptop:

Die Standardsteuerung des Autos funktioniert über einen Laptop. Mit den Pfeiltasten der Tastatur kann das Auto vorwärts, rückwärts, links und rechts gesteuert werden. Da das Signal nur gedrückt oder nicht gedrückt sein kann, wird das Auto mit vollen Lenkeinschlag, bzw. Vollgas reagieren. Optional wird ein Regler implementiert, mit dem die Empfindlichkeit der Steuerung angepasst werden kann.

### PS3 Controller (optional):

Die optionale Steuerung über einen über Bluetooth angeschlossenen PS3 Controller ist komfortabler als mit dem Laptop. Durch die analogen Bedienelemente des Controllers kann das Signal genauer abgelesen und umgesetzt werden, was verschiedene Lenkwinkel und Geschwindigkeiten ermöglicht.

### Smartphone (optional):

Eine weitere optionale Steuerungsoption ist die Steuerung über eine App auf einem Smartphone. Die Steuerung erfolgt mit Hilfe der Kippsensoren des Smartphones. Um diese Steuerung zu realisieren, muss eine eigene App geschrieben werden. Die Steuerung wäre analog, würde aber viel Übung benötigen, um sie zu meistern.

## 4.2 Webcam

Eine weitere Option wäre das Anbringen einer Webcam und das Übertragen des Fahrtbildes auf eines der Endgeräte. Dies würde die verschiedenen Steuerungsoptionen verbessern und den Einsatzradius erhöhen.

Die Webcam müsste am Auto nach vorne verbaut sein und ihr Bild an den Raspberry Pi über das WLAN-Signal an den Laptop bzw. das Smartphone weitergeben.

## 5. Produktleistung

Zuverlässigkeit, Korrektheit:

Die über das WLAN übertragenen Signale müssen zuverlässig übertragen werden.

Dabei sind verschiedene Gesichtspunkte wichtig:

- Die Signale müssen zeitnah übertragen werden, damit ein flüssiger und sicherer Fahrbetrieb gewährleistet ist. Dies ist insbesondere bei Ausweichmanövern wegen Hindernissen wichtig.
- Die Signale müssen immer korrekt übertragen werden. Auch hier spielt der physische Sicherheitsaspekt die größte Rolle.
- Beim Betrieb mit der Webcam muss der Empfang entsprechend gut sein, dass das Kamerabild klar übertragen wird, da dies sonst bei Fahrten außerhalb des eigenen Sichtfelds zu Problemen führt.

Bedienbarkeit:

Die Steuerung muss bedienbar sein. Mit der Vorauswahl (Tastaturtasten auf dem Laptop/PS3 Controller/Kippsensoren des Smartphones) wurden Eingabegeräte gewählt, die einen flüssigen und flotten Betrieb des Autos gewährleisten können.

Allerdings muss beachtet werden, dass die Steuerung über den Laptop schwieriger ist wie die anderen beiden Möglichkeiten, da über die Tastatur nur 2 Zustände gibt.

Vollgas oder Stehen bleiben bzw. Beim Lenken Volleinschlag (Korrigierbar mit Empfindlichkeitsregler). Da beim Smartphone Übung dazu gehört, mit den Kippsensoren zu arbeiten, dürfte der PS3 Controller (eventuell mit einem Kamerabild auf dem Bildschirm des Laptops) die komfortabelste Möglichkeit sein.



## 6. Hardware

Die verwendeten Hardwarekomponenten sind:

- Mercedes AMG C Klasse Cassis mit Elektromotor 9,6V
- Servo 3 Pin Anschluss
- H-Brücke SEOK DC-Motortreiber Brett Modul L298N
- Raspberry Pi B+ Modell
- 8 Gb MicroSD Karte
- WLAN Stick 300 Mbit/s
- Akkupack Raspberry Pi Anker 5V/3A/10Ah
- Akkupack Auto 9,6V/2A/750mAh
- Laptop Toshiba Satellite 5850
- PS3 Controller
- Webcam
- Smartphone
- Bluetooth Empfänger (USB Anschluss)
- 4-Pin Ultraschallsensor HC-SR04
- Diverse Jumperkabel

## 7. Qualitätszielbestimmung

Sehr wichtig:

- Korrektheit: wichtigstes Kriterium zur Unfallvermeidung. Wenn man vor einem Hindernis oder einer Person bremst, darf das Auto nicht beschleunigen.
- Zuverlässigkeit: wichtiges Kriterium zur Unfallvermeidung.  
Wichtiges Kriterium für den Spaßfaktor.

Wichtig:

- Robustheit: Software und Hardware müssen robust realisiert sein.
- Bedienbarkeit: Die Bedienbarkeit der verschiedenen Steuerungsmöglichkeiten ist essentiell für den Spaßfaktor des Autos, aber nicht sicherheitsrelevant.
- Kompatibilität: die Erweiterbarkeit des Systems zur Realisierung weiterer Funktionen ist ein wichtiges Feature des Projekts.

Weniger wichtig:

- Effizienz
- Portierbarkeit

## 8. Tests

Jeder Funktionsabschnitt wird parallel zur Entwicklung getestet:

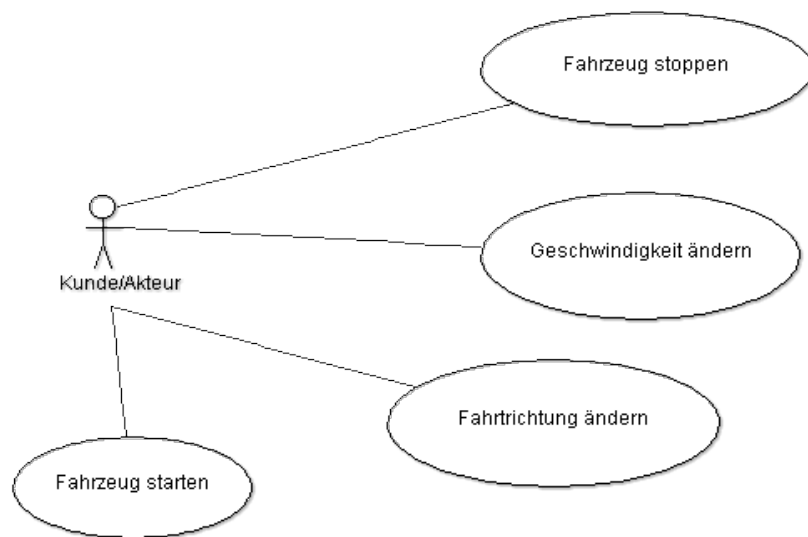
- Kommunikation Raspberry Pi – Netzwerk
- Kommunikation Raspberry Pi – Auto
- Kommunikation Raspberry Pi – Laptop
- Stromversorgung Raspberry Pi, Motor und Servos
- PS3 Controller
- Funktion Webcam
- Smartphone App

## 9. Entwicklungsumgebung

Folgende Tools werden bei der Entwicklung benötigt:

- Notepad ++
- VIM

## 10. Use Case Diagramm



Geschwindigkeit ändern (vorwärts):

Drücken der W-Taste (Tastatur), X/R2-Taste (Controller)

Geschwindigkeit ändern (rückwärts):

Drücken der S-Taste (Tastatur), Kreis/L2-Taste (Controller)

Fahrtrichtung ändern (links):

Drücken der A-Taste (Tastatur), Linken Joystick nach links (Controller)

Fahrtrichtung ändern (rechts):

Drücken der D-Taste (Tastatur), Linken joystick nach rechts (Controller)

Fahrzeug starten/stoppen:

Programm starten/stoppen