TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**DỰ ÁN CÔNG NGHỆ THÔNG TIN1**

**XÂY DỰNG TRANG TÌM KIẾM VIỆC LÀM DỰA TRÊN NỀN TẢNG ANGULAR, NODEJS**

*Giảng viên giám sát:* **MAI VĂN MẠNH**

*Người hướng dẫn*: **BÙI THANH PHƯƠNG**

*Người thực hiện*: **ĐÀO CẨM AN - 51603386**

Lớp **: 16050302**

Khoá  **: 20**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2019**

TỔNG LIÊN ĐOÀN LAO ĐỘNG VIỆT NAM

**TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

**KHOA CÔNG NGHỆ THÔNG TIN**



**DỰ ÁN CÔNG NGHỆ THÔNG TIN 1**

**XÂY DỰNG TRANG TÌM KIẾM VIỆC LÀM DỰA TRÊN NỀN TẢNG ANGULAR, NODEJS**

*Giảng viên giám sát:* **MAI VĂN MẠNH**

*Người hướng dẫn*: **BÙI THANH PHƯƠNG**

*Người thực hiện*: **ĐÀO CẨM AN - 51603386**

Lớp **: 16050302**

Khoá  **: 20**

**THÀNH PHỐ HỒ CHÍ MINH, NĂM 2019**

LỜI CẢM ƠN

Em xin chân thành cảm ơn khoa Công nghệ thông tin trường Đại học Tôn Đức Thắng đã tạo điều kiện để em có cơ hội làm việc với môi trường thực tế.

Em xin cảm ơn các cán bộ hướng dẫn: Bùi Thanh Phương công ty cổ phần AMIT GROUP đã nhiệt tình hướng dẫn, hỗ trợ chúng em trong thời gian vừa qua. Đồng thời, các cán bộ đã tạo môi trường làm việc chuyên nghiệp, năng động giúp chúng em phát huy khả năng của mình, trau dồi thêm kiến thức và kỹ năng thuyết trình. Em chúc các cán bộ luôn khỏe mạnh để tiếp bước cho các thế hệ sau và thành công trong công việc.

**ĐỒ ÁN ĐƯỢC HOÀN THÀNH**

**TẠI TRƯỜNG ĐẠI HỌC TÔN ĐỨC THẮNG**

Tôi xin cam đoan đây là sản phẩm đồ án của riêng chúng tôi và được sự hướng dẫn của các cán bộ công ty;. Các nội dung nghiên cứu, kết quả trong đề tài này là trung thực và chưa công bố dưới bất kỳ hình thức nào trước đây. Những số liệu trong các bảng biểu phục vụ cho việc phân tích, nhận xét, đánh giá được chính tác giả thu thập từ các nguồn khác nhau có ghi rõ trong phần tài liệu tham khảo.

Ngoài ra, trong đồ án còn sử dụng một số nhận xét, đánh giá cũng như số liệu của các tác giả khác, cơ quan tổ chức khác đều có trích dẫn và chú thích nguồn gốc.

**Nếu phát hiện có bất kỳ sự gian lận nào tôi xin hoàn toàn chịu trách nhiệm về nội dung đồ án của mình.** Trường đại học Tôn Đức Thắng không liên quan đến những vi phạm tác quyền, bản quyền do tôi gây ra trong quá trình thực hiện (nếu có).

*TP. Hồ Chí Minh, ngày tháng năm*

*Tác giả*

*(ký tên và ghi rõ họ tên)*

*Đào Cẩm An*

PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN

**Phần xác nhận của GV hướng dẫn**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

**Phần đánh giá của GV chấm bài**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tp. Hồ Chí Minh, ngày tháng năm

(kí và ghi họ tên)

TÓM TẮT

Nhu cầu tìm kiếm việc làm bao giờ cũng là vấn đề được xã hội quan tâm trong nhiều thế kỷ. Để đáp ứng nhu cầu đó trong thời đại 4.0, các trang tìm kiếm việc làm xuất hiện nhằm đáp ứng phục vụ, hỗ trợ người dùng tìm kiếm công việc, đăng công việc. Ở đây, em xây dựng trang tìm kiếm việc làm của mình dựa trên các nhu cầu phát sinh của người dùng trong thực tế, đồng thời ứng dụng công nghệ như NodeJS, RestAPI,… vào bài làm của mình.

