

Phần mềm quản lý công việc

Todo

Mục lục

Mục Tiêu:	1
Công nghệ:	1
Chức Năng:	1
Thiết lập:	2
1.Database:	2
2.Backend:	3
3.Frontend:	4
Khởi chạy ứng dụng:	5
1.backend:	5
Frontend:	6

Mục Tiêu:

Phát triển một ứng dụng web Todo đơn giản với giao diện người dùng phía front-end được xây dựng bằng React hoặc Vue, back-end được xây dựng bằng Node.js và Express, và hỗ trợ quản lý người dùng cùng hệ thống đăng nhập/đăng xuất.

Công nghệ:

- **Front-end:** React.js
- **Back-end:** Node.js, Express
- **Cơ sở dữ liệu:** MongoDB
- **Xác thực:** JWT (JSON Web Tokens)
- **công cụ:** visual studio code, mongodb, mongoDB Compass

Chức Năng:

- **Quản lý người dùng:**
 - Người dùng có thể đăng ký, đăng nhập và đăng xuất.
 - Thông tin người dùng bao gồm ít nhất tên người dùng và mật khẩu.
 - Mật khẩu cần được mã hóa trước khi lưu trữ trong cơ sở dữ liệu.

- **Quản lý công việc:**
 - Người dùng có thể tạo, xem, chỉnh sửa và xóa các công việc sau khi đăng nhập.
 - Mỗi công việc bao gồm tiêu đề và trạng thái hoàn thành.
 - Có khả năng lọc các công việc dựa trên trạng thái hoàn thành (All, Done Pending).
- **Đa người dùng:**
 - Hỗ trợ đa người dùng
 - Mỗi người dùng chỉ có thể truy cập và quản lý các công việc của riêng mình..

Thiết lập:

cài đặt visual studio code và các công nghệ node, npm và mongodb để có thể khởi chạy dự án

1.Database:

- Sử dụng MongoDB để quản trị cơ sở dữ liệu
- Khởi chạy MongoDB trên môi trường phát triển tại cổng **mongodb://localhost:27017/**

Cơ sở dữ liệu(Database):

- Cơ sở dữ liệu gồm 2 bảng:

Bảng người dùng:

- **_id** (string): mã người dùng, làm khóa chính
- **fullname** (string): tên người dùng
- **email** (string): email người dùng
- **username** (string): tên đăng nhập
- **password** (string): mật khẩu
- **mobile** (string): số điện thoại

trong đó **username** và **email** có ràng buộc unique, để không người dùng nào có tên đăng nhập và email giống nhau

