

Smart Mirror

Baris,Tikir, Leon Dodrimong

19. Februar 2020

Inhaltsverzeichnis

1	Grundidee	3
1.1	Konzept Benutzerfreundlichkeit	3
1.2	Spiegel mit integriertem Display	3
2	Aufbau	3
2.1	Strukturierung	3
2.2	Materialien	3
3	Software	3
3.1	Prereuisites	3
3.2	Frontend - SmartMirrorWeb	4
3.3	Backend - SmartMirror.WebApi	4
4	Erweiterungen	5
4.1	API's	5
4.2	Hardware	5
5	Über uns	5
6	Anhang	5
6.1	Verweise	5

1 Grundidee

1.1 Konzept Benutzerfreundlichkeit

1.2 Spiegel mit integriertem Display

2 Aufbau

2.1 Strukturierung

Notizen: Diagramme und erklärungen der Funktionsweise

2.2 Materialien

Der Spiegel besteht aus einem Einwegspiegelglass, welches von einer Seite spiegelt und von der anderen Seite reflektiert. Außerdem haben wir einen Holzrahmen, welcher aus Fassung für das Spiegelglass dient, genutzt. Für die Technik haben wir ein Display für die Anzeigen , ein RasperryPi für die Steuerung und die nötigen Verbindungskabel, wie Spannungsversorgung und Videokabel (HDMI) zur Übertragung der Videosignals zum Display, eingesetzt.

Für die optimalen Erweiterungen würde man je nachdem welches Feature gewünscht ist, noch eine Picamera (für Facerecognition), RasperryPi Bewegungssensor (Bewegungserkennung)¹, Gesture Sensor (Gestiksteuerung)² oder ein kleines Mikrophone zur Sprachsteuerung ³

Notizen: Später eintragen welches Raspi Modell, Display, Glass verwendet wurde

3 Software

3.1 Prerequisites

- Node.js (Javascript runtime)
- Angular
- IDE (Visual Studio Code)
- Datenbank ()

¹ *Rasperry Pi Infrarot Bewegungsmelder*: <https://www.reichelt.de/rasperry-pi-infrarot-bewegungsmelder-hc-sr501-rpi-hc-sr501-p224216.html?&nbc=1>

² *3D Gesture Tracking Shield for Rasperry Pi*: <http://wiki.seeedstudio.com/3D-Gesture-Tracking-Shield-for-Rasperry-Pi-MGC3130/>

³ *Ansteckmikrofon über Klinke*: https://www.amazon.de/dp/B073GJQKL1/ref=psdc_1384055031_t1_B07WQFNVVQ

3.2 Frontend - SmartMirrorWeb

Wir haben unser Frontend mithilfe von Angular 9. Für uns war dies eine Neue Erfahrung, da wir noch nie zuvor damit gearbeitet haben. Das Frontend soll sich lediglich um die Anzeige der Daten kümmern und folgende Funktionen übernehmen:

- Daten vom Backend (SmartMirror.WebApi) anfragen
 - alle möglichen Dienste (Widgets)
 - vom User angemeldeten Dienste
- Daten senden
 - Dienst aktivieren/ deaktivieren
 - Benutzereingaben zur Erstellung eines neuen Benutzers
- User zur Anmeldung von weiteren externen Diensten, zur jeweiligen Website weiterleiten⁴

3.3 Backend - SmartMirror.WebApi

Das Backend besteht aus einer eigens Entwickelten Web API, welche zum einen die Anfragen und Daten vom Frontend (SmartMirror.Web) entgegennimmt und bearbeitet, und zum anderen den Datenaustausch mit der eigenen Datenbank und Kommunikation mit den externen API kommuniziert. Das Backend besteht zum einen Client-Server. Zum einen stellt er als Server eine eigene API dar, welche vom Frontend genutzt wird. Zum anderen fungiert dieser auch als Client und nutzt die externen API Schnittstellen von Drittanbietern. Besitzt dieser eine extra Komponente, welche auf die interne Datenbank zugreift

Client Funktion Da der Smart Mirror möglichst viele externe Features verbinden soll, muss die Schnittstelle zu diesen gut strukturiert werden. Daher haben wir uns dazu entschieden, dass jede externe Kommunikation ihre eigene Komponente bekommt, welche bestimmte Standards (Interfaces) bedient. Somit können auch später leicht neue externe API's eingebunden werden, indem die Kommunikation in einer neuen Komponente implementiert wird. Dort müssen dann nur die vorgegebenen Interfaces definiert sein und dann anschließend bekannt gemacht werden, indem diese in eine interne Liste von allen möglichen externen Diensten eingetragen wird.

⁴zum Beispiel: wenn der User den neuen Dienst Google Kalender für sich registrieren möchte, muss er sich auf der Website von Google Kalender anmelden um sich zu zertifizieren. Diese sendet dann die zur Authentifizierung notwendigen Credentials, welche vom Backend gespeichert werden müssen

Server Funktion

Kommunikation zur Datenbank

4 Erweiterungen

4.1 API's

- Google Kalender
- Todoist
- BVG
- Wetter

4.2 Hardware

- Face Recognition
- Voice Control
- Bewegungssensor

5 Über uns

Wir sind zwei Studenten, Baris Tikir und Leon Dodrimong, von der HTW - Hochschule für Technik und Wirtschaft Berlin aus dem Fachbereich der Ingenieurwissenschaften und Technik.

6 Anhang

6.1 Verweise