

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC HOA SEN
KHOA CÔNG NGHỆ THÔNG TIN**

ĐỒ ÁN CHUYÊN NGÀNH A

Tên đề tài: *Xây dựng website hỗ trợ chia sẻ và hỏi đáp kiến thức về lập trình*

Giảng viên hướng dẫn : Phạm Thị Kim Ngôn

Lớp : PM1511

Thời gian thực hiện : Từ 26/07/2021 đến 29/10/2021

Nhóm sinh viên thực hiện : Trần Xuân Hoàng

MSSV: 2152205

Số nhóm : 01

THÁNG 10 / NĂM 2021

TRÍCH YẾU

Trong thời đại công nghệ hiện nay, nhu cầu của thế giới về lập trình viên giỏi đang càng ngày càng tăng cao. Chính vì thế, việc tiếp thu kiến thức phần mềm là nhu cầu vô cùng thiết yếu của những người đang muốn đi theo ngành lập trình hay những lập trình viên tập sự đang cần trau dồi kiến thức của mình. Việc lập diễn đàn để những lập trình viên có kinh nghiệm chia sẻ kiến thức học được, cộng với nhu cầu truy vấn kiến thức lập trình sẽ trở nên vô cùng quý giá cho những bạn đang mong muốn theo đuổi sự nghiệp của một lập trình viên.

Nắm bắt được nhu cầu đó, khoa Khoa học và Công Nghệ đã đề ra đề án phát triển Web “Xây dựng website hỗ trợ chia sẻ và hỏi đáp kiến thức về lập trình” cho sinh viên ngành công nghệ thông tin và kỹ thuật phần mềm có cơ hội học hỏi thêm những kiến thức về phát triển ứng dụng Web và xây dựng một website QnA phù hợp với môi trường Web.

Mục tiêu của tôi là tạo ra “website hỗ trợ chia sẻ và hỏi đáp kiến thức về lập trình” để có thể hỗ trợ phần nào cho việc tìm tòi, học hỏi kiến thức lập trình của những bạn có đam mê về công nghệ, cho phép các bạn sử dụng website tiếp cận kiến thức phần mềm qua những bài viết liên quan đến các chủ đề lập trình, được sắp xếp theo các chủ đề khác nhau. Đồng thời cũng cho phép họ đặt câu hỏi liên quan đến kiến thức lập trình, cho phép những người dùng khác trên website có thể thảo luận, chia sẻ kiến thức để giải quyết những vấn đề khó.

Bởi vì ứng dụng web được sử dụng toàn cầu, tôi đã chọn kết hợp những công nghệ, nền tảng front-end và sử dụng nền tảng ASP.NET Core được viết trên C# làm back-end cho đề án này.

LỜI CẢM ƠN

Thành công lớn nhất trong lần thực hiện đồ án lần này không chỉ đơn thuần dừng lại ở việc thành công về mặt điểm số, tín chỉ,...mà là việc gặt hái được nhiều kiến thức cũng như kinh nghiệm, tích lũy kiến thức để có thể ứng dụng vào công việc thực tiễn.

Và nhân tố quan trọng nhất trong sự thành công này không phải là bản thân của nhóm mà là nhờ vào công lao của giảng viên hướng dẫn.

Lời đầu tiên, xin gửi tới cô **Phạm Thị Kim Ngôn** lời cảm ơn chân thành và sâu sắc. Cảm ơn cô trong quãng thời gian qua đã đồng hành cùng tôi trên con đường gian nan này. Cảm ơn cô dù có những lúc rất bận vẫn sắp xếp thời gian để gặp gỡ, hướng dẫn từng chi tiết để tôi có thể thực hiện và hoàn thành tốt đồ án lần này. Một lần nữa xin chân thành gửi lời cảm ơn sâu sắc đến cô.

Tiếp đến, xin gửi lời cảm ơn tới khoa **Công Nghệ thông tin** đã tạo điều kiện cho tôi có thể tiếp xúc với những kiến thức hoàn toàn mới mẻ mà chưa bao giờ được giảng dạy và học trên giảng đường Đại học. Từ đó nâng cao kỹ năng cũng như tinh thần tự học, tự tìm hiểu, nghiên cứu của tôi. Cuối cùng, xin gửi lời cảm ơn đến các bạn và cũng như các thầy cô khác đã hết lòng hỗ trợ nhóm trong quá trình thực hiện đồ án. Các bạn và thầy cô đã hướng dẫn tôi phương pháp để thực hiện đồ án hiệu quả cũng như có những góp ý, chia sẻ những kinh nghiệm hay về cách sử dụng ngôn ngữ lập trình và công cụ được sử dụng trong đồ án lần này.

