

# Projekt: Pi-lot

## Projektmitglieder:

MS: Bager Bingöl (2239338), Andre Till (2239431)

MT: Philipp Dreyer (2249310), Frederik Gutbrod (2248407)

## Ziele und Aufgaben der Anwendung

### Projektziel

Auf der Jahresausstellung sollen die Besucher die Möglichkeit haben durch unser Projekt, "Pi-lot", einen ferngesteuerten Wagen zu lenken. Eine für dieses Projekt entwickelte App, die den Besucher den Platz des Fahrers einnehmen lässt, soll den Fahrspaß erhöhen. Auch die eines Autos nachempfundene Konstruktion soll eine intuitive Lenkung ermöglichen. Damit der Besucher das Interesse am Fahren nicht verliert, hat er die Möglichkeit auf einer kleinen Rennstrecke sich mit anderen Besuchern zu messen.

### Anforderungsanalyse

Unsere Aufgabe ist es ein Fahrzeug so zu entwerfen, dass jeder, der schon einmal ein Auto per Controller in einem Spiel gesteuert hat, auch unseren Wagen intuitiv fahren kann. Hierbei liegt die Schwierigkeit bei der Konstruktion einer Lenkachse. Der Besucher soll Hindernissen ausweichen oder vorgegebene Strecken folgen können.

Der Besucher sitzt als Fahrer in der vordersten Reihe. Dazu muss eine bewegliche Kamera am Wagen angebracht werden. Der Wagen wird, durch einen Motor, in Bewegung gesetzt, der am hinteren Teil befestigt ist.

Der Motor wird mittels eines Motortreibers gesteuert. Parallel zur Steuerung des Motors überwacht der Motortreiber auch die Spannungsversorgung.

Die für den Motor notwendigen Steuersignale werden vom Raspberry Pi erzeugt. Die Steuerung des Wagens wird mittels eines Gamepads vom Besucher betätigt.

Die angebrachte Kamera kann in der App, durch eine View, betrachtet werden. Hierbei wird das Bild von der Kamera erzeugt, im Raspberry Pi kodiert, über das Netzwerk verschickt und von der App dekodiert und angezeigt.

# Technische Umsetzung

## Technische Rahmenbedingung

### Wagen

Der Antrieb des Wagens wird über ein DC-Motor gesteuert, welcher mit einem Keilriemen oder einem Gummiband direkt mit der Antriebsachse verbunden ist. Die Übersetzung zwischen Motor und Antriebsachse ist noch nicht festgelegt und kann mit der Ritzelgröße variiert werden.

Die Lenksteuerung besteht zum wesentlichen aus einem Servomotor, einer Lenkachse, einer Umlenkung und der Radaufhängung. Der Servo wird per Servoarm und Öse an der Achse befestigt, welche wiederum in die Umlenkung eingehängt wird. Die Räder sind direkt mit der Umlenkung verbunden.

Die Kamera wird samt 3D gedruckter Halterung auf einem 90° gedrehten Servomotor montiert. Somit kann sich die Kamera nur auf einer Achse (Pan-Achse) bewegen, was aber auch durch das verwendete Flachbandkabel limitiert wird.

### Motortreiber

Der Motortreiber besteht aus einer H-Brücke und einem Arduino.

Die H-Brücke dient zur Steuerung des Motors. Der Arduino erzeugt die dafür notwendigen Signale, die zur H-Brücke weitergeleitet werden. Vom Raspberry Pi werden Servopulssignale empfangen. Diese müssen dekodiert und auf die im Arduino enthaltene PWM Hardware weitergeleitet werden. Zusätzlich erfolgt die Spannungsüberwachung des Akkumulators. Ein Spannungsteiler an einem analogen Eingang erzeugt hierfür das notwendige Spannungsniveau. Kurz vor der Entladeschlussspannung signalisiert der Arduino dem Raspberry Pi den niedrigen Akkustand.