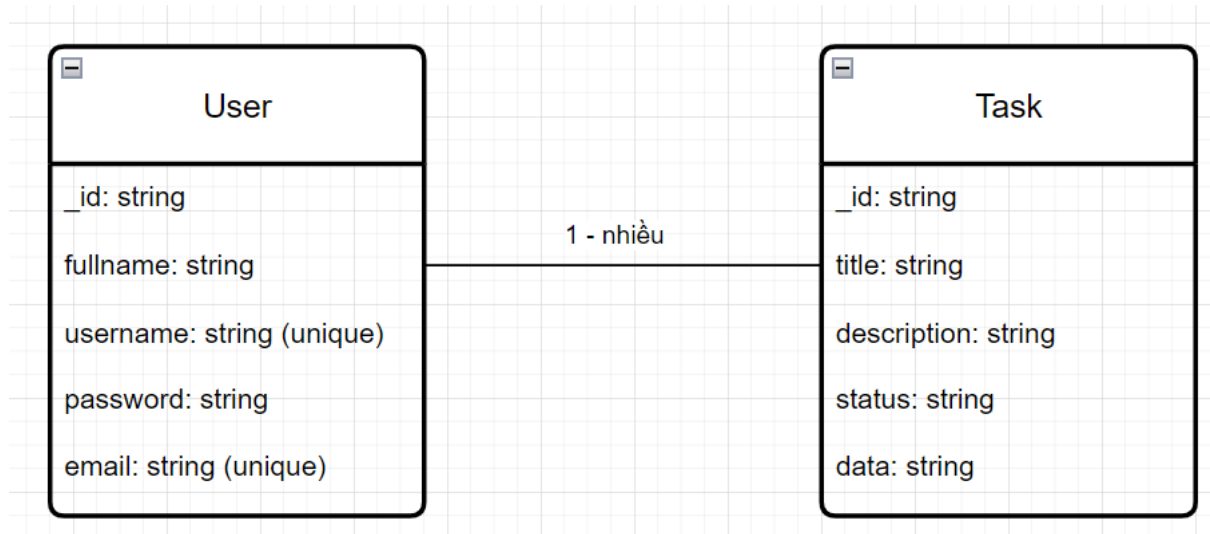
quan hệ: 1 users có thể có nhiều task

Bảng công việc:

- **_id** (string): mã công việc, làm khóa chính
- **title** (string): tiêu đề
- **description** (string): mô tả công việc
- **status** (string): trạng thái, gồm 2 trạng thái là Pending và Done
- **date** (string): thời gian tạo

quan hệ: một task chỉ ứng với 1 user

Diagram:



2.Backend:

- Công nghệ: nodejs, expressjs để xây dựng hệ thống API
- hệ thống api gồm các api liên quan đến:

Người dùng:

- **Khởi tạo người dùng:** người dùng sẽ đăng kí tài khoản trên hệ thống để có thể sử dụng các chức năng trong hệ thống
- **Lấy dữ liệu người dùng:** khi đăng nhập vào hệ thống thành công, hệ thống sẽ lấy thông tin của người dùng theo mã id đã được tạo tự động trong cơ sở dữ liệu
- **Cập nhật thông tin người dùng:** người dùng có thể cập nhật thông tin người dùng trong tab profile, thông tin người dùng có thể cập nhật gồm tên đầy đủ, email, số điện thoại
- **Lấy danh sách công việc người dùng:** tại tab Task hệ thống sẽ lấy danh sách các công việc của người dùng đã đăng nhập

Đăng nhập:

- **Đăng nhập:** người dùng sẽ đăng nhập bằng tài khoản đã đăng kí, khi người dùng đăng nhập thành công hệ thống sẽ trả về token gồm userId và thời gian sử dụng của token đã được mã hóa, và mã token này được sử dụng để hệ thống front-end call đến các api liên quan đến thông người dùng và công việc

Công việc:

- **Khởi tạo công việc:** người dùng có thể tạo mới công việc của mình
- **Cập nhật thông tin công việc:** người dùng có thể cập nhật thông tin công việc
- **Xóa công việc:** người dùng có thể xóa công việc
- **Lọc công việc:** người dùng có thể lọc danh sách công việc theo tiêu đề hoặc trạng thái

3.Frontend:

- công nghệ: reactjs, vitejs, taiwind, các thư viện ui của react

Khởi tạo các trang:

Đăng nhập:

- trang đăng nhập gồm form điền thông tin đăng nhập
- khi đăng nhập thành công người dùng sẽ chuyển hướng đến trang quản lí công việc

Đăng ký:

- nếu người dùng chưa có tài khoản, người dùng sẽ phải điền form đăng kí trên trang đăng kí
- khi đăng kí thành công người dùng sẽ được chuyển hướng đến trang đăng nhập để đăng nhập vào hệ thống

Quản lý công việc:

- trên trang quản lí công việc người dùng có thể thêm, sửa, xóa, tìm kiếm công việc
- nếu người dùng chưa đăng nhập hoặc hết phiên đăng nhập truy cập đến trang quản lí công việc sẽ bị đẩy đến trang đăng nhập

Thông tin người dùng:

- tại trang thông tin người dùng có thể xem, sửa thông tin người dùng
- nếu người dùng chưa đăng nhập hoặc hết phiên đăng nhập truy cập đến trang thông tin người dùng sẽ bị đẩy đến trang đăng nhập

Đăng xuất:

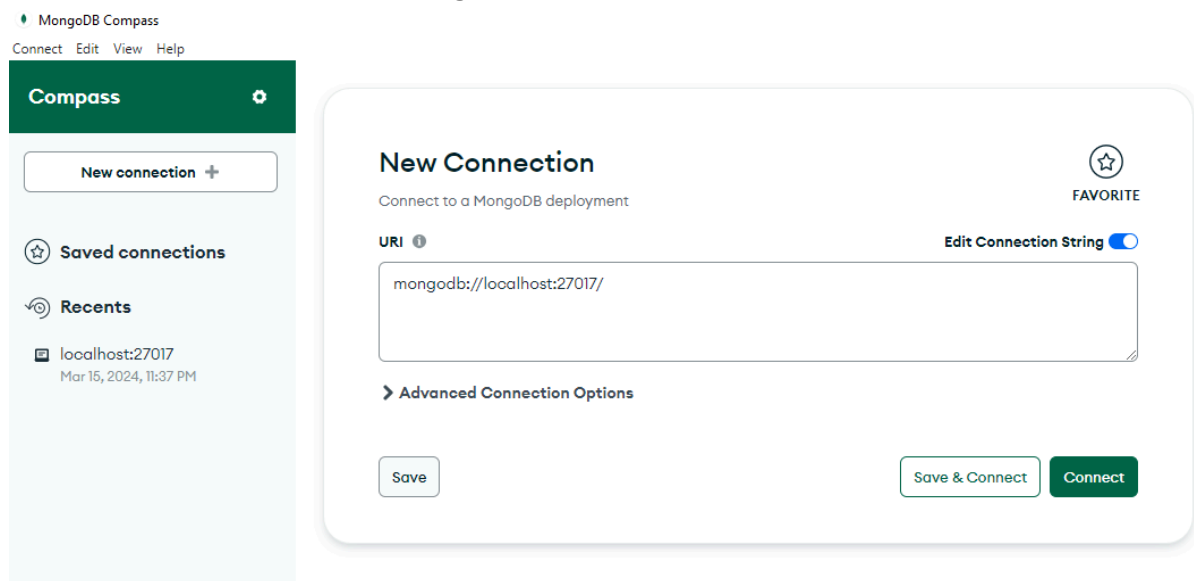
khi người dùng ấn nút đăng xuất ở góc phải màn hình, thì hệ thống sẽ xóa token và thông tin người dùng sau đó chuyển hướng đến trang đăng nhập để có thể đăng nhập tài khoản

khác

Khởi chạy ứng dụng:

mở ứng dụng visual studio code truy cập vào thư mục dự án database:

mở ứng dụng mongo compass để khởi tạo connection đến database
khởi tạo connection với url là **mongodb://localhost:27017**



1.backend:

Bước 1: mở terminal và trở đến đường dẫn toto-test\backend

Bước 2: sử dụng câu lệnh **npm install** để cài đặt node-modules

```
PS G:\TodoTest\toto-test\backend> npm install

up to date, audited 159 packages in 784ms

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

found 0 vulnerabilities
```

Bước 3: mở file .env trong dự án để thay đổi đường dẫn đến database

thay đổi `DB_CONNECTION_STRING` thành uri connection đã tạo trên mongodb và thêm `/todo` để trở đến database

```
PORT=3001
DB_CONNECTION_STRING=mongodb://0.0.0.0:27017/todo
```

Bước 3: chạy tiếp câu lệnh ***npm run start*** để khởi chạy dự án

```
PS G:\TodoTest\toto-test\backend> npm run start

> backend@1.0.0 start
> node ./app.js

(node:17688) [MONGODB DRIVER] Warning: useNewUrlParser is a
(Use `node --trace-warnings ...` to show where the warning
(node:17688) [MONGODB DRIVER] Warning: useUnifiedTopology i
Server is running on port 3001
Connected to MongoDB
```

sau khi server chạy trên port 3001 và connect tới được mongodb là đã thành công chạy backend

có thể nhập liệu dữ liệu test bằng cách chạy ***node seeddata.js***

Frontend:

sau khi chạy backend và connect thành công đến database, có thể khởi chạy hệ thống frontend để thao tác

Bước 1: mở terminal và trở đến đường dẫn toto-test\frontend

Bước 2: sử dụng câu lệnh ***npm install*** để cài đặt node-modules

```

PS G:\TodoTest\toto-test\frontend> npm i

up to date, audited 489 packages in 1s

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

1 high severity vulnerability

To address all issues, run:
  npm audit fix --force

Run `npm audit` for details.

```

Bước 3: chạy câu lệnh ***npm run dev*** để khởi chạy hệ thống, hệ thống sẽ chạy trên cổng 3000

```

PS G:\TodoTest\toto-test\frontend> npm run dev

> todo-dashboard@2.1.0 dev
> vite

VITE v4.5.0 ready in 339 ms

→ Local:   http://localhost:3000/
→ Network: use --host to expose
→ press h to show help

```

truy cập trình duyệt với url: <http://localhost:3000/> để mở ứng dụng

Sign In

Enter your email and password to Sign In.

Your username

user1

Password

.....

SIGN IN

☐ Subscribe me to newsletter
 [Forgot Password](#)

Not registered? [Create account](#)



Thao tác:

- sau khi người dùng khởi chạy và truy cập url <http://localhost:3000/> lần đầu hệ thống sẽ đẩy người dùng ra trang login

Sign In

Enter your email and password to Sign In.

Your username

Password

SIGN IN

☐ Subscribe me to newsletter [Forgot Password](#)

[Not registered? Create account](#)



- nếu chưa có tài khoản người dùng có thể ấn nút **Create account** để đi đến trang đăng kí

Sign In

Enter your email and password to Sign In.

Your username

Password

SIGN IN

☐ Subscribe me to newsletter [Forgot Password](#)

[Not registered? Create account](#)

- khi đến trang đăng kí người dùng có thể nhập thông tin để đăng kí tài khoản
- người dùng đăng kí tài khoản với điều kiện email và username chưa được người dùng nào khác sử dụng

Join Us Today

Enter your email and password to register.

Your email

Your Username

Your Password

Confirm Password

REGISTER NOW

Already have an account? [Sign in](#)

sau khi đăng kí thành công người dùng sẽ được chuyển hướng đến trang đăng nhập để tiến hành đăng nhập

- sau khi đăng nhập thành công người dùng sẽ được chuyển hướng đến trang quản lí công việc
- trên trang quản lí công việc người dùng có thể xem, sửa, xóa, tìm kiếm công việc
- xem danh sách

Authors Table					
ADD NEW TASK		Search	All		
TITLE	DESCRIPTION	STATUS		DATE	
Task 1	Description of Task 1	PENDING		19/03/2024 23:17:49	Edit Delete
Task 2	Description of Task 2	PENDING		19/03/2024 23:17:49	Edit Delete
Task 3	Description of Task 3	PENDING		19/03/2024 23:17:49	Edit Delete
Task 4	Description of Task 4	PENDING		19/03/2024 23:17:49	Edit Delete

- tạo công việc mới

Create Task

Status

Pending

title

description

Create

cancel

- chỉnh sửa công việc

All

Task info

Status

Pending

Title

Task 1

description


Description of Task 1

update

cancel

- xóa công việc

TITLE	DESCRIPTION	STATUS	DATE		
Task 1	Description of Task 1	PENDING	19/03/2024 23:17:49	Edit	Delete
Task 3	Description of Task 3	PENDING	19/03/2024 23:17:49	Edit	Delete
Task 4	Description of Task 4	PENDING	19/03/2024 23:17:49	Edit	Delete

© 2024, made with  by Creative Tim for a better web.

Creative Tim

About Us

Blog

- tìm kiếm công việc theo tiêu đề hoặc trạng thái


<div>Search</div>	<div>All</div> <div> <div>All</div> <div>✓ Pending</div> <div>Done</div> </div>	
DESCRIPTION		STATUS
Description of Task 1		PENDING
Description of Task 3		PENDING
Description of Task 4		PENDING

- người dùng có thể chuyển hướng đến trang thông tin để xem và chỉnh sửa thông tin người dùng

Todo

Profile

Tasks



User 1

Profile Information

Hi, I'm Alec Thompson, Decisions: If you can't decide, the answer is no. If two equally difficult paths, choose the one more painful in the short term (pain avoidance is creating an illusion of equality).

Name:

User 1

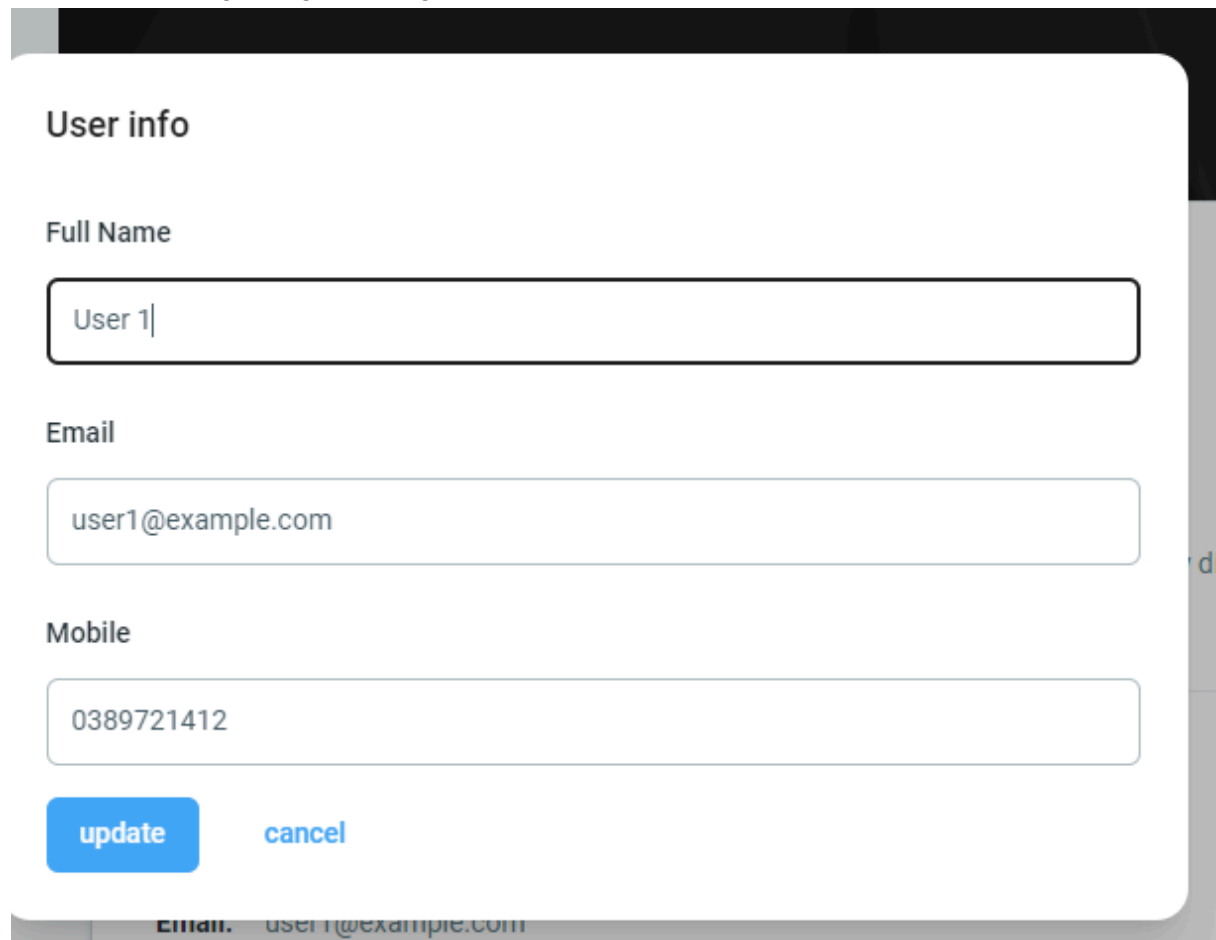
Mobile:

0389721412

Email:

user1@example.com

- chỉnh sửa thông tin người dùng



A screenshot of a 'User info' form. The form has a title 'User info' at the top. Below it are three input fields: 'Full Name' with the value 'User 1', 'Email' with the value 'user1@example.com', and 'Mobile' with the value '0389721412'. At the bottom of the form are two buttons: a blue 'update' button and a 'cancel' button.

User info

Full Name

User 1

Email

user1@example.com

Mobile

0389721412

update cancel

người dùng có thể đăng xuất ra khỏi ứng dụng để đăng nhập tài khoản khác

