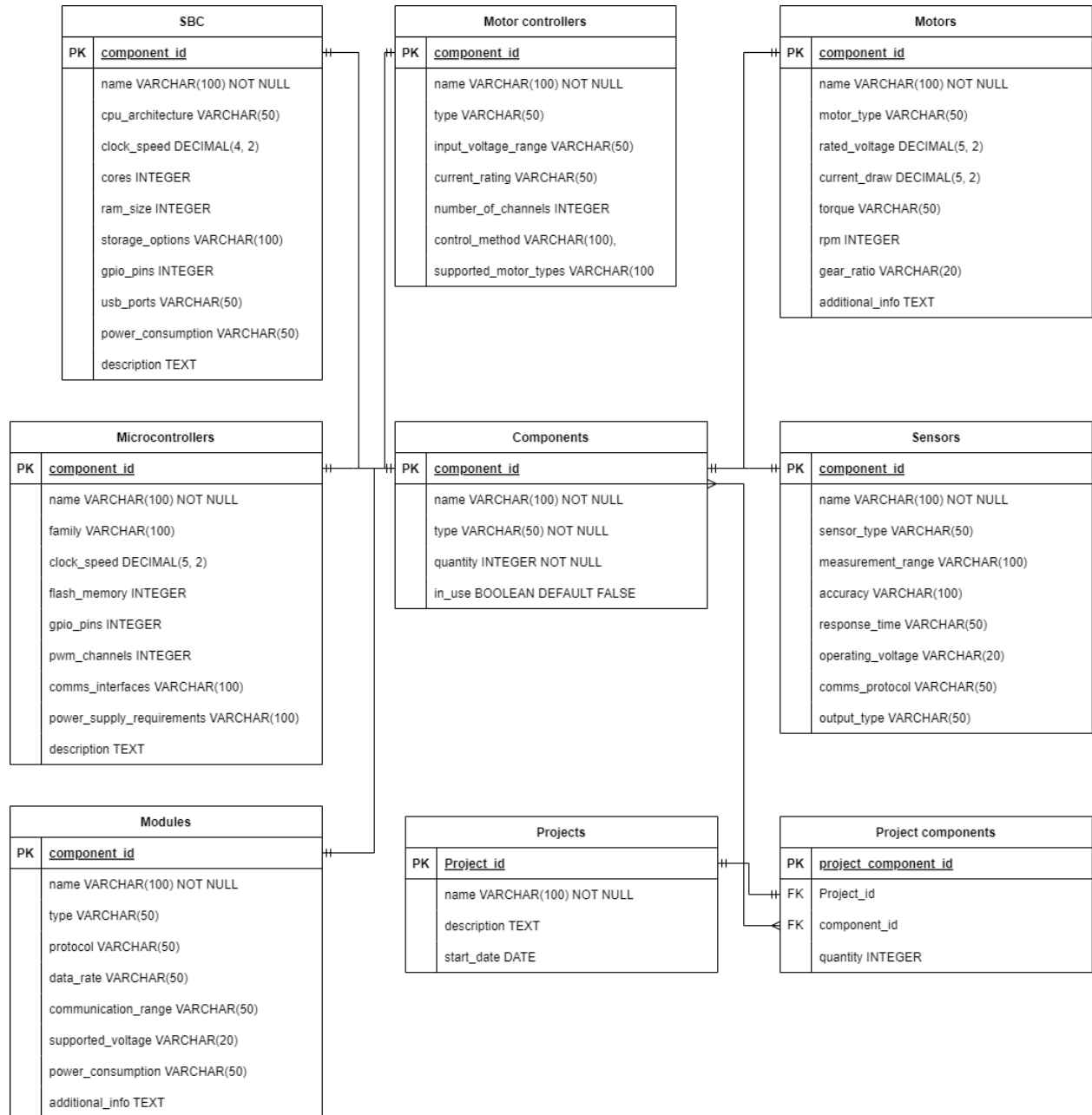


Andmebaasi dokumentatsioon

ERD



Eesmärk

Projekti eesmärgiks oli kataloogida elektroonika komponente isiklikuks kasutamiseks, et oleks hea ülevaade mis on olemas ja mis on nende nõuded ja võimalused.

