

## Contents

<b>1 Pengantar</b>	<b>1</b>
<b>2 Desain</b>	<b>1</b>
2.1 Client . . . . .	1
2.1.1 Run Time . . . . .	2
2.1.2 Install dependency . . . . .	2
2.1.3 Clone repository . . . . .	2
2.1.4 Membuat directory installer . . . . .	2
2.1.5 Untuk membuild program . . . . .	3
2.1.6 Log Error . . . . .	3
2.1.7 Database Device . . . . .	3
2.1.8 Konfigurasi Lainnya . . . . .	3
2.2 Server . . . . .	4
2.2.1 Contoh data . . . . .	4
2.3 Link API pada server . . . . .	5
2.3.1 Link API untuk GET semua data . . . . .	5
2.3.2 Link API untuk POST data . . . . .	5
2.3.3 Link untuk Status Server . . . . .	6
<b>3 Logbook</b>	<b>6</b>
3.1 Masalah . . . . .	6
3.1.1 AutoInsert API . . . . .	6

oo#+TITLE: GFW Doc

## 1 Pengantar

Dokument ini merupakan penjelasan kerja dan beragam setting konfigurasi serta lokasi log error untuk melakukan perbaikan atau ketika menjalankan aplikasi yang ditanamkan pada dragonboard ataupun aplikasi yang dijalankan di server.

## 2 Desain

### 2.1 Client

Untuk Clientnya dibangun dengan menggunakan bahasa C. Menggunakan pthread, curl, sqlite3, dan libgpsd

### 2.1.1 Run Time

Karena menggunakan dinamic linking maka beberapa runtime harus tersedia di sbc dragonboard. Tetapi jika anda ingin melakukan development langsung pada device maka berikut yang harus diinstall.

1. GNU build system (autoconf, automake, libtool)
2. Curl
3. pthread
4. sqlite3

### 2.1.2 Install dependency

```
sudo apt-get install cmake make unzip
sudo apt-get install libcurl4-openssl-dev
sudo apt-get install libgps-dev libsqlite3-dev
sudo apt-get install qdsp-start
wget https://s3.amazonaws.com/json-c_releases/releases/json-c-0.15.tar.gz
tar -zxvf json-c-0.15.tar.gz
cd json-c-0.15
mkdir build &&
cd build &&

cmake -DCMAKE_INSTALL_PREFIX=/usr \
      -DCMAKE_BUILD_TYPE=Release \
      -DBUILD_STATIC_LIBS=OFF \
      .. &&
make && sudo make install
```

### 2.1.3 Clone repository

```
wget https://github.com/faoziaziz/api-gfw/archive/latest.zip
unzip latest.zip
cd api-gfw-latest/
```

### 2.1.4 Membuat directory installer

sebelum membuat membuild program pertama buatlah directory

```
sudo mkdir -p /opt/gfw
sudo chown $USER:$USER /opt/gfw/
```

### 2.1.5 Untuk membuild program

```
autoreconf -i
./configure --prefix=/opt/gfw/
make
make install
```

setelah terinstall maka akan ada file

`/opt/gfw/bin/api-gfw-client`

dengan mengeksekusi file tersebut maka program akan berjalan.

### 2.1.6 Log Error

log error tersedia pada

`/opt/gfw/error.log`

### 2.1.7 Database Device

Jika tidak terdapat koneksi untuk mengirimkan data ke server maka file akan disimpan pada database local

`/opt/gfw/gfw_data_gps.db`

### 2.1.8 Konfigurasi Lainnya

Anda juga bisa melakukan setting terhadap nama device, link url API atau lokasi database local dengan mengedit file `src/conf.h` pada code yang di download.

```
#ifndef __conf_h
#define __conf_h

/* this debug setting */
#define DEBUG 1
/* url api configuration*/
#define URL_API "http://server.faozi.tech:8083/api-gfw/v1"

#define DEVICE "DRAGON001"
#define DATABASE_LOC "/opt/gfw/gfw_data_gps.db"

#endif
```

## 2.2 Server

Server menggunakan Spring-boot untuk aplikasi pengolah data dan Redis sebagai data storagenya. Data yang dikirimkan dari device menuju server menggunakan perantara JSON.

### 2.2.1 Contoh data

Contoh data yang di tampilkan dalam bentuk JSON

```
{
  "id": "27d48518-218a-11eb-9790-3f50feae72b1",
  "online": 1.6048162822154794E9,
  "status": 1.0,
  "sateliteUsed": 15.0,
  "mode": 3.0,
  "time": 1.604816282E9,
  "altitude": 199.31,
  "speed": 0.0,
  "track": 96.5,
  "pdop": 1.3,
  "device": "DRAGON001",
  "longitude": 106.725318833,
  "latitude": -6.555990667,
  "dateStamp": "Sun Nov 8 06:18:02 2020\n"
}
```

dari data tersebut anda bisa melihat beragam variable berikut penjelasannya

variable	keterangan
id	ID merupakan Identitas dari data yang dikirim. Nilai ini dibuat dengan menggunakan UUID.
online	Merupakan variable yang dihasilkan dari status online pada gpsmon nilai ini juga memberikan nilai waktu.
status	Status gps
sateliteUsed	Merupakan jumlah satelite yang digunakan.
mode	
time	Merupakan waktu yang digunakan oleh GPS untuk menerima data. Waktu ini dalam format float.
altitude	Nilai ketinggian dari data GPS,
speed	Kecepatan perangkat yang tertanggap GPS.
track	
pdop	
longitude	Nilai longitude yang diberikan GPS.
latitude	nilai latitude yang diberikan GPS.
dateStamp	Merupakan waktu kirim data dari device ke server.

## 2.3 Link API pada server

### 2.3.1 Link API untuk GET semua data

Untuk mendapatkan data dari seluruh data yang tersimpan pada server anda bisa melakukan transaksi seperti berikut.

URL	<code>http://server.faozi.tech:8083/api-gfw/v1/all</code>
METHODE	GET
RESPONSE	list Semua Data

### 2.3.2 Link API untuk POST data

Untuk menambahkan data GPS pada server anda bisa melakukan transaksi dengan metode seperti berikut

```

URL      http://server.faozi.tech:8083/api-gfw/v1/add
METHODE  POST
REQ_BODY {
    "id":"27d48518-218a-11eb-9790-3f50feae72b1",
    "online":1.6048162822154794E9,
    "status":1.0,
    "sateliteUsed":15.0,
    "mode":3.0,
    "time":1.604816282E9,
    "altitude":199.31,
    "speed":0.0,
    "track":96.5,
    "pdop":1.3,
    "device":"DRAGON001",
    "longitude":106.725318833,
    "latitude":-6.555990667,
    "dateStamp":"Sun Nov 8 06:18:02 2020"
}
RESPONSE Sama seperti REQ_BODY

```

### 2.3.3 Link untuk Status Server

Untuk keperluan tertentu anda bisa melakukan checking server dengan melakukan GET pada URL berikut

```

URL      http://server.faozi.tech:8083/api-gfw/v1/status
METHODE  GET
RESPONSE {"kode":1,"status":"OK"}

```

## 3 Logbook

### 3.1 Masalah

#### 3.1.1 AutoInsert API

Tanggal kasus : *<2020-10-01 Thu 12:31>* . Deskripsi : Masih ada masalah dengan methode insert id di api. Setiap insert data belum bisa autoinsert id.