**Ideenskizze für das IoT-Projekt**

**1. Allgemeine Informationen**

* **Projektname:** Cocktailautomat
* **Datum:** 18.03.2025 bis 06.05.2025

**2. Projektbeschreibung**

**Kurzbeschreibung:**  
Es soll ein Automat gebaut werden, der selbstständig Getränke mischt. An einen Touchdisplay kann man auswählen, was man haben möchte. Es soll auch ein Reinigungsprogramm geben, um die Schläuche zu reinigen.

**3. Anforderungen und Funktionalitäten**

✅ **Sensorik**

Der Füllstand der Flaschen soll mit einem VL53L0X Sensor überwacht werden. Außerdem soll überwacht werden, ob ein Glas unter dem Auslass steht. Dies geschieht mit einer Reflexionslichtschranke.

✅ **Aktoren-Steuerung**

Die Peristaltik Pumpen zum Pumpen der Getränke werden über Relais Module mit Optokopplern angesteuert. Ein LED-Ring zeigt an auf welcher Position das Glas stehen soll.

✅ **Webinterface & Benutzerinteraktion**

Die Bedienung soll über ein Touchdisplay direkt am Automaten laufen.

**4. Benötigte Komponenten**

**4.1 Hardware**

| **Komponente** | **Modell/Typ** | **Funktion** |
| --- | --- | --- |
| **Mikrocontroller** | ESP 32- S3 | Steuerung |
| **Sensor 1** | VL53L0X | Füllstands Überwachung Flaschen |
| **Sensor 2** | Datalogic950811290 | Überwachung Glas vorhanden |
| **Aktor 1** | Peristaltik Pumpen | Dosierung der Flüssigkeiten |
| **Aktor 2** | LED-Ring | Beleuchtung vom GLS |
| **Stromversorgung** | MW HDR-100-12N  MW HDR-60-5  RPI PS 27W BK EU | 12V Netzteil  5V Netzteil  Raspberry Pi Netzteil 5V |
| **Weitere Bauteile** | Raspberry Pi 5,  Touch Display für Raspberry Pi  PC817 Optokoppler  Relais Modul 3,3V | Für Node-RED und MQTT-Server  12V Schaltsignal auf 3,3V Spannung bringen  Ansteuerung der Pumpen |

**4.2 Software & Datenbank**

| **Komponente** | **Technologie** | **Funktion** |
| --- | --- | --- |
| **Microcontroller-Code** | Micro Python | Steuerung der Dosierung und des LED-Rings sowie auslesen der Sensoren |
| **Webinterface** | Node-RED | Bedienung des Automaten, Auswahl der Dosiermengen |
| **Datenbank** | Maria DB |  |

**5. Systemarchitektur**

**Ein Bild, das Text, Elektronik, Screenshot enthält.

KI-generierte Inhalte können fehlerhaft sein.**

**6. Zeitplanung (Meilensteine)**

| **Datum** | **Aufgabe** |
| --- | --- |
| KW 12 - 13 | Materialbeschaffung, ggf. Hardwareaufbau |
| KW 14 | Hardwareaufbau, ggf. ESP 32 und Raspberry Pi einrichten |
| KW 15 - 16 | ESP 32 und Raspberry Pi einrichten  Programmierung |
| KW 17 | Restarbeiten |
| KW 18 | Finale Testphase |
| KW 19 | Abgabe |

**7. Offene Fragen & Herausforderungen**

Die Herausforderungen sind, dass einige Oberflächen leicht zu reinigen sein müssen. Außerdem wird mir Flüssigkeiten gearbeitet. Davor muss die Elektronik geschützt werden. Die Flüssigkeiten müssen genau dosiert werden. Außerdem muss Node-RED auf dem Raspberry Pi eingerichtet werden und mit dem Touchdisplay bedienbar sein.

**8. Fazit & Zielsetzung**

Das Projekt war zeitlich knapp bemessen. Dadurch, dass ich 2 Wochen krank war und eine Woche voll arbeiten musste, konnten einige Zusatz Features nicht umgesetzt werden. Diese können als mögliche Erweiterung umgesetzt werden. Es wird nur der Füllstand von zwei Flaschen gemessen. Dieses kann man auf alle acht Flaschen erweitern. Außerdem kann man noch viel mehr LED-Effekte Programmieren. Aktuell kann man in der Touch Eingabe nur die Flüssigkeiten auswählen. Auch dort kann man vorgefertigte Rezepte verwenden.

Eine größere Herausforderung war die Installation vom Raspberry PI. Auf diesem läuft Node-RED, der MQTT-Broker, sowie Maria DB. Dies hat nach der Einarbeitung erfolgreich funktioniert.

Allgemein war die Integration der MQTT-Kommunikation am herausforderndsten.

**9. Installationsanleitung**

Der Cocktailautomat muss an einem Ort mit WLAN aufgestellt werden. Danach müssen der Raspberry Pi und der ESP 32 mit dem WLAN verbunden werden. Der Raspberry Pi muss ein fest IP-Adresse vom Router erhalten. Diese feste IP-Adresse muss im ESP 32 eingetragen werden, um die Kommunikation mit dem MQTT-Broker sicherzustellen.

**9. Bedienungsanleitung**

Der Cocktailautomat muss mit dem Strom verbunden werden. Danach muss der FI/LS Schalter auf der Rückseite eingeschaltet werden. Nun fährt der Cocktailautomat hoch. Nachdem alles gestartet ist, muss auf dem Raspberry Pi das Node RED Dashboard gestartet werden.

Nun werden die Flaschen mit den Zutaten in den Automaten gestellt und die Schläuche kommen in die Flaschen und der Füllstands Sensor kommt auf den Flaschenkopf.

Als nächstes müssen die Schläuche des Cocktailautomaten befüllt werden. Dafür wir ein Glas unter den Auslass des Cocktailautomaten gestellt. Danach werden im Handbetrieb alle Zutaten nacheinander angesteuert, bis die Zutat aus dem Auslass kommt.

Ist dieser Schritt abgeschlossen kann auf die Zubereitung gewechselt werden. Hier können die Mengen der Zutaten ausgewählt werden. Spätestens jetzt sollte man das Glas unter den Auslass stellen. Mit Betätigung des Startbutton beginnt der Automat nacheinander die Zutaten in das Glas zu füllen. Nach diesem Vorgang kann das Glas entnommen werden.

Dies kann nun so lange wiederholt werden, bis die Zutaten leer sind. Diese können dann gewechselt werden und dann geht es weiter.

Soll der Automat werden außer Betrieb genommen werden, müssen die Schläuche gereinigt und entleert werden. Dafür werden die Schläuche aus den Flaschen mit den Zutaten genommen. Diese kommen jetzt in Behälter mit lauwarmem Wasser. Nun werden alle Schläuche, die benutzt wurden, durchgespült. Dafür muss wieder auf den Handbetrieb gewechselt werden. Pro Zutat soll ein halbes Glas Wasser ca. 15cl durchlaufen. Dafür ein Glas unter den Auslass stellen und die Zutaten nacheinander auf der Bedienüberfläche anwählen. Ist das Wasser einer Zutat durchgelaufen, wird der Zutatenschlauch aus dem Wasser gezogen und die Pumpe saugt den Schlauch leer. Kommt keine Flüssigkeit mehr aus dem Auslass vom Gerät kann die Zutat abgeschaltet werden und die Reinigung vom nächsten kann beginnen. Sind alle Schläuche gereinigt, können die Oberflächen unter dem Auslass sowie die Stellplätze der Flaschen mit einem feuchten Lappen gereinigt werden.