

GmapZip

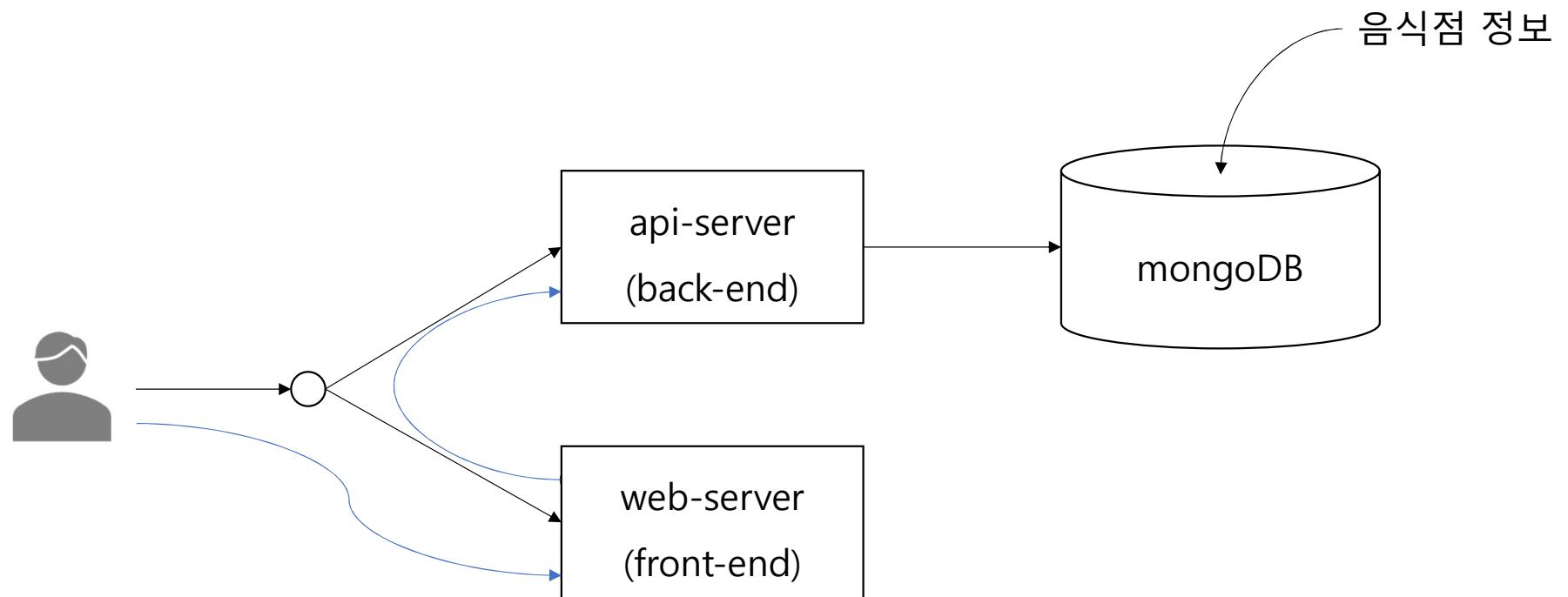
Hands On

Step #1

written by whatwant

2020.12.05

실습 목표



실습 환경



Express 



 mongoose
elegant mongodb object modeling for node.js

Common

Docker 설치

- Ubuntu 18.04 환경에서 최신 버전 docker 설치
- 최신 버전 확인 후 Download
 - <https://download.docker.com/linux/ubuntu/dists/bionic/pool/stable/amd64/>

version은 직접 웹에서 확인하는 것을 추천한다.

```
$ wget https://download.docker.com/linux/ubuntu/dists/bionic/pool/stable/amd64/containerd.io_1.3.7-1_amd64.deb  
$ wget https://download.docker.com/linux/ubuntu/dists/bionic/pool/stable/amd64/docker-ce-cli_19.03.13~3-0~ubuntu-bionic_amd64.deb  
$ wget https://download.docker.com/linux/ubuntu/dists/bionic/pool/stable/amd64/docker-ce_19.03.13~3-0~ubuntu-bionic_amd64.deb
```

- 설치

```
$ sudo dpkg --install ./containerd.io_1.3.7-1_amd64.deb  
$ sudo dpkg --install ./docker-ce-cli_19.03.13~3-0~ubuntu-bionic_amd64.deb  
$ sudo dpkg --install ./docker-ce_19.03.13~3-0~ubuntu-bionic_amd64.deb
```

- 현재 계정에서 바로 docker 사용할 수 있도록 권한(그룹) 설정

```
$ sudo usermod -aG docker $USER && newgrp docker
```

- hello-world

```
$ docker run hello-world
```

```
> docker run hello-world  
Unable to find image 'hello-world:latest' locally  
latest: Pulling from library/hello-world  
0e03bdcc26d7: Pull complete  
Digest: sha256:4cf9c47f86df71d48364001ede3a4fc85ae80ce0ebad74156906caff5378bc  
Status: Downloaded newer image for hello-world:latest  
  
Hello from Docker!  
This message shows that your installation appears to be working correctly.  
To generate this message, Docker took the following steps:  
1. The Docker client contacted the Docker daemon.  
2. The Docker daemon pulled the "hello-world" image from the Docker Hub.  
(amd64)  
3. The Docker daemon created a new container from that image which runs the  
executable that produces the output you are currently reading.  
4. The Docker daemon streamed that output to the Docker client, which sent it  
to your terminal.  
  
To try something more ambitious, you can run an Ubuntu container with:  
$ docker run -it ubuntu bash  
  
Share images, automate workflows, and more with a free Docker ID:  
https://hub.docker.com/  
  
For more examples and ideas, visit:  
https://docs.docker.com/get-started/
```

Node.JS 설치

- <https://nodejs.org/>
- <https://github.com/nodesource/distributions/blob/master/README.md#debinstall>

```
$ curl -sL https://deb.nodesource.com/setup_14.x | sudo -E bash -
```

```
$ sudo apt-get install -y nodejs
```

실습은 굳이 하지 않아도 된다.

- node.js 샘플 작성

```
$ nano /srv/workspace/html/index-1.js
```

- node 실행

```
$ cd /srv/workspace/html/
```

```
$ node ./index-1.js
```

```
var http = require('http');
var os = require('os');

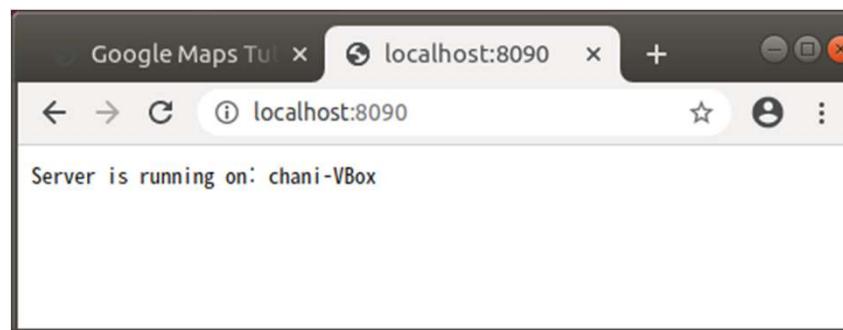
var host = os.hostname();

var handleRequest = function(request, response) {
  console.log('Received request for URL: ' + request.url);
  response.writeHead(200);
  response.end('Server is running on: ' + host);
};

var www = http.createServer(handleRequest);
www.listen(8090);
```

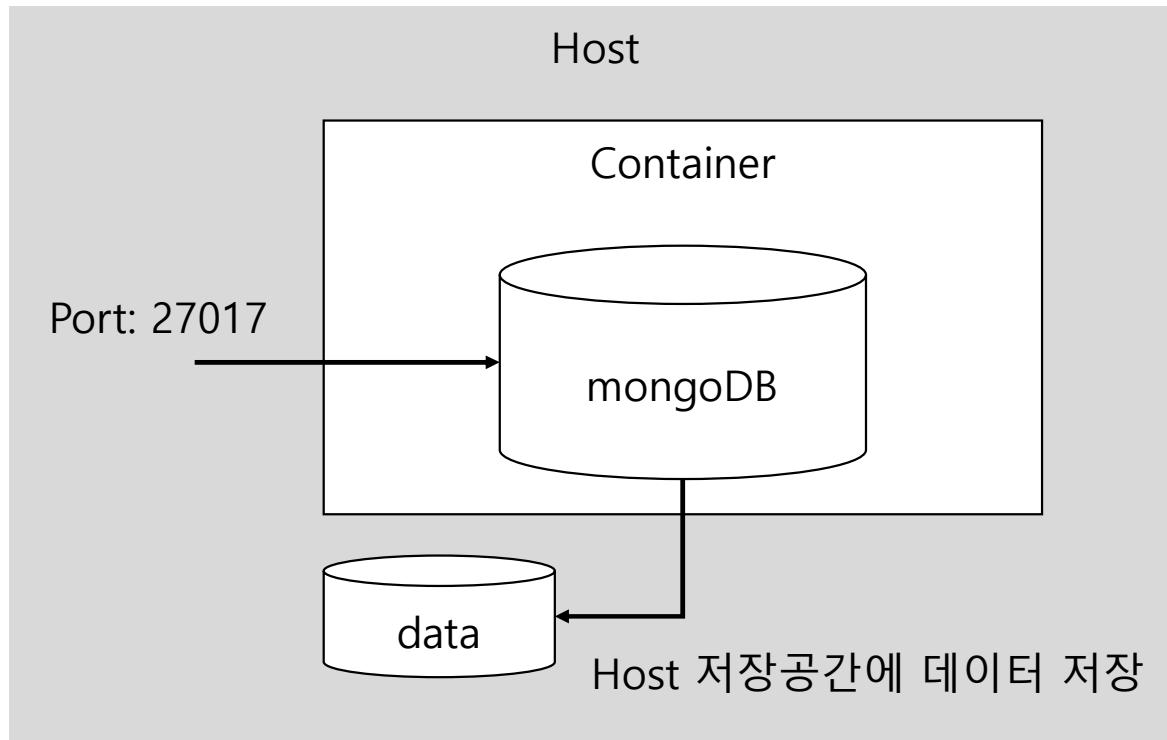
- 웹페이지 접속

```
http://localhost:8090/
```



