-- Step 1: Create the ENUM type for role

```
CREATE TYPE user_role AS ENUM ('admin', 'parent', 'doctor', 'special_educator');
```

-- Step 2: Create the users table : Stores all users (admin, parent, doctor, special educator)

```
CREATE TABLE users (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    username VARCHAR(255) NOT NULL,
    email VARCHAR(255) UNIQUE NOT NULL,
    password_hash VARCHAR(255) NOT NULL,
    role user_role NOT NULL,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    updated_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
);
```

Step 3:

gen_random_uuid() requires the pgcrypto extension. You can enable it with:

```
CREATE EXTENSION IF NOT EXISTS "pgcrypto";
```

Step 4:

```
SELECT table_name FROM information_schema.tables
WHERE table_schema = 'public' AND table_type = 'BASE TABLE';
```

Or

you can use this command inside the psql terminal:

```
\dt schema_name.*
```

Step 5: Stores information about children registered by parents

```
CREATE TABLE child_details (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
```

```
    parent_id UUID REFERENCES users(id),
    name VARCHAR(255) NOT NULL,
    date_of_birth DATE NOT NULL,
    gender VARCHAR(50),
    special_abilities TEXT,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    updated_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
);
```

gen_random_uuid() requires the pgcrypto extension. Enable it with:

```
CREATE EXTENSION IF NOT EXISTS "pgcrypto";
```

Step 6: Stores screening results for each child.

```
CREATE TABLE screening (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    child_id UUID REFERENCES child_details(id),
    screening_date TIMESTAMP WITH TIME ZONE NOT NULL,
    condition_identified UUID REFERENCES condition(id),
    screening_result TEXT,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    updated_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
);
```

Step 7: Stores screening questions for each condition

```
CREATE TABLE condition (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    name VARCHAR(255) NOT NULL,
    description TEXT,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
```

```
    updated_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
);
```

Step 8: Stores Screening questions for each condition

```
CREATE TABLE question (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    condition_id UUID REFERENCES condition(id),
    question_text TEXT NOT NULL,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    updated_time TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
);
```

Step 9: Stores Recommendations (medication, education,exercises) for each condition

Create ENUM type for recommendation type

```
CREATE TYPE recommendation_type AS ENUM ('medication', 'education', 'tutorial', 'exercise');
```

Create the recommendation table

```
CREATE TABLE recommendation (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    condition_id UUID REFERENCES condition(id),
    type recommendation_type NOT NULL,
    description TEXT,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    update_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
```

);

Step 10: Service condition table

Tracks condition between admin, parents, doctors, educators.

```
 Create ENUM type for status
CREATE TYPE service_status AS ENUM ('pending', 'active', 'completed');

CREATE TABLE service_condition (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    child_id UUID REFERENCES child_details(id),
    admin_id UUID REFERENCES users(id),
    doctor_id UUID REFERENCES users(id),
    educator_id UUID REFERENCES users(id),
    status service_status NOT NULL,
    notes TEXT,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    updated_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
);
```

Step 11: Stores medication details prescribed to a child

```
CREATE TABLE medication (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    child_id UUID REFERENCES child_details(id),
    doctor_id UUID REFERENCES users(id),
    medication_name VARCHAR(255) NOT NULL,
    dosage_exercises TEXT,
    instructions TEXT,
    start_date DATE NOT NULL,
```

```
    end_date DATE,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
);
```

Step 12: Stores Education plans or exercise for a child

Create ENUM type for plan_type

```
CREATE TYPE plan_type_enum AS ENUM ('education', 'exercise', 'tutorial');

CREATE TABLE education_exercise (
    id UUID PRIMARY KEY DEFAULT gen_random_uuid(),
    child_id UUID REFERENCES child_details(id),
    educator_id UUID REFERENCES users(id),
    plan_type plan_type_enum NOT NULL,
    description TEXT,
    start_date DATE NOT NULL,
    end_date DATE,
    created_date TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    updated_time TIMESTAMP WITH TIME ZONE DEFAULT CURRENT_TIMESTAMP,
    created_by UUID REFERENCES users(id),
    updated_by UUID REFERENCES users(id),
    is_active BOOLEAN DEFAULT TRUE
);
```