

CoffeeBot

Die Kaffeemaschine, die über eine App einstellbar ist und am Morgen zur gewünschten Zeit frisch gekochten Kaffee ans Bett liefert.

1. Einleitung

Der Wunsch nach einer frischen Tasse Kaffee am Morgen, ohne das Bett zu verlassen und ohne sich zu bewegen, ist in der smarten Welt, wie wir sie aktuell kennen, noch nicht gängig. Um dieser Problemstellung entgegenzuwirken, wird der *CoffeeBot* entwickelt.

2. Grundbegriffe

2.1. Raspberry Pi

Der Raspberry Pi ist ein kleiner Computer. Der Rechner enthält ein Ein-Chip-System und einen Mikroprozessor, die alle auf einer Platine verbaut sind. Das Betriebssystem des Raspberry Pi ist ein Unix-System.

2.2. React Native

Mit React Native können native Apps für Android und iOS mithilfe von Javascript entwickelt werden. Es handelt sich hierbei um native Apps und keine Hybrid-, HTML5- oder Web-Apps. Wann immer ein nativer Code, wie beispielsweise Java, Objective-C oder Swift, benötigt wird, kann dieser integriert werden.

2.3. NodeJS

NodeJS ist eine serverseitige Plattform. Mittels JavaScript und NodeJS lassen sich Webserver abbilden. Dieses Framework ist plattformunabhängig und bietet eine sehr ressourcensparende Architektur.

2.4. Xcode & Android Studio

Xcode ist eine Entwicklungsumgebung in der native Apps des Betriebssystems iOS entwickeln werden können. Xcode verwendet aktuell die Programmiersprache Swift

3. Android Studio ist eine Entwicklungsumgebung in der native Apps für das Betriebssystem Android entwickeln werden. Hier wird mit der Programmiersprache Java gearbeitet.

2.5. Orange Pi

Der Orange Pi ist das Pendant zum bekannteren Raspberry Pi. Allerdings ist dieser günstiger und hat geringfügig mehr Leistung als ein Raspberry Pi. Jedoch fehlt hier die große Community im Gegensatz zum Raspberry Pi und das große Angebot an Software.

2.6. Arduino/Genuino

Der Begriff "Genuino" ist gleichzusetzen mit "Arduino". Arduino ist eine aus Soft- und Hardware bestehende Physical-Computing-Plattform. Beide Komponenten sind im Sinne von Open Source quelloffen.

2.7. SSH

SSH auch Secure Shell genannt, ist ein Unix basiertes Kommandozeilen-Interface und Protokoll mit dem der sichere Zugriff auf einen entfernten Computer möglich ist.

3. Anforderungsanalyse und Priorisierung

Der *CoffeeBot* soll ein System, das aus einer Kaffeemaschine, einem Roboter und einer App/Webseite besteht, sein. Die Kaffeemaschine, die automatisch zur konfigurierten Uhrzeit frischen Kaffee kocht, ist über die App einstellbar. Nachdem die Kaffeemaschine den Vorgang beendet hat, schickt sie den Roboter mit der frischen Tasse Kaffee zum Bett.

3.1. Kaffeemaschine

3.1.1. Musskriterien

Die Kaffeemaschine muss den Kaffee innerhalb von 4 Minuten fertig kochen, so dass der Roboter den Kaffee zu der in der App eingestellten Zeit zum Bett bringen kann, ohne dass der Kaffee kalt wird.

Damit kein Wasser ausläuft oder die Kaffeemaschine gar umfällt, muss diese einen festen Stand besitzen.

Außerdem muss die Elektronik vor Wasser geschützt sein, so dass Kurzschlüsse oder ähnliche Gefahren beim Kochvorgang ausgeschlossen werden können.

3.1.2. Wunschkriterien

Der Kaffee sollte den Ansprüchen des Nutzers in Sachen Geschmack entsprechen.

Ein weiteres Wunschkriterium ist das Einhalten der DIN-Norm 40050, so dass die Kaffeemaschine Spritzwassergeschützt ist.

Dem Benutzer soll mit Tee eine weitere Möglichkeit anstatt des Kaffees geboten werden.

3.1.3. Abgrenzungskriterien

Die Kaffeemaschine soll im Rahmen dieses Projekts nicht die Möglichkeit bereitstellen dem Kaffee Milch oder Zucker dazuzugeben.

3.2. App

3.2.1. Musskriterien

Es muss in der App die Möglichkeit bestehen, dass man eine Uhrzeit einstellen kann. Zu dieser Uhrzeit muss der Kaffee an das Bett geliefert worden sein.

Damit viele Nutzer die App verwenden können, muss die App für die iOS und Android Plattform zur Verfügung gestellt werden.