Trong đợt báo cáo này, em xin trình bày cách xây dựng hệ thống, các công cụ đã dùng và cũng như cách vận hành hệ thống

MỤC LỤC

[LỜI CẢM ƠN i](#_Toc50212160)

[PHẦN XÁC NHẬN VÀ ĐÁNH GIÁ CỦA GIẢNG VIÊN iii](#_Toc50212161)

[TÓM TẮT iv](#_Toc50212162)

[MỤC LỤC 1](#_Toc50212163)

[DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT 2](#_Toc50212164)

[DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ 3](#_Toc50212165)

[CHƯƠNG 1 – THIẾT KẾ API (BACKEND) 4](#_Toc50212166)

[1.1 Giới thiệu 4](#_Toc50212167)

[1.2 Các công cụ sử dụng 4](#_Toc50212168)

[1.3 Cơ sở dữ liệu 6](#_Toc50212169)

[1.3.1 Thông tin về cơ sở dữ 6](#_Toc50212170)

[1.3.2 Kết nối cơ sở dữ liệu 6](#_Toc50212171)

[1.3.3 Các bảng trong cơ sở dữ liệu 7](#_Toc50212172)

[1.3.3.1 Bảng User 7](#_Toc50212173)

[1.3.3.2 Bảng Job 7](#_Toc50212174)

[1.4 Danh sách API 8](#_Toc50212175)

[1.4.1 Các API về bảng user 8](#_Toc50212176)

[1.4.2 Các API về bảng job 10](#_Toc50212177)

DANH MỤC KÍ HIỆU VÀ CHỮ VIẾT TẮT

**CÁC KÝ HIỆU**

**CÁC CHỮ VIẾT TẮT**

HTTP HyperText Transfer Protocol (giao thức truyền tải siêu văn bản)

REST Representational State Transfer

JS JavaScript

MVC Model-View-Controller

API Application Programming Interface (giao diện lập trình ứng dụng)

DANH MỤC CÁC BẢNG BIỂU, HÌNH VẼ, ĐỒ THỊ

**DANH MỤC HÌNH**

**DANH MỤC BẢNG**

[Bảng 1.1: Danh sách các module sử dụng trong backend 5](#_Toc50209051)

[Bảng 1.2: Table User trong cơ sở dữ liệu 7](#_Toc50209052)

CHƯƠNG 1 – THIẾT KẾ API (BACKEND)

* 1. Giới thiệu

Web tìm kiếm việc làm được thiết kế dạng REST API. API được thiết kế chính dựa trên nền tảng Node.js sử dụng Express.js để tạo các url truy cập lấy thông tin từ hệ thống và sử dụng cơ sở dữ liệu thuần túy của web MySQL. Việc thiết kế kiến trúc API khiến cho việc kiểm soát code tốt hơn, phân vùng các lỗi theo các đường dẫn, dễ nắm bắt và sửa chữa. Bên cạnh đó việc thiết kế API giúp cho việc thiết kế website được tách biệt giữa thiết kế giao diện (frontend) và thiết kế, truy xuất dữ liệu (backend). Nhờ đó việc xây dựng website sẽ không bị ảnh hưởng nhiều từ backend và frontend. Đây cũng là mô hình nhiều website lớn, dự án lớn được xây dựng giảm thiểu thời gian và đạt hiểu qua cao.

* 1. Các công cụ sử dụng

API được viết dựa trên ngôn ngữ Javascript là ngôn ngữ chính. IDE hỗ trợ chính trong quá trình thiết kế là Visual studio code. Cũng như sử dụng công Xampp hỗ trợ thiết kế cơ sở dữ liệu cho website. Ngoài ra còn sử dụng một số module khác được cung cấp bởi node.

|  |  |  |  |
| --- | --- | --- | --- |
| STT | Tên module | Phiên bản | Mục đích |
| 1. | Express.js | 4.17.1 | Xây dựng API cho web site, các đường dẫn route truy xuát thông tin |
| 2. | MySQL | 2.18.1 | Giúp giao tiếp với hệ cơ sở dữ liệu MySQL từ ngôn ngữ JS. |
| 3. | Body-parser | 1.19.0 | Hỗ trợ lấy các thông tin từ các request từ website, các dữ liệu để tiến hành truy xuất database và trả về thông tin phù hợp |
| 4. | Nodemoon | 2.0.4 | Hỗ trợ việc reload lại server trong việc debug và thay đổi file trong hệ thống. |
| 5. | Cors | 2.8.5 | Hỗ trợ việc giải quyết vấn đề cors từ request. |
| 6. | Crypto-js | 4.0 | Hỗ trợ việc mã hóa mật khẩu để không bị đánh cắp trên cơ sở dữ liệu. Mã hóa MD5 được sử dụng chính. |