MongoDB

mongoDB – Docker – Goal



mongoDB – Docker – run & client

- 데이터 저장할 디렉토리 미리 생성 후, docker 실행

```
$ mkdir /srv/workspace/db  
$ docker run -d -p 27017:27017 -v /srv/workspace/db:/data/db --name mongo-db mongo:latest
```

- mongoDB client 설치

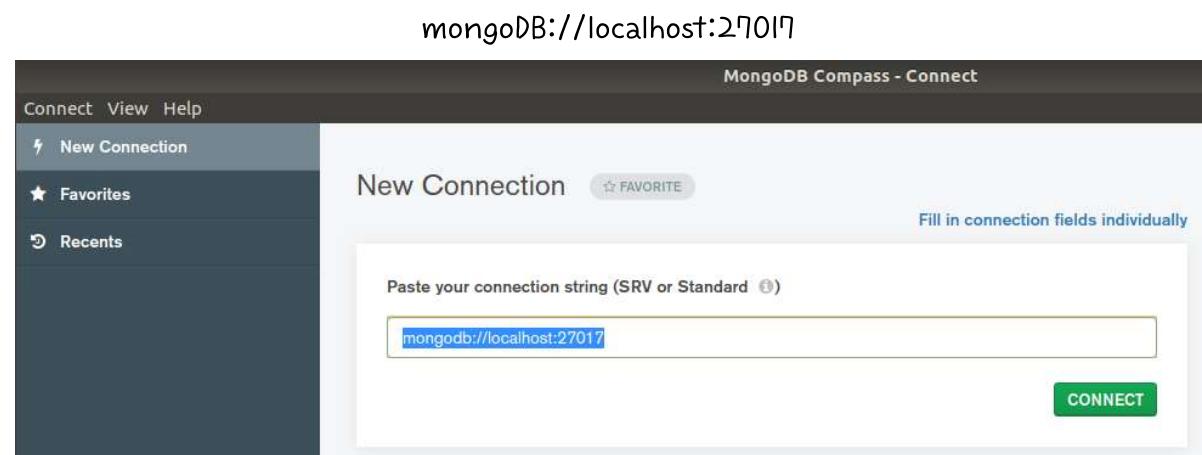
```
https://www.mongodb.com/try/download/compass
```

```
$ wget https://downloads.mongodb.com/compass/mongodb-compass_1.23.0_amd64.deb  
$ sudo dpkg --install ./mongodb-compass_1.23.0_amd64.deb  
$ sudo apt --fix-broken install  
$ sudo dpkg --install ./mongodb-compass_1.23.0_amd64.deb
```

- 의존성 패키지 때문에 --fix-broken 중간에 실행 후 continue

mongoDB – Docker – compass

"MongoDB Compass" 설치되어 있음을 확인



CONNECT 성공하면 볼 수 있는 화면

MongoDB Compass - localhost:27017

Database Name	Storage Size	Collections	Indexes
admin	20.0KB	0	1
config	4.0KB	0	2
local	20.0KB	1	1

mongoDB – Docker – get Data

- 공공데이터포털 : 파일데이터 상세 – 경기도_안심식당 정보

<https://www.data.go.kr/data/15065521/fileData.do>

- 경기데이터드림 : 데이터셋 – 안심식당 정보

<https://data.gg.go.kr/portal/data/service/selectServicePage.do?infId=YBSSB8F816M3B966AB6K30423820&infSeq=1>

The screenshot shows the 'File Data' section of the Gyeonggi-do Safety Restaurant dataset. It includes a summary table with various details like location, contact information, and download links, and a detailed table with columns such as 'Category', 'Title', 'Content', and 'Download'.

JSON 타입으로 다운로드 받기 !!!

The screenshot shows the 'Dataset' section of the Gyeonggi-do Data Dream website. It displays the same dataset information as the previous screenshot but includes a prominent 'Download as JSON' button at the bottom of the page.

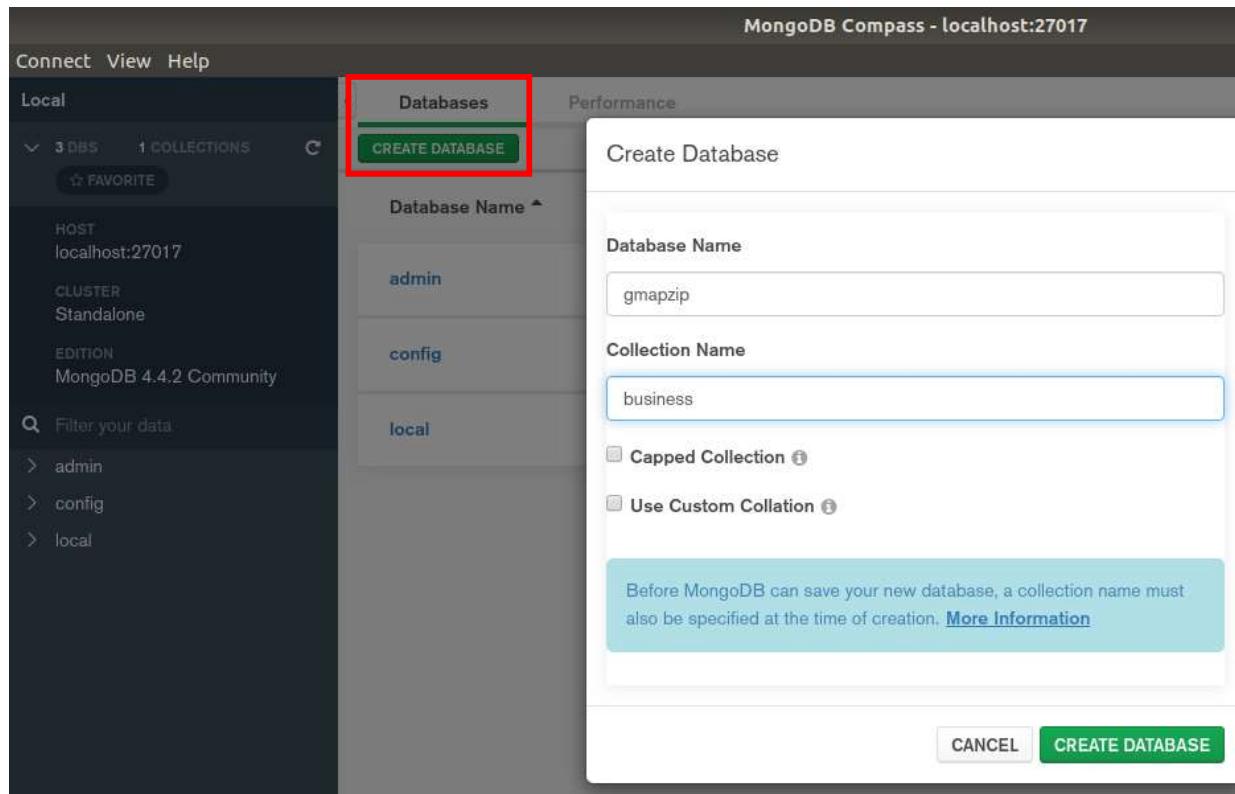
mongoDB – Docker – create Database & Collection

- Create Database & Collection

Database Name : gmapzip

당연히 다른 이름으로 해도 되지만, 실습의 일관성을 위해 같은 이름으로 하기를 권장한다.

Collection Name : business



mongoDB – Docker – import Data

- “business” collection에서 “Import Data” 선택하여, 앞에서 다운로드 받은 JSON 파일 지정
- “Select Input File Type”은 당연하게도 “JSON”으로 지정하고 IMPORT

The image shows two screenshots of the MongoDB Compass interface. The left screenshot displays the 'business' collection in the 'gmapzip' database, showing 0 documents and 1 index. The 'Import Data' button is highlighted with a red box. The right screenshot shows an 'Import To Collection gmapzip.business' dialog. It includes fields for 'Select File' (set to '/srv/workspace/SDI-K8s/hands-on/gmapzip/안심식당정보.json'), 'Select Input File Type' (set to 'JSON', also highlighted with a green box), and 'Options' (checkboxes for 'Ignore empty strings' and 'Stop on errors').

