Design Document

version 0.0.1

Chaojie Wang

1. **Purpose**

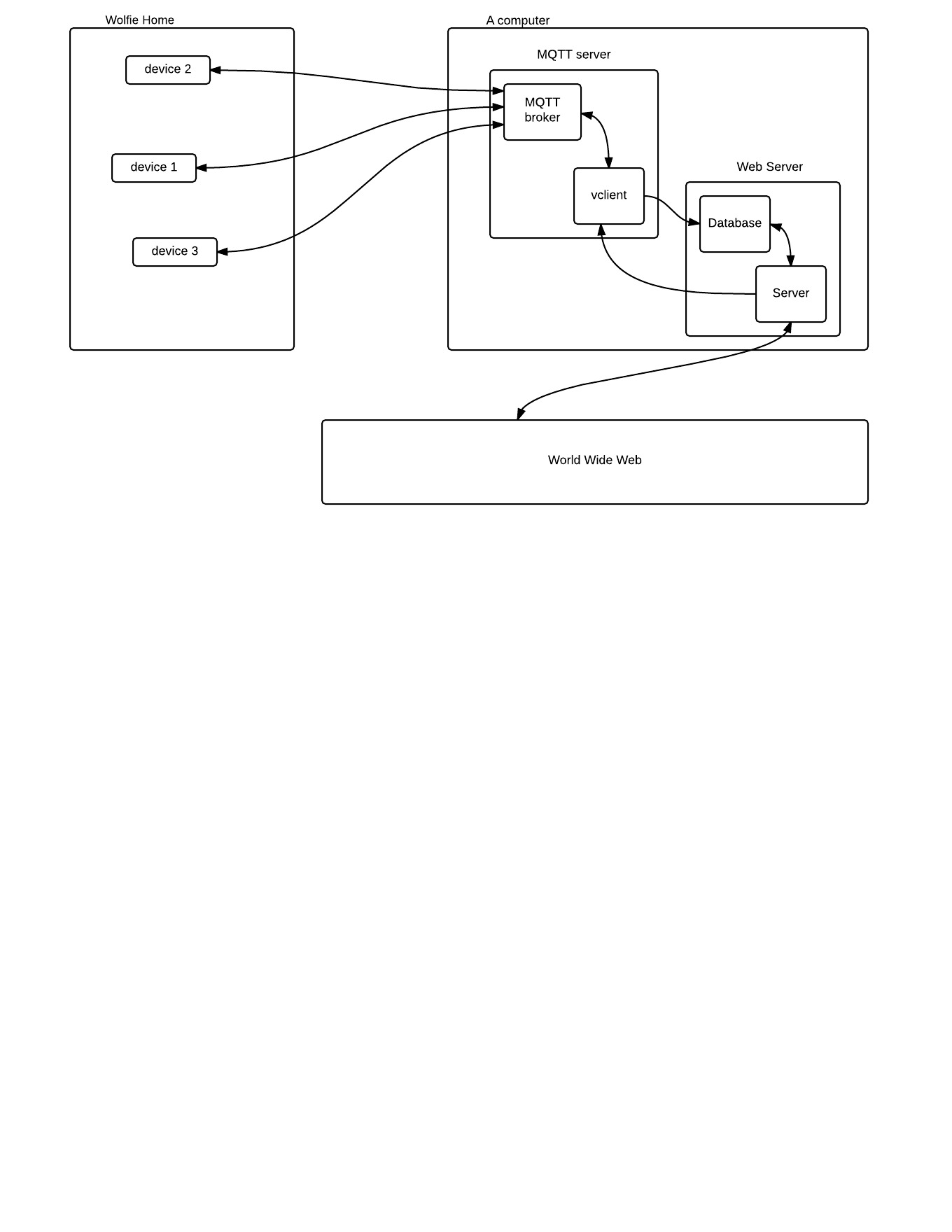
This document is a technical design document. This document specifies components developers need to implement and how pieces work together. This should be used by developers as a reference for implementing Wolfie Home.