Bảng 1.1: Danh sách các module sử dụng trong backend

* 1. Cơ sở dữ liệu

1.3.1 Thông tin về cơ sở dữ

- Tên cơ sở dữ liệu: job

- User: root

- Password:

- Có tổng cổng 2 bảng chính (user, job)

1.3.2 Kết nối cơ sở dữ liệu

+ API được khởi chạy từ localhost với cổng 3000

|  |
| --- |
| const http = require('http');  const app = require('./app');  const port = process.env.PORT || 3000;  const server = http.createServer(app);  server.listen(port); |

+ Kết nối cơ sở dữ liệu

|  |
| --- |
| var mysql = require('mysql');  var connection = mysql.createConnection({  host : 'localhost',  user : 'root',  password : '',  database : 'job'  });  // connect to database  connection.connect(function(err) {if (err) throw err;});  module.exports = connection; |

1.3.3 Các bảng trong cơ sở dữ liệu

1.3.3.1 Bảng User

|  |  |  |
| --- | --- | --- |
| STT | Tên | Kiểu dữ liệu |
| 1 | Id | Int(11), primary key, auto increment |
| 2 | First\_name | Varchar(50) |
| 3 | Last\_name | Varchar(50) |
| 4 | Email | Varchar(100) |
| 5 | Password | Varchar(100) |
| 6 | Type | Int(11) |

Bảng 1.2: Table User trong cơ sở dữ liệu

1.3.3.2 Bảng Job

|  |  |  |
| --- | --- | --- |
| STT | Tên | Kiểu dữ liệu |
| 1. | Id | Int(11) primary key, auto increment |
| 2. | Id\_user | Int(11) |
| 3. | Job\_title | Varchar(100) |
| 4. | Job\_des | Longtext |
| 5. | Type | Int(11) |
| 6 | Salary | Int(11) |
| 7 | Location | Varchar(50) |
| 8 | Category | Int(11) |
| 9 | Expire\_date | Date |
| 10 | Company\_name | Varchar(30) |
| 11 | Company\_des | Longtext |
| 12 | Web\_site | Varchar(50) |
| 13 | Email | Varchar(50) |

Bảng 1.3: Bảng Job trong cơ sở dữ liệu

* 1. Danh sách API

1.4.1 Các API về bảng user

- API lấy danh sách user:

|  |  |
| --- | --- |
| Đường dẫn | “localhost:3000/user/list” |
| Phương thức | POST |
| Dữ liệu đầu vào | {‘page’:1, ‘display’:5} |
| Dữ liệu đầu ra | {  message: "ok",  users: [],  \_meta: {  total: 1,  num\_pages: 1,  current\_page: 1,  previous\_page: 1,  next\_page: 1,  num\_per\_page: 5  } |
| Mô tả | Dùng để lấy danh sách user |

Bảng 1.4: API lấy danh sách đăng nhập

- API tạo mới user:

|  |  |
| --- | --- |
| Đường dẫn | “localhost:3000/user/create” |
| Phương thức | POST |
| Dữ liệu đầu vào | user |
| Dữ liệu đầu ra | Thành công:  {  message: "ok",  user: user  } |
| Thất bại:{message: 'Tài khoản đã tồn tại'} |
| Mô tả | Dùng để tạo mới tài khoản để đăng bài tuyển dụng |

Bảng 1.5: API tạo mới tài khoản

- API đăng nhập:

|  |  |
| --- | --- |
| Đường dẫn | “localhost:3000/user/login” |
| Phương thức | POST |
| Dữ liệu đầu vào | user |
| Dữ liệu đầu ra | Thành công:  {  message: "ok",  user: user  } |
| Thất bại không tồn tài khoản  {message: ‘Không tồn tại tài khoản'} |
| Thất bại sai mật khẩu  { message: ‘Mật khẩu không đúng'} |
| Mô tả | Dùng để đăng nhập vào hệ thống |