MongoDB Compass - localhost:27017/gmapzip.business

Documents

gmapzip.business

Documents Aggregations Schema Explain Plan Indexes Validation

DOCUMENTS 0 TOTAL SIZE 0B AVG. SIZE 0B INDEXES 1 TOTAL SIZE 4.0KB AVG. SIZE 4.0KB

0 FILTER ADD DATA

This collection has no data

Import Data

MongoDB Compass - localhost:27017/gmapzip.business

Import To Collection gmapzip.business

Select File /srv/workspace/SDI-K8s/hands-on/gmapzip/안심식당정보.json BROWSE

Select Input File Type JSON CSV

Options

Ignore empty strings

Stop on errors

CANCEL IMPORT

mongoDB – Docker – convert Data

- 앞에서 실행한 mongoDB에 있는 mongo를 실행하도록 하자

```
$ docker exec -it mongo-db mongo
```

```
> use gmapzip

>db.business.find().forEach(function(biz){
  db.geodata.insert({ "geometry": { "type": "Point",
    "coordinates": [ Number(biz.REFINE_WGS84_LOGT), Number(biz.REFINE_WGS84_LAT) ] },
  "properties": { "business_id": biz._id,
    "name": biz.BIZPLC_NM,
    "tel": biz.TELNO,
    "address": biz.REFINE_LOTNO_ADDR,
    "category": biz.INDUTYPE_DETAIL_NM } });
});
```

- "business" collection 데이터를 이용해서, 우리가 사용할 새로운 collection을 만드는 과정이다.

mongoDB – Docker – test Data

- Index 생성해보고, 검색도 한 번 해보자.

```
> db.geodata.createIndex( {"properties.category" : "text"} );
```

```
> db.geodata.find({ "properties.category": "한식" })
```

```
> exit
```

- Compass에서 Find 하고 싶은 경우, 원하는 collection의 Document에서 FILTER에 아래와 같이 입력하면 된다.

```
({"properties.category": "한식"})
```

mongoDB – Docker – Compass browse

MongoDB Compass - localhost:27017

Local

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 4.4.2 Community

business

business

Documents Aggregations Schema Explain Plan Indexes Validation

Displaying documents 1 ~ 20 of 14430

ADD DATA VIEW FILTER FIND RESET

Filter your data

admin config gmapzip business local

_id: ObjectId("5fc65019b6df553f75b0f9e4")
SAFETY_RESTRICT_NO: "2390"
SIDO_NM: "경기도"
SJUNGMYUNG_NM: "고양시"
BIZPLC_NM: "국내(내)"
REFINE_ROADNM_ADDR: "경기도 군포시 산본로323번길 16-30"
REFINE_LOTNO_ADDR: "경기도 군포시 산본동 1333-1번지"
DETAIL_ADDR: "성보빌딩 208호"
INDUSTRY_NM: "일반음식점"
INDUSTRY_DETAIL_NM: "일식"
TELNO: ""
SLCTM_YN_DIV: "y"
CANC1_DE: ""
RK_MAKR: ""
UPD_DAY: ""
REFINE_ZIPNO: "15865"
REFINE_WGS84_LAT: "37.3081888763"
REFINE_WGS84_LOGT: "126.9316075911"

MongoDB Compass - localhost:27017/gmapzip.geodata

Local

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB 4.4.2 Community

business

geodata

Documents Aggregations Schema Explain Plan Indexes Validation

Query returned 10194 documents. This report is based on a sample of 1000 documents (9.81%).

OPTIONS ANALYZE RESET

Filter your data

admin config gmapzip business geodata local

_id
objectid

first: 2020-12-04 19:36:16 last: 2020-12-04 19:36:20

geometry
coordinates

Document with 5 nested fields.

Query returned 10194 documents. This report is based on a sample of 1000 documents (9.81%).

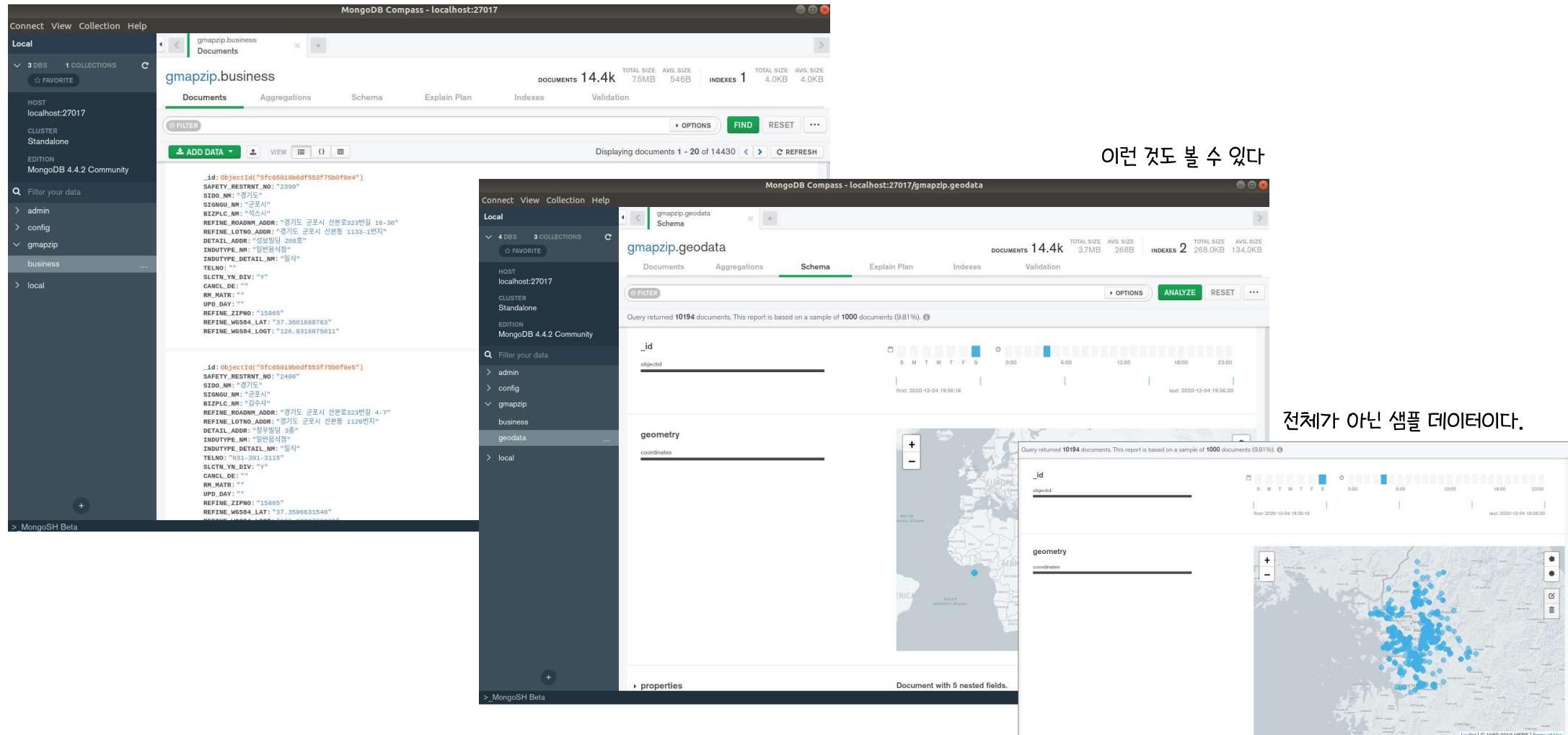
first: 2020-12-04 19:36:16 last: 2020-12-04 19:36:20

geometry
coordinates

Leaflet | © 1987-2019 HERE | Terms of Use

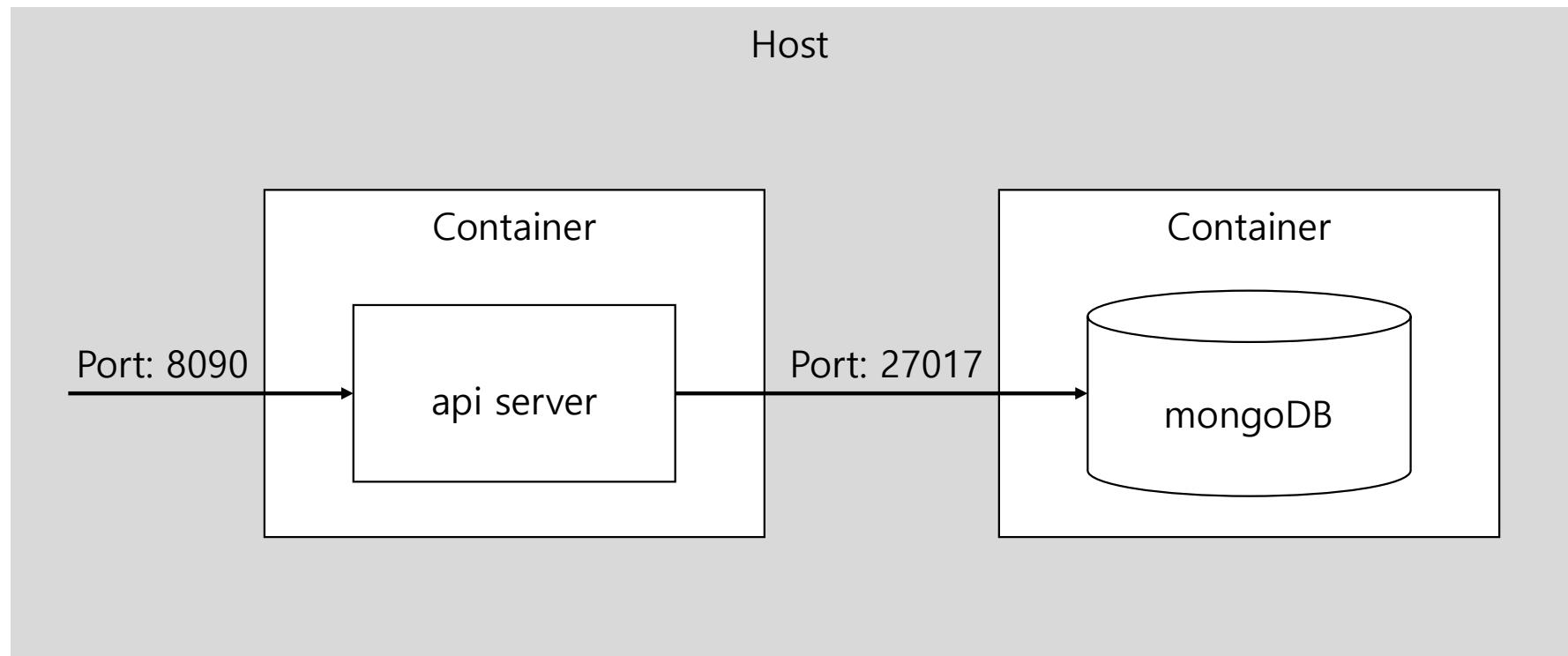
이런 것도 볼 수 있다

전체가 아닌 샘플 데이터이다.