Andmebaasi struktuur

Andmebaas koosneb järgnevatest tabelitest:

- Components : peatabel mis sisaldab komponentide põhiandmeid mis rakenduvad kõikidele tüüpidele.
- Sbc
- Microcontrollers
- Motor controllers
- Motors
- Sensors
- Modules

Igas tabelis on konkreetse komponendi tüübi andmed

- Projects : sisaldab infot projekti kohta
- Project components : suhtetabel mis seob projekti komponendiga

Andmebaasi andmed

Selleks, et andmebaasi andmetega täita kasutasin Python script'e, sest ei leidnud valmisolevat mis minu ideega kattuks ning käsitsi täitmine võtaks liiga kaua aega. Kuna tegemist on isiklikuks kasutamiseks mõeldud andmebaasiga, siis tulevikus realsed andmed on lisatud veebirakenduse kaudu käsitsi.

Andmebaasi funktsionaalsused

Hetkel on andmebaasis võimalik lisada uusi komponente, otsida komponente nime või tüübi järgi ning näha komponentide informatsiooni. Tulevikus on võimalik lisada projekte ning otsida parameetrite järgi.

Triggerid ja funktsioonid

Selleks et veebiliidese kaudu lisatud komponendi oleks nähtavad oli vaja luua trigger, et uuendada components tabelit.

```
CREATE OR REPLACE FUNCTION update_components_table()
RETURNS TRIGGER AS $$
DECLARE
    component_type TEXT;
BEGIN
    component_type := CASE TG_TABLE_NAME
        WHEN 'sbc' THEN 'SBC'
        WHEN 'microcontrollers' THEN 'Microcontroller'
        WHEN 'motor_controllers' THEN 'Motor Controller'
        WHEN 'motors' THEN 'Motor'
        WHEN 'sensors' THEN 'Sensor'
        WHEN 'modules' THEN 'Module'
    END;

    INSERT INTO components (component_id, name, type, quantity)
    VALUES (NEW.component_id, NEW.name, component_type, 1)
    ON CONFLICT (name, type)
    DO UPDATE SET quantity = components.quantity + 1;

    RETURN NEW;
END;
$$ LANGUAGE plpgsql;

CREATE TRIGGER sbc_insert_trigger
AFTER INSERT ON sbc
FOR EACH ROW EXECUTE FUNCTION update_components_table();

CREATE TRIGGER microcontroller_insert_trigger
AFTER INSERT ON microcontrollers
FOR EACH ROW EXECUTE FUNCTION update_components_table();

CREATE TRIGGER motor_controller_insert_trigger
AFTER INSERT ON motor_controllers
FOR EACH ROW EXECUTE FUNCTION update_components_table();

CREATE TRIGGER motor_insert_trigger
AFTER INSERT ON motors
FOR EACH ROW EXECUTE FUNCTION update_components_table();

CREATE TRIGGER sensor_insert_trigger
AFTER INSERT ON sensors
FOR EACH ROW EXECUTE FUNCTION update_components_table();

CREATE TRIGGER module_insert_trigger
AFTER INSERT ON modules
FOR EACH ROW EXECUTE FUNCTION update_components_table();
```

Tabelite loomine

```
CREATE SEQUENCE global_component_id_seq;
```

```
CREATE TABLE components (  
    component_id INTEGER PRIMARY KEY DEFAULT nextval('global_component_id_seq'),  
    name VARCHAR(100) NOT NULL,  
    type VARCHAR(50) NOT NULL,  
    quantity INTEGER NOT NULL,  
    in_use BOOLEAN DEFAULT FALSE,  
    CONSTRAINT unique_name_type UNIQUE (name, type)  
);
```

