



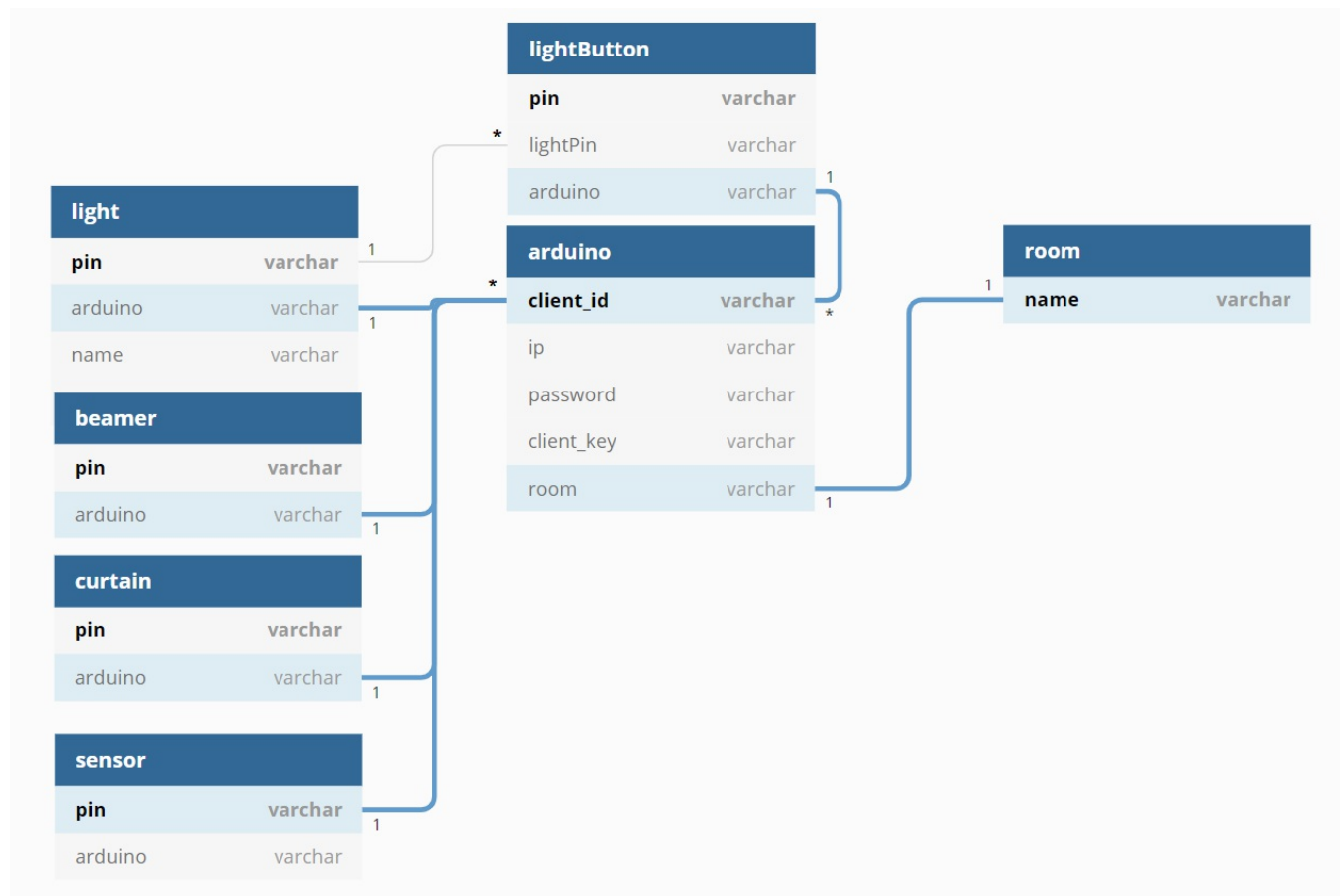
Guida Modulo

domotics database

Capitolo	Pagina
Indice	2
Funzionamento	3
Implementazione	4

domotics database

Funzionamento



Questo è il design del database, che è formato da sette tabelle. La tabella room rappresenta le aule di cui viene salvato il loro identificatore, per ogni aula viene associato un arduino, di esso si vuole sapere l'indirizzo ip, la password e la chiave per il client. Ogni arduino gestisce i moduli che devono essere controllati all'interno dell'aula che poi potranno essere aggiunti e implementati ma che al momento sono solamente gli interruttori delle luci, di cui vengono memorizzati i suoi pin e le luci a cui sono collegati. Delle luci si memorizzano i loro pin e stessa cosa per il beamer, le tende e i sensori. In questo modo grazie alla chiave che ogni arduino ha, il server riesce a riconoscerli e identificarli per poi ricavare lo stato di tutti i suoi moduli, quindi luci, tende, beamer e sensori andando a ricercare le varie informazione nel database. Grazie al database si possono tenere tutte le informazioni in modo ordinato, pulito e reperibile per quando si ha bisogno.