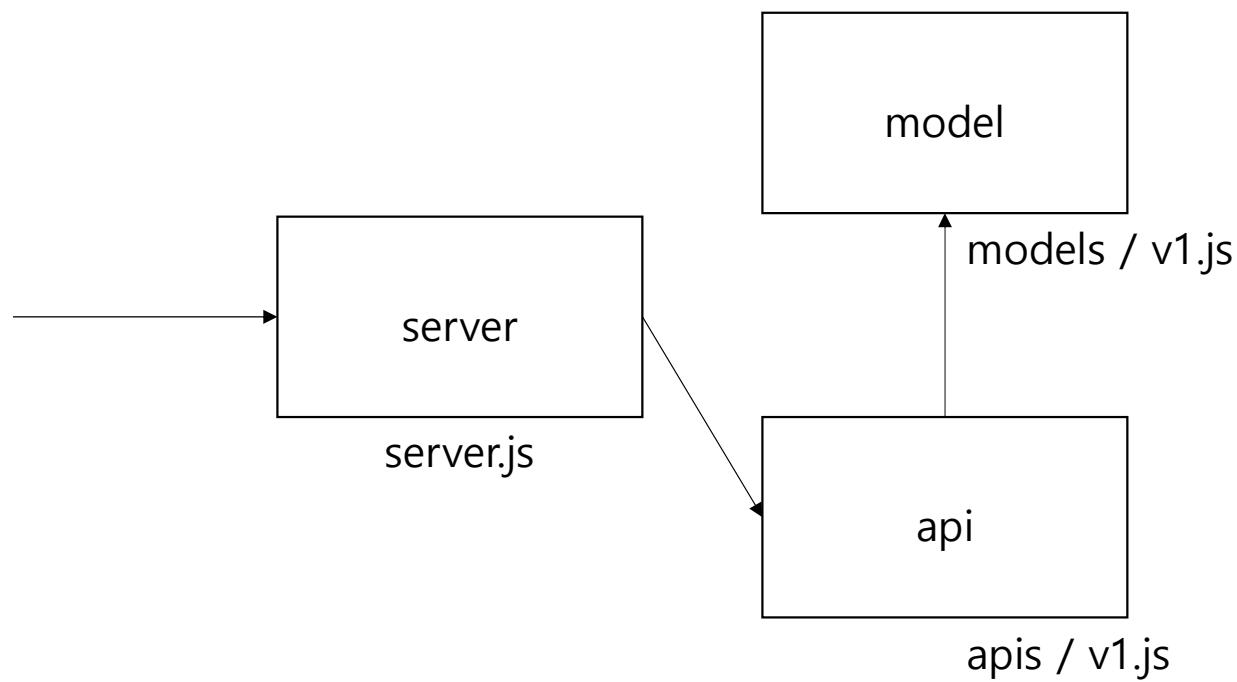


**api server
(backend)**

backend – Docker – Goal



backend – Docker – Architecture



backend – Docker – server.js

```
$ cd /srv/workspace/gmapzip  
$ nano ./server.js
```

```
const mongoose = require('mongoose');  
mongoose.connect('mongodb://mongo-db/gmapzip',  
    {useNewUrlParser: true, useUnifiedTopology: true, useCreateIndex: true});  
mongoose.set('debug', true);  
  
const express = require('express');  
const app = express();  
const port = 8090;  
  
app.use('/api/v1', require('./apis/v1'));  
  
app.listen(port);
```

backend – Docker – apis/v1.js

```
$ cd /srv/workspace/gmapzip  
$ nano ./apis/v1.js
```

```
const express = require('express');  
const router = express.Router();  
  
const geodata = require('../models/v1');  
  
router.get('/within', function(request, response) {  
    geodata.findInBox([Number(request.query.left), Number(request.query.lower)],  
        [Number(request.query.right), Number(request.query.upper)])  
        .exec(function(error, result){  
            if (error) return response.status(500).json(error);  
            return response.json(result);  
        });  
});  
  
router.get('/:id', function(request, response) {  
    geodata.findById(request.params.id).exec(function(error, result){  
        if (error) return response.status(500).json(error);  
        return response.json(result[0]);  
    });  
});  
  
module.exports = router;
```

backend – Docker – models/v1.js

```
$ cd /srv/workspace/gmapzip  
$ nano ./models/v1.js
```

```
const mongoose = require('mongoose');

const pointSchema = new mongoose.Schema({
  type: { type: String, enum: ['Point'], required: true },
  coordinates: { type: [Number], required: true }
});

const geodataSchema = new mongoose.Schema({
  geometry: pointSchema,
  properties: {
    business_id: { type: String, unique: true, index: true, required: true },
    name: String,
    tel: String,
    address: String,
    categories: [String],
  },
  { collection: 'geodata' }
});
```

```
geodataSchema.index({properties.name: 'text', properties.categories: 'text'});

geodataSchema.statics.findById = function(ID){
  return this.find().where('properties.business_id', ID);
};

geodataSchema.index({geometry: '2dsphere'});

geodataSchema.statics.findInBox = function(lowerLeft, upperRight){
  return this.find().where('geometry').box(lowerLeft, upperRight);
};

module.exports = mongoose.model('geodata', geodataSchema);
```

backend – Docker – package.json

```
$ cd /srv/workspace/gmapzip  
$ nano ./package.json
```

```
{  
  "name": "gmapzip-backend",  
  "version": "0.0.1",  
  "description": "gmapzip DB API SERVER",  
  "author": "whatwant <whatwant@whatwant.com>",  
  "scripts": {  
    "start": "node server.js"  
  },  
  "dependencies": {  
    "express": "latest",  
    "mongoose": "latest"  
  }  
}
```

backend – Docker – Dockerfile

```
$ cd /srv/workspace/gmapzip  
$ nano ./Dockerfile
```

```
FROM node:latest
```

```
WORKDIR /src
```

```
COPY . .
```

```
RUN npm install
```

```
EXPOSE 8090
```

```
CMD npm start
```

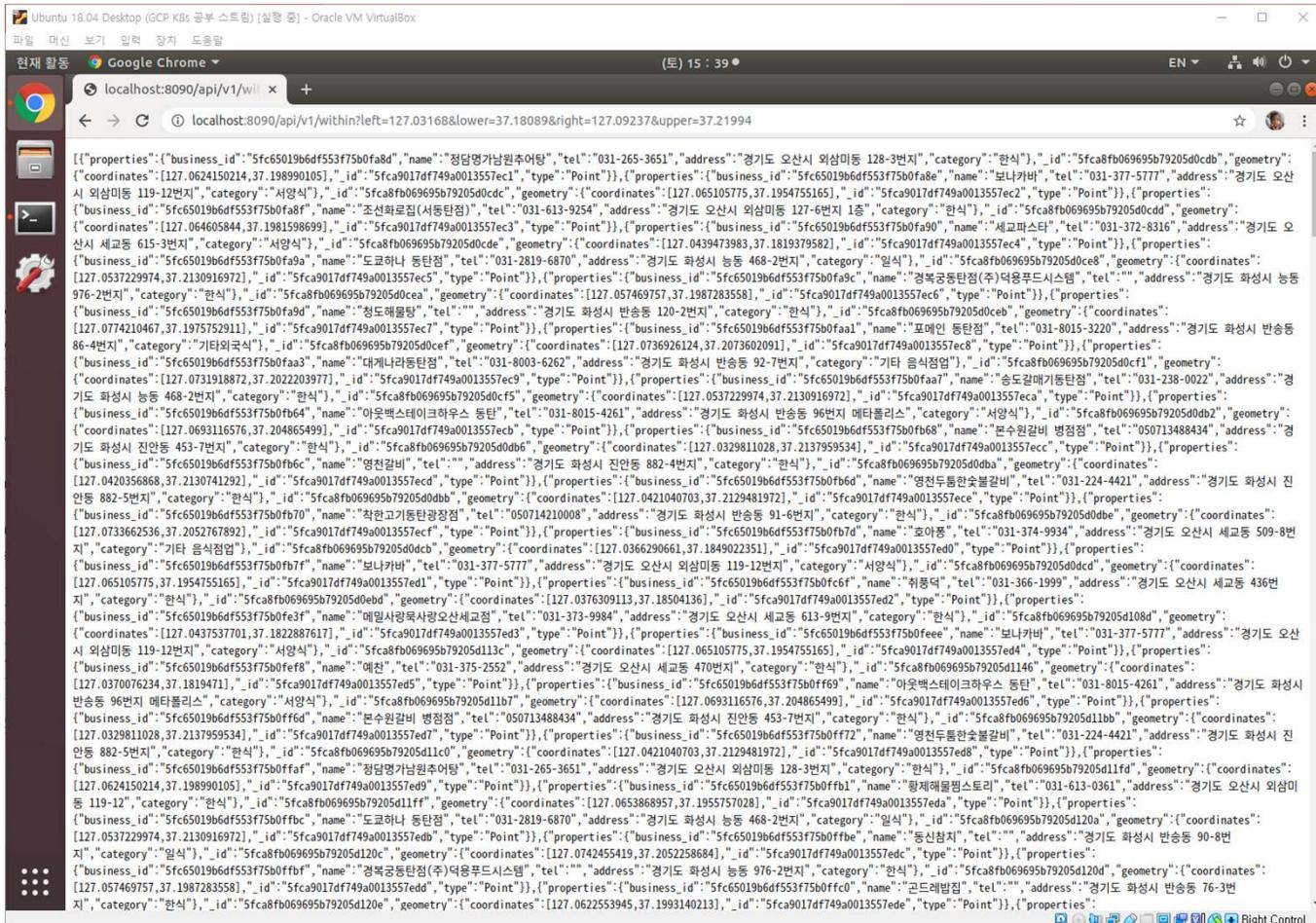
```
$ docker build -t gmapzip-backend:v1 .
```

```
$ docker run -d -p 8090:8090 --link mongo-db:mongo-db gmapzip-backend:v1
```

mongo-db Container IP를 대체하기 위해 "--link" 옵션을 이용하여 hostname을 지정해준다.

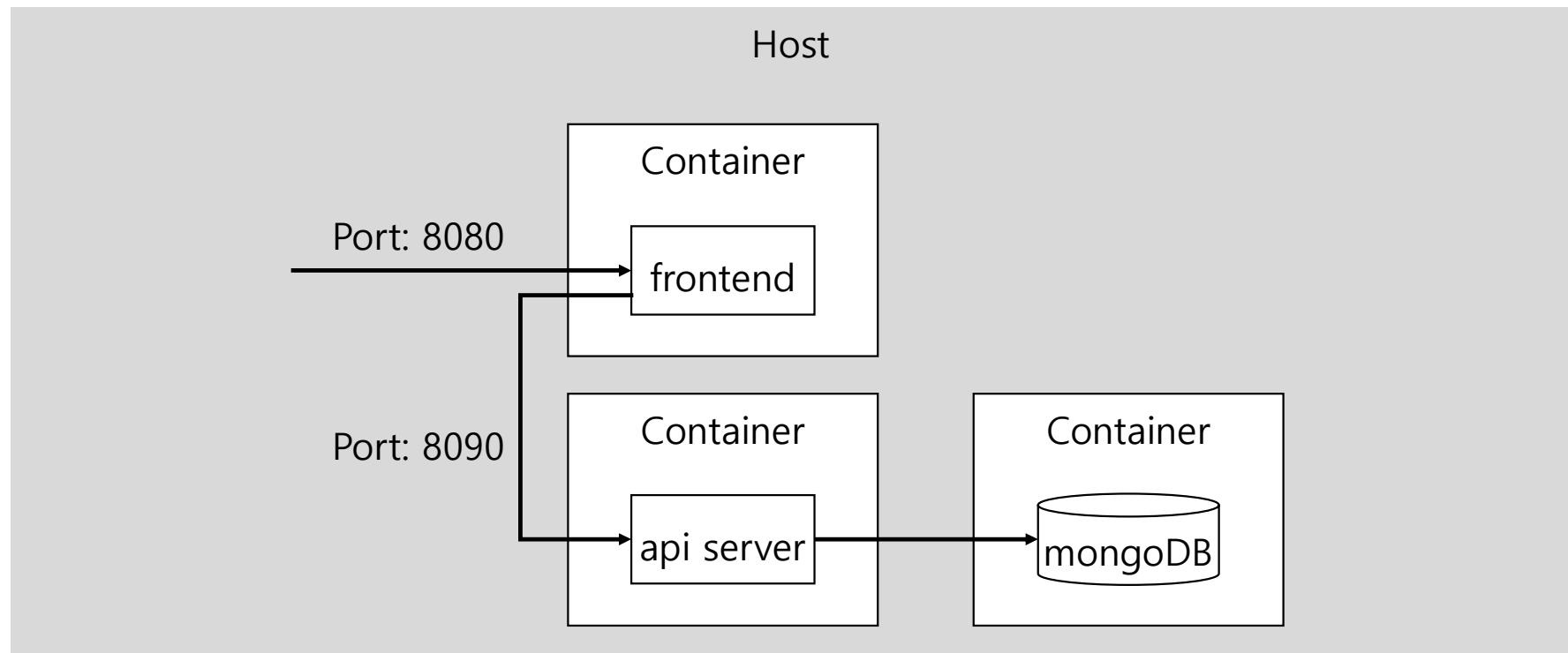
backend – Docker – results

<http://localhost:8090/api/v1/within?left=127.03168&lower=37.18089&right=127.09237&upper=37.21994>



GmapZip Service (frontend)

frontend – Docker – Goal



frontend – Local – create Nuxt Project

```
$ cd /srv/workspace/  
$ npx create-nuxt-app gmapzip
```

```
create-nuxt-app v3.4.0  
✨ Generating Nuxt.js project in gmapzip  
? Project name: gmapzip  
? Programming language: JavaScript  
? Package manager: Npm  
? UI framework: Vuetify.js  
? Nuxt.js modules: Axios, Progressive Web App (PWA), Content  
? Linting tools: (Press <space> to select, <a> to toggle all, <i> to invert selection)  
? Testing framework: None  
? Rendering mode: Universal (SSR / SSG)  
? Deployment target: Server (Node.js hosting)  
? Development tools: jsconfig.json (Recommended for VS Code if you're not using typescript)  
? What is your GitHub username? whatwant  
? Version control system: None
```

```
create-nuxt-app v3.4.0  
✨ Generating Nuxt.js project in gmapzip  
? Project name: gmapzip  
? Programming language: JavaScript  
? Package manager: Npm  
? UI framework: Vuetify.js  
? Nuxt.js modules: Axios, Progressive Web App (PWA), Content  
? Linting tools: (Press <space> to select, <a> to toggle all, <i> to invert selection)  
? Testing framework: None  
? Rendering mode: Universal (SSR / SSG)  
? Deployment target: Server (Node.js hosting)  
? Development tools: jsconfig.json (Recommended for VS Code if you're not using typescript)  
? What is your GitHub username? whatwant  
? Version control system: Git
```

Git은 선택하지 않는 것이 낫다

frontend – Local – add page

```
$ cd /srv/workspace/gmapzip  
$ nano ./pages/map.vue
```

GoogleMap을 화면에 보여주는 페이지 구성이다.

```
<template>  
  <div id="map"></div>  
</template>  
  
<script>  
export default {  
  mounted(){  
    const map = new google.maps.Map(document.getElementById('map'), {  
      center: { lat: 37.19, lng: 127.08 },  
      zoom: 14  
    });  
  }  
}  
</script>
```

```
<style>  
#map {  
  width: 100%;  
  height: 100%;  
}  
</style>
```

frontend – Local – modify layout

```
$ cd /srv/workspace/gmapzip  
$ nano ./layouts/default.vue
```

메뉴를 하나 추가하고, Map이 화면에 제대로 채워지게 하기 위해서 layout 수정

```
...  
<v-main>  
  <v-container fluid fill-height>  
    <nuxt />  
  </v-container>  
</v-main>  
...  
items: [  
...  
  },  
  {  
    icon: 'mdi-map-search',  
    title: '맛집',  
    to: '/map'  
  }  
,  
],
```

frontend – Local – add script for GoogleMap

```
$ cd /srv/workspace/gmapzip  
$ nano ./nuxt.config.js
```

GoogleMap을 위한 Script 파일이 1번만 호출되도록 여기에 명시해준다.

```
...  
head: {  
  titleTemplate: '%s - gmapzip',  
  title: 'gmapzip',  
  meta: [  
    { charset: 'utf-8' },  
    { name: 'viewport', content: 'width=device-width, initial-scale=1' },  
    { hid: 'description', name: 'description', content: "" }  
],  
  link: [  
    { rel: 'icon', type: 'image/x-icon', href: '/favicon.ico' }  
],  
  script: [  

```

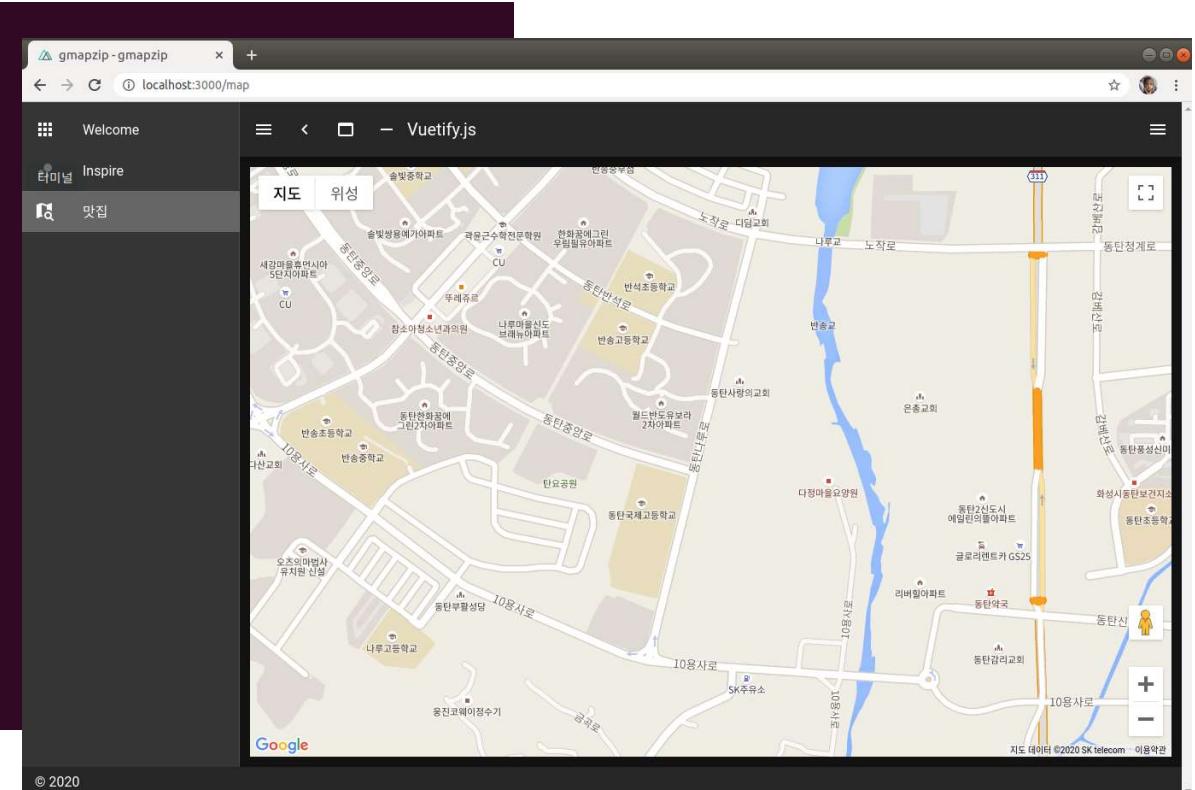
frontend – Local – Execute

```
$ cd /srv/workspace/gmapzip  
$ npm run dev
```

Nuxt 빌드 후 호스팅해준다. Map이 잘 나오는 것을 확인할 수 있다.

<http://localhost:3000/>

```
> npm run dev  
> gmapzip@1.0.0 dev /srv/workspace/gmapzip  
> nuxt  
  
i Parsed 1 files in 0,4 seconds  
  
Nuxt.js @ v2.14.7  
▶ Environment: development  
▶ Rendering: server-side  
▶ Target: server  
  
Listening: http://localhost:3000/  
  
i Preparing project for development  
i Initial build may take a while  
✓ Builder initialized  
✓ Nuxt files generated  
  
✓ Client  
Compiled successfully in 6.94s  
  
✓ Server  
Compiled successfully in 6.33s  
  
i Waiting for file changes  
i Memory usage: 357 MB (RSS: 575 MB)  
i Listening on: http://localhost:3000/
```



frontend – Docker – modify package

```
$ cd /srv/workspace/gmapzip  
$ nano ./package.json
```

Docker Image를 만들기 위한 준비

```
...  
"private": true,  
"config": {  
  "nuxt": {  
    "host": "0.0.0.0", 이렇게 해주지 않으면 localhost에서만 접근된다.  
    "port": "8080"  
  }  
},  
"scripts": {
```

frontend – Docker – create Dockerfile & run

```
$ cd /srv/workspace/gmapzip  
$ nano ./Dockerfile
```

```
FROM node:14.15.1      version 호환성이 맞지 않아서 latest로 하면 안된다.
```

```
WORKDIR /src  
COPY . .
```

```
RUN npm install  
RUN npm run build
```

```
EXPOSE 8080  
CMD npm run start
```

```
$ docker build -t gmapzip-frontend:v1 .
```

```
$ docker run -d -p 8080:8080 gmapzip-frontend:v1
```

But, 식당 정보는 아직 가져오지 않았다!

frontend – Docker – update page

```
$ cd /srv/workspace/gmapzip  
$ nano ./pages/map.vue
```

api server로부터 현재 화면의 위경도 안에 포함된 식당 정보를 받아서 출력하는 부분을 추가한다.

```
...  
const map = new google.maps.Map(document.getElementById('map'), {  
  center: { lat: 37.19, lng: 127.08 },  
  zoom: 14  
});  
  
map.addListener('idle', function(){  
  map.data.forEach(function(feature){  
    map.data.remove(feature);  
  });  
  
  const bounds = map.getBounds();  
  const sw = bounds.getSouthWest();  
  const ne = bounds.getNorthEast();  
  
  const request = new XMLHttpRequest();  
  const url = 'http://localhost:8090/api/v1/within?'  
    + 'left=' + sw.lng()  
    + '&lower=' + sw.lat()  
    + '&right=' + ne.lng()  
    + '&upper=' + ne.lat();  
  request.open('GET', url);  
  request.responseType = 'json';  
  request.send();  
  
  request.onload = function() {  
    const geodata = request.response;  
    for(var i = 0; i < geodata.length; i++){  
      const latLng = new google.maps.LatLng({ lng: geodata[i].geometry.coordinates[0], lat:  
        geodata[i].geometry.coordinates[1] });  
      map.data.add(new google.maps.Data.Feature({ geometry: latLng, properties: geodata[i].properties }));  
    }  
  }  
});  
  
const infowindow = new google.maps.InfoWindow();  
map.data.addListener('click', function(e){  
  const feature = e.feature;  
  const name = feature.getProperty("name");  
  const category = feature.getProperty("category");  
  
  let content = '<div style="color:black"><b>' + name + '</b><br>' + category + '</div>';  
  
  infowindow.setPosition(e.latLng);  
  infowindow.setContent(content);  
  infowindow.open(map);  
});  
...
```

frontend – Docker – rebuild & update & run

```
$ docker build -t gmapzip-frontend:v2 .
```

v2로 이미지 새로 빌드하고,

```
$ docker ps -a
```

```
$ docker stop xxx
```

지금 실행중인 Container 멈춘 다음에, 삭제해버리자.

```
$ docker rm xxx
```

```
$ docker run -d -p 8080:8080 gmapzip-frontend:v2
```

그리고, 새로운 이미지로 다시 container 실행하자.

frontend – Docker – CORS

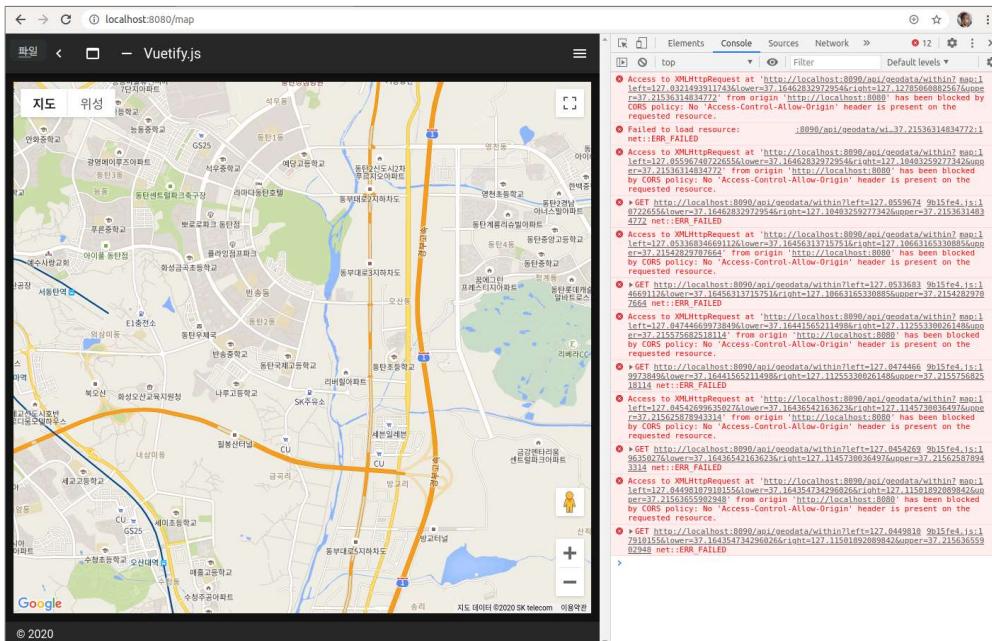
<http://localhost:8080>

그런데, 지도에 원하는 식당 정보가 나오지 않는다.

Chrome에서 F12 눌러서 개발자 메뉴 中 Console을 보면 여러 화면을 볼 수 있다.

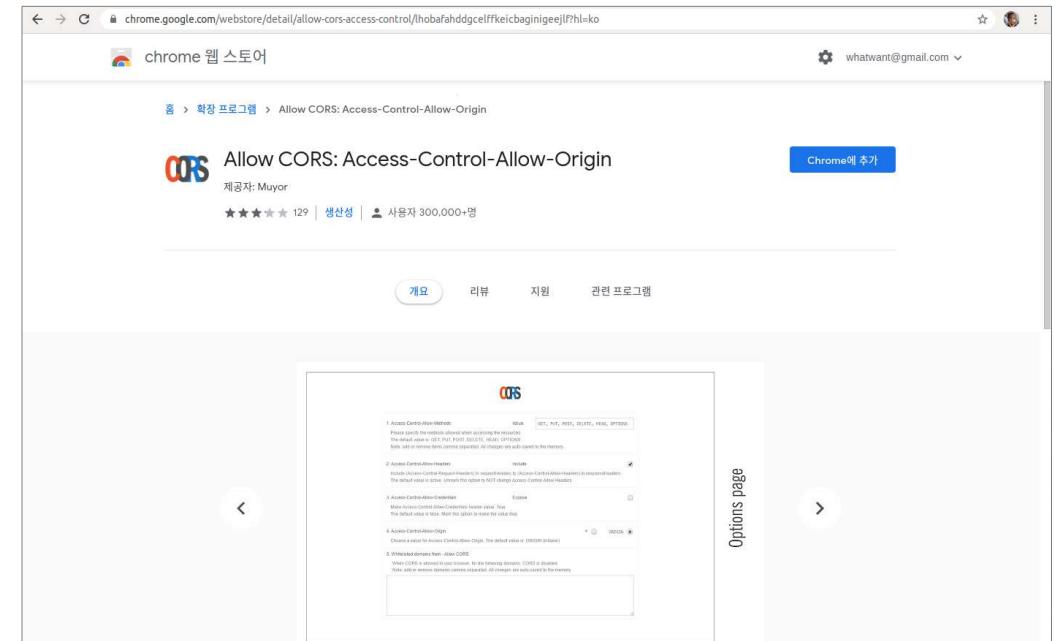
api-server를 호출하는 부분에서 다른 웹사이트와 통신하는 부분을

Chrome에서는 보안상 위험하다고 보고 차단을 해버린 것이다.



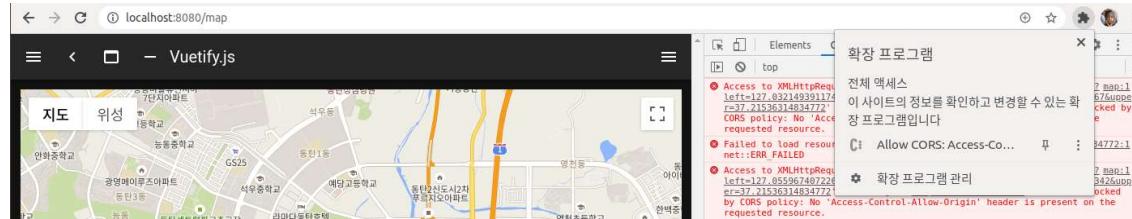
Chrome Extension(확장 프로그램)에서

"Allow CORS: Access-Control-Allow-Origin" 을 찾아서 설치(추가)하자.