Bảng 1.6: API đăng nhập

1.4.2 Các API về bảng job

- API lấy tất cả job (phía người dùng không đăng nhập, admin)

|  |  |
| --- | --- |
| Đường dẫn | “localhost:3000/job/list” |
| Phương thức | POST |
| Dữ liệu đầu vào | {‘page’:1, ‘display’:5, text: ‘finding’} |
| Dữ liệu đầu ra | Thành công:  {  message: "ok",  jobs: [],  \_meta: {  total: 1,  num\_pages: 1,  current\_page: 1,  previous\_page: 1,  next\_page: 1,  num\_per\_page: 5  }  } |
| Mô tả | Dùng để lấy danh sách công việc |

Bảng 1.7: API lấy danh sách công việc (khách và admin)

- API lấy tất cả job (phía người sử dụng có đăng nhập)

|  |  |
| --- | --- |
| Đường dẫn | “localhost:3000/job/list/user/:userId” |
| Phương thức | POST |
| Dữ liệu đầu vào | {‘page’:1, ‘display’:5 }, |
| Dữ liệu đầu ra | Thành công:  {  message: "ok",  jobs: [],  \_meta: {  total: 1,  num\_pages: 1,  current\_page: 1,  previous\_page: 1,  next\_page: 1,  num\_per\_page: 5  }  } |
| Mô tả | Dùng để lấy danh sách công việc |

Bảng 1.8: API lấy danh sách công việc (user đăng nhập)

- API lấy một công việc cụ thể

|  |  |
| --- | --- |
| Đường dẫn | “localhost:3000/job/list/:id” |
| Phương thức | GET |
| Dữ liệu đầu vào |  |
| Dữ liệu đầu ra | Thành công:  {  message: "ok",  job: job  } |
| Mô tả | Dùng để lấy công việc cụ thể |

Bảng 1.9: API lấy công việc cụ thể

- API lấy một công việc cụ thể

|  |  |
| --- | --- |
| Đường dẫn | “localhost:3000/job/update” |
| Phương thức | PUT |
| Dữ liệu đầu vào | user |
| Dữ liệu đầu ra | Thành công:  {  message: "ok",  } |
| Mô tả | Cập nhật công việc |

