1. File App.vue

Có 3 mục:

* Template: Giao diện chính của ứng dụng.
* Script: Xử lý logic của ứng dụng.
* Style: Thiết lập giao diện.

Template:



**<header-app>**:

* Chỉ hiển thị khi showHeader = true (khi user đã đăng nhập).
* Nhận sự kiện @logout để gọi hàm hideHeader (ẩn header).

**<router-view>**:

* Nơi hiển thị các trang con theo Vue Router (dùng để định hướng).

**<ChatBox>**:

* Chỉ hiển thị khi isOpenChat = true.
* Nhận các props dataFriend và isOpenChat truyền vào.
* Sử dụng v-model để gán giá trị message.
* Có các sự kiện:
  + @clearMessage: Xóa nội dung message (cho giá trị message = rỗng).
  + @exitChat: Gọi hàm exitChat.

**<Spinner>**:

* Hiển thị khi isSpinner = true (hiển thị hiệu ứng loading trang lấy dữ liệu).

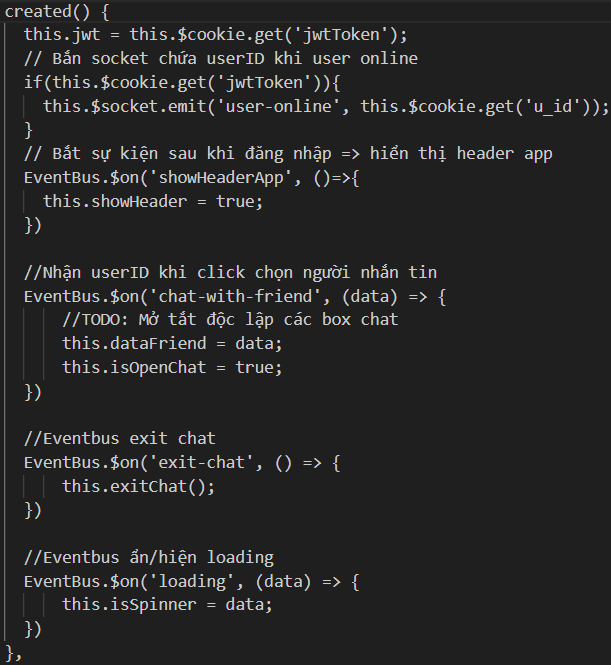
Script:



Các import để lấy dữ liệu của các thư viện và gọi tới các components để sử dụng, export để xuất các file ra cho các file khác nếu cần sử dụng thì gọi tới.

**data()**:

* showHeader: Điều khiển việc hiển thị header-app, đang là false nên sẽ không hiển thị.
* jwt: Lưu JWT Token của user, đang là rỗng.
* message: Nội dung tin nhắn trong chatbox, mặc định là rỗng (với các tài khoản mới).
* isOpenChat: Xác định chatbox có mở hay không, đang là false nên không hiển thị.
* dataFriend: Lưu thông tin bạn bè đang chat, mặc định là rỗng.
* isSpinner: Điều khiển hiệu ứng loading, mặc định là false.

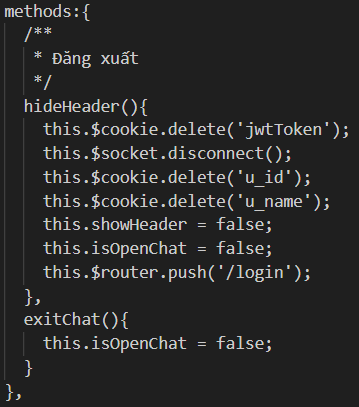


**JWT Token**:

* Lấy token từ cookie của web, nếu có thì gửi sự kiện user-online (người dùng đang online) qua WebSocket.

**EventBus - Bắt sự kiện giữa các component**:

* showHeaderApp: Hiển thị header khi user đăng nhập.
* chat-with-friend: Nhận thông tin và lấy dữ liệu bạn bè đang chat và mở chatbox.
* exit-chat: Đóng chatbox.
* loading: Hiển thị hoặc ẩn spinner.

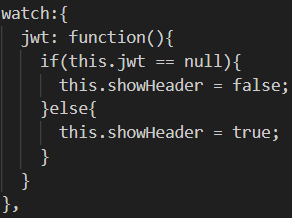


**hideHeader()** (Xử lý đăng xuất)

* Xóa cookies liên quan đến user.
* Ngắt kết nối WebSocket.
* Ẩn header đi.
* Chuyển hướng về trang đăng nhập.

**exitChat()** (Đóng chatbox)

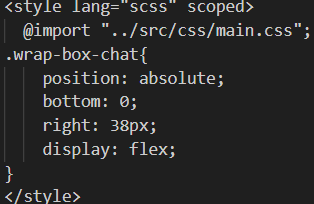
* Đặt isOpenChat = false.



**Theo dõi thay đổi của jwt**:

* Nếu jwt bị xóa (user đăng xuất), showHeader = false, ẩn header.
* Nếu jwt có giá trị, showHeader = true, hiện header.

Style:

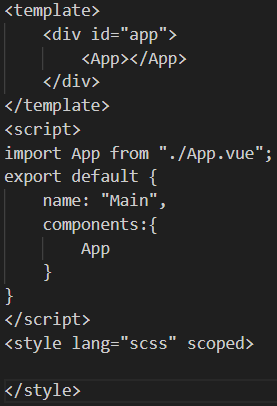


Import file CSS main.css.

**wrap-box-chat**:

* Vị trí **absolute**, đặt chatbox ở góc phải dưới màn hình.

1. File Main.vue



Dùng để bọc toàn bộ ứng dụng và render file App.vue.

Template:

* **<div id="app">**: Thẻ chứa toàn bộ ứng dụng Vue (thường là phần tử root trong index.html).
* **<App></App>**: Gọi component App.vue, thành phần chính của ứng dụng.

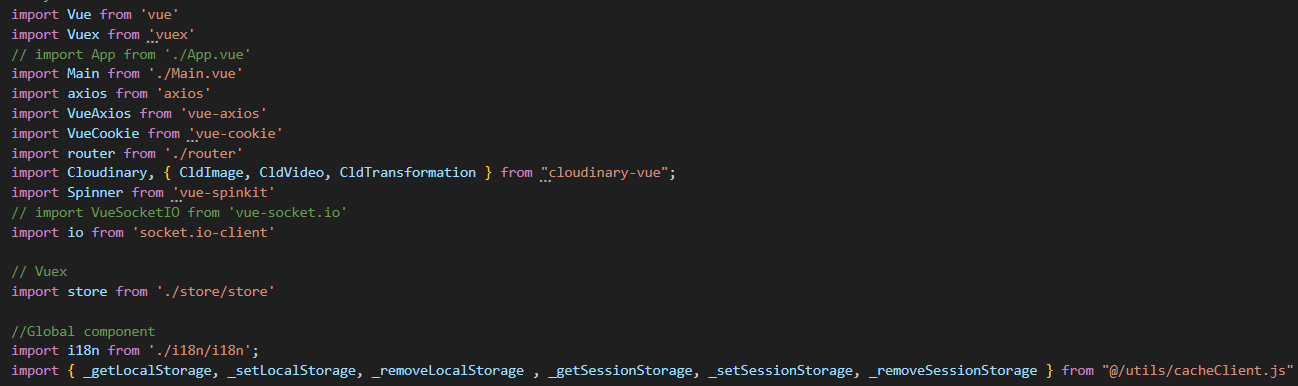
Script:

* **Import App.vue** để sử dụng trong component này.

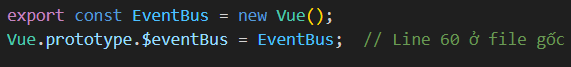
Style:

* **lang="scss"**: Cho phép sử dụng SCSS thay vì CSS thuần.
* **scoped**: CSS chỉ áp dụng cho component này, không ảnh hưởng toàn bộ ứng dụng.

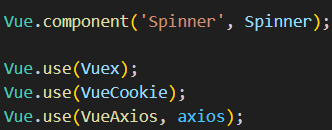
1. File main.js



Import các thư viện cần sử dụng



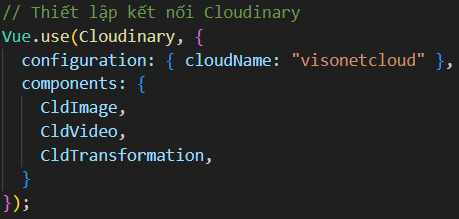
* **EventBus** là một đối tượng Vue dùng để truyền dữ liệu giữa các component không liên quan trực tiếp.
* Gán EventBus vào prototype để có thể gọi trong toàn bộ ứng dụng thông qua this.$eventBus.



Đăng ký component Spinner là global component.

Kích hoạt các plugin Vue như:

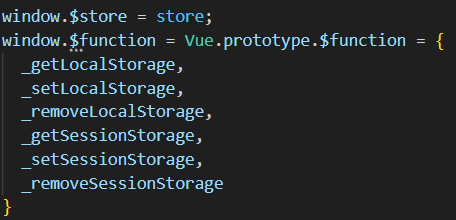
* Vuex (quản lý trạng thái).
* VueCookie (quản lý cookie).
* VueAxios (gọi API).



Thiết lập cấu hình kết nối với Cloudinary bằng cloudName.

Đăng ký các component của Cloudinary như:

* CldImage: Hiển thị hình ảnh.
* CldVideo: Hiển thị video.
* CldTransformation: Biến đổi hình ảnh (resize, crop, filter).

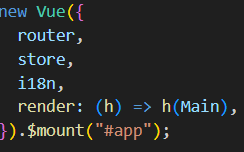


Gọi tới store để lưu trữ các dữ liệu của state và props.

Đăng ký các hàm lưu trữ cục bộ (**localStorage** và **sessionStorage**) làm global function.



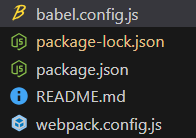
* Tạo kết nối WebSocket tới server tại <http://localhost:3000>.
* **$socket** có thể được sử dụng trong các component qua this.$socket.
* Chỉnh config.productionTip = false để ngăn console cảnh báo khi đang chạy trong môi trường Production.



* **router**: Khai báo Vue Router.
* **store**: Quản lý trạng thái với Vuex.
* **i18n**: Đa ngôn ngữ.
* **render**: Render component Main.vue và mount vào phần tử có id="app".

1. File .gitignore

Ngăn không cho các file được gọi tới được push lên repo trong git.

1. 

Các file này mục đích để hiển thị các phiên bản của các thư viện sử dụng và được cài đặt khi cần dùng (dạng thế).

Readme.md là file của git để ghi tài liệu hướng dẫn sử dụng.

1. Folder node\_modules

Được cài đặt sau khi npm install để chứa các thư viện được cài đặt để sử dụng.

1. Folder public

Có sẵn khi bắt đầu tạo 1 ứng dụng Vue mới.