Selleks et tabelitesse lisatud komponendid ei jagaks ühte ID'd sai loodud Global ID mis tagab et iga uus komponent saab uue ID, mistahes tabelis.

```
CREATE TABLE sbc (  
    component_id INTEGER DEFAULT nextval('global_component_id_seq') PRIMARY KEY,  
    name VARCHAR(100) NOT NULL,  
    cpu_architecture VARCHAR(50),  
    clock_speed DECIMAL(4, 2),  
    cores INTEGER,  
    ram_size INTEGER,  
    storage_options VARCHAR(100),  
    gpio_pins INTEGER,  
    usb_ports VARCHAR(50),  
    power_consumption VARCHAR(50),  
    description TEXT  
);
```

```
CREATE TABLE projects (  
    project_id SERIAL PRIMARY KEY,  
    name VARCHAR(100) NOT NULL,  
    description TEXT,  
    start_date DATE DEFAULT CURRENT_DATE  
);
```

```
CREATE TABLE project_components (  
    project_component_id SERIAL PRIMARY KEY,  
    project_id INTEGER REFERENCES projects(project_id) ON DELETE CASCADE,  
    component_id INTEGER REFERENCES components(component_id) ON DELETE CASCADE,  
    quantity INTEGER DEFAULT 1  
);
```

Veebirakendus

Veebirakendus on tehtud Flaski peal.

Component List

ASUS Tinker Board S (SBC)

Quantity: 72

Raspberry Pi 4 Model B 8GB (SBC)

Quantity: 71

BeagleBone Black (SBC)

Quantity: 72

Raspberry Pi 4 Model B 2GB (SBC)

Quantity: 72

Raspberry Pi 4 Model B 4GB (SBC)

ASUS Tinker Board S (SBC)

Quantity: 72

Details:

- **Component ID:** 6698
- **Name:** ASUS Tinker Board S
- **CPU Architecture:** ARM Cortex-A17
- **Clock Speed:** 1.80 GHz
- **Cores:** 4
- **RAM:** 2 GB
- **Storage Options:** 16GB eMMC, microSD
- **GPIO Pins:** 40
- **USB Ports:** 4x USB 2.0
- **Power Consumption:** 5-10W
- **Description:** ASUS Tinker Board S offers enhanced performance for media playback and GPIO-intensive applications with its Rockchip processor and HDMI output.

Veebirakenduses on näha komponentide nimekiri, sulgudes on kirjas mis tüüpi komponendiga tegu, palju seda komponenti on ning peale vajutades avaneb detailne info komponendi kohta. Sealhulgas on igal komponendi tüübil eri detaili väljad.

Component List

The screenshot shows a web application interface. On the left, there is a sidebar with a search bar and a list of components. The main area displays a table of components. A modal window titled 'Add New Component' is open, showing a form to add a new component. The form includes a dropdown for 'Component Type' (set to 'SBC'), and input fields for 'Name', 'CPU Architecture', 'Clock Speed (GHz)', 'Cores', 'RAM Size (MB)', 'Storage Options', 'GPIO Pins', 'USB Ports', 'Power Consumption', and 'Description'. An 'Add Component' button is at the bottom of the modal.

Search by name or...

Add New Component

ASUS Tinker Board

Quantity: 72

Details:

- Component Name: ASUS Tinker Board
- CPU Architecture: Rockchip RK3328
- Clock Speed: 1.8GHz
- Cores: 4
- RAM: 2GB
- Storage: 16GB
- GPIO Pins: 40
- USB Ports: 2
- Power Consumption: 1.5W
- Description: A small, powerful, and affordable single-board computer (SBC) for hobbyists and developers. It features a Rockchip RK3328 processor, 2GB of RAM, and 16GB of storage. It is designed for media playback and GPIO-intensive applications with its Rockchip processor and HDMI output.

Add Component

Veel on võimalik lisada komponente otse veebirakenduse kaudu ning vastavalt komponendi tüübile avanevad täidetavad väljad.

Edasised arendused

Tulevikus on plaanis lisada detailide järgi otsing(nt. Komponent millel on RAM'i rohkem kui 2GB). Projektide lisamine lõpetada. Components tabel ühendused viia üle foreign key'dele. Pildid lisada.

Kokkuvõte

Projekti käigus omandasin praktilise kogemuse andmebaaside loomise ja haldamise alal. Samuti kasutades Flaski õppisin looma veebirakendust. Lisaks õppisin kasutama PostgreSQL'i ja PGAdmin 4 rakendust.