### Fernsteuerung

Für die Fernsteuerung des Wagens wird als Eingabegerät ein Playstation 3 Controller genutzt. Dieser verbindet sich mit dem Raspberry Pi mittels Bluetooth und nutzt die Linux Joystick-API als Software Schnittstelle. Die Software wandelt die erhaltenen Werte vom Controller in Servopulssignale um und gibt diese auf den GPIO Header des Raspberry Pi's aus. Hierfür eignet sich pigpio lib, die mittels DMA und einem PWM Taktgeber präzise Servosignale auf fast allen GPIO Pins erzeugen kann. Die Fernsteuerungssoftware wird in der Programmiersprache C realisiert.

## First Person View

Ein Raspberry Pi erzeugt mit der Pi-Kamera einen Videostream. Dieser wird mittels der Software "MJPEG-Streamer" in ein MJPEG-Stream umgewandelt und der Netzwerkschnittstelle bereitgestellt. Wir behalten uns vor weitere Möglichkeiten zu evaluieren und gegebenenfalls umzusetzen. Dieser Stream wird von unserer App, die auf einem Android Smartphone läuft, empfangen und auf einem Surface-View wiedergegeben. Unsere App wird mit Android Studio realisiert und mit der Programmiersprache Java entwickelt.

## Technisches Konzept

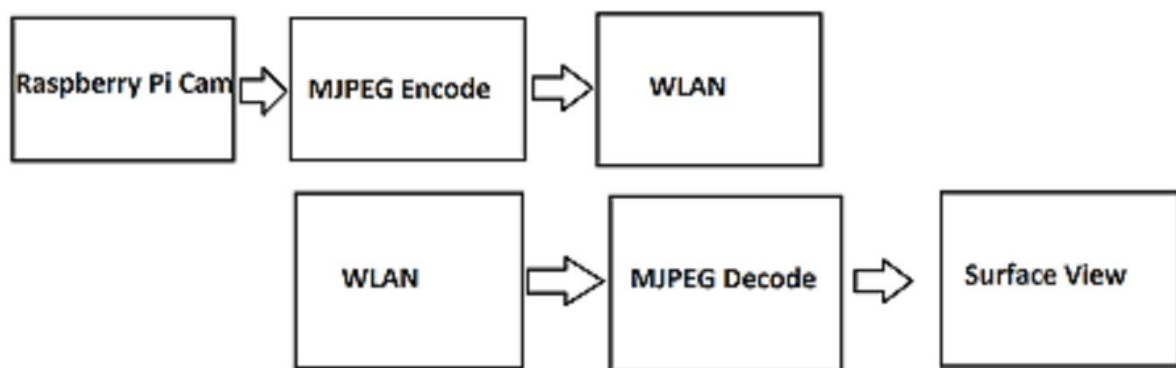