1. **Overview**

There are three components in this project: microcontrollers that are able to connect to the Internet, a MQTT server, and a webserver with a database.

Microcontrollers communicate among each other with the MQTT server. Inside MQTT server, there are two components: one is MQTT broker, and the other one is called vclient. The MQTT broker handles the inter-communications among microcontrollers and vclient. The vclient is the virtual client that communicates with database and the webserver. The webserver communicates with users and the MQTT server.

The following topology illustrates the idea:



1. **Microcontrollers**

To be filled

1. **MQTT Server**

Both the web server and the MQTT are running on Linux computer.

**3.1 MQTT broker**

The MQTT broker is implemented according to MQTT protocol. There is an open source MQTT broker called Moquitto, which is used in this project.

**3.1 vclient**

Moquitto also provides library for writing MQTT client. It is used for writing vclient. vclient is short for virtual client. vclient is responsible for two tasks. The first task is obtaining all status data from devices and store them in the database. The second task is creating a channel for accepting commands from the web server. The channel is a FIFO pipe. vclient passively listens commands and act accordingly.

1. **MQTT Server**

**4.1 MySQL**

The database vendor is MySQL. Use account *chaojie*, and password *dummypass* to login. The MySQL has a database called *wolfie\_home*. *wolfie\_home* has two types of table: users type and home type.

There is only one users type table and it is called *users* (for lack of better name). It stores information about *users*. For now, main purpose of *users* is to authenticate users.

|  |  |  |
| --- | --- | --- |
| COLUMN NAME | SQL TYPE | DESCRIPTION |
| id | INT | unique identifier of the user |
| username | VARCHAR(128) | username |
| password | VARCHAR(128) | password |
| email | VARCHAR(128) | email |
| wolfie\_home | VARCHAR(128) | wolfie\_home table name associated with the user |

Each user has a *wolfie\_home* table. The naming convention is wolfie\_home\_[username]. The table stores status of all devices in user’s house. Data stores in a *wolfie\_home* table are generated by vclient.

|  |  |  |
| --- | --- | --- |
| COLUMN NAME | SQL TYPE | DESCRIPTION |
| dev\_name | VARCHAR(128) | Device name |
| dev\_id | VARCHAR(128) | Unique identifier of the device |
| topic | VARCHAR(128) | Topic of the value |
| sensor\_value | VARCHAR(256) | Sensor value, for example temperature of a room |
| time | TIMESTAMP | Time the sensor value was recorded |

**4.2 Web Server**

When a user log in, the web server queries data from *wolfie\_home* table data, and display to the user. It also accepts commands from user to control devices.

1. **Data Format**

**6.1 Between devices and the MQTT broker**

there are two types: control message and status message.

• Status Message

Status message is a message published from devices to the broker. Following topic is used: status/[room]/[type]/[dev\_id].

• [room] is unique identifier of a room

• [type] is type of the information, such as temperature, LED status

• [dev\_id], unique identifier of the device

The payload is csv formatted: [dev\_name],[value],[time]

• [dev\_name]: user friendly name of the device

• [value]: value of the status

• [time]: time stamp of when the value was recorded.

For example, a LED is in room1, and is off. It can publish with topic

“status/room1/led/led1”. The payload could be: “LED,off,2016-02-29 02:45:31”.

Note that vclient subscribes to these data and store them in the database.

• Control Message

To control a device, format of topic is used:

control/[dev\_id]

The payload format is simply:

[value]

For example, in order to turn off a LED with led1 as the dev\_id, topic is control/led1, and the payload will be “on”

**6.1 From the web server to vclient**

It is a FIFO pipe. vclient reads and web server writes.

Data format:

[dev\_id],[control value]

• dev\_id: unique identifier of a room

• control value: value being overwritten in the device.

1. **Security**

Security is very important in this project. To be filled.