A. Quy tắc xác thực:

- Sử dụng header "Authorization"

- Áp dụng cho mọi đường dẫn có dấu \*.

- Định dạng: "Bearer " + jwt

B. Các cấu trúc JSON:

1. Thuộc nhóm phản hồi (response):

- Từ server -> response.Messenger:

{

"content": ...

}

- Một trang chứa object -> response.Page:

{

"current": trang hiện tại

"total": tổng số trang

"limit": giới hạn mỗi trang

"count": tổng số toàn thể

"content": một list các object

}

2. Thuộc nhóm đơn (form):

- Đơn của Article (Thêm và sửa) -> form.ArticleForm:

{

"title": Bắt buộc, không rỗng, tối đa 100 ký tự

"image": Bắt buộc, không rỗng, tối đa 255 ký tự

"shortIntro": Bắt buộc, không rỗng, tối đa 500 ký tự

"content": Bắt buộc, không rỗng

"categoryId": id của category, cần phải tồn tại

"tagIds": một list các id của các tag, cần phải tồn tại, không được lặp lại, không bắt buộc

}

- Đơn của Account (Thêm) -> form.AccountForm:

{

"fullname": Bắt buộc, độ dài 3 - 30

"username": Bắt buộc, độ dài 4 - 15, ký tự đầu tiên là chữ cái in thường, chỉ cho phép chữ in thường và chữ số, phải là độc nhất (unique)

"password": Bắt buộc, dộ dài 6 - 20, ký tự đầu tiên k được là ký tự đặc biệt, chứa ít nhất 1 chữ in thường, in hoa, 1 chữ số và 1 ký tự đặc biệt

}

- Đơn của Comment (thêm và sửa) -> form.CommentForm:

{

"content": Nội dung

}

3. Thuộc nhóm cấu trúc hiển thị (sendover):

- Một article rút gọn (dùng khi query nhiều) -> sendover.article.TruncatedArticleSendover:

{

"id":

"title":

"image":

"shortIntro":

"dateCreated":

"account": {

"id":

"fullname":

"role":

}

"commentCount": Tổng số comment liên quan trong bài báo

}

C. Các đường dẫn:

1. Đường dẫn gốc: /test

2. Đường dẫn của Article: /articles

\* **GET** - /:articleId/get

- Trả về một bài báo chi tiết, status 200 OK

- Ngoại lệ:

+ Nếu k tồn tại, trả về null, status 404 NOT FOUND

\* **GET** - /index (page, limit)

- Các biến param:

+ page: Số trang hiện tại (k bắt buộc, mặc định là 1)

+ limit: Số bài tối đa cho mỗi trang (k bắt buộc, mặc định là 10)

- Trả về một trang các đầu báo (ngắn gọn hơn so với bài báo chi tiết), status 200 OK

- Ngoại lệ:

+ Nếu page <= 0 hoặc limit <= 0, trả về null, status 406 NOT ACCEPTABLE

+ Nếu page vượt quá số trang tối đa, vẫn trả về một trang báo, nhưng content là null, status 400 BAD REQUEST

\* **GET** - /categories/:categoryId (page, limit)

- Các biến param:

+ page: Số trang hiện tại (k bắt buộc, mặc định là 1)

+ limit: Số bài tối đa cho mỗi trang (k bắt buộc, mặc định là 10)

- Trả về một trang các đầu báo thuộc về category có id như trên, status 200 OK

- Ngoại lệ:

+ Nếu page <= 0 hoặc limit <= 0, trả về null, status 406 NOT ACCEPTABLE

+ Nếu category không tồn tại, trả về null, status 404 NOT FOUND

+ Nếu page vượt quá số trang tối đa, vẫn trả về một trang báo, nhưng content là null, status 400 BAD REQUEST

\* **GET** - /tags/:tagId (page, limit)

- Các biến param:

+ page: Số trang hiện tại (k bắt buộc, mặc định là 1)

+ limit: Số bài tối đa cho mỗi trang (k bắt buộc, mặc định là 10)

- Trả về một trang các đầu báo thuộc về tag có id như trên, status 200 OK

- Ngoại lệ:

+ Nếu page <= 0 hoặc limit <= 0, trả về null, status 406 NOT ACCEPTABLE

+ Nếu tag không tồn tại, trả về null, status 404 NOT FOUND

+ Nếu page vượt quá số trang tối đa, vẫn trả về một trang báo, nhưng content là null, status 400 BAD REQUEST

\* **GET** - /accounts/:accountId (page, limit)

- Các biến param:

+ page: Số trang hiện tại (k bắt buộc, mặc định là 1)

+ limit: Số bài tối đa cho mỗi trang (k bắt buộc, mặc định là 10)

- Trả về một trang các đầu báo thuộc về tag có id như trên, status 200 OK

- Ngoại lệ:

+ Nếu page <= 0 hoặc limit <= 0, trả về null, status 406 NOT ACCEPTABLE

+ Nếu account không tồn tại, trả về null, status 404 NOT FOUND

+ Nếu page vượt quá số trang tối đa, vẫn trả về một trang báo, nhưng content là null, status 400 BAD REQUEST

\* **GET** - /:articleId/relevant (limit)

- Các biến param:

+ limit: Số bài tối đa định lấy (k bắt buộc, mặc định là 3)

- Trả về một trang các đầu báo thuộc về tag có id như trên, status 200 OK

- Ngoại lệ:

+ Nếu page <= 0 hoặc limit <= 0, trả về null, status 406 NOT ACCEPTABLE

+ Nếu bài báo không tồn tại, trả về null, status 404 NOT FOUND

\* **PUT** - /insert (\*)

- Cấu trúc body:

+ ArticleForm

+ Nếu tagIds để rỗng hoặc null, tức là sẽ k thêm tag cho bài báo

- Trả về một bài báo mới thêm, status 201 CREATED

- Ngoại lệ:

+ Nếu form vi phạm 1 trong các điều trên, trả về content là tin nhắn, status 400 BAD REQUEST

+ Nếu không thể thêm được bài báo vào CSDL, trả về null, status 500 INTERNAL SERVER ERROR

\* **PATCH** - /:articleId/edit (\*)

- Cấu trúc body:

+ ArticleForm

+ Nếu tagIds để null, tức là k chỉnh sửa tag của bài báo

- Trả về bài báo vừa mới chỉnh sửa, status 200 OK

- Ngoại lệ:

+ Nếu bài báo k tồn tại, trả về null, status 404 NOT FOUND

+ Nếu bài báo k phải do tài khoản gửi request lên viết (so sánh id), trả về null, status 403 FORBIDDEN

+ Nếu form vi phạm 1 trong các điều trên, trả về content là tin nhắn, status 404 BAD REQUEST

+ Nếu chỉnh sửa không thành công, trả về null, status 500 INTERNAL SERVER ERROR

\* **DELETE** - /:articleId/delete (\*)

- Xóa một bài báo, trả về bài báo trước khi bị xóa, status 200 OK

- Ngoại lệ:

+ Nếu bài báo k tồn tại, trả về null, status 404 NOT FOUND

+ Nếu bài báo k phải do tài khoản gửi request lên viết (so sánh id), trả về null, status 403 FORBIDDEN

+ Nếu xóa không thành công, trả về null, status 500 INTERNAL SERVER ERROR

2. Đường dẫn của Category: /categories

\* GET - /basic/index

- Trả về toàn bộ các category, status 200 OK

3. Đường dẫn liên quan tới tag: /tags

\* GET - /basic/index

- Trả về toàn bộ các tag, status 200 OK

4. Đường dẫn của Comment: /comments

\* GET - /articles/:articleId (page, limit)

- Biến param:

+ page: Trang hiện tại

+ limit: Giới hạn mỗi trang

- Trả về một trang comment của bài báo có id trên, status 200 OK

- Ngoại lệ:

+ page hoặc limit <= 0 -> null, 406

+ page vượt quá total -> Page có content là null, 400

\* PUT - /articles/:articleId/insert (\*)

- Cấu trúc body: CommentForm

- Trả về comment mới thêm nếu thành công, status 201 CREATED

- Ngoại lệ:

+ Article k tồn tại -> null, 404

+ Form vi phạm -> tin nhắn, 400

+ Thêm thất bại -> null, 500

\* PATCH - /:commentId/edit (\*)

- Cấu trúc body: CommentForm

- Trả về comment mới sửa nếu thành công, status 200 OK

- Ngoại lệ:

+ Comment k tồn tại -> null, 400

+ K đúng account -> null, 403

+ Form vi phạm -> tin nhắn, 400

+ Sửa thất bại -> null, 500

\* DELETE - /:commentId/delete (\*)

- Trả về comment mới xóa nếu thành công, status 200 OK

- Ngoại lệ:

+ Comment k tồn tại -> null, 400

+ K đúng account -> null, 403

+ Xóa thất bại -> null, 500

5. Đường dẫn của Account: /accounts

\* PUT - /insert

- Cấu trúc body: AccountForm

- Trả về một Account (k chứa password) nếu thành công, status 201 CREATED

- Mã xác thực jwt sẽ được đính kém vào header

- Ngoại lệ:

+ Form vi phạm -> Tin nhắn, 400

+ Thêm thất bại -> null, 500

\* POST - /auth

- Cấu trúc body: LoginForm

- Trả về một Account (k chứa password) nếu thành công, status 200 OK

- Mã xác thực jwt sẽ được đính kém vào header

- Ngoại lệ:

+ Thông tin sai -> null, 400

D. Các lưu ý:

- Do hiện tại đã triển khai password hash, các password cũ do chưa sửa lại nên muốn test phải tự tạo account mới.