3.2.2. Wunschkriterien

Es ist wünschenswert, dass der Roboter über die App konfiguriert werden kann. Das bedeutet dem Roboter kann eine bestimmte Route beigebracht werden. Diese Funktion sollte in einer eigenen View in der App umgesetzt werden.

Des Weiteren soll die Größe der Kaffeetasse über die App eingestellt werden können, damit die Kaffeemaschine weiß wie viel Kaffee benötigt wird.

Ein weiteres Wunschkriterium ist, dass der Nutzer verschiedene Weckzeiten angeben und verwalten kann.

Damit der Benutzer die Kommunikation zwischen App und Roboter, wie auch Kaffeemaschine, nicht manuell einstellen muss, soll die App automatisch nach den Kommunikationspartner suchen.

3.2.3. Abgrenzungskriterien

Die App wird im Rahmen der Projektarbeit nicht für die Windows Phone Geräte umgesetzt. Sie soll auch nicht im Google PlayStore und im AppStore erhältlich sein.

3.3. Webseite

3.3.1. Wunschkriterien

Die Webseite selbst ist ein Wunschkriterium. Sie soll über zwei Funktionen verfügen. Zum einen kann über die Webseite die Weckzeit gesetzt werden. Zum anderen soll der Benutzer die Größe der Tasse mit angeben sollen.

Die Webseite sollte ein ansprechendes, modernes Design besitzen und leicht bedienbar sein.

3.3.2. Abgrenzungskriterien

Die Webseite soll die Konfiguration und Steuerung des Roboters nicht zur Verfügung stellen.

3.4. Roboter

3.4.1. Musskriterien

Roboter befördert den Kaffee geradlinig von Kaffeemaschine zu einem vom Nutzer festgelegten Punkt. Der Roboter muss die vom Nutzer festgelegte Zeitschranken einhalten.

3.4.2. Wunschkriterien

Der Roboter hat die Fähigkeit eine vom Nutzer bestimmte Route abzufahren, um Hindernissen ausweichen zu können und von einem Raum zum anderen gelangen zu können.

Der Roboter besitzt die Fähigkeit automatisiert Hindernissen auszuweichen ohne die Fähigkeit zu verlieren das Ziel zu erreichen.

Der Roboter besitzt Lautsprecher und eine Sprachunterstützung um Warnungen oder Hinweise dem Nutzer zu übermitteln.

Die Sprachausgabe kann wahlweise auch in klingonisch oder sächsisch erfolgen.

3.4.3. Abgrenzungskriterien

Der Roboter sollte auf Bewaffnung verzichten können um zum Ziel zu gelangen.

4. Lösungsvorschläge

4.1. Kaffeemaschine

Material-Grundkonstruktion:

1. Lego-Bausteine
2. Holzkonstruktion
3. Plastik-Aufbau

Komponenten der Kaffeemaschine:

1. Komponenten aus einer anderen Maschine ausbauen
2. Komplett neue Komponenten kaufen

Steuerung der Kaffeemaschine:

1. RaspberryPi
2. Orange Pi
3. Arduino/Genuino

4.2. App

Umsetzungsmethoden:

1. Native App mittels Xcode & Android Studio
2. Native App mithilfe von React Native
3. Web-App durch HTML5

4.3. Webseite

Server-Hardware:

1. RaspberryPi

2. Orange Pi
3. Genuino/Arduino

Server-Software:

1. NodeJS
2. Apache2
3. Microsoft(IIS)

Umsetzung der Webseite:

1. HTML und CSS
2. WebApp mittels React
3. Content-Management-System

4.4. Roboter

Karosserie und Antrieb:

1. Lego-Technic Bausteine und Lego Powerfunction Motoren
+ Fantasie, Kreativität und jede Menge Spass Spass
2. Robot Car Chassis-Kit für Arduino

Steuerteil:

1. Raspberry Pi
2. Lego Mindstorm Nxt
3. Arduino Uno

Stromversorgung:

1. Lego Technic Netzteil 9V mit 6x AA Batterien
2. USB Powerbank
3. Plutoniumantrieb

Ansteuerung der Motoren / Controller:

1. Lego Motoren: Infrarot Transceiver
2. Lego Motoren oder anderer DC Motor: L293D Chip + Kabel

4.5. Kommunikation der Komponenten

Kommunikation zwischen App/Webseite und Kaffeemaschine:

1. SSH - Verbindung
2. Webserver auf dem ein CGI ausgeführt wird
3. Bluetooth

Kommunikation zwischen App und Roboter:

1. SSH - Verbindung zum Roboter
2. Webserver auf Roboter
3. Bluetooth

Kommunikation zwischen Kaffeemaschine und Roboter:

1. SSH - Verbindung zwischen Kaffeemaschine und Roboter
2. Webserver auf Roboter
3. Bluetooth - Verbindung zwischen Kaffeemaschine und Roboter

5. Auswahl Lösung anhand der Anforderungen

5.1. Kaffeemaschine

Material-Grundkonstruktion:

Wahl:

Die Grundkonstruktion wird aus (2) Holz gebaut.