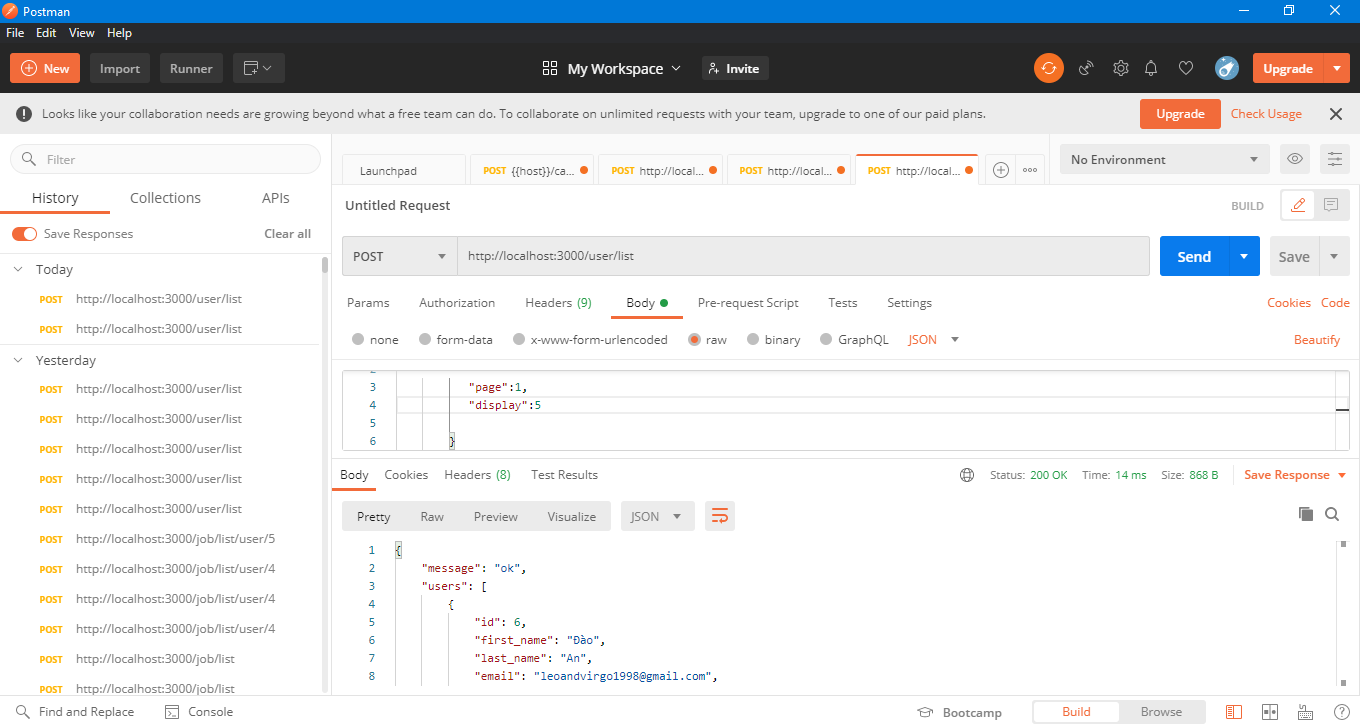
Bảng 1.10: API cập nhật công việc

- API xóa một công việc cụ thể

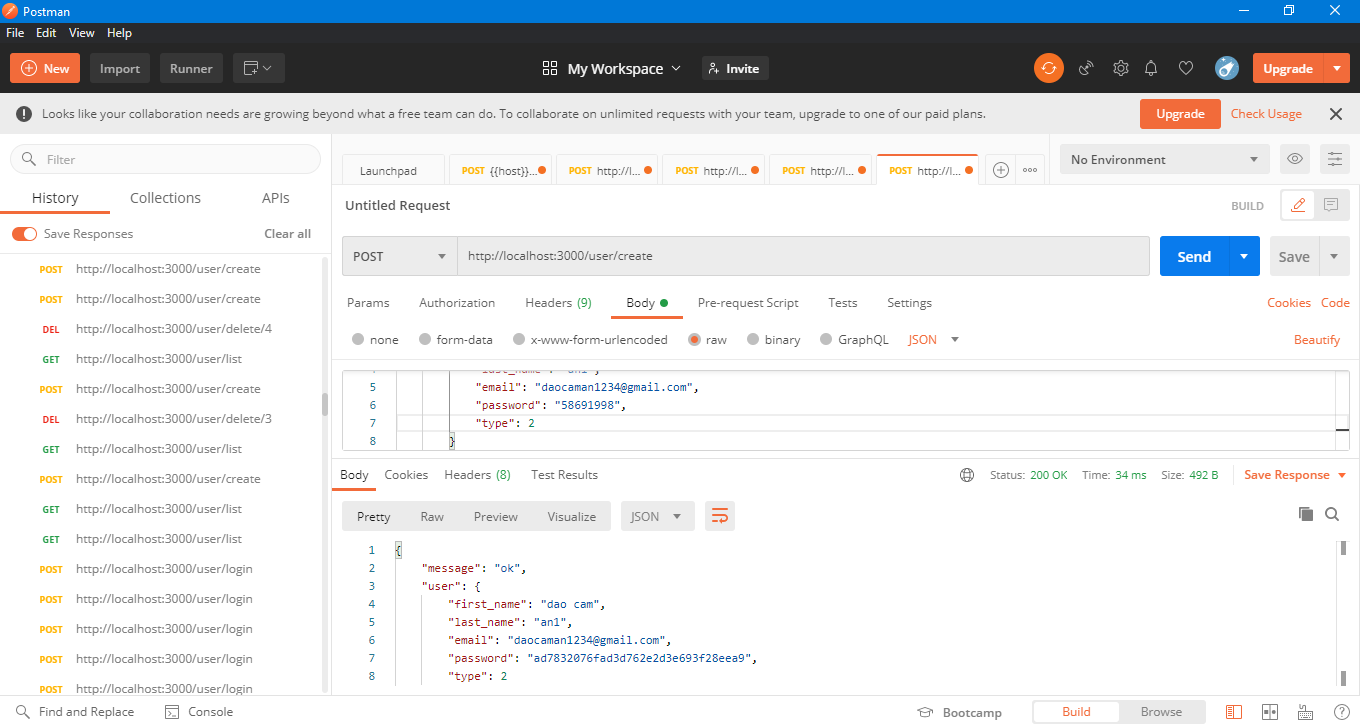
|  |  |
| --- | --- |
| Đường dẫn | “localhost:3000/job/delete/:id” |
| Phương thức | DELETE |
| Dữ liệu đầu vào |  |
| Dữ liệu đầu ra | Thành công:  {  message: "delete success",  } |
| Mô tả | Xóa công việc |