Abbildung 1:First Person View App

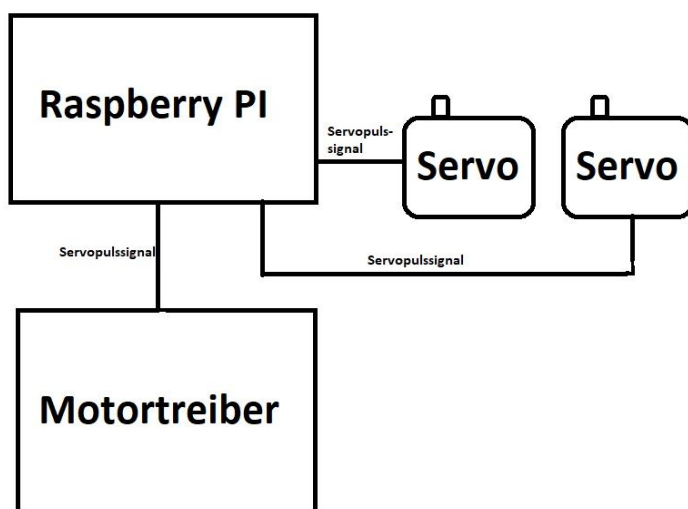


Abbildung 2:Fernsteuerung

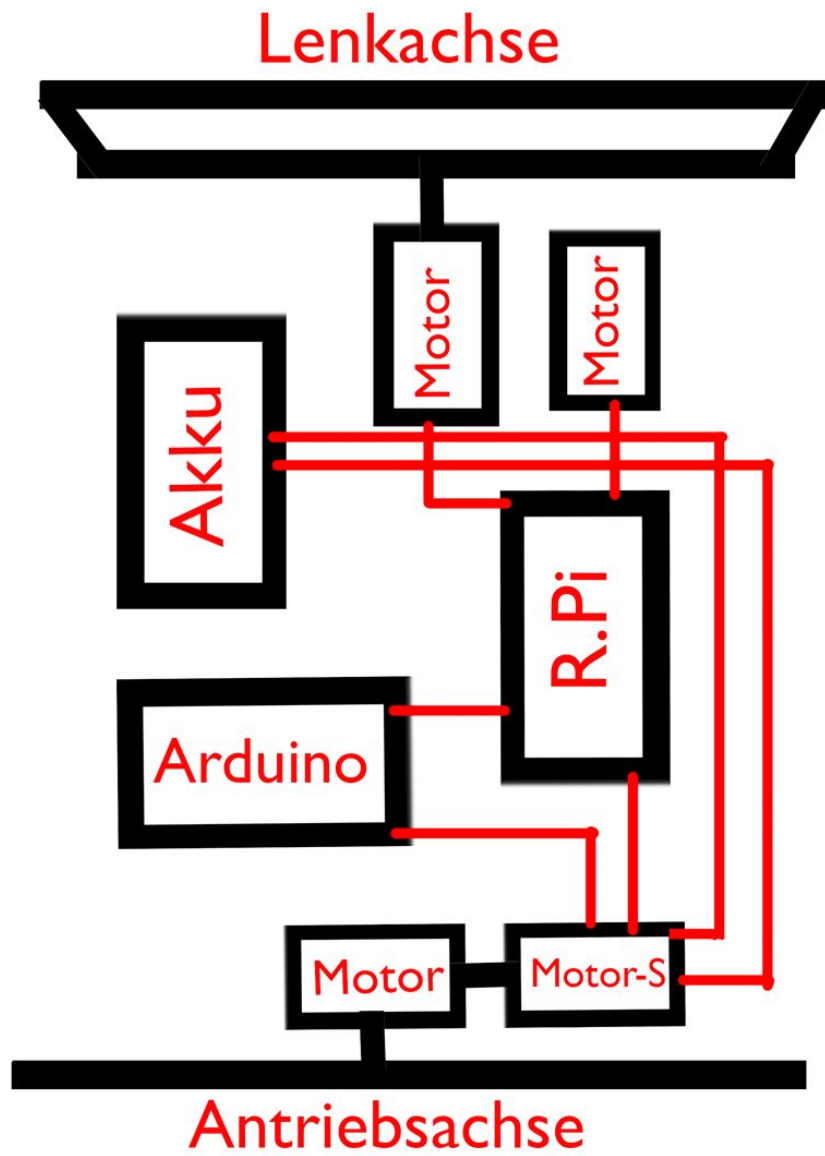


Abbildung 3: Skizze Wagen

## Bedienkonzept

Der Benutzer setzt das CardBoard auf, welches das Smartphone beinhaltet.  
Es wird pro Auge jeweils ein Surface dargestellt, die den Video-Feed enthalten. Der Augenabstand kann mit dem Lautstärkeregler am Handy angepasst werden.  
Anschließend wird das Handy am CardBoard fixiert und auf dem Kopf des Benutzers angebracht.

Daraufhin bekommt der Anwender einen Playstation 3 Controller in die Hand.  
Mit der hinteren, rechten Schultertaste (R2) kann er das Auto in Bewegung setzen.  
Durch die hintere, linke Schultertaste (L2) kann er den Rückwärtsgang betätigen.  
Der linke Analogstick steuert die Lenkachse des Wagens zur rechten und linken Seite.  
Die Ausrichtung der Kamera, kann mithilfe des rechten Analogsticks horizontal verändert werden.

Um den Akku vor Tiefentladung zu schützen, fährt der Raspberry Pi, kurz vor der Entladeschlussspannung, herunter.  
Hier muss der Akku gewechselt bzw. neu aufgeladen werden.

# Zeit- und Ressourcenplan

## Zeitplan

### Mechanische Meilensteine:

- Fertigstellung des Grundgerüsts (08.05.)
  - Das Grundgerüst ist fertiggestellt, wenn alle benötigten Bauteile ihren Platz auf dem Gerüst gefunden haben
- Fertigstellung Antriebsmechanik (15.05.)
  - Der Antriebsmechanismus ist fertiggestellt, wenn der DC Motor erfolgreich die Hinterachse drehen kann und keine Schwierigkeit mit dem Keilriemen auftreten.
- Fertigstellung Lenkmechanismus (29.05.)
  - Der Lenkmechanismus ist fertiggestellt, wenn der Servomotor auch mit belasteten Rädern die Umlenkung über die 3D gedruckten Teile auf die Reifen übertragen kann.
- Fertigstellung Spannungsversorgung (22.05.)
  - Die Spannungsversorgung ist fertiggestellt, wenn wir an allen mit Spannung versorgten Bauteilen, die benötigte Spannung ohne Schwankungen messen können.
- Fertigstellung Verkabelung (29.05.)
  - Die Verkabelung ist fertiggestellt, wenn alle benötigten Daten und Stromleitungen fest mit ihren zugehörigen Pins verbunden sind und der Kabelbau einen sicheren Platz auf der Grundplatte gefunden hat.
- Fertigstellung Motorsteuerung (29.05.)
  - Die Motorsteuerung ist fertiggestellt, wenn der Programmcode des Arduino die Geschwindigkeit und das Bremsen des Motors steuern kann.
- Fertigstellung der Kamera-Mechanik (29.05.)
  - Die Kamera-Mechanik ist fertiggestellt, wenn der Programmcode das Raspberry Pi die Kamera auf ihrer Achse bewegen kann.

### Software Meilensteine:

- Fertigstellung Spannungskontrolle Akku (29.05.)
  - Die Spannungskontrolle ist fertiggestellt, wenn wir über den an den Spannungsteiler angeschlossenen Arduino die Restspannung in der Akkuzelle auslesen können
- Fertigstellung Software (29.05.)
  - Die Software ist fertig, wenn ein Bild vom Raspberry Pi zum App, mit geringerer Verzögerung, angezeigt wird.
- Fertigstellung Bluetooth Funkfernsteuerung (29.05.)
  - Die Software ist fertiggestellt, wenn Tastendrücke von dem Playstation 3 Controller in Servopulssignale umgewandelt werden.

### Allgemeine Meilensteine:

- Erster "Sicker Jump" überlebt (12.7)

# Einkaufsliste

**Projektname:** Pi-Lot

**Antrieb:**

- Motor (DC) - Privat / FH?
- Gummiband - Extern
- Metallstange (4mm) - Extern
- 2x Kugellager ((innen 4mm), (außen 7mm)) - Extern

**Lenkung:**

- 2x Kugellager ((innen 4mm), (außen 7mm)) - Extern
- Servo - Reichelt - Artikel-Nr.: MASTER S4020
- Metallstange- Extern
- 2x Ösen für Lenkstangenende - Extern

**Kamera:**

- Servo - Reichelt - Artikel-Nr.: GRV SERVO
- Raspberry Pi-Cam - FH/Privat

**Technik:**

- Motortreiber 78M05 - FH
- Raspberry-Pi (SD-Karte) - FH
- Arduino - FH
- Akku - FH
- DC-DC Spannungswandler LM2596 - FH

**Wagen:**

- Holz (6mm) - Extern
- Holz - Extern
- Leiterplattenabstandshalter (Gewindestange + Hülsen + Mutter) - Extern

**Allgemein:**

- Schrauben - FH

**App:**

- PS3-Controller - FH/Privat
- Google CardBoard mit Kopfband -FH/Extern

**Fancy Extras:**

- LED (RGB / Weiß) - Extern
- Piezo Summer - Reichelt - Artikel-Nr.: SUMMER AL-60P12