Begründung:

Holz ist hitzeresistenter als Plastik. Außerdem ist es für den Prototypenbau gut geeignet, da Holz zum einen billig und einfach zu verarbeiten ist. Auch durch das Gewicht einer Holzkonstruktion wird die Standfestigkeit der Maschine gewährleistet.

Komponenten der Kaffeemaschine:

Wahl:

Für die Komponenten wird eine Mischung aus vorhandenen und neuen Teilen verwendet. Alle Teile die hygienisch unbedenklich sind, wie zum Beispiel das Heizelement können in der selbstgebauten Kaffeemaschine verwendet werden. Teile wie die Halterung für den Kaffeefilter müssen neu gekauft werden.

Begründung:

Die Verwendung einer fertigen Kaffeemaschine kommt nicht in Frage, da diese keine Möglichkeit der Anpassung bietet. Die Elektronik ist bei einer fertigen Maschine komplett versteckt. Um aber den Raspberry Pi und das Relaisboard anzuschliessen braucht man Zugriff auf diese Elektronik und Kabel. Da auf die Hygiene geachtet werden muss, müssen gezwungenermaßen auch neue Teile wie die Halterung des Kaffeefilters verbaut werden. Deshalb werden gebrauchte wie auch neuen Teile für die Umsetzung der Kaffeemaschine gewählt.

Steuerung der Kaffeemaschine:**Wahl:**

Die Kaffeemaschine wird mit einem (1) Raspberry Pi verbunden.

Begründung:

Da der Raspberry Pi schon vorhanden ist. Die anderen Vorschläge könnte man hier genauso verwenden, da nur eine geringe Kapazität an Rechenleistung des Geräts benutzt wird und somit alle Auswahlmöglichkeiten in Frage kommen würden.

5.2. App

Umsetzungsmethoden:**Wahl:**

Die Umsetzung der App wird mit (2) React Native durchgeführt.

Begründung:

Da die App für iOS und Android Plattformen zur Verfügung gestellt werden muss wird React Native als Unterstützung verwendet. Durch React Native kann im Gegensatz zur nativen Entwicklung mit Xcode und Android Studio mithilfe von JavaScript durch den gleichen Code eine native App für iOS wie auch Android implementiert werden. Für eine Web-Application mit HTML5 wurde in diesem Projekt aus Performancegründen verzichtet.

5.3. Webseite

Server-Hardware:**Wahl:**

(1) Raspberry Pi

Begründung:

Da der Webserver auch auf dem Raspberry Pi der Kaffeemaschine läuft. Somit ist kein weiteres Gerät notwendig. Der Zugriff und die Konfiguration der Kaffeemaschine ist direkt am Gerät gewährleistet.

Server-Software:**Wahl:**

(2) Apache2

Begründung:

Die Konfiguration eines Apache2-Webserver wurde schon in der Vorlesung Systemadministration besprochen und deshalb fällt hier die Wahl anhand von Wissen auf Apache2. Außerdem ist der Apache2-Webserver der meistgenutzte Webserver weltweit.

Umsetzung der Webseite:**Wahl:**

WebApp wird mittels (2) React umgesetzt.

Begründung:

Durch die Einarbeitung in React im Wahlfach Mobile Anwendungen 3, wird die Webseite mit React umgesetzt. Des Weiteren finden sich viele Parallelen zwischen React und React Native. Die fertig implementierte Webseite lässt sich zudem einfach auf einem Apache2-Webserver installieren.

5.4. Roboter

Karosserie und Antrieb:**Wahl:**

Für die Karosserie und Antrieb werden die (1) Lego-Technic Bausteine und Lego Powerfunction Motoren sowie jede Menge Fantasie und Kreativität verwendet.

Begründung:

Da das notwendige Lego, Kreativität und jede Menge Spieltrieb gegeben ist, wird kein fertiges Robot Car Chassis-Kit für Arduino verwendet, obwohl bei einem Neukauf zweiteres kostengünstiger wäre und weniger Zeitaufwand beim Aufbau benötigen würde. Dafür bietet die gewählte Lösung auch eine gewisse Flexibilität um die speziellen Anforderungen an das Gehäuse des Roboters zu erfüllen, um z.B. die Halterung für eine Kaffeetasse zu konstruieren. Besonders aber für die Umsetzung der Kann-Ziele bietet Lego viele Freiheiten um schnell neue Komponenten wie Sensoren zu befestigen.