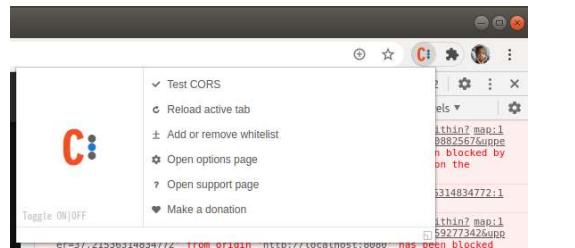


frontend – Docker – rebuild & update & run

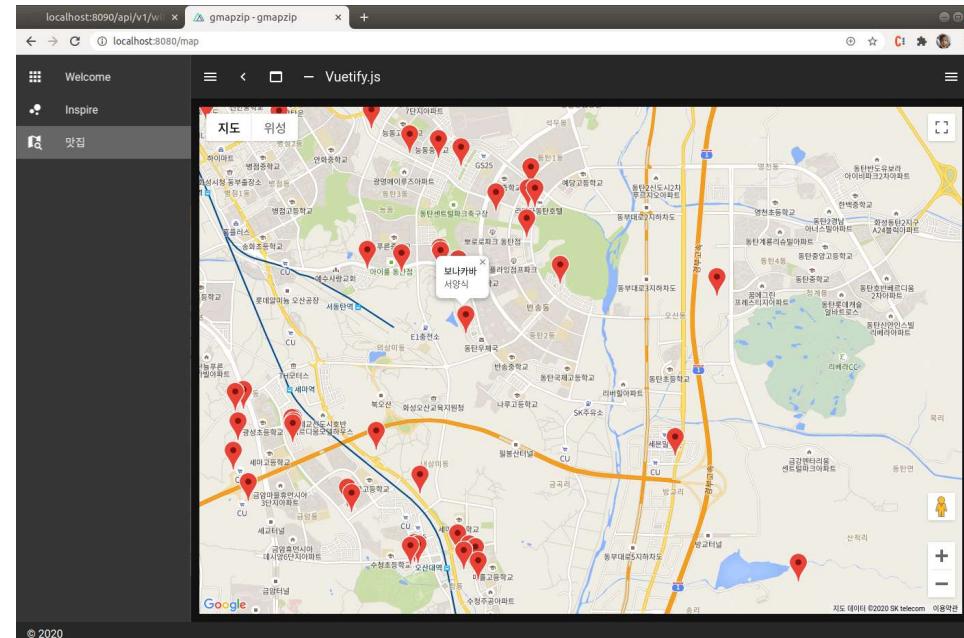
① 확장 프로그램에서 Allow CORS를 편셋으로 고정하고



③ 지도에 마커(Marker)가 잘 나오는 것을 볼 수 있다.



② Toggle ON 해주면,



지금까지의 결과

```
chanı ➤ /srv/workspace/gmapzip/backend
➤ docker ps -a
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
08cc315f479e	gmapzip-frontend:v2	"docker-entrypoint.s..."	20 hours ago	Up 5 seconds	0.0.0.0:8080->8080/tcp	angry_nightingale
c2236831a419	gmapzip-backend:v1	"docker-entrypoint.s..."	2 days ago	Up 14 seconds	0.0.0.0:8090->8090/tcp	suspicious_einstein
81983857ee44	mongo:latest	"docker-entrypoint.s..."	2 days ago	Up 22 seconds	0.0.0.0:27017->27017/tcp	mongo-db

```
➤ docker images
```

REPOSITORY	TAG	IMAGE ID	CREATED	SIZE
gmapzip-frontend	v2	d54b18624647	21 hours ago	1.09GB
gmapzip-backend	v1	d9e662334ac8	2 days ago	952MB
gmapzip-frontend	v1	de9b68e643a2	5 days ago	1.09GB
mongo	latest	ef5c2207766e	10 days ago	493MB
node	latest	2d840844f8e7	11 days ago	935MB
node	14.15.1	4c7835fa737e	2 weeks ago	943MB
hostname	v1	c64a791a11e8	8 weeks ago	943MB
node	14.13.1	f47907840247	8 weeks ago	943MB
gcr.io/k8s-minikube/kicbase	v0.0.12-snapshot3	25ac91b9c8d7	3 months ago	952MB
hello-world	latest	bf756fb1ae65	11 months ago	13.3kB

MongoDB Compass - localhost:27017/gmapzip.geodata

Connect View Collection Help

Local

4 DBS 3 COLLECTIONS

FAVORITE

HOST localhost:27017

CLUSTER Standalone

EDITION MongoDB Compass 4.4.2 Community

Filter your data

admin config gmapzip business geodata local

gmapzip.geodata

Documents Aggregations Schema Explain Plan Indexes Validation

DOCUMENTS 14.4k TOTAL SIZE 3.7MB AVG. SIZE 268B INDEXES 4 TOTAL SIZE 756.0KB AVG. SIZE 189.0KB

ADD DATA FILTER FIND RESET ...

Displaying documents 1 - 20 of 10194 < > C REFRESH

`_id: ObjectId("5fca8fb069695b79205d0c34")
> geometry: Object
> properties: Object
 business_id: ObjectId("5fc65019b6df553f75b0f9e6")
 name: "이학갈비"
 tel: "031-392-2095"
 address: "경기도 군포시 산본동 1123-3번지 센트로 601호"
 category: "한식"`

`_id: ObjectId("5fca8fb069695b79205d0c36")
> geometry: Object
> properties: Object`

`_id: ObjectId("5fca8fb069695b79205d0c37")
> geometry: Object
> properties: Object`

`_id: ObjectId("5fca8fb069695b79205d0c38")
> geometry: Object
> properties: Object`

`_id: ObjectId("5fca8fb069695b79205d0c3a")
> geometry: Object
> properties: Object`

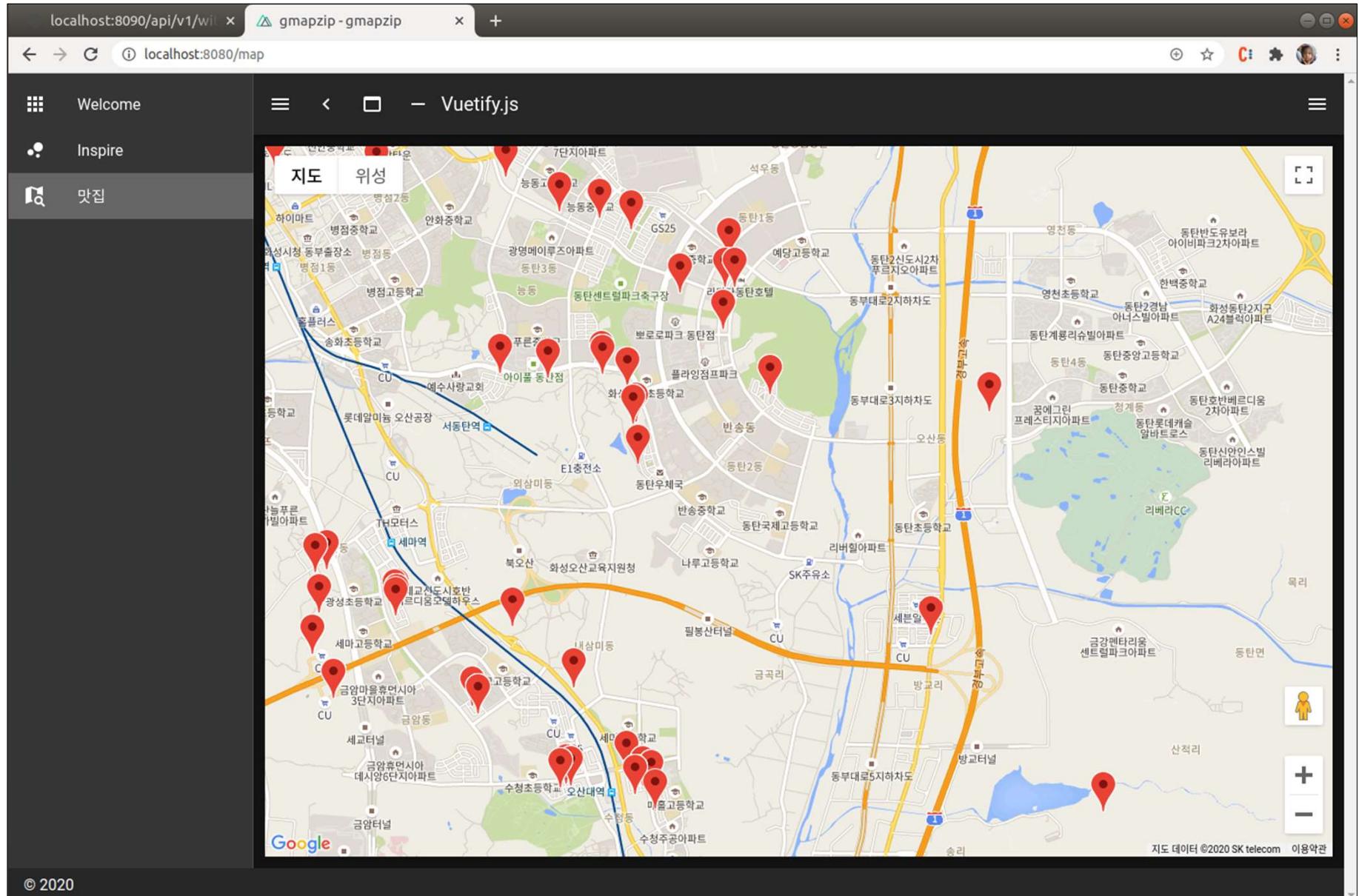
`_id: ObjectId("5fca8fb069695b79205d0c3b")
> geometry: Object
> properties: Object`

`_id: ObjectId("5fca8fb069695b79205d0c3c")`

>_MongoSH Beta

```
localhost:8090/api/v1/wit x gmapzip - gmapzip
localhost:8090/api/v1/within?left=127.03168&lower=37.18089&right=127.09237&upper=37.21994

[{"properties": {"business_id": "5fc65019b6df553f75b0fa8d", "name": "정답명가남원추어탕", "tel": "031-265-3651", "address": "경기도 오산시 외삼미동 128-3번지", "category": "한식"}, "id": "5fc8fb069695b79205d0cdb", "geometry": {"coordinates": [127.0624150214, 37.198990105], "id": "5fc67ea5eee3e00130139b0", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fa8e", "name": "보나카비", "tel": "031-377-5777", "address": "경기도 오산시 외삼미동 119-12번지", "category": "서양식"}, "id": "5fc67ea5eee3e00130139b1", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fa8f", "name": "조선화로집(서동탄점)", "tel": "031-613-9254", "address": "경기도 오산시 외삼미동 127-6번지 1층", "category": "한식"}, "id": "5fc8fb069695b79205d0cdd", "geometry": {"coordinates": [127.065105775, 37.1954755165], "id": "5fc67ea5eee3e00130139b1", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fa9", "name": "도쿄나하 동탄점", "tel": "031-2819-6870", "address": "경기도 화성시 능동 468-2번지", "category": "일식"}, "id": "5fc8fb069695b79205d0ce8", "geometry": {"coordinates": [127.0537229974, 37.2130916972], "id": "5fc67ea5eee3e00130139b4", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fa9c", "name": "경복궁동탄점(주)덕용푸드시스템", "tel": "", "address": "경기도 화성시 능동 976-2번지", "category": "한식"}, "id": "5fc8fb069695b79205d0cea", "geometry": {"coordinates": [127.057469757, 37.1987283558], "id": "5fc67ea5eee3e00130139b5", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fa9d", "name": "청도해물탕", "tel": "", "address": "경기도 화성시 반송동 120-2번지", "category": "한식"}, "id": "5fc8fb069695b79205d0ceb", "geometry": {"coordinates": [127.0774210467, 37.1975752911], "id": "5fc67ea5eee3e00130139b6", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fa1", "name": "포메인 동탄점", "tel": "031-8015-3220", "address": "경기도 화성시 반송동 86-4번지", "category": "기타외국식"}, "id": "5fc8fb069695b79205d0cef", "geometry": {"coordinates": [127.0736926124, 37.2073602091], "id": "5fc67ea5eee3e00130139b7", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fa3", "name": "대개나라동탄점", "tel": "031-8003-6262", "address": "경기도 화성시 반송동 92-7번지", "category": "기타 음식점점업"}, "id": "5fc8fb069695b79205d0cf1", "geometry": {"coordinates": [127.0731918872, 37.2022230977], "id": "5fc67ea5eee3e00130139b8", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fa9a", "name": "송도갈매기동탄점", "tel": "031-238-0022", "address": "경기도 화성시 능동 468-2번지", "category": "한식"}, "id": "5fc8fb069695b79205d0cf5", "geometry": {"coordinates": [127.0537229974, 37.2130916972], "id": "5fc67ea5eee3e00130139b9", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fb64", "name": "아웃백스테이크하우스 동탄", "tel": "031-8015-4261", "address": "경기도 화성시 반송동 96번지 메타폴리스", "category": "서양식"}, "id": "5fc8fb069695b79205d0db2", "geometry": {"coordinates": [127.0693116576, 37.204865499], "id": "5fc67ea5eee3e00130139b9", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fb68", "name": "본수원갈비 병점점", "tel": "050713488434", "address": "경기도 화성시 진안동 453-7번지", "category": "한식"}, "id": "5fc8fb069695b79205d0db6", "geometry": {"coordinates": [127.0329811028, 37.2137959534], "id": "5fc67ea5eee3e00130139bb", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fbcc", "name": "영천갈비", "tel": "", "address": "경기도 화성시 진안동 882-4번지", "category": "한식"}, "id": "5fc8fb069695b79205d0db9", "geometry": {"coordinates": [127.04202356868, 37.2130741292], "id": "5fc67ea5eee3e00130139bc", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fb6d", "name": "영천두툼한숯불갈비", "tel": "031-224-4421", "address": "경기도 화성시 진안동 882-5번지", "category": "한식"}, "id": "5fc8fb069695b79205d0dbd", "geometry": {"coordinates": [127.0421040703, 37.2129481972], "id": "5fc67ea5eee3e00130139bd", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fb70", "name": "착한고기동탄광장점", "tel": "050714210008", "address": "경기도 화성시 반송동 91-6번지", "category": "한식"}, "id": "5fc8fb069695b79205d0dbe", "geometry": {"coordinates": [127.0733662536, 37.2052767892], "id": "5fc67ea5eee3e00130139be", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fb7d", "name": "호아ונג", "tel": "031-374-9934", "address": "경기도 오산시 세교동 509-8번지", "category": "기타 음식점점업"}, "id": "5fc8fb069695b79205d0cf", "geometry": {"coordinates": [127.0366290661, 37.1849022351], "id": "5fc67ea5eee3e00130139bt", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fb7f", "name": "보나카비", "tel": "031-377-5777", "address": "경기도 오산시 외삼미동 119-12번지", "category": "서양식"}, "id": "5fc8fb069695b79205d0cd", "geometry": {"coordinates": [127.065105775, 37.1954755165], "id": "5fc67ea5eee3e00130139co", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ffcd", "name": "취풍더", "tel": "031-366-1999", "address": "경기도 오산시 세교동 436번지", "category": "한식"}, "id": "5fc8fb069695b79205d0eb", "geometry": {"coordinates": [127.0376309113, 37.18504136], "id": "5fc67ea5eee3e00130139cl", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fe3f", "name": "메밀사랑夙사랑오산세교점", "tel": "031-373-9984", "address": "경기도 오산시 세교동 613-9번지", "category": "한식"}, "id": "5fc8fb069695b79205d108d", "geometry": {"coordinates": [127.0437537701, 37.1822887617], "id": "5fc67ea5eee3e00130139c2", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0feee", "name": "보나카바", "tel": "031-377-5777", "address": "경기도 오산시 외삼미동 119-12번지", "category": "서양식"}, "id": "5fc8fb069695b79205d113c", "geometry": {"coordinates": [127.065105775, 37.1954755165], "id": "5fc67ea5eee3e00130139c3", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0feff8", "name": "예잔", "tel": "031-375-2552", "address": "경기도 오산시 세교동 470번지", "category": "한식"}, "id": "5fc8fb069695b79205d1146", "geometry": {"coordinates": [127.0370076234, 37.1819471], "id": "5fc67ea5eee3e00130139c4", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ff69", "name": "아웃백스테이크하우스 동탄", "tel": "031-8015-4261", "address": "경기도 화성시 반송동 96번지 메타폴리스", "category": "한식"}, "id": "5fc8fb069695b79205d11b7", "geometry": {"coordinates": [127.0693116576, 37.204865499], "id": "5fc67ea5eee3e00130139c5", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0fffd", "name": "본수원갈비 병점점", "tel": "050713488434", "address": "경기도 화성시 진안동 453-7번지", "category": "한식"}, "id": "5fc8fb069695b79205d11bb", "geometry": {"coordinates": [127.0329811028, 37.2137959534], "id": "5fc67ea5eee3e00130139c6", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ff72", "name": "영천두툼한숯불갈비", "tel": "031-224-4421", "address": "경기도 화성시 진안동 882-5번지", "category": "한식"}, "id": "5fc8fb069695b79205d11c0", "geometry": {"coordinates": [127.0421040703, 37.2129481972], "id": "5fc67ea5eee3e00130139c7", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ff88", "name": "정답명가남원추어탕", "tel": "031-265-3651", "address": "경기도 오산시 외삼미동 128-3번지", "category": "한식"}, "id": "5fc8fb069695b79205d11fd", "geometry": {"coordinates": [127.0624150214, 37.198990105], "id": "5fc67ea5eee3e00130139c8", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ffbf1", "name": "황제해물찜스토리", "tel": "031-613-0361", "address": "경기도 오산시 외삼미동 119-12", "category": "한식"}, "id": "5fc8fb069695b79205d11ff", "geometry": {"coordinates": [127.0653868957, 37.1955757028], "id": "5fc67ea5eee3e00130139c9", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ffbf", "name": "정답명가남원추어탕", "tel": "031-265-3651", "address": "경기도 오산시 외삼미동 128-3번지", "category": "한식"}, "id": "5fc8fb069695b79205d120a", "geometry": {"coordinates": [127.0537229974, 37.2130916972], "id": "5fc67ea5eee3e00130139ca", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ffbe", "name": "동신참치", "tel": "", "address": "경기도 화성시 반송동 90-8번지", "category": "일식"}, "id": "5fc8fb069695b79205d120c", "geometry": {"coordinates": [127.0742455419, 37.2052258684], "id": "5fc67ea5eee3e00130139cb", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ffbf", "name": "경복궁동탄점(주)덕용푸드시스템", "tel": "", "address": "경기도 화성시 능동 976-2번지", "category": "한식"}, "id": "5fc8fb069695b79205d120d", "geometry": {"coordinates": [127.057469757, 37.1987283558], "id": "5fc67ea5eee3e00130139cc", "type": "Point"}, {"properties": {"business_id": "5fc65019b6df553f75b0ffcc", "name": "곤드레밥집", "tel": "", "address": "경기도 화성시 반송동 76-3번지", "category": "한식"}, "id": "5fc8fb069695b79205d120e", "geometry": {"coordinates": [127.0622553945, 37.1993140213], "id": "5fc67ea5eee3e00130139cd", "type": "Point"}]
```



<https://github.com/whatwant-school/gmapzip>

지금까지는 **Local**에서
Docker로 구성은 했지만,
이제는
Kubernetes를 사용해보자!!!