

**TRƯỜNG ĐẠI HỌC CẦN THƠ**  
**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**



**BÁO CÁO THỰC HÀNH**  
**HỌC PHẦN:**  
**PHÁT TRIỂN ỨNG DỤNG WEB**  
**CT449 NHÓM 05**

\*\*\*

**HỌ TÊN:** NGUYỄN THANH TÍN  
**MSSV:** B1910312  
**NHÓM:** 01

**Giáo viên hướng dẫn:**  
Nguyễn Minh Trung

# Buổi 1

## Ứng dụng Contactbook - Backend - Phần 1

### Bước 1: Tạo ứng dụng Node

- Tạo thư mục dự án, ví dụ "contactbook-backend".
- Trong thư mục này, ta khởi tạo ứng dụng Node:

#### ○ Terminal

```
package name: (contactbook-backend) contactbook-backend
version: (1.0.0)
description:
entry point: (index.js) server.js
test command:
git repository:
keywords:
author: B1910312 Nguyen Thanh Tin
license: (ISC)
About to write to D:\Phat trien ung dung web\contactbook-backend\package.json:

{
  "name": "contactbook-backend",
  "version": "1.0.0",
  "description": "",
  "main": "server.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "B1910312 Nguyen Thanh Tin",
  "license": "ISC"
}

Is this OK? (yes) █
```

#### ○ File package.json

```
{ } package.json > ...
1  {
2    "name": "contactbook-backend",
3    "version": "1.0.0",
4    "description": "",
5    "main": "server.js",
6    "scripts": {
7      "test": "echo \"Error: no test specified\" && exit 1"
8    },
9    "author": "B1910312 Nguyen Thanh Tin",
10   "license": "ISC"
11 }
12
```

- Cài đặt các package cần thiết:

```
PS D:\Phat trien ung dung web\contactbook-backend> npm install express cors
added 59 packages, and audited 60 packages in 5s

7 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS D:\Phat trien ung dung web\contactbook-backend>
```

## Bước 2: Quản lý mã nguồn dự án với git và GitHub

- Tạo tập tin .gitignore:

```
PS D:\Phat trien ung dung web\contactbook-backend> npx gitignore node
Need to install the following packages:
  gitignore@0.7.0
Ok to proceed? (y) y
Created .gitignore file for flag type node.
PS D:\Phat trien ung dung web\contactbook-backend> S
```

- chạy lệnh git init . Gõ tiếp git status

```
PS D:\Phat trien ung dung web\contactbook-backend> git init
Initialized empty Git repository in D:/Phat trien ung dung web/contactbook-backend/.git/
PS D:\Phat trien ung dung web\contactbook-backend> git status
On branch master

No commits yet

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        .gitignore
        package-lock.json
        package.json

nothing added to commit but untracked files present (use "git add" to track)
PS D:\Phat trien ung dung web\contactbook-backend>
```

- Cho git quản lý 3 tập tin .gitignore, package-lock.json và package.json, sau đó kiểm tra lại

```
PS D:\Phat trien ung dung web\contactbook-backend> git add .gitignore package-lock.json package.json
warning: in the working copy of '.gitignore', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it
PS D:\Phat trien ung dung web\contactbook-backend> git status
On branch master

No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   .gitignore
        new file:   package-lock.json
        new file:   package.json
```

- Thực thi git commit -m "Cai dat du an" để lưu các thay đổi. Tùy chọn -m trong câu lệnh cho phép tạo ghi chú cho commit này.

```
PS D:\Phat trien ung dung web\contactbook-backend> git commit -m "Cai dat du an"
[master (root-commit) dfb657f] Cai dat du an
3 files changed, 1198 insertions(+)
 create mode 100644 .gitignore
 create mode 100644 package-lock.json
 create mode 100644 package.json
PS D:\Phat trien ung dung web\contactbook-backend> █
```

- Đăng ký tài khoản và tạo dự án trên GitHub.

```
PS D:\Phat trien ung dung web\contactbook-backend> git remote add origin https://github.com/b1910312/b1910312-NguyenThanhTin-Buoil.git
PS D:\Phat trien ung dung web\contactbook-backend> git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 8.43 KiB | 2.11 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
remote:
remote: Create a pull request for 'master' on GitHub by visiting:
remote:   https://github.com/b1910312/b1910312-NguyenThanhTin-Buoil/pull/new/master
remote:
To https://github.com/b1910312/b1910312-NguyenThanhTin-Buoil.git
 * [new branch]      master -> master
PS D:\Phat trien ung dung web\contactbook-backend> █
```

- Link GitHub: [https://github.com/b1910312/B1910312\\_NguyenThanhTin\\_ContactBook.git](https://github.com/b1910312/B1910312_NguyenThanhTin_ContactBook.git)
- Thực hiện chứng thực tài khoản và kiểm tra lại các tập tin đã được tải lên GitHub:

	<b>b1910312</b> Cai dat du an	e832844 1 minute ago	🕒 1 commit
	.gitignore	Cai dat du an	1 minute ago
	package-lock.json	Cai dat du an	1 minute ago
	package.json	Cai dat du an	1 minute ago

### Bước 3: Cài đặt Express

- Trong thư mục gốc của dự án, tạo tập tin app.js và server.js
  - o App.js

```
JS app.js U X JS server.js U ●
JS app.js > ...
1 const express = require("express");
2 const cors = require("cors");
3 const res = require("express/lib/response");
4
5 const app = express();
6
7 app.use(cors());
8 app.use(express.json());
9
10 app.get("/", (req, res) => {
11   res.json({ message: "Welcome to contact book application." });
12 });
13
14 module.exports = app;
```

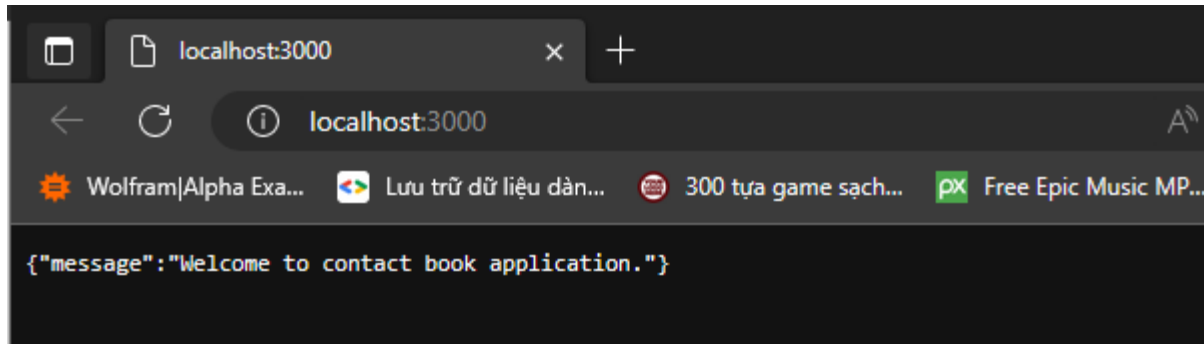
- Server.js

```
JS app.js U JS server.js U X
JS server.js > ...
1 const app = require("./app");
2 const config = require("./app/config");
3
4 //start server
5 const PORT = config.app.port;
6 app.listen(PORT, () => {
7   console.log(`Server is running on port ${PORT}.`);
8 });
```

- Lần lượt tạo thư mục app, app/config và tập tin app/config/index.js

```
JS app.js U X JS server.js U JS index.js U ●
app > config > JS index.js > ...
1 const config = {
2   app: {
3     port: process.env.PORT || 3000,
4   }
5 };
6 module.exports = config;
7
```

- thực hiện lệnh chạy server: node server.js . Truy cập <http://localhost:3000/> để kiểm tra kết quả.



- cài đặt thêm gói hỗ trợ nodemon

```
Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS D:\Phat trien ung dung web\contactbook-backend> npm install nodemon --save-dev

changed 33 packages, and audited 92 packages in 8s

10 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS D:\Phat trien ung dung web\contactbook-backend> █
```

- mở tập tin package.json và thay đổi cấu hình mục "scripts"

```
1 package.json > ...
2 {
3   "name": "contactbook-backend",
4   "version": "1.0.0",
5   "description": "",
6   "main": "server.js",
7   "scripts": {
8     "start": "nodemon server.js"
9   },
10  "author": "B1910312 Nguyen Thanh Tin",
11  "license": "ISC",
12  "dependencies": {
13    "cors": "^2.8.5",
14    "express": "^4.18.2"
15  },
16  "devDependencies": {
17    "nodemon": "^2.0.20"
18  }
19 }
```

- Lưu các thay đổi trên vào git:

```
PS D:\Phat trien ung dung web\contactbook-backend> git add -u
warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it
PS D:\Phat trien ung dung web\contactbook-backend> git add app/ app.js server.js
PS D:\Phat trien ung dung web\contactbook-backend> git commit -m "Cai dat express chay thong bao xin chao"
[master 777cc9a] Cai dat express chay thong bao xin chao
5 files changed, 684 insertions(+), 14 deletions(-)
create mode 100644 app.js
create mode 100644 app/config/index.js
create mode 100644 server.js
PS D:\Phat trien ung dung web\contactbook-backend> █
```

## Bước 4: Định nghĩa Controller và các route

- Trong thư mục app/controllers, tạo tập tin contact.controller.js

```
app > controllers > JS contact.controller.js > ...
1  exports.create = (req, res) => {
2    res.send({ message: "create handler" });
3  };
4  exports.findAll = (req, res) => {
5    res.send({ message: "findAll handler" });
6  };
7  exports.findOne = (req, res) => {
8    res.send({ message: "findOne handler" });
9  };
10 exports.update = (req, res) => {
11   res.send({ message: "update handler" });
12 };
13 exports.delete = (req, res) => {
14   res.send({ message: "delete handler" });
15 };
16 exports.deleteAll = (req, res) => {
17   res.send({ message: "deleteAll handler" });
18 };
19 exports.findAllFavorite = (req, res) => {
20   res.send({ message: "findAllFavorite handler" });
21 };
22
```

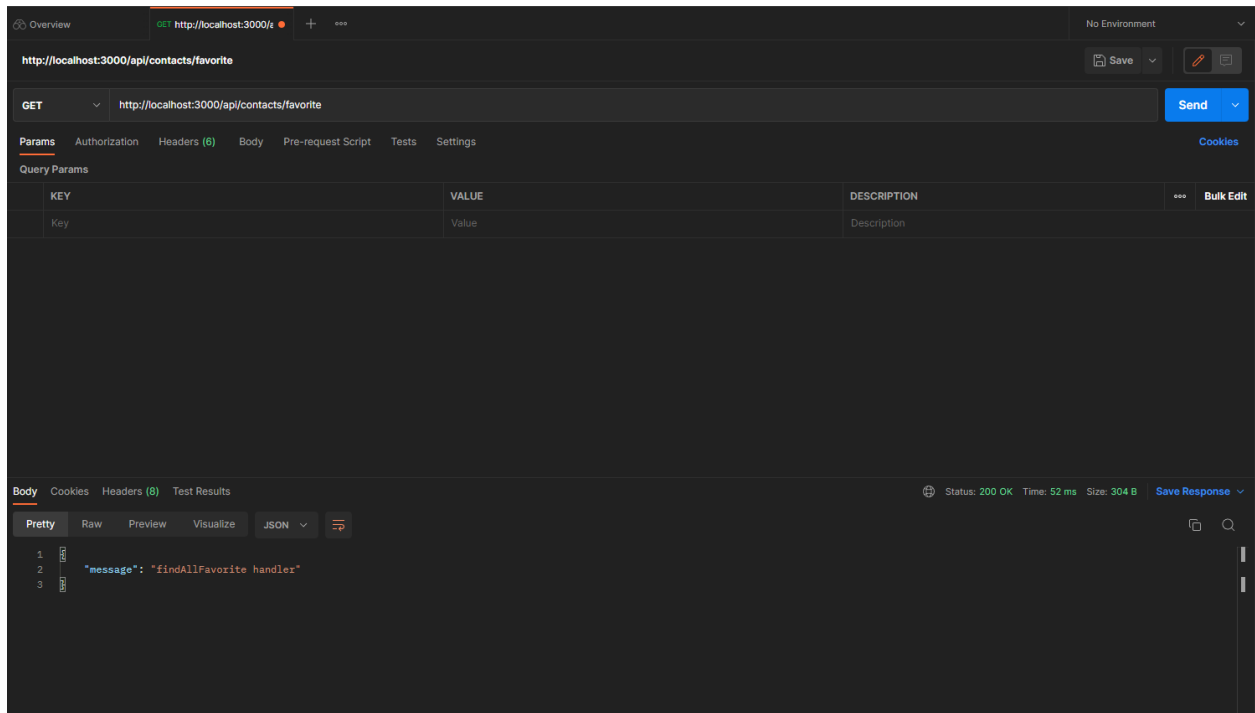
- Trong thư mục app/routes, tạo tập tin contact.route.js

```
app > routes > JS contact.route.js > ...
1  const express = require("express");
2  const contacts = require("../controllers/contact.controller");
3  const router = express.Router();
4
5  router.route("/")
6    .get(contacts.findAll)
7    .post(contacts.create)
8    .delete(contacts.deleteAll);
9
10 router.route("/favorite")
11   .get(contacts.findAllFavorite);
12
13 router.route("/:id")
14   .get(contacts.findOne)
15   .put(contacts.update)
16   .delete(contacts.delete);
17
18 module.exports = router;
```

- đăng ký các route vừa tạo với express app.

```
JS app.js > ...
1  const express = require("express");
2  const cors = require("cors");
3  const contactsRouter = require("./app/routes/contact.route");
4  const app = express();
5
6  app.use(cors());
7  app.use(express.json());
8  app.use("/api/contacts", contactsRouter);
9
10 app.get("/", (req, res) => {
11   res.json({ message: "Welcome to contact book application." });
12 });
13 module.exports = app;
```

- Sử dụng ứng dụng Postman hoặc curl để kiểm tra các route vừa tạo.
  - o findAllFavorite



- lưu các thay đổi lên git:

```
PS D:\Phat trien ung dung web\contactbook-backend> git add -u
PS D:\Phat trien ung dung web\contactbook-backend> git add app/controllers app/routes
PS D:\Phat trien ung dung web\contactbook-backend> git commit -m "Định nghĩa các route cho tài nguyên Contact"
[master aec1745] Định nghĩa các route cho tài nguyên Contact
3 files changed, 41 insertions(+), 1 deletion(-)
create mode 100644 app/controllers/contact.controller.js
create mode 100644 app/routes/contact.route.js
```

## Bước 5: Cài đặt xử lý lỗi

- Tạo tập tin `app/api-error.js`:

```
app > JS api-error.js > ...
1 class ApiError extends Error {
2   constructor(statusCode, message) {
3     super();
4     this.statusCode = statusCode;
5     this.message = message;
6   }
7 }
8 module.exports = ApiError;
```

- Mở tập tin `app.js` và thêm vào các middleware xử lý lỗi



```
app.js > ...
8 |
9 | app.use(cors());
10 | app.use(express.json());
11 | app.use("/api/contacts", contactsRouter);
12 | // handle 404 response
13 | app.use((req, res, next) => {
14 |   // Code ở đây sẽ chạy khi không có route được định nghĩa nào
15 |   // khớp với yêu cầu. Gọi next() để chuyển sang middleware xử lý lỗi
16 |   return next(new ApiError(404, "Resource not found"));
17 | });
18 | // define error-handling middleware last, after other app.use() and routes calls
19 | app.use((err, req, res, next) => {
20 |   // Middleware xử lý lỗi tập trung.
21 |   // Trong các đoạn code xử lý ở các route, gọi next(error)
22 |   // sẽ chuyển về middleware xử lý lỗi này
23 |   return res.status(error.statusCode || 500).json({
24 |     message: error.message || "Internal Server Error",
25 |   });
26 | });
27 | app.get("/", (req, res) => {
28 |   res.json({ message: "Welcome to contact book application." });
29 | });
30 | module.exports = app;
```

- Sử dụng một HTTP client (Chrome, Postman, curl, ...) và gửi yêu cầu đến một URL không được định nghĩa trong ứng dụng, kiểm tra nhận được lỗi 404 và thông báo "Resource not found".

The screenshot shows a Postman interface with a GET request to `http://localhost:3000/api/contacts/b1910312`. The response status is `500 Internal Server Error`. The response body is displayed in HTML format, showing a detailed error message from the application.

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>Error</title>
</head>
<body>
  <pre>ReferenceError: error is not defined: <br> <nbsp;<br>at D:\Phat trien ung dung web\contactbook-backend\app.js:23:23<br> <nbsp;<br>at Layer.handle_error (D:\Phat trien ung dung web\contactbook-backend\node_modules\express\lib\router\layer.js:71:5)<br> <nbsp;<br>at trim_prefix (D:\Phat trien ung dung web\contactbook-backend\node_modules\express\lib\router\index.js:326:13)<br> <nbsp;<br>at D:\Phat trien ung dung web\contactbook-backend\node_modules\express\lib\router\index.js:286:9<br> <nbsp;<br>at Function.process_params (D:\Phat trien ung dung web\contactbook-backend\node_modules\express\lib\router\index.js:346:12)<br> <nbsp;<br>at next (D:\Phat trien ung dung web\contactbook-backend\node_modules\express\lib\router\index.js:288:10)<br> <nbsp;<br>at D:\Phat trien ung dung web\contactbook-backend\app.js:16:12<br> <nbsp;<br>at Layer.handle [as handle_request] (D:\Phat trien ung dung web\contactbook-backend\node_modules\express\lib\router\layer.js:95:5)<br> <nbsp;<br>at trim_prefix (D:\Phat trien ung dung web\contactbook-backend\node_modules\express\lib\router\index.js:286:9</pre>
```

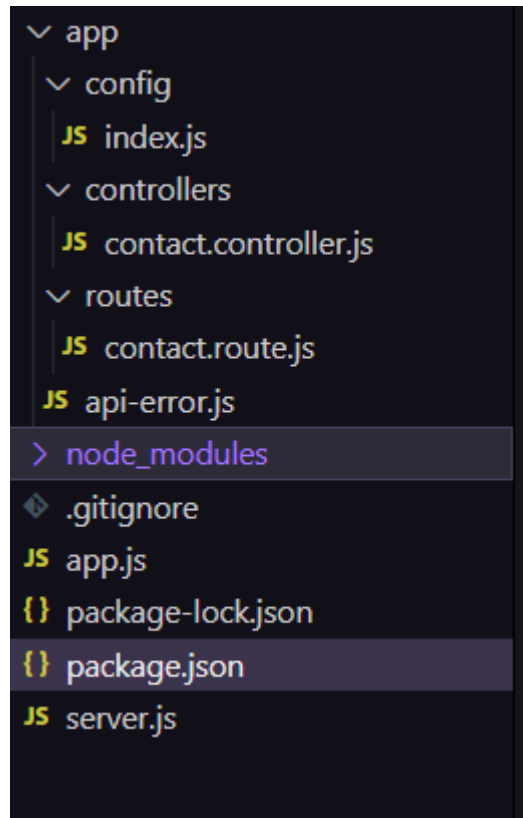
- Lưu các thay đổi vào git:

```
PS D:\Phat trien ung dung web\contactbook-backend> git add -u
PS D:\Phat trien ung dung web\contactbook-backend> git add app/api-error.js
PS D:\Phat trien ung dung web\contactbook-backend> git commit -m "Cai dat xu ly loi"
[master 1797a67] Cai dat xu ly loi
2 files changed, 26 insertions(+), 1 deletion(-)
create mode 100644 app/api-error.js
PS D:\Phat trien ung dung web\contactbook-backend>
```

- Đẩy các thay đổi lên GitHub bằng lệnh:

```
PS D:\Phat trien ung dung web\contactbook-backend> git push origin master
Enumerating objects: 8, done.
Counting objects: 100% (8/8), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 1.12 KiB | 575.00 KiB/s, done.
Total 5 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (1/1), completed with 1 local object.
To https://github.com/b1910312/B1910312_NguyenThanhTin_ContactBook.git
   aec1745..1797a67  master -> master
PS D:\Phat trien ung dung web\contactbook-backend> █
```

- Cấu trúc thư mục dự án đến thời điểm này sẽ như sau:



## Buổi 2

### Ứng dụng Contactbook - Backend - Phần 2

#### Bước 1: Cài đặt thư viện mongodb, định nghĩa hàm trợ giúp kết nối và lớp dịch vụ truy xuất cơ sở dữ liệu (CSDL)

- Cài đặt thư viện mongodb vào dự án:

```
PS D:\Phat trien ung dung web\contactbook-backend> npm install mongodb

added 16 packages, and audited 108 packages in 11s

10 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
PS D:\Phat trien ung dung web\contactbook-backend> █
```

- Trong thư mục app/config, hiệu chỉnh tập tin index.js:

```
const config = {}
  app: {
    port: process.env.PORT || 3000,
  },
  db: {
    uri: process.env.MONGODB_URI || "mongodb://127.0.0.1:27017/contactbook"
  }
};
module.exports = config;
```

- Định nghĩa lớp trợ giúp kết nối đến MongoDB app/utils/mongodb.util.js:

```
app > utils > JS mongodb.util.js > ...
1  const { MongoClient } = require("mongodb");
2
3  class MongoDB {
4    static connect = async(uri) => {
5      if (this.client) return this.client;
6      this.client = await MongoClient.connect(uri);
7      return this.client;
8    };
9  }
10
11 module.exports = MongoDB;
```

- Thực hiện kết nối đến CSDL MongoDB khi chạy server, thay toàn bộ nội dung tập tin server.js

```

JS server.js > ...
1  const app = require("./app");
2  const config = require("./app/config");
3  const MongoDB = require("./app/utis/mongodb.util");
4
5  async function startServer() {
6      try {
7          await MongoDB.connect(config.db.uri);
8          console.log("Connectd to the database!");
9
10         const PORT = config.app.port;
11         app.listen(PORT, () => {
12             console.log(`server is running on port ${PORT}`);
13         });
14     } catch (error) {
15         console.log("Cannot connect to the database!", error);
16         process.exit();
17     }
18 }
19
20 startServer();

```

- Định nghĩa lớp dịch vụ ContactService (trong tập tin app/services/contact.service.js) sử dụng các API của thư viện mongodb để thực hiện các thao tác với CSDL MongoDB:

```

app > services > JS contact.service.js > ...
1  const { ObjectId } = require("mongodb");
2  class ContactService {
3      constructor(client) {
4          this.Contact = client.db().collection("contacts");
5      }
6      // Định nghĩa các phương thức truy xuất CSDL sử dụng mongodb API
7  }
8  module.exports = ContactService;

```

- Kết nối server thành công

```

PS D:\Phat trien ung dung web\contactbook-backend> npm start

> contactbook-backend@1.0.0 start
> node server.js

Connectd to the database!
server is running on port 3000

```

## Bước 2: Cài đặt các handler

### Cài đặt handler create

- Hiệu chỉnh tập tin app/controllers/contact.controller.js:

```

app > controllers > JS contact.controller.js
1 exports.create = async(req, res, next) => {
2   if (!req.body?.name) {
3     return next(new ApiError(400, "Name can not be empty"));
4   }
5
6   try {
7     const contactService = new ContactService(MongoDB.client);
8     const document = await contactService.create(req.body);
9     return res.send(document);
10  } catch (error) {
11    return next(
12      new ApiError(500, "An error occurred while creating the contact")
13    );
14  }
15 };
16 exports.findAll = (req, res) => {
17   res.send({ message: "findAll handler" });
18 };

```

- Lỗi gọi `contactService.create()` lưu thông tin đối tượng contact xuống CSDL. Phương thức `create()` được định nghĩa trong lớp `ContactService` (`app/services/contact.service.js`)

```

app > services > JS contact.service.js > ContactService > create
1 const { ObjectId } = require("mongodb");
2 class ContactService {
3   constructor(client) {
4     this.Contact = client.db().collection("contacts");
5   }
6   // Định nghĩa các phương thức truy xuất CSDL sử dụng mongodb API
7   extractContactData(payload) {
8     const contact = {
9       name: payload.name,
10      email: payload.email,
11      address: payload.address,
12      phone: payload.phone,
13      favorite: payload.favorite,
14    };
15    Object.keys(contact).forEach(
16      (key) => contact[key] === undefined && delete contact[key]
17    );
18    return contact;
19  }
20  async create(payload) {
21    const contact = this.extractContactData(payload);
22    const result = await this.Contact.findOneAndUpdate(
23      contact, { $set: { favorite: contact.favorite === true } }, { returnDocument: "after", upsert: true }
24    );
25    return result.value;
26  }
27 }
28 module.exports = ContactService;

```

- Nếu có lỗi xảy ra sẽ chuyển cho middleware xử lý lỗi đã định nghĩa trong `app.js` (thông qua lời gọi `next(error)`). Chú ý require các hàm/lớp được sử dụng ở đầu tập tin `app/controllers/contact.controller.js`:

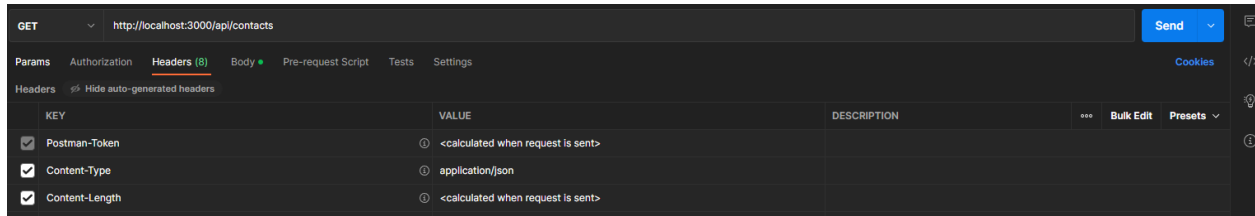
```

app > controllers > JS contact.controller.js > ...
1 const ContactService = require("../services/contact.service");
2 const MongoDB = require("../utils/mongodb.util");
3 const ApiError = require("../api-error");
4

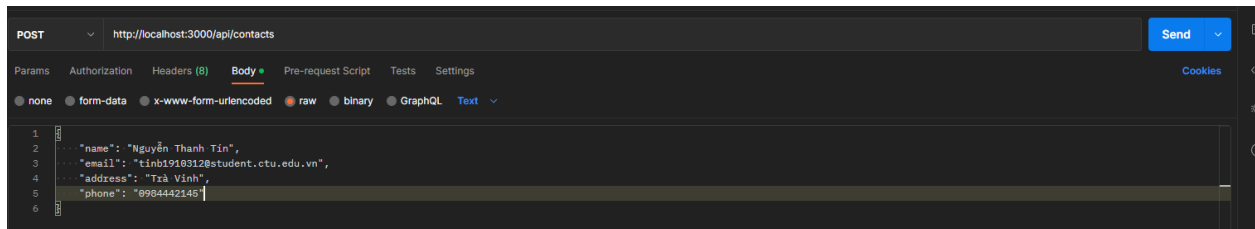
```

- Dùng Postman hoặc curl kiểm tra handler hoạt động đúng.

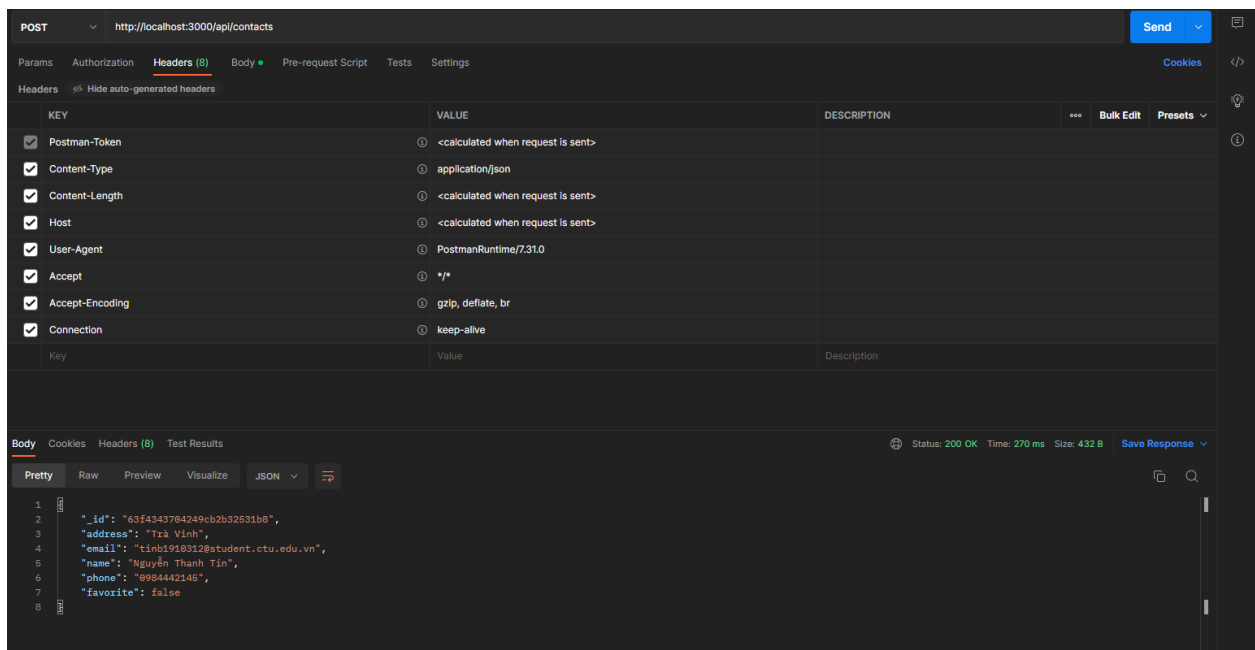
- Để gửi dữ liệu JSON về server với Postman hoặc curl, cần đặt Header "Content-Type: application/json" và đặt dữ liệu JSON trong phần Body của yêu cầu



*Hiệu chỉnh Header "Content-Type: application/json"*



*Đặt dữ liệu JSON trong phần Body*



*Kết quả trả về*

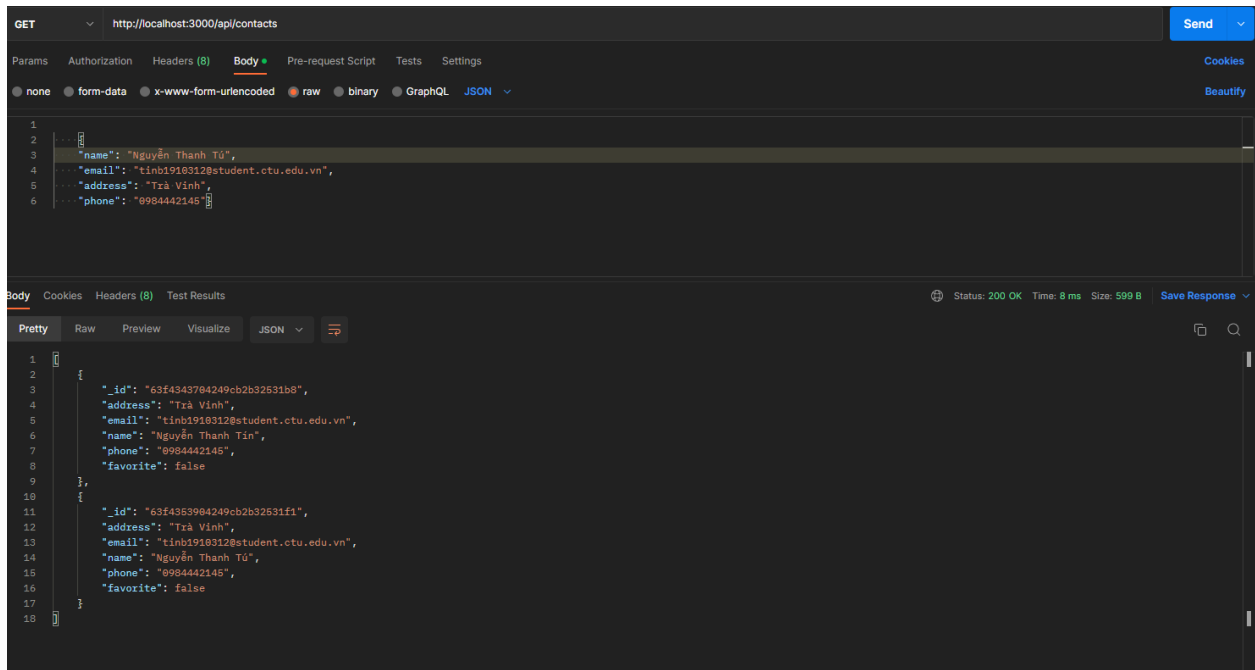
Cài đặt handler findAll:

```
app > controllers > JS contact.controller.js > create > create
1
20 exports.findAll = async (req, res, next) => {
21   let documents = [];
22
23   try {
24     const contactService = new ContactService(MongoDB.client);
25     const { name } = req.query;
26     if (name) {
27       documents = await contactService.findByName(name);
28     } else {
29       documents = await contactService.find({});
30     }
31   } catch (error) {
32     console.log(error);
33     return next(
34       new ApiError(500, "An error occurred while retrieving contacts")
35     );
36   }
37   return res.send(documents)
38 };
```

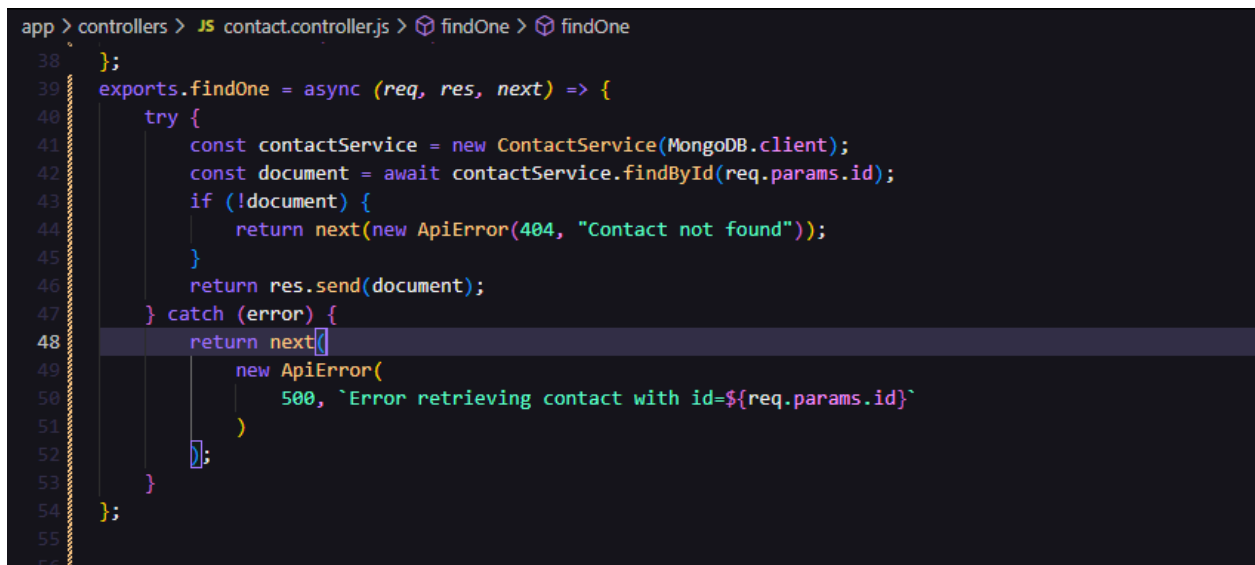
- contactService.find(condition) và contactService.findByName(name) lần lượt tìm kiếm các tài liệu thỏa điều kiện chỉ định trong đối tượng condition, theo tên name. Hai phương thức này có thể được định nghĩa như sau:

```
app > services > JS contact.service.js > ContactService > findByName
25   return result.value;
26 }
27 async find(filter) {
28   const cursor = await this.Contact.find(filter);
29   return await cursor.toArray();
30 }
31 async findByName(name) {
32   return await this.find({
33     name: { $regex: new RegExp(name), $options: "i" },
34   });
35 }
36 }
```

- Trong đoạn code của phương thức findByName(name), ta dùng biểu thức chính quy không phân biệt hoa thường (\$option: "i") để so khớp tên contact cần tìm kiếm.
- Dùng Postman hoặc curl kiểm tra handler hoạt động đúng.



### Cài đặt handler findOne:



- contactService.findById(id) tìm kiếm tài liệu theo Id. Phương thức findById(id) có thể được định nghĩa như sau:

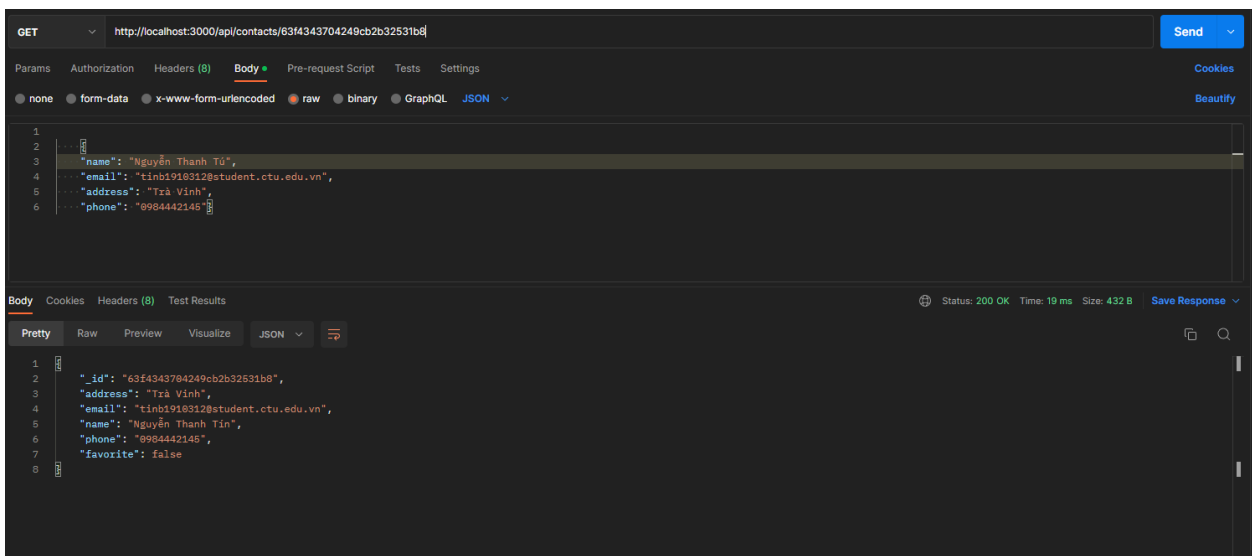


```

app > services > JS contact.service.js > ContactService > findById
30   }
31   async findByName(name) {
32     return await this.find({
33       name: { $regex: new RegExp(name), $options: "i" },
34     });
35   }
36   async findById(id) {
37     return await this.Contact.findOne({
38       _id: ObjectId.isValid(id) ? new ObjectId(id) : null,
39     });
40   }
41 }
42 module.exports = ContactService;

```

- Dùng Postman hoặc curl kiểm tra handler hoạt động đúng.



### Cài đặt handler update:

```

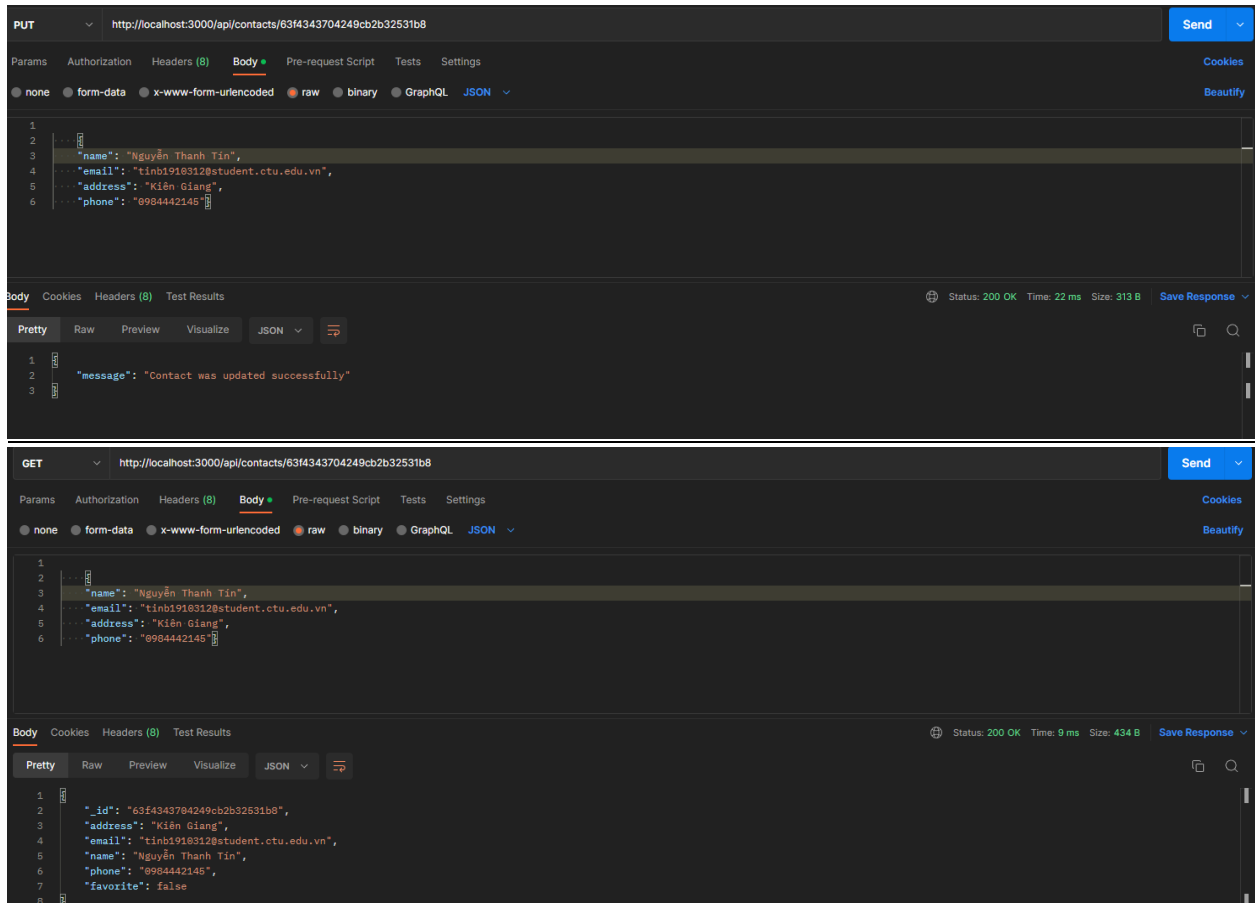
app > controllers > JS contact.controller.js > findOne > findOne
56
57 exports.update = async (req, res, next) => {
58   if (Object.keys(req.body).length === 0) {
59     return next(new ApiError(400, "Data to update can not be empty"));
60   }
61   try {
62     const contactService = new ContactService(MongoDB.client);
63     const document = await contactService.update(req.params.id, req.body);
64     if (!document) {
65       return next(new ApiError(404, "Contact not found"));
66     }
67     return res.send({ message: "Contact was updated successfully" });
68   } catch (error) {
69     return next(
70       new ApiError(500, `error updating contact with id-${req.params.id}`)
71     );
72   }
73 };
74

```

- contactService.update(id, document) tìm kiếm tài liệu theo Id và cập nhật tài liệu này với dữ liệu trong đối tượng document. Phương thức update(id, document) có thể được định nghĩa như sau:

```
app > services > JS contact.service.js > ContactService
41   }
42   async update(id, payload) {
43     const filter = {
44       _id: ObjectId.isValid(id) ? new ObjectId(id) : null,
45     };
46     const update = this.extractConactData(payload);
47     const result = await this.Contact.findOneAndUpdate(
48       filter, { $set: update }, { returnDocument: "after" }
49     );
50     return result.value;
51   }
```

- Dùng Postman hoặc curl kiểm tra handler hoạt động đúng.



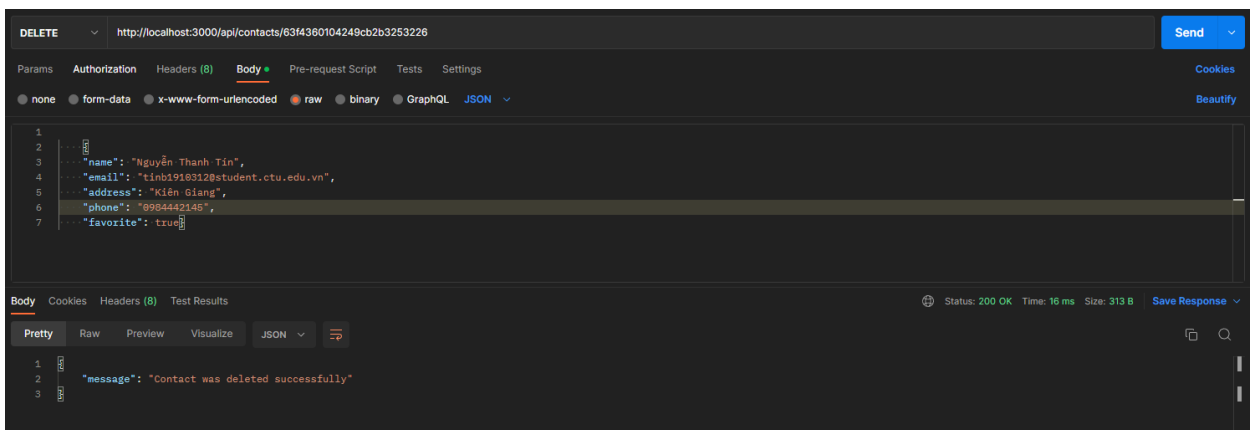
Cài đặt handler delete:

```
app > controllers > JS contact.controller.js > findOne > findOne
75 exports.delete = async (req, res, next) => {
76   try {
77     const contactService = new ContactService(MongoDB.client);
78     const document = await contactService.delete(req.params.id);
79     if (!document) {
80       return next(new ApiError(404, "Contact not Found"));
81     }
82     return res.send({ message: "Contact was deleted successfully" });
83   } catch (error) {
84     return next(
85       new ApiError(
86         500,
87         `Could not delete contact with id=${req.params.id}`
88       )
89     );
90   }
91 };
92 }
```

- contactService.delete(id) tìm kiếm tài liệu theo Id và xóa tài liệu này. Phương thức delete(id) có thể được định nghĩa như sau:

```
app > services > JS contact.service.js > ContactService
51
52 async delete(id) {
53   const result = await this.Contact.findOneAndDelete({
54     _id: ObjectId.isValid(id) ? new ObjectId(id) : null,
55   });
56   return result.value;
57 }
58 }
```

- Dùng Postman hoặc curl kiểm tra handler hoạt động đúng.



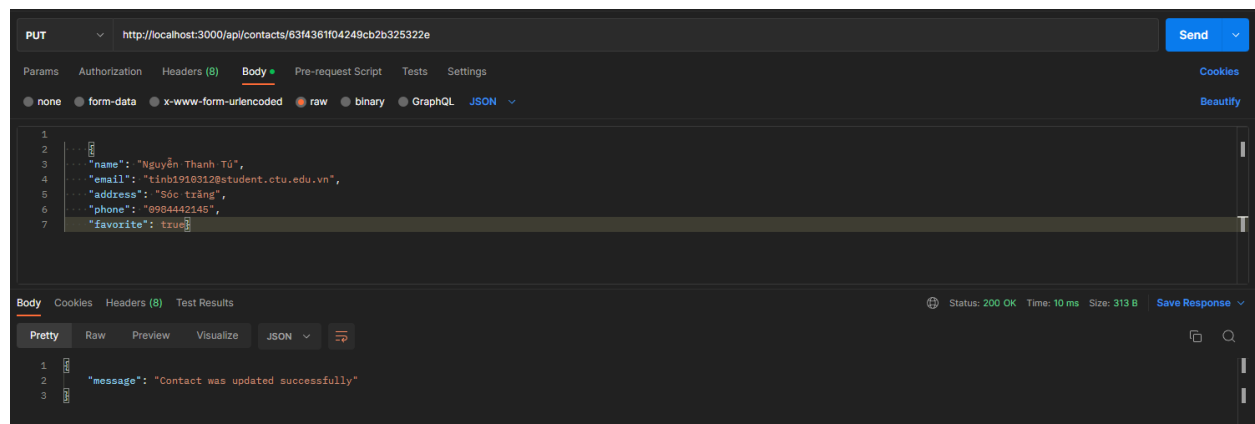
Cài đặt handler findAllFavorite:

```
app > controllers > JS contact.controller.js > findOne > findOne
91 };
92
93 exports.findAllFavorite = async (req, res, next) => {
94   try {
95     const contactService = new ContactService(MongoDB.client);
96     const documents = await contactService.findFavorite();
97     return res.send(documents)
98   } catch (error) {
99     return next(
100       500,
101       "An error occurred while retrieving favorite contacts"
102     )
103   };
104 };
105
```

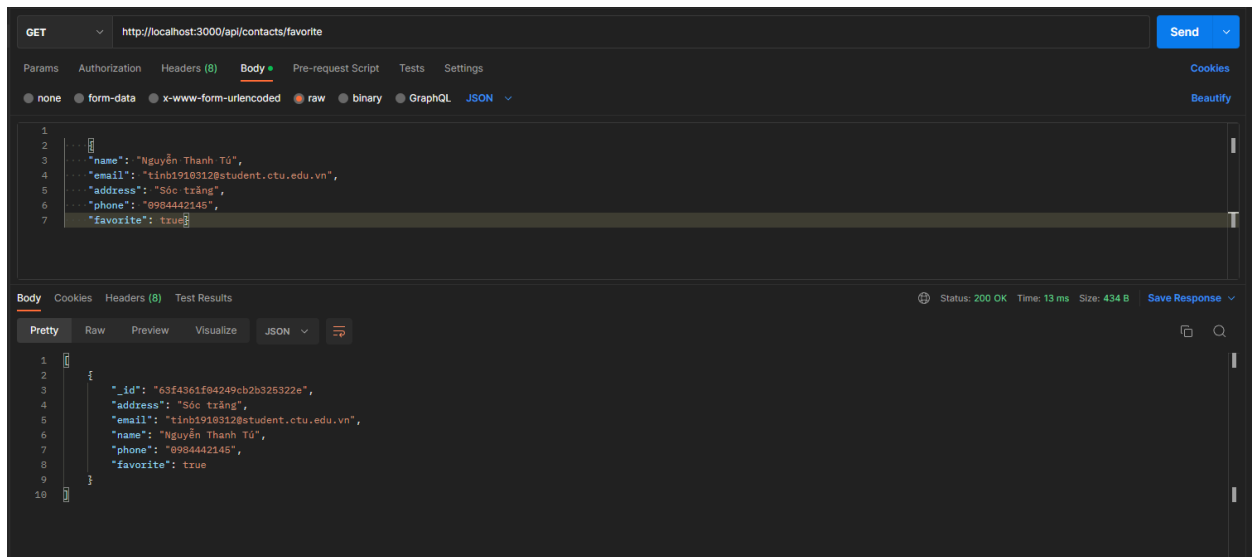
- Phương thức findFavorite() trong lớp ContactService có thể được định nghĩa như sau:

```
app > services > JS contact.service.js > ContactService
59   async findFavorite() {
60     return await this.find({ favorite: true });
61   }
62
```

- Dùng Postman hoặc curl kiểm tra handler hoạt động đúng.



- Cập nhật favorite cho đối tượng



- Kết quả trả về

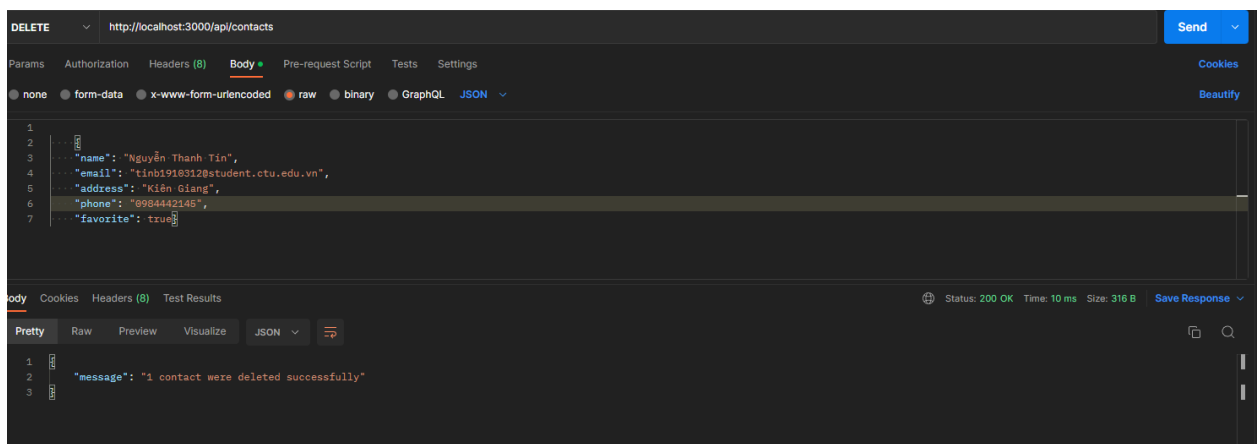
Cài đặt handler deleteAll:

```
app > controllers > JS contact.controller.js > ...  
104   }  
105   }  
106   exports.deleteAll = async (req, res, next) => {  
107     try {  
108       const contactService = new ContactService(MongoDB.client);  
109       const deletedCount = await contactService.deleteAll();  
110       return res.send({  
111         message: `${deletedCount} contact were deleted successfully`,  
112       });  
113     } catch (error) {  
114       return next(  
115         500,  
116         "An error occurred while removing all contacts"  
117       );  
118     }  
119   }  
120 }  
121  
122  
123
```

- contactService.deleteMany() xóa tất cả các đối tượng trong collection. Phương thức deleteAll() có thể được định nghĩa như sau:

```
app > services > JS contact.service.js > ContactService
61      }
62
63      async deleteAll() {
64          const result = await this.Contact.deleteMany({});
65          return result.deletedCount;
66      }
67
68  }
69  module.exports = ContactService;
```

- Dùng Postman hoặc curl kiểm tra handler hoạt động đúng.

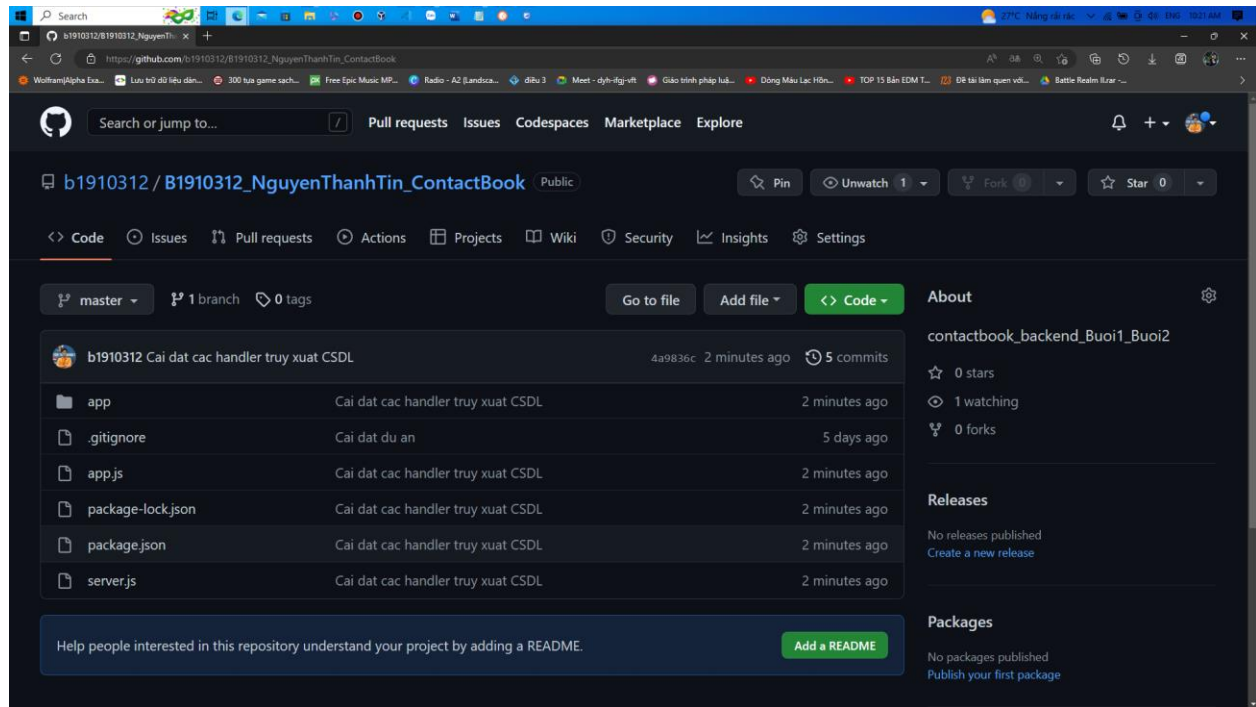


- Sau khi kiểm tra các handler hoạt động đúng, chúng ta có thể lưu các thay đổi vào git:

```
PS D:\Phat trien ung dung web\contactbook-backend> git add -u
warning: in the working copy of 'package-lock.json', LF will be replaced by CRLF the next time Git touches it
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it
PS D:\Phat trien ung dung web\contactbook-backend> git add app/utils app/services
PS D:\Phat trien ung dung web\contactbook-backend> git commit -m "Cai dat cac handler truy xuat CSDL"
[master 4a9836c] Cai dat cac handler truy xuat CSDL
 8 files changed, 2180 insertions(+), 1708 deletions(-)
 create mode 100644 app/services/contact.service.js
 create mode 100644 app/utils/mongodb.util.js
PS D:\Phat trien ung dung web\contactbook-backend>
```

Đẩy các thay đổi lên GitHub:

```
PS D:\Phat trien ung dung web\contactbook-backend> git push origin master
Enumerating objects: 25, done.
Counting objects: 100% (25/25), done.
Delta compression using up to 8 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (15/15), 9.37 KiB | 872.00 KiB/s, done.
Total 15 (delta 3), reused 0 (delta 0), pack-reused 0
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/b1910312/B1910312_NguyenThanhTin_ContactBook.git
 1797a67..4a9836c master -> master
PS D:\Phat trien ung dung web\contactbook-backend>
```



- Cấu trúc thư mục dự án đến thời điểm này sẽ như sau:

