

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC CẦN THƠ
TRƯỜNG CÔNG NGHỆ THÔNG TIN & TRUYỀN THÔNG**



BÁO CÁO PHÁT TRIỂN ỨNG DỤNG WEB

**Giáo viên hướng dẫn:
Ths.GVC Nguyễn Minh Trung**

**Sinh viên thực hiện:
Ngô Thị Thu Thảo
MSSV: B2207566
Khóa: 48**

Cần Thơ, 09/2024

MỤC LỤC

Catalog	
Ứng dụng Contactbook - Backend - Phần 1	2
Bước 0: Cài đặt Node và git	2
Bước 1: Tạo ứng dụng Node	2
Bước 2: Quản lý mã nguồn dự án với git và github	3
Bước 3: Cấu hình Visual code, ESLint và Prettier	4
Bước 4: Cài đặt Express	6
Bước 5: Định nghĩa controller và các router	7
Bước 6: Cài đặt xử lý lỗi	10
Ứng dụng Contactbook - Backend - Phần 2	13
7. Cài đặt MongoDB	13
8. Cài đặt thư viện MongoDB, định nghĩa hàm trợ giúp kết nối và lớp dịch vụ truy xuất cơ sở dữ liệu (CSDL)	13
Bước 9: Cài đặt các handler	15
- Cài đặt handler create	15
- Cài đặt handler findAll	16
- Cài đặt handler findOne	18
- Cài đặt handler update	19
- Cài đặt handler delete	20
- Cài đặt handler findAllFavorite	21
- Cài đặt handler deleteAll	22

Ứng dụng Contactbook - Backend - Phần 1

Bước 0: Cài đặt Node và git

Đã cài node và git

```
PS C:\Users\thuth> node -v
v20.11.0
PS C:\Users\thuth> git --version
git version 2.45.2.windows.1
PS C:\Users\thuth>
```

Bước 1: Tạo ứng dụng Node

- Tạo thư mục dự án: ContactBook-backend
- Gõ lệnh : git init

```
PS C:\Users\thuth\ContactBook-backend> npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See 'npm help init' for definitive documentation on these fields
and exactly what they do.

Use 'npm install <pkg>' afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (contactbook-backend)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author: Thao Ngo
license: (ISC)
About to write to C:\Users\thuth\ContactBook-backend\package.json:

{
  "name": "contactbook-backend",
  "version": "1.0.0",
  "main": "index.js",
  "scripts": {
    "test": "echo \"Error: no test specified\" && exit 1"
  },
  "author": "Thao Ngo",
  "license": "ISC",
  "description": ""
}

Is this OK? (yes)
PS C:\Users\thuth\ContactBook-backend> |
```

- Cài đặt các package cần thiết

```
PS C:\Users\thuth\ContactBook-backend> npm install express cors
added 67 packages, and audited 68 packages in 4s

13 packages are looking for funding
  run 'npm fund' for details

found 0 vulnerabilities
PS C:\Users\thuth\ContactBook-backend>
```

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

- Tạo tập tin '.gitignore'

```
PS C:\Users\thuth\ContactBook-backend> npx gitignore node
Created .gitignore file for flag type node.
```

- Gõ lệnh: 'git init' để khởi tạo dự án Git. Sau đó kiểm tra với 'git status'.

```
PS C:\Users\thuth\ContactBook-backend> git init
Initialized empty Git repository in C:/Users/thuth/ContactBook-backend/.git/
PS C:\Users\thuth\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 C:\Users\thuth\ContactBook-backend>
```

- Cho git quản lý 3 dự án .gitignore, package-lock.json, package.json

```
PS C:\Users\thuth\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 C:\Users\thuth\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

PS C:\Users\thuth\ContactBook-backend>
```

- Thực thi lệnh: `git commit -m "Cài đặt dự án"`

```
PS C:\Users\thuth\ContactBook-backend> git commit -m "Cài đặt dự án"
[master (root-commit) 4d6ef87] Cài đặt dự án
3 files changed, 874 insertions(+)
create mode 100644 .gitignore
create mode 100644 package-lock.json
create mode 100644 package.json
```

- Upload dự án lên GitHub với các lệnh:

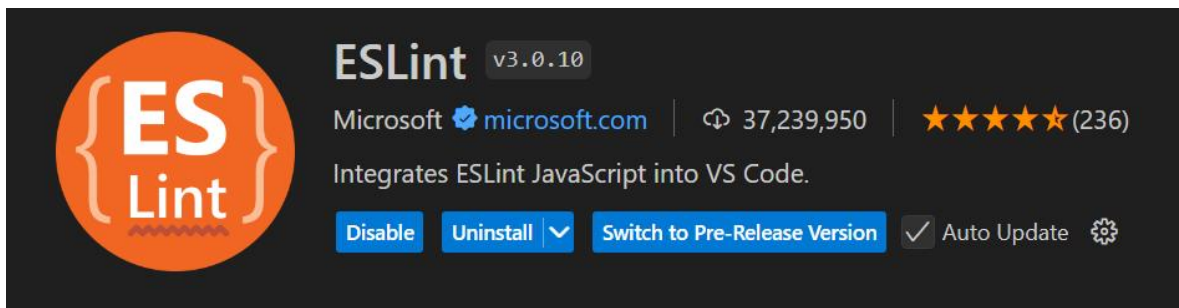
`git remote add origin https://github.com/ThuThao527/CT449/ContackBook.git`