Mục lục

TRÍCH YẾU	2
LỜI CẢM ƠN.....	3
Mục lục	4
Danh mục hình ảnh.....	5
Nhập Đề.....	6
1. Tổng quan công nghệ sử dụng.....	7
2. Phân tích đề án.....	11
2.1. Mô tả đề án	11
2.2. Đặc tả yêu cầu đề án	11
2.2.1. Yêu cầu chức năng.....	11
2.2.2. Yêu cầu phi chức năng.....	12
3. Lược đồ use-case.....	13
4. Đặc tả use-case	14
5. Activity Diagram	32
6. Class diagram.....	35
7. Thiết kế database	36
8. Thiết kế giao diện	37
9. Kết luận.....	47
NHẬN XÉT CỦA GIẢNG VIÊN	48

Danh mục hình ảnh

Hình 1: Logo ASP.NET Core	7
Hình 2: Mô Hình MVC	8
Hình 3: Logo EFCore	9
Hình 4: Logo W3CSS	9
Hình 5: Logo Ckeditor.....	10
Hình 6: Logo SQL Server.....	10
Hình 7: Giao diện đăng nhập tài khoản	37
Hình 8: Giao diện chức năng	37
Hình 9: Giao diện trang chủ.....	38
Hình 10: Giao diện một bài viết	39
Hình 11: Bình luận	40
Hình 12: Giao diện đăng bài viết.....	40
Hình 13: Chỉnh sửa, gỡ bỏ bài viết.....	41
Hình 14: Giao diện chỉnh sửa bài viết	41
Hình 15: Giao diện một câu hỏi.....	42
Hình 16: Giao diện đặt câu hỏi	42
Hình 17: thông tin người dùng khác	43
Hình 18: Giao diện chức năng theo dõi	43
Hình 19: Giao diện thịnh hành.....	44
Hình 20: Thông tin người dùng khác.....	44
Hình 21: Giao diện Bookmark.....	45
Hình 22: Giao diện phân loại tìm kiếm.....	45
Hình 23: Giao diện trả kết quả do người dùng tìm kiếm	46
Hình 24: Giao diện lọc theo tag.....	46

Nhập Đề

Bước vào những năm cuối của quãng đời sinh viên ngành kỹ thuật phần mềm, đã dần định hướng được hướng đi cho nghề nghiệp trong tương lai, tôi đã được trường Đại học Hoa Sen, cụ thể là khoa Khoa học và Công nghệ tạo cơ hội được thực hiện một đồ án mang tính thực tiễn cao trong vòng 14 tuần. Việc được thực hiện đồ án này giúp tôi có được những kinh nghiệm quý báu, những kiến thức, kỹ năng cần có để phục vụ cho hướng đi chuyên ngành của mình khi tốt nghiệp mai sau.

Tôi đã được khoa Công Nghệ thông tin và Cô Phạm Thị Kim Ngôn giao cho một đề tài đồ án về phát triển một “website hỗ trợ chia sẻ và hỏi đáp kiến thức về lập trình” trên nền tảng web, tạo điều kiện cho nhóm tôi có cơ hội áp dụng những kiến thức đã học ở trường qua các năm vào thực tế.

Lý do chọn đề tài: Vì đề tài “website hỗ trợ chia sẻ và hỏi đáp kiến thức về lập trình” là một dạng website Q&A phổ biến và đa dụng ở nhiều góc ngách trên internet, một phần vì tôi cũng có kiến thức về lập trình web nên muốn vận dụng các kiến thức có được để thử sức mình vào đề tài lần này. Những mục tiêu mà tôi đã đề ra khi thực hiện đồ án:

Mục tiêu 1: Nâng cao khả năng học hỏi, tự nghiên cứu.

Mục tiêu 2: củng cố vững chắc kiến thức về lập trình, xây dựng website.

Mục tiêu 3: củng cố vững chắc kiến thức về lập trình web cơ sở dữ liệu, phân tích thiết kế hướng đối tượng, kỹ thuật phát triển website với Front-end, Back-end.

Mục tiêu 4: Trang bị kinh nghiệm phát triển phần mềm Website với đề tài mang tính thực tiễn cao.

1. Tổng quan công nghệ sử dụng

Khi tôi nhận đề tài, tôi đã rất phân vân. Do đề tài đi theo hướng web, có rất nhiều lựa chọn liên quan đến các nền tảng phát triển. Nhưng sau khi xem xét tất cả các lựa chọn, tôi quyết định lấy ASP.NET Core làm nền tảng phát triển đề án lần này, một phần vì tôi cũng đã có kinh nghiệm khi làm các project của ASP.NET (ASP.NET Core là tiến hóa của ASP.NET) trước đó, một phần cũng vì tôi muốn phát triển kỹ năng C# của mình. Kèm theo đó là một ORM như Entity Framework Core để hỗ trợ cho việc xử lý dữ liệu dễ dàng. Đề án áp dụng những framework front-end lẫn back-end, gồm có: W3CSS, Ckeditor, ASP.NET Core, Entity Framework Core, SQL Server.

ASP.NET Core

- Framework đa nền tảng (cross-platform), có mã nguồn mở, được phát triển bởi Microsoft, phục vụ cho việc xây dựng những ứng dụng hiện tại dựa trên kết nối đám mây.
- Được thiết kế để cung cấp và tối ưu development framework cho những ứng dụng được triển khai trên đám mây (cloud) hoặc chạy on-premise.
- Bao gồm các thành phần theo hướng module nhằm tối thiểu tài nguyên và chi phí phát triển.

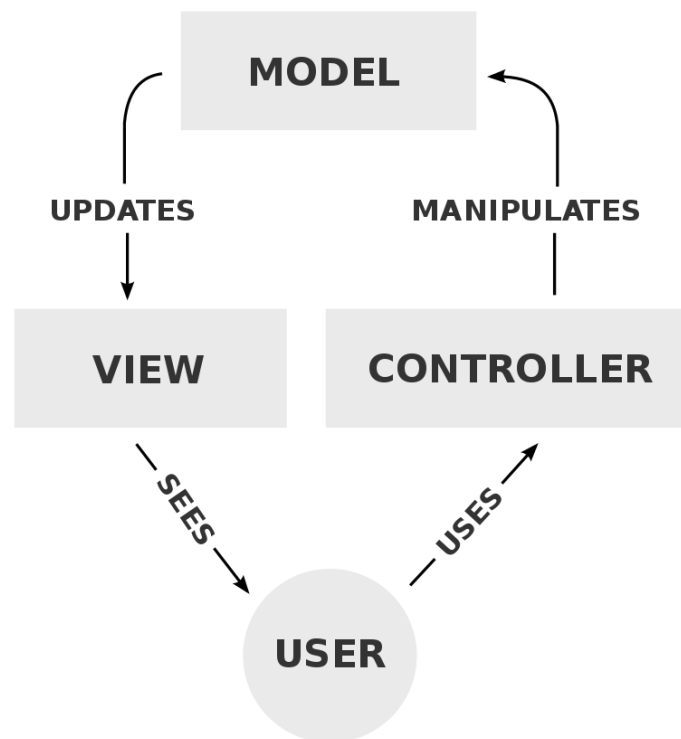


Hình 1: Logo ASP.NET Core

Kiến trúc MVC trong ASP.Net Core.

MVC viết tắt của Model – View – Controller. Là mô hình thiết kế được sử dụng trong kỹ thuật phần mềm, dùng để phân bố source code thành 3 phần. Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác :

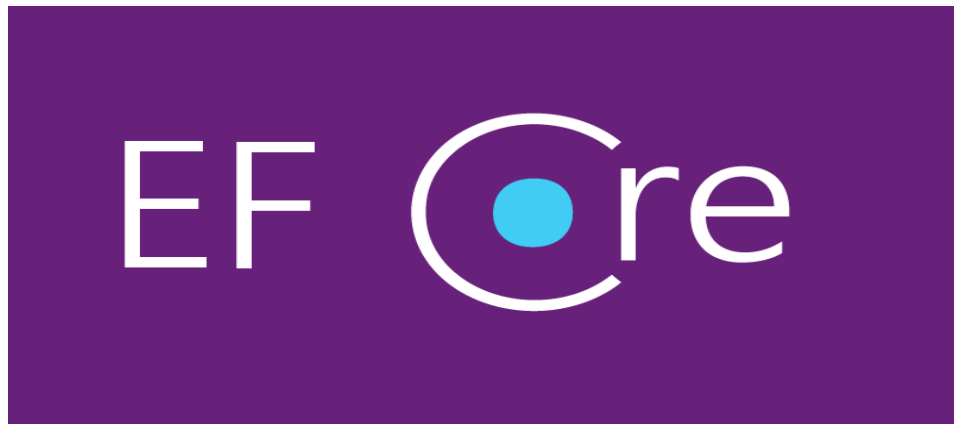
- Controller giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử lý chúng.
- Model là thành phần chứa tất cả các nghiệp vụ logic, phương thức xử lý, truy xuất database, đối tượng mô tả dữ liệu như các Class, hàm xử lý...
- View đảm nhận việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images.



Hình 2: Mô Hình MVC

Entity Framework Core

- Một ánh xạ quan hệ đối tượng (ORM), là một loại công cụ giúp đơn giản hóa ánh xạ giữa các đối tượng trong phần mềm với các bảng và cột của cơ sở dữ liệu quan hệ.
- Giúp theo dõi các thay đổi của đối tượng và cập nhật các thay đổi đó trở lại cơ sở dữ liệu.
- Đảm nhận việc tạo các kết nối cơ sở dữ liệu và thực thi các lệnh, cũng như lấy kết quả truy vấn và tự động ánh xạ các kết quả đó thành các đối tượng trong Website Đồ Án.



Hình 3: Logo EFCore

W3.CSS

- Một trang web cần phải có giao diện đẹp, đó là nơi mà W3.CSS tỏa sáng. Nó là framework front-end thuần css được dùng để trang trí bố cục cho website và là framework được chọn cho đồ án.



Hình 4: Logo W3CSS

Ckeditor

- Do mục tiêu của đồ án cần nhận các input của người dùng và hiển thị chúng dưới văn bản có định dạng. Ckeditor được chọn để tích hợp vào đồ án để trở thành trình soạn thảo văn bản trực tuyến dạng WYSIWYG chính cho project



Hình 5: Logo Ckeditor

SQL Server

- Là một hệ quản trị cơ sở dữ liệu quan hệ (Relational Database Management System (RDBMS) sử dụng câu lệnh SQL (**Transact-SQL**) để trao đổi dữ liệu giữa máy Client và máy cài SQL Server. Một RDBMS bao gồm databases, database engine và các ứng dụng dùng để quản lý dữ liệu và các bộ phận khác nhau trong RDBMS.
- Là công cụ quản lý cơ sở dữ liệu cho Website.



Hình 6: Logo SQL Server

2. Phân tích đề án

2.1. Mô tả đề án

Đề án “website hỗ trợ chia sẻ và hỏi đáp kiến thức về lập trình” được xây dựng dựa trên mục đích là nơi chia sẻ kiến thức lập trình qua các bài viết, đồng thời cũng là nơi đặt những câu hỏi về các kiến thức lập trình, để những lập trình viên khác có thể cùng nhau hợp tác giải quyết vấn đề. Ngoài ra, người dùng website còn có thể sử dụng tính năng tìm kiếm các bài viết, lọc bài theo tag, xem bài các bài viết và cuộc thảo luận thịnh hành. Người dùng còn có thể theo dõi những người dùng khác với mục đích xem những bài viết và cuộc thảo luận từ các đối tượng được theo dõi

2.2. Đặc tả yêu cầu đề án

2.2.1. Yêu cầu chức năng

Xây dựng một website Q&A theo mô hình MVC và chạy theo nền tảng web application với cơ sở dữ liệu được lưu trữ tại SQL Server.

Phần mềm giúp các lập trình viên chia sẻ kiến thức lập trình của mình qua những bài viết, đồng thời họ cũng có thể đặt những câu hỏi liên quan đến kiến thức lập trình.

Để có thể sử dụng website, người dùng cần phải truy cập vào đúng đường dẫn :

- Hạng mục 1: Đối với người dùng không có tài khoản, thì chỉ được phép xem các bài viết trên trang.
- Hạng mục 2: Đối với người dùng đã tạo tài khoản và cần phải trải qua bước đăng nhập thành công, thì thực hiện được nhiều chức năng trong trang.

Yêu cầu chức năng của phần mềm gồm các chức năng sau :

Phần mềm :

- Kiểm tra tài khoản và mật khẩu người dùng
- Phân chức năng theo từng hạng mục :
 - Liệt kê bài viết (theo tiêu chí thịnh hành, theo tag).
 - Liệt kê câu hỏi (theo tiêu chí thịnh hành, theo tag).
 - Xem chi tiết (xem chi tiết bài viết, xem chi tiết câu hỏi).
 - Quản lý bài viết (đăng bài, hiển thị bài viết, chỉnh sửa bài, xóa bài).
 - Quản lý câu hỏi (đặt câu hỏi, hiển thị câu hỏi, chỉnh sửa câu hỏi).
 - Quản lý tag (thêm tag mới, xóa tag cho bài viết, câu hỏi).
 - Theo dõi quá trình chỉnh sửa thông tin của các bài viết và câu hỏi.
 - Đăng lời phản hồi.
 - Theo dõi người dùng (liệt kê các bài viết và câu hỏi từ những người dùng đang được theo dõi).
 - Đánh dấu bài viết (liệt kê các bài viết đã được đánh dấu).
 - Đánh dấu câu hỏi (liệt kê các câu hỏi đã được đánh dấu).
 - Hỗ trợ đăng hình ảnh cho bài viết và câu hỏi.

- Tìm kiếm (tìm theo tên, tìm theo tag).

Người dùng không có tài khoản:

- Xem, tìm kiếm.