Implementazione

Per mettere sul il seguente data base bisogna seguire i seguenti passaggi.

Innanzitutto bisogna creare la tabella principale che rappresenta le aule che vengono identificate dal loro nome.

```
CREATE TABLE domotics.room (  
    name VARCHAR(255) PRIMARY KEY  
);
```

Poi si crea una tabella che rappresenta l'arduino che sarà posizionato all'interno dell'aula. L'arduino viene identificato da un id e si vuole sapere il suo indirizzo ip, la password della root, la chiave del client che servirà al server per riconoscerlo e l'aula in cui sono.

```
CREATE TABLE domotics.arduino (  
    client_id VARCHAR(255) PRIMARY KEY,  
    ip VARCHAR(255),  
    root_password VARCHAR(255),  
    client_key VARCHAR(255),  
    room VARCHAR(255),  
  
    FOREIGN KEY (room)  
        REFERENCES domotics.room (name)  
);
```

Poi si andranno a creare le tabelle che rappresentano i moduli all'interno delle aule che vengono controllati dal'arduino.

I valori da memorizzare sono molto simili per i moduli tranne qualche eccezione.

Delle luci si vogliono sapere i loro pin, l'arduino alla quale appartengono e quindi da cui vengono controllate e il loro stato quindi se sono accese o spente.

```
CREATE TABLE domotics.light (  
    pin VARCHAR(255),  
    arduino VARCHAR(255),  
    name VARCHAR(255),  
    status INT(1),  
    PRIMARY KEY (pin, arduino),  
    FOREIGN KEY (arduino)  
        REFERENCES domotics.arduino (client_id)  
);  
  
CREATE TABLE domotics.beamer (  
    pin VARCHAR(255),  
    arduino VARCHAR(255),  
    name VARCHAR(255),  
    PRIMARY KEY (pin, arduino),  
    FOREIGN KEY (arduino)  
        REFERENCES domotics.arduino (client_id)  
);
```

Il tipo di tenda permette di riconoscere se è la tenda di sinistra o di destra.

```
CREATE TABLE domotics.curtain (  
    pin VARCHAR(255),  
    arduino VARCHAR(255),  
    name VARCHAR(255),  
    PRIMARY KEY (pin, arduino),  
    FOREIGN KEY (arduino)  
        REFERENCES domotics.arduino (client_id)  
);
```

Dei sensori si vogliono sapere anche i pin del arduino a cui appartengono grazie al quale poi sarà facile ricavare tutte le informazioni necessarie. SensorType serve a evitare che ci siano sensori uguali con nomi diversi.

```
CREATE TABLE domotics.sensor (  
    pin VARCHAR(255),  
    arduino VARCHAR(255),  
    type VARCHAR(255),  
    name VARCHAR(255),  
    PRIMARY KEY (pin, arduino),  
    FOREIGN KEY (arduino)  
        REFERENCES domotics.arduino (client_id),  
    FOREIGN KEY (type)  
        REFERENCES domotics.sensorType (name)  
);  
  
CREATE TABLE domotics.sensorType (  
    name VARCHAR(255) PRIMARY KEY  
);
```

Le luci possono essere gestite anche trammite dei pulsanti di cui si salva il pin, il pin della luce che gestisce, l'arduino a cui è collegato e il nome.

```
CREATE TABLE domotics.lightButton (  
    pin VARCHAR(255),  
    lightPin VARCHAR(255),  
    arduino VARCHAR(255),  
    name VARCHAR(255),  
    PRIMARY KEY (pin, arduino),  
    FOREIGN KEY (arduino)  
        REFERENCES domotics.arduino (client_id),  
    FOREIGN KEY (lightPin)  
        REFERENCES domotics.light (pin)  
);
```