`git push origin main`

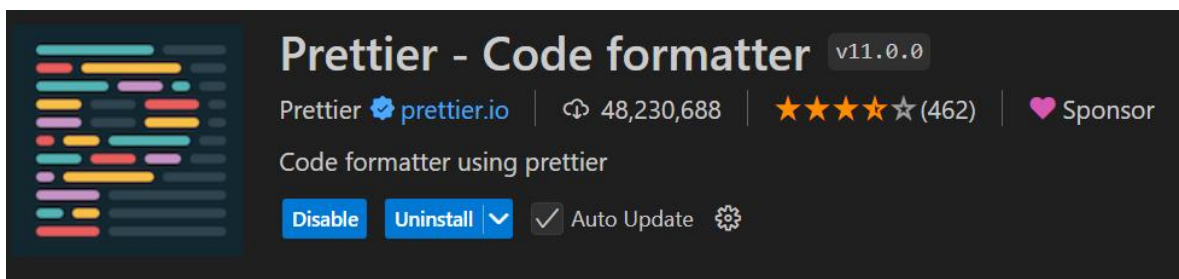
```
PS C:\Users\thuth\ContactBook-backend> git remote add origin https://github.com/ThuThao527/CT449-ContactBook.git
PS C:\Users\thuth\ContactBook-backend> git push origin master
Enumerating objects: 5, done.
Counting objects: 100% (5/5), done.
Delta compression using up to 8 threads
Compressing objects: 100% (5/5), done.
Writing objects: 100% (5/5), 8.76 KiB | 4.38 MiB/s, done.
Total 5 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/ThuThao527/CT449-ContactBook.git
 * [new branch]      master -> master
PS C:\Users\thuth\ContactBook-backend>
```

Bước 3: Cấu hình Visual code, ESLint và Prettier

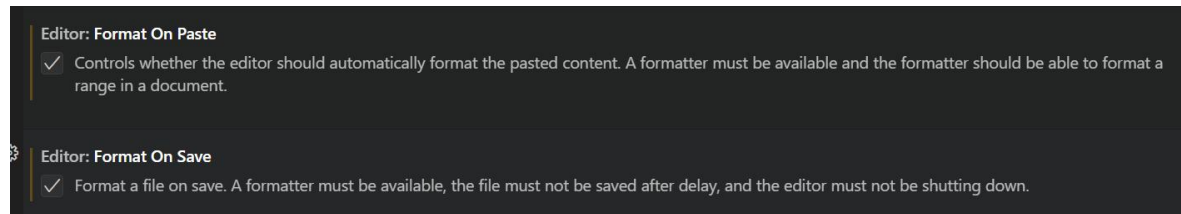
- Cài đặt ESLint extension. ESLint là một công cụ phân tích tĩnh mã Javascript, giúp tìm và sửa các vấn đề trong mã nguồn.



- Cài đặt Prettier extension. Prettier là công cụ hỗ trợ định dạng tự động các tập tin mã nguồn.



- Cấu hình VSCode cho phép định dạng mã lệnh khi save và paste.



- Cài đặt gói thư viện ESLint và Prettier với lệnh sau: `npm i -C eslint prettier eslint-config-prettier`

sau đó tạo một tập tin `eslinttrc.js` : `npx eslint --init`



Và lưu thay đổi trên git:

`git add -u`

`git add .eslinttrc.js`

`git commit -m "Cau hình ESLint cho du an"`

```
PS C:\Users\thuth\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  
warning: in the working copy of 'package.json', LF will be replaced by CRLF the next time Git touches it  
PS C:\Users\thuth\ContactBook-backend> git add .eslinttrc.js  
fatal: pathspec '.eslinttrc.js' did not match any files  
PS C:\Users\thuth\ContactBook-backend> git commit -m "Cau hình ESLint cho du an"  
[master 26d2741] Cau hình ESLint cho du an  
 2 files changed, 2588 insertions(+), 102 deletions(-)  
PS C:\Users\thuth\ContactBook-backend>
```


Bước 4: Cài đặt Express

- Tạo tập tin app.js, server.js và index.js

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

```
JS server.js > app.listen() callback
1  const app = require("./app");
2  const config = require("./app/config");
3
4  const PORT = config.app.port;
5  app.listen(PORT, () => {
6    console.log(`Server is running on port ${PORT}`);
7  });
8
9
```

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

Kiểm tra kết quả:

```
PS C:\Users\thuth\ContactBook-backend> node server.js
Server is running on port 3000
```

- Cài đặt nodemon

```
PS C:\Users\thuth\ContactBook-backend> npm install nodemon --save-dev

added 24 packages, and audited 276 packages in 3s

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

found 0 vulnerabilities
PS C:\Users\thuth\ContactBook-backend> 
```

```
"scripts": {
  "test": "echo \"Error: no test specified\" && exit 1",
  "start" : "nodemon server.js"
},
"author": "Thao Ngo",
```

- Lưu các thay đổi vào git:

```
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 C:\Users\thuth\ContactBook-backend> git add app/app.js server.js
fatal: pathspec 'app/app.js' did not match any files
PS C:\Users\thuth\ContactBook-backend> git add app/ app.js server.js
PS C:\Users\thuth\ContactBook-backend> git commit -m "Cai dat express chay thong bao xin chao"
[master fb8789d] Cai dat express chay thong bao xin chao
5 files changed, 331 insertions(+), 3 deletions(-)
create mode 100644 app.js
create mode 100644 app/config/index.js
create mode 100644 server.js
PS C:\Users\thuth\ContactBook-backend> 
```

Bước 5: Định nghĩa controller và các router

- Tạo file contact.controller.js


```
controllers > JS contact.controller.js > ...
1  exports.create = (req, res) => {
2    res.send({ message: "create handler" });
3  };
4
5  exports.findAll = (req, res) => {
6    res.send({ message: "findAll handler" });
7  };
8
9  exports.findOne = (req, res) => {
10   res.send({ message: "findOne handler" });
11 };
12
13 exports.update = (req, res) => {
14   res.send({ message: "update handler" });
15 };
16
17 exports.delete = (req, res) => {
18   res.send({ message: "delete handler" });
19 };
20
21 exports.deleteAll = (req, res) => {
22   res.send({ message: "deleteAll handler" });
23 };
24
25 exports.findAllFavorite = (req, res) => {
26   res.send({ message: "findAllFavorite handler" });
27 };
28
```

- Tạo file contact.route.js

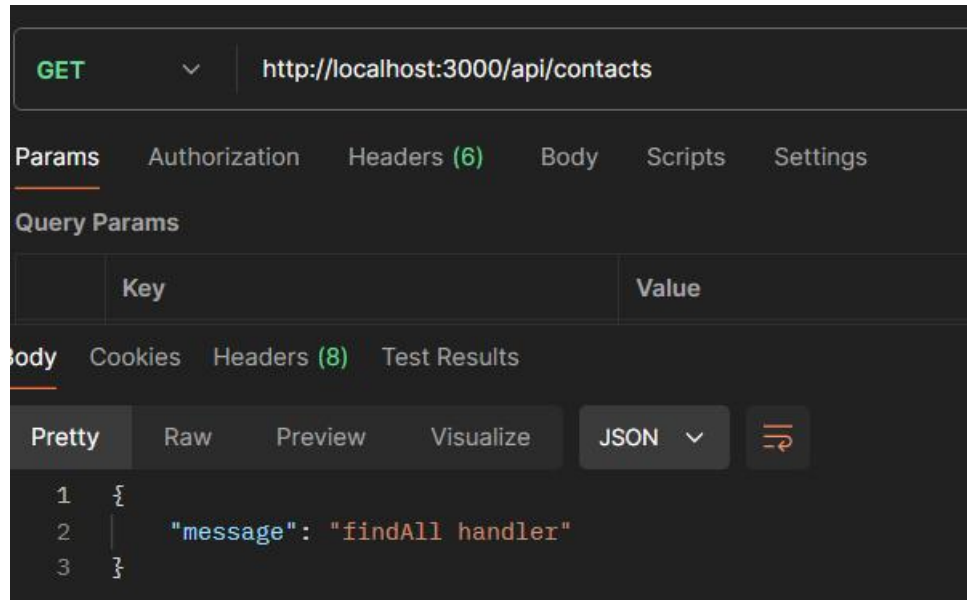
```
app > routes > JS contact.route.js > ...
1  const express = require("express");
2  const contacts = require("../controllers/contact.controller");
3
4  const router = express.Router();
5
6  router
7    .route("/")
8    .get(contacts.findAll)
9    .post(contacts.create)
10   .delete(contacts.deleteALL);
11
12  router.route("/favorite").get(contacts.findAllFavorite);
13
14  router
15    .route("/:id")
16    .get(contacts.findOne)
17    .put(contacts.update)
18    .delete(contacts.delete);
19
20  module.exports = router;
21
```

- Cập nhật lại app.js

```
JS app.js > ...
1  const express = require("express");
2  const cors = require("cors");
3  const contactsRouter = require("./app/routes/contact.route");
4
5  const app = express();
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
14 module.exports = app;
```

- Kiểm tra ví dụ

```
1 {  
2   "message": "findAllFavorite handler"  
3 }
```



Bước 6: Cài đặt xử lý lỗi

- Tạo tập tin `app/api-error.js`

```
app > JS apt-error.js > ...  
1 class ApiError extends Error {  
2   constructor(statusCode, message) {  
3     super();  
4     this.statusCode = statusCode;  
5     this.message = message;  
6   }  
7 }  
8  
9 module.exports = ApiError;  
10
```

- Kiểm tra ví dụ:

- Thêm vào các middleware xử lý lỗi trong app.js

```
JS app.js > ...
1  const express = require("express");
2  const cors = require("cors");
3  const contactsRouter = require("./app/routes/contact.route");
4  const ApiError = require("./app/api-error");
5
6  const app = express();
7  app.use(cors());
8  app.use(express.json());
9  app.use("/api/contacts", contactsRouter);
10
11
12  app.use((req, res, next) => {
13    return next(new ApiError(404, "Resource not found"));
14  });
15
16  app.use((err, req, res, next) => {
17    return res.status(error.statusCode || 500).json({
18      message: error.message || "Internal Server Error",
19    });
20  });
21
22  app.get("/", (req, res) => {
23    res.json({ message: "Welcome to contact book application." });
24  });
25
26  module.exports = app;
```

- Kiểm tra ví dụ:

POST http://localhost:3000/api/112233

Params Authorization Headers (7) Body Scripts Settings

Query Params

Key	Value
-----	-------

Body Cookies Headers (8) Test Results

Pretty Raw Preview Visualize JSON

```
1  {
2    "message": "Resource not found"
3  }
```

- Lưu thay đổi trên git:

```
PS C:\Users\thuth\ContactBook-backend> git add -u
PS C:\Users\thuth\ContactBook-backend> git add app/api-error.js
PS C:\Users\thuth\ContactBook-backend> git commit -m "Cai dat va xu ly loi"
It : A parameter cannot be found that matches parameter name 'm'.
At line:1 char:11
+ git commit -m "Cai dat va xu ly loi"
+ ~~~~~
+ CategoryInfo          : InvalidArgument: (:) [It], ParameterBindingException
+ FullyQualifiedErrorId : NamedParameterNotFound,It

PS C:\Users\thuth\ContactBook-backend> git commit -m "Cai dat va xu ly loi"
[master 151487a] Cai dat va xu ly loi
3 files changed, 22 insertions(+), 1 deletion(-)
create mode 100644 app/api-error.js
PS C:\Users\thuth\ContactBook-backend>
```

```
PS C:\Users\thuth\ContactBook-backend> git push origin master
Enumerating objects: 31, done.
Counting objects: 100% (31/31), done.
Delta compression using up to 8 threads
Compressing objects: 100% (23/23), done.
Writing objects: 100% (28/28), 33.30 KiB | 5.55 MiB/s, done.
Total 28 (delta 9), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (9/9), completed with 1 local object.
To https://github.com/ThuThao527/CT449-ContactBook.git
4d6ef87..151487a master -> master
PS C:\Users\thuth\ContactBook-backend>
```

- Cây thư mục hiện tại:

```

v CONTACTBOOK-BACKEND
  v app
  > config
  v controllers
  JS contact.contr... 7
  v routes
  JS contact.route.js 3
  JS api-error.js 1
  v eslint
  > node_modules
  {} package-lock.js... U
  {} package.json U
  > node_modules
  .gitignore
  JS app.js 6
  JS eslint.config.mjs U
  {} package-lock.json
  {} package.json
  JS server.js 2

```

Link mã nguồn: <https://github.com/ThuThao527/CT449-ContactBook>

Ứng dụng Contactbook - Backend - Phần 2



7. Cài đặt MongoDB

```
$ brew install mongodb-atlas
$ atlas setup
```

Version
8.0.0 (current) ▾

Platform
Windows x64 ▾

Package
msi ▾

Download   Copy link More Options ...

8. Cài đặt thư viện MongoDB, định nghĩa hàm trợ giúp kết nối và lớp dịch vụ truy xuất cơ sở dữ liệu (CSDL)

- Gõ lệnh: `npm install mongodb`

```
PS C:\Users\thuth\ContactBook-backend> npm install mongodb
added 11 packages, and audited 287 packages in 10s
102 packages are looking for funding
  run `npm fund` for details
found 0 vulnerabilities
PS C:\Users\thuth\ContactBook-backend> 
```

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

```
app > config > JS index.js > [x] config > db > uri
1  const config = {
2    app: {
3      port: process.env.PORT || 3000,
4    },
5    db: {
6      uri : process.env.MONGODB_URI || "mongodb://127.0.0.1:27017/contactbook"
7    }
8  };
9
10 module.exports = config;
```


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

```
app > utils > JS mongodb.utils.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;
12
```

- Thay toàn bộ tập tin server.js

```
JS server.js > ...
1  const app = require("./app");
2  const config = require("./app/config");
3  const MongoDB = require("./app/utils/mongodb.util");
4
5  async function startServer() {
6    try {
7      await MongoDB.connect(config.db.uri);
8      console.log("Connected 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();
21
```

- Định nghĩa các lớp dịch vụ ContactServices (trong tập tin app/services/services.mjs) chứa các API của thư viện MongoDB để thực hiện thao tác với CSDL MongoDB:

```
1  const { ObjectId } = require("mongodb");
2
3  class ContactService {
4    constructor(client) {
5      this.Contact = client.db().collection("contacts");
6    }
7
8    // Định nghĩa các phương thức truy xuất CSDL sử dụng mongodb API
9  }
10
11  module.exports = ContactService;
12
```

Bước 9: Cài đặt các handler

- Cài đặt handler create

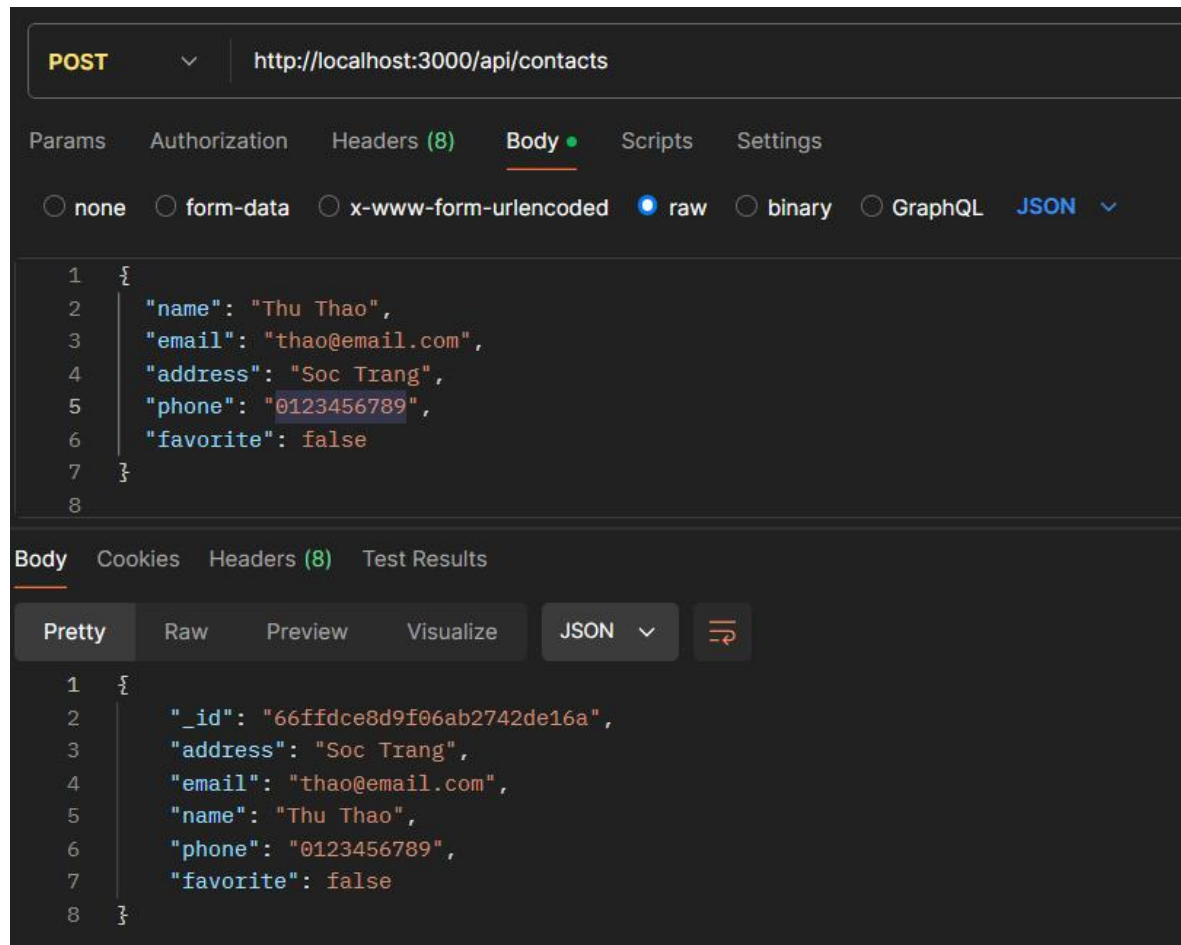
- Hiệu chỉnh tập tin `app/controllers/contact.controller.js`

```
app > controllers > JS contact.controller.js > ...
25 exports.findALLFavorite = (req, res) => {
26   // ...
27 };
28
29 // Create and Save a new Contact
30 exports.create = async (req, res, next) => {
31   if (!req.body?.name) {
32     return next(new ApiError(400, "Name can not be empty"));
33   }
34
35   try {
36     const contactService = new ContactService(MongoDB.client);
37     const document = await contactService.create(req.body);
38     return res.send(document);
39   } catch (error) {
40     return next(new ApiError(500, "An error occurred while creating the contact"));
41   }
42 };
43
```

- Tạo phương thức `create()` trong lớp `ContactService`

```
1 const { ObjectId } = require("mongodb");
2
3 class ContactService {
4   constructor(client) {
5     this.Contact = client.db().collection("contacts");
6   }
7   // Định nghĩa các phương thức truy xuất CSDL sử dụng mongodb API
8   extractContactData(payload) {
9     const contact = {
10       name: payload.name,
11       email: payload.email,
12       address: payload.address,
13       phone: payload.phone,
14       favorite: payload.favorite,
15     };
16
17     // Remove undefined fields
18     Object.keys(contact).forEach(
19       (key) => contact[key] === undefined && delete contact[key]);
20
21     return contact;
22   }
23
24   async create(payload) {
25     const contact = this.extractContactData(payload);
26     const result = await this.Contact.findOneAndUpdate(
27       contact,
28       { $set: { favorite: contact.favorite === true } },
29       { returnDocument: "after", upsert: true }
30     );
31     return result;
32   }
33 }
34
35 module.exports = ContactService;
```

● Kiểm tra với Postman



- Cài đặt handler findAll

```
8
9 exports.findAll = async (req, res, next) => {
10   let documents = [];
11
12   try {
13     const contactService = new ContactService(MongoDB.client);
14     const { name } = req.query;
15     if (name) {
16       documents = await contactService.findByName(name);
17     } else {
18       documents = await contactService.find({});
19     }
20   } catch (error) {
21     return next(new ApiError(500, "An error occurred while retrieving contacts"));
22   }
23
24   return res.send(documents);
25 };
```

```
33   async find(filter) {
34     const cursor = await this.Contact.find(filter);
35     return await cursor.toArray();
36   }
37
38   async findByName(name) {
39     return await this.find({
40       name: { $regex: new RegExp(name), $options: "i" },
41     });
42   }
43 }
44
```

● Kiểm tra với Postman

The screenshot shows the Postman interface for a POST request to `http://localhost:3000/api/contacts`. The request body is a JSON object with the following fields: `name` ("Thu Thao"), `email` ("thao@email.com"), `address` ("Soc Trang"), `phone` ("0123456789"), and `favorite` (false). The response is also a JSON object, showing the same data plus an `_id` field with the value "66ffdce8d9f06ab2742de16a".

Request:

```
{
  "name": "Thu Thao",
  "email": "thao@email.com",
  "address": "Soc Trang",
  "phone": "0123456789",
  "favorite": false
}
```

Response:

```
{
  "_id": "66ffdce8d9f06ab2742de16a",
  "address": "Soc Trang",
  "email": "thao@email.com",
  "name": "Thu Thao",
  "phone": "0123456789",
  "favorite": false
}
```


- Cài đặt handler findOne

```
24 exports.findOne = async (req, res, next) => {
25   try {
26     const contactService = new ContactService(MongoDB.client);
27     const document = await contactService.findById(req.params.id);
28     if (!document) {
29       return next(new ApiError(404, "Contact not found"));
30     }
31     return res.send(document);
32   } catch (error) {
33     return next(new ApiError(500, `Error retrieving contact with id=${req.params.id}`));
34   }
35 };
```

```
42 }
43 async findById(id) {
44   return await this.Contact.findOne({
45     _id: ObjectId.isValid(id) ? new ObjectId(id) : null,
46   });
47 }
48 }
49
50
51 module.exports = ContactService;
```

● Kiểm tra Postman

GET http://localhost:3000/api/contacts/66ffdce8d9f06ab2742de16a

Params Authorization Headers (6) **Body** Scripts Settings

☒ none ☐ form-data ☐ x-www-form-urlencoded ☐ raw ☐ binary ☐ GraphQL

This request does not have a body.

Body Cookies Headers (8) Test Results

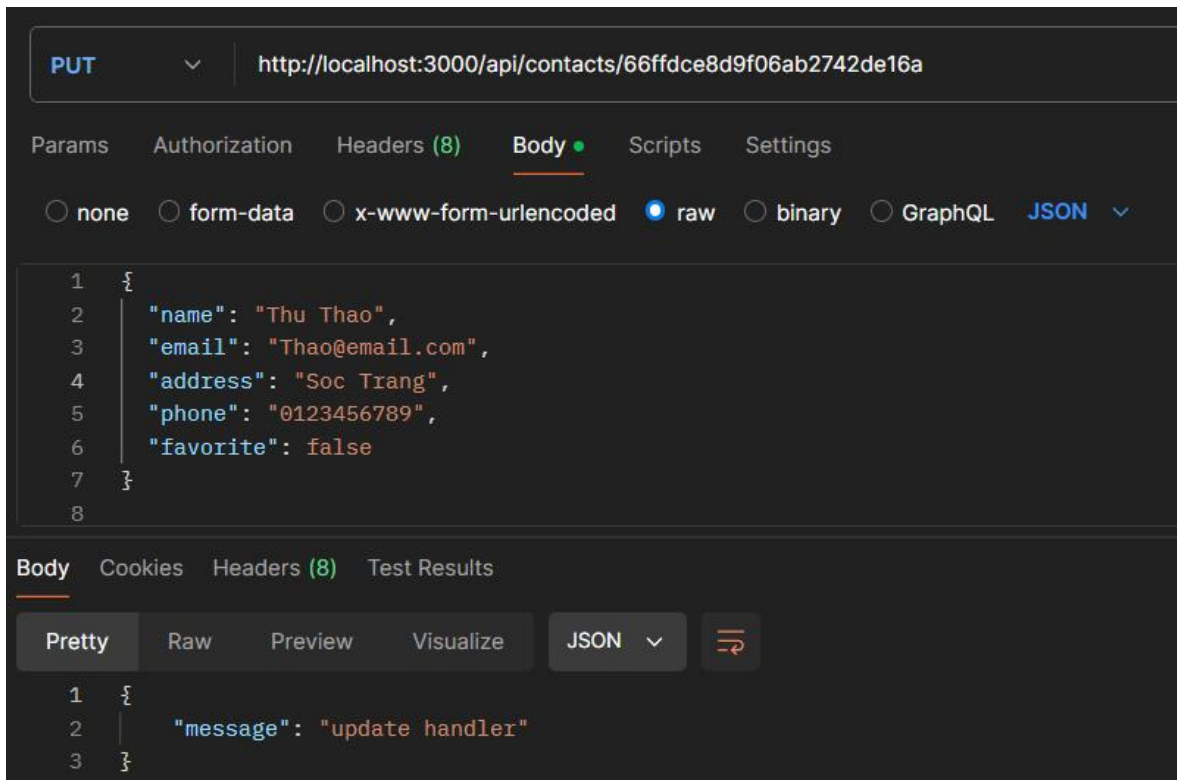
Pretty Raw Preview Visualize JSON

```
1 {
2   "message": "findOne handler"
3 }
```

- Cài đặt handler update

```
36
37 exports.update = async (req, res, next) => {
38   if (Object.keys(req.body).length === 0) {
39     return next(new ApiError(400, "Data to update can not be empty"));
40   }
41
42   try {
43     const contactService = new ContactService(MongoDB.client);
44     const document = await contactService.update(req.params.id, req.body);
45     if (!document) {
46       return next(new ApiError(404, "Contact not found"));
47     }
48
49     return res.send({ message: "Contact was updated successfully" });
50   } catch (error) {
51     return next(new ApiError(500, `Error updating contact with id=${req.params.id}`));
52   }
53 };
54
55 async update(id, payload) {
56   const filter = {
57     _id: ObjectId.isValid(id) ? new ObjectId(id) : null,
58   };
59   const update = this.extractContactData(payload);
60   const result = await this.Contact.findOneAndUpdate(
61     filter,
62     { $set: update },
63     { returnDocument: "after" }
64   );
65   return result.value; //return the updated document
66 }
```

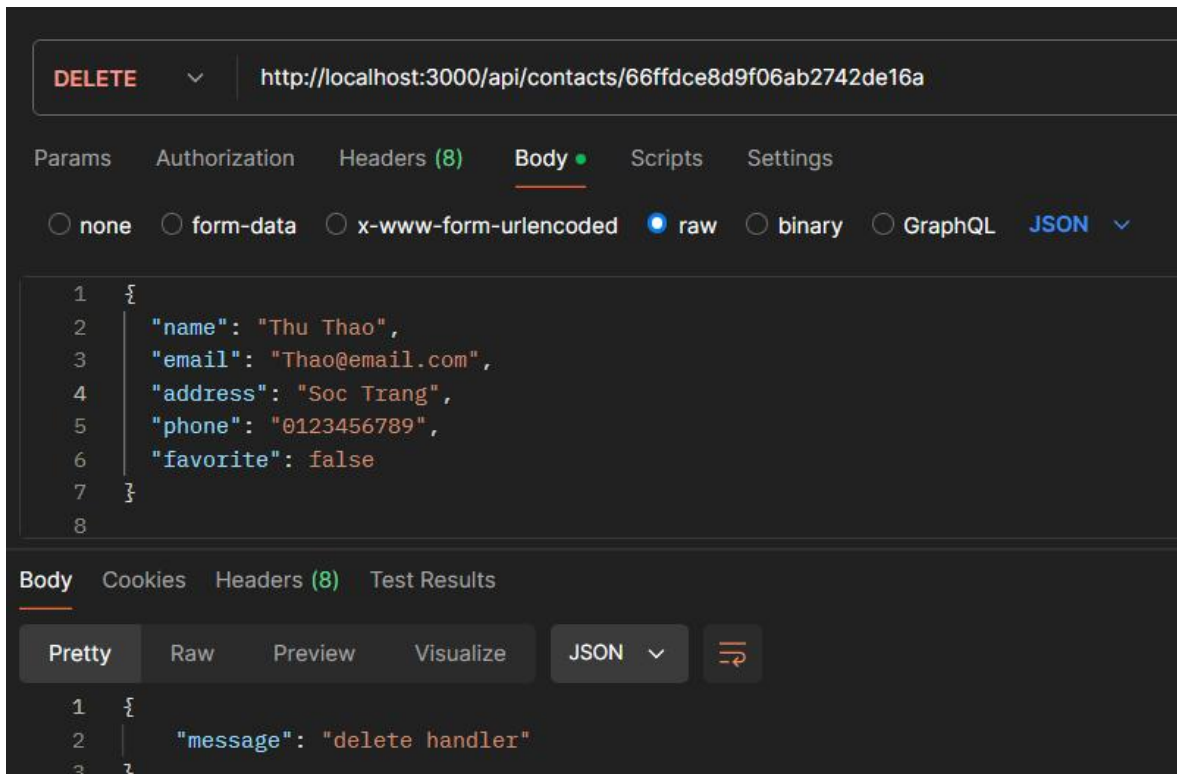
Kiểm tra postman



- Cài đặt handler delete

```
55 // Delete a contact with the specified id in the request
56 exports.delete = async (req, res, next) => {
57   try {
58     const contactService = new ContactService(MongoDB.client);
59     const document = await contactService.delete(req.params.id);
60     if (!document) {
61       return next(new ApiError(404, "Contact not found"));
62     }
63
64     return res.send({ message: "Contact was deleted successfully" });
65   } catch (error) {
66     return next(new ApiError(500, `Could not delete contact with id=${req.params
67   }
68 };
```

```
60   async delete(id) {
61     const result = await this.Contact.findOneAndDelete({
62       _id: ObjectId.isValid(id) ? new ObjectId(id) : null,
63     });
64     return result;
65   }
```



The screenshot shows a REST client interface with a DELETE request to `http://localhost:3000/api/contacts/66ffdce8d9f06ab2742de16a`. The request body is a JSON object representing a contact. The response body is a JSON object with a message indicating the delete handler was successful.

Request:

```
1 {
2   "name": "Thu Thao",
3   "email": "Thao@email.com",
4   "address": "Soc Trang",
5   "phone": "0123456789",
6   "favorite": false
7 }
```

Response:

```
1 {
2   "message": "delete handler"
3 }
```

- Cài đặt handler `findAllFavorite`

```
exports.findAllFavorite = async (req, res, next) => {
  try {
    const contactService = new ContactService(MongoDB.client);
    const documents = await contactService.findFavorite();
    return res.send(documents);
  } catch (error) {
    return next(new ApiError(500, "An error occurred while retrieving favorite c"));
  }
};

async findFavorite() {
  return await this.find({ favorite: true });
}
```

- Cài đặt handler deleteAll