Steuerteil:**Wahl:**

(1) Raspberry Pi mit OS Raspbian

Begründung:

Die Wahl für das Steuerteil für den Roboter fällt zugunsten des Raspberry Pi's aus. Zum einen ist der Bezug zur Vorlesung viel mehr gegeben und zum anderen bietet der Raspberry in Kombination mit dem Betriebssystem Raspbian viel mehr Flexibilität beim Programmieren und eine höhere Konnektivität zu anderen Komponenten des Gesamtsystems. Raspbian bietet alle Vorzüge eines Linux Betriebssystems.

Durch die zahlreichen GPIO-Pins sind genügend Ein- und Ausgänge für zusätzliche Sensoren oder Aktuatoren vorhanden. Man ist nicht eingeschränkt durch Spezifikation eines Herstellers sondern kann die Komponenten des Systems frei wählen. Auch die Anzahl der Komponenten ist nicht beschränkt.

Stromversorgung:

Wahl:

- (1) Lego Technic Netzteil 9V mit 6x AA Batterien zur Versorgung der Motoren
- +
- (2) USB Powerbank zur Versorgung des Raspberry

Begründung:

Die Wahl der Stromversorgung für die Motoren fällt auf ein Lego Technic 9V Netzteil mit einer Spannung von 9V und für den Raspberry Pi wird eine USB Powerbank verwendet. Durch die Nutzung von 2 Netzteilen spart man sich einen Spannungswandler und eine zusätzliche Schaltung. Auch um die Kabelverbindung braucht man sich so keine zusätzlichen Gedanken machen da das Lego-Netzteil die nötigen Steckverbindungen für die Motoren bietet und die Powerbank den Raspberry einfach über einen USB-Port mit Strom versorgt. Auch ist die Laufzeit entsprechend länger. Ein Nachteil besteht am höheren Gewicht und am erhöhten Platzbedarf.

Als alternative Stromversorgung ist noch der Plutoniumantrieb zu erwähnen, zum Beispiel erfolgreich verwendet bei der Cassini-Huygens-Mission. Doch stellt die Beschaffung des Plutoniums

und die hohen Kosten ein gewisses Problem dar. Auch besteht bei einer USB-Powerbank eine höhere gesellschaftliche Akzeptanz als bei einem Plutoniumantrieb. Das eigentlich Ausschlusskriterium besteht am Ende aber nur an dem zu hohen Gewicht.

Ansteuerung der Motoren / Controller:

Wahl:

(1) Infrarot Transceiver

Begründung:

Die Entscheidung des Infrarot Transceivers wird damit gerechtfertigt, da dieser recht kostengünstig und einfach zu implementieren ist. Auch ein großer Vorteil liegt darin, dass keine zusätzliche Schaltung benötigt wird. Jedoch ist die Reaktionszeit der Motoren etwas langsamer, was sich aber nicht merklich negativ auf das Fahrverhalten des Roboters auswirkt.

5.5. Kommunikation der Komponenten

Kommunikation zwischen App/Webseite und Kaffeemaschine:

Wahl:

(2) Webserver auf dem ein CGI ausgeführt wird.

Begründung:

Da für die Umsetzung der Webseite ein Webserver nötig ist, wird auch dieser für die Kommunikation zwischen Kaffeemaschine und App genutzt. Die Webseite, wie auch die App, sollen die Weckzeit setzen können. Um diese Funktion zu gewährleisten wird ein CGI - Programm in den Webserver eingebunden, das die Konfiguration der Kaffeemaschine übernimmt.

Kommunikation zwischen App und Roboter:

Wahl:

(1) SSH - Verbindung zum Roboter

Begründung:

Die Kommunikation zwischen App und Roboter wird über SSH realisiert. Ein Grund für die Entscheidung ist, dass über SSH Scripte, die auf dem entfernten Rechner liegen, ausgeführt werden können. Die Scripte sind für die Steuerung und Programmierung des Roboters gedacht. Ein anderer Grund ist die Sicherheit, die durch SSH gewährleistet wird. Durch die Verwendung von SSH wird auch kein zusätzlicher Webserver auf dem Roboter benötigt.

Kommunikation zwischen Kaffeemaschine und Roboter:

Wahl:

(1) SSH - Verbindung zwischen Kaffeemaschine und Roboter

Begründung:

Auf der einen Seite ist SSH extrem sicher und andererseits benötigt man keine weiteren Schnittstellen, wie beispielsweise ein Webserver, für die Kommunikation zwischen der Kaffeemaschine und dem Roboter. Zum Starten des Roboters muss ein Script ausgeführt werden, was mit SSH einfach umgesetzt werden kann.