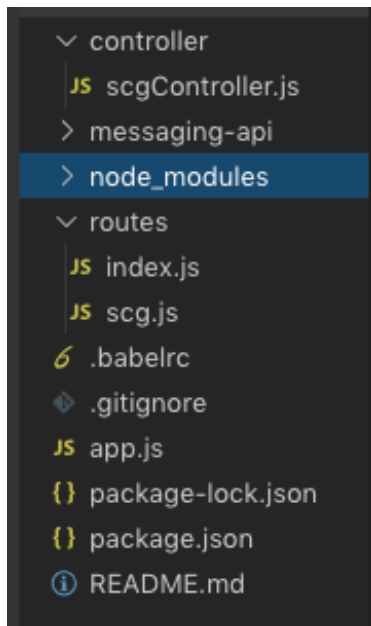


# Back-End API

**github:** <https://github.com/cartoon-s/backend.git>



## โครงสร้าง structure

- 1) controller (เก็บ logic code ต่างๆ ของแต่ละ controller)
- 2) routes (Path เพื่อเรียกไปยังแต่ละ controller)
- 3) app.js (หน้าหลัก)

## หน้าหลัก (app.js)

```
JS app.js > [x] bodyParser
1  import routes from './routes';
2
3  const express = require('express')
4  const app = express()
5  const cors = require('cors');
6  const path = require('path');
7
8  // body parser module
9  // to pass json to api server
10 const bodyParser = require('body-parser')
11
12 app.use(cors());
13
14 app.use(express.static(path.join(__dirname, 'build')));
15
16 app.use('/.well-known', express.static('.well-known'));
17
18 app.use(bodyParser.json())
19
20 app.use(bodyParser.urlencoded({ extended: true }))
21
22 app.use(function (req, res, next) {
23   res.header('Cache-Control', 'private, no-cache, no-store, must-revalidate');
24   res.header('Expires', '-1');
25   res.header('Pragma', 'no-cache');
26   next()
27 });
28
29 app.use('/api/scg', routes.scg);
30
31 // listening to port 3000 (localhost)
32 app.listen(3001, () => {
33   console.log('Start server at port 3001.')
34 })
35
```

Import routes ต่างๆ

บอกว่า path นี้ไปที่ไหน

## Routes folder

— หน้า index.js และ scg.js

### Index.js ของ folder routes

```
1 import scg from './scg';
2 export default {
3   scg
4 };
```

Import และ export scg routes

### scg.js ของ folder routes

```
routes > JS scg.js > ...
1 const express = require('express');
2 const router = express.Router()
3 var request = require('request');
4 var _ = require('lodash');
5 var scgController = require('../controller/scgController');
6
7 router.get('/findxyz', scgController.findParameter);
8 router.get('/findingRestaurant', scgController.findResByPlace);
9
```

Path แต่ละ Function

## Controller

— scg.js

มีสองฟังก์ชันประกอบไปด้วย 1) findParameter (ค้นหาค่า xyz แล้ว return ค่ากลับ), 2) findResByPlace (ยิง api search place of google แล้วส่งค่าร้านอาหารในบางชื่อกลับมา)



```
controller > JS scgController.js > ...
1  var request = require('request');
2  var _ = require('lodash');
3
4  > exports.findParameter = function (req, res) { ...
52  };
53
54  > exports.findResByPlace = function (req, res) { ...
100 }
```

### \* Function findParameter()



```
controller > JS scgController.js > findParameter > findParameter
exports.findParameter = function (req, res) {
  const arr = ['x', 5, 9, 15, 23, 'y', 'z'];
  let mul = 2; let indexOfX; let indexOfY; let indexOfZ;
  arr.find(function (value, index) {
    if (value === 'x') {
      indexOfX = index;
      if (indexOfX > 0) {
        let value = arr[indexOfX - 1] + (mul * indexOfX);
        arr[indexOfX] = value;
      } else {
        let value = arr[indexOfX + 1] - (mul * (indexOfX + 1));
        arr[indexOfX] = value;
      }
    }
    if (value === 'y') {
      indexOfY = index;
      if (indexOfY > 0) {
        let value = arr[indexOfY - 1] + (mul * indexOfY);
        arr[indexOfY] = value;
      } else {
        let value = arr[indexOfY + 1] - (mul * (indexOfY + 1));
        arr[indexOfY] = value;
      }
    }
    if (value === 'z') {
      indexOfZ = index;
      if (indexOfZ > 0) {
        let value = arr[indexOfZ - 1] + (mul * indexOfZ);
        arr[indexOfZ] = value;
      } else {
        let value = arr[indexOfZ + 1] - (mul * (indexOfZ + 1));
        arr[indexOfZ] = value;
      }
    }
  })
  let data = {
    'x': arr[indexOfX],
    'y': arr[indexOfY],
    'z': arr[indexOfZ]
  }
  return res.send({ result: data });
};
```

นำค่าทั้งหมดมาทำการวนลูปเช็คค่าทุกคน เพื่อค้นหา Index ของแต่ละตัว จากนั้นนำ index มาคำนวณค่าของ x y z ตามสมการ

$$T_n = T(n-1) + 2(n)$$

โดยที่ n คือ index

แต่หากว่า index = 0 จะใช้สมการ

$$T_n = T(n+1) - 2(n+1)$$

จากนั้นเมื่อได้ค่าของ x y z ทั้งหมดก็ return ค่ากลับไปเป็น object

## **\*\* Function findResByPlace()**

```
exports.findResByPlace = function (req, res) {
  let YOUR_API_KEY = 'AIzaSyBSiLVNYMbdKdh-Ika4Zvn0CztHWhItYKw';
  let lat = 13.810577;
  let lng = 100.530278;
  let url = `https://maps.googleapis.com/maps/api/place/textsearch/json?query=restaurants+in+BangSue&type=restaurant&key=${YOUR_API_KEY}`;
  let data = [];
  request(url, function (error, response, body) {
    if (error) {
      throw error; //error happen
    }
    var obj = JSON.parse(body);
    for (let i of obj.results) {
      console.log(i.name);
      let temp = {
        vicinity: i.formatted_address,
        rating: i.rating,
        name: i.name
      }
      data.push(temp);
    }
  })
  let url2 = `https://maps.googleapis.com/maps/api/place/textsearch/json?location=${lat},${lng}&type=restaurant&keyword=Bang Sue&key=${YOUR_API_KEY}`;
  request(url2, function (error, response, body) {
    if (error) {
      console.log(error);
      throw error; //error happen
    }
    var obj = JSON.parse(body);
    for (let i of obj.results) {
      console.log(i.name);
      let temp = {
        vicinity: i.formatted_address,
        rating: i.rating,
        name: i.name
      }
      data.push(temp);
    }
  })
  return res.send(_.uniqBy(data, 'name'));
});
```

ทำการค้นหาร้านอาหารในบางซื่อ โดยการยิงไปที่ api ของ google (search place) โดยทำการยิงไปที่ search place 2 รอบ (เนื่องจาก Limited ของ api เมื่อใช้บริการฟรี response ที่ได้กลับมามีแค่ 20 ร้านเท่านั้น) เพื่อให้ได้ร้านในบางซื่อมากกว่า 20 ร้าน จากนั้นนำร้านแต่ละร้านที่ได้มาคัดเลือกร้านที่ซ้ำออก แล้ว return ร้านค้าทั้งหมดที่มีกลับออกมา