```

69
70 exports.deleteAll = async (req, res, next) => {
71   try {
72     const contactService = new ContactService(MongoDB.client);
73     const deletedCount = await contactService.deleteAll();
74     return res.send({
75       message: `${deletedCount} contacts were deleted successfully`,
76     });
77   } catch (error) {
78     return next(new ApiError(500, "An error occurred while removing all contacts"));
79   }
80 };

```

```

69   async deleteAll() {
70     const result = await this.Contact.deleteMany({});
71     return result.deletedCount;
72   }
73 }

```

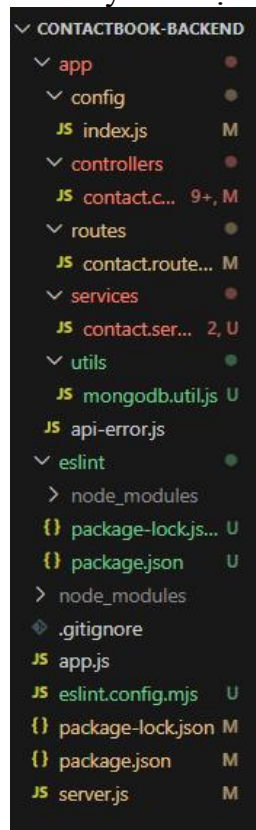
● Đưa lên github:

```

PS C:\Users\thuth\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 C:\Users\thuth\ContactBook-backend> git add app/utils app/services
PS C:\Users\thuth\ContactBook-backend> git commit -m "Cài đặt các handler truy xuất CSDL"
[master 0628eae] Cài đặt các handler truy xuất CSDL
 8 files changed, 333 insertions(+), 24 deletions(-)
 create mode 100644 app/services/contact.service.js
 create mode 100644 app/utils/mongodb.util.js
PS C:\Users\thuth\ContactBook-backend> git push origin master
Enumerating objects: 27, done.
Counting objects: 100% (27/27), done.
Delta compression using up to 8 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (16/16), 4.64 KiB | 951.00 KiB/s, done.
Total 16 (delta 3), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (3/3), completed with 3 local objects.
To https://github.com/ThuThao527/CT449-ContactBook.git
 151487a..0628eae master -> master
PS C:\Users\thuth\ContactBook-backend> 

```

● Cây thư mục:



Ứng dụng Contactbook - FrontEnd - Phần 3

Bước 10: Chuẩn bị môi trường làm việc

Bước 11: Tạo ứng dụng Vue

```
PS C:\Users\thuth\ContactBook-FrontEnd> npm init vue@3
Need to install the following packages:
create-vue@3.10.4
Ok to proceed? (y) y

> npx
> create-vue

Vue.js - The Progressive JavaScript Framework

✓ Project name: ... ContactBook-FrontEnd
✓ Package name: ... contactbook-frontent
✓ Add TypeScript? ... No / Yes
✓ Add JSX Support? ... No / Yes
✓ Add Vue Router for Single Page Application development? ... No / Yes
✓ Add Pinia for state management? ... No / Yes
✓ Add Vitest for Unit Testing? ... No / Yes
✓ Add an End-to-End Testing Solution? » No
✓ Add ESLint for code quality? ... No / Yes
✓ Add Vue DevTools 7 extension for debugging? (experimental) ... No / Yes

Scaffolding project in C:\Users\thuth\ContactBook-FrontEnd\ContactBook-FrontEnd...

Done. Now run:

  cd ContactBook-FrontEnd
  npm install
  npm run dev

PS C:\Users\thuth\ContactBook-FrontEnd> 
```

- Xóa tập tin src/components và hiệu chỉnh tập tin App.vue như sau:

```
1  <script setup>
2  </script>
3
4  <template>
5    <h1>Hello, Vue.js</h1>
6
7  </template>
8
9  <style scoped>
10   .page{
11     max-width: 400px;
12     margin: auto;
13   }
14 </style>
15
```


- Hiệu chỉnh tập tin vite.config.js

```
ContactBook-FrontEnd > JS vite.config.js > [?] default > server
1  import { fileURLToPath, URL } from 'node:url'
2
3  import { defineConfig } from 'vite'
4  import vue from '@vitejs/plugin-vue'
5  import vueDevTools from 'vite-plugin-vue-devtools'
6
7  // https://vitejs.dev/config/
8  export default defineConfig({
9    plugins: [
10     vue(),
11     vueDevTools(),
12   ],
13   resolve: {
14     alias: {
15       '@': fileURLToPath(new URL('./src', import.meta.url))
16     }
17   },
18   server: {
19     port: 3001,
20   }
21 })
```

- Cài đặt các phụ thuộc và kiểm tra:

```
PS C:\Users\thuth\ContactBook-FrontEnd> npm install

changed 186 packages, and audited 149 packages in 17s

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

found 0 vulnerabilities
PS C:\Users\thuth\ContactBook-FrontEnd> npm run dev

> contactbook-frontend@0.0.0 dev
> vite

VITE v5.4.8 ready in 1581 ms

→ Local:   http://localhost:3001/
→ Network: use --host to expose
→ Vue DevTools: Open http://localhost:3001/__devtools__/ as a separate window
→ Vue DevTools: Press Alt(⌘)+Shift(⇧)+D in App to toggle the Vue DevTools
→ press h to enter to show help
```

Hello, Vue.js

Bước 12: Quản lý mã nguồn dự án với git và git hub