Bảng 1.11: API xóa công việc

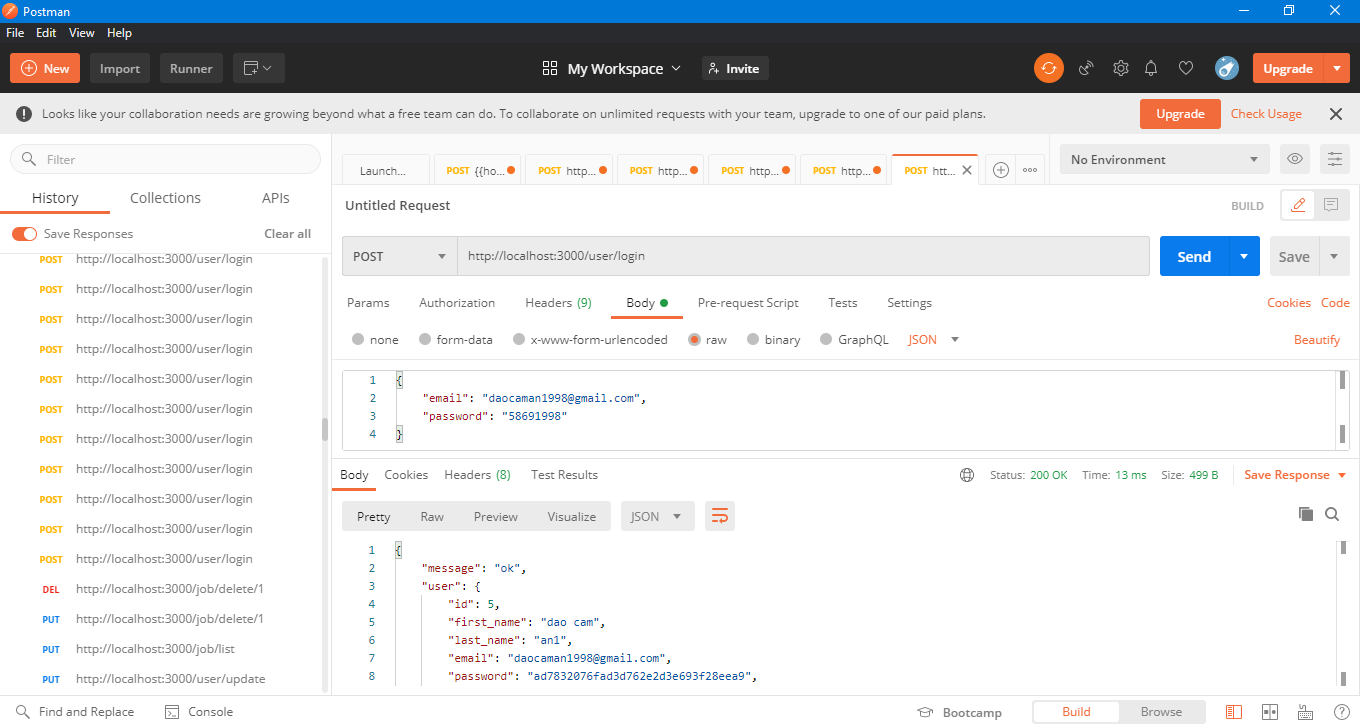
* 1. Các hình ảnh chạy API



Hình 1.1: Demo API danh sách user



Hình 1.2: Demo API tạo user



Hình 1.3: Demo API đăng nhập

**TÀI LIỆU THAM KHẢO**

**Tiếng Việt**

1. <https://topdev.vn/blog/web-server/>

**Tiếng Anh**

1. <https://whatis.techtarget.com/definition/Web-server>
2. <https://dev.mysql.com/doc/refman/8.0/en/what-is-mysql.html>
3. <https://dev.mysql.com/doc/refman/8.0/en/data-type-overview.html>
4. <https://www.journaldev.com/7462/node-js-architecture-single-threaded-event-loop>
5. <https://www.altexsoft.com/blog/engineering/the-good-and-the-bad-of-node-js-web-app-development/>
6. <https://github.com/expressjs/express>
7. <https://techtalk.vn/tat-tan-tat-ve-api.html>
8. <https://topdev.vn/blog/restful-api/>
9. <https://techtalk.vn/ban-co-chac-la-da-hieu-ro-ve-api.html>
10. <https://movan.vn/rest-api-gioi-thieu-rest-api/>
11. <https://restfulapi.net/>
12. <https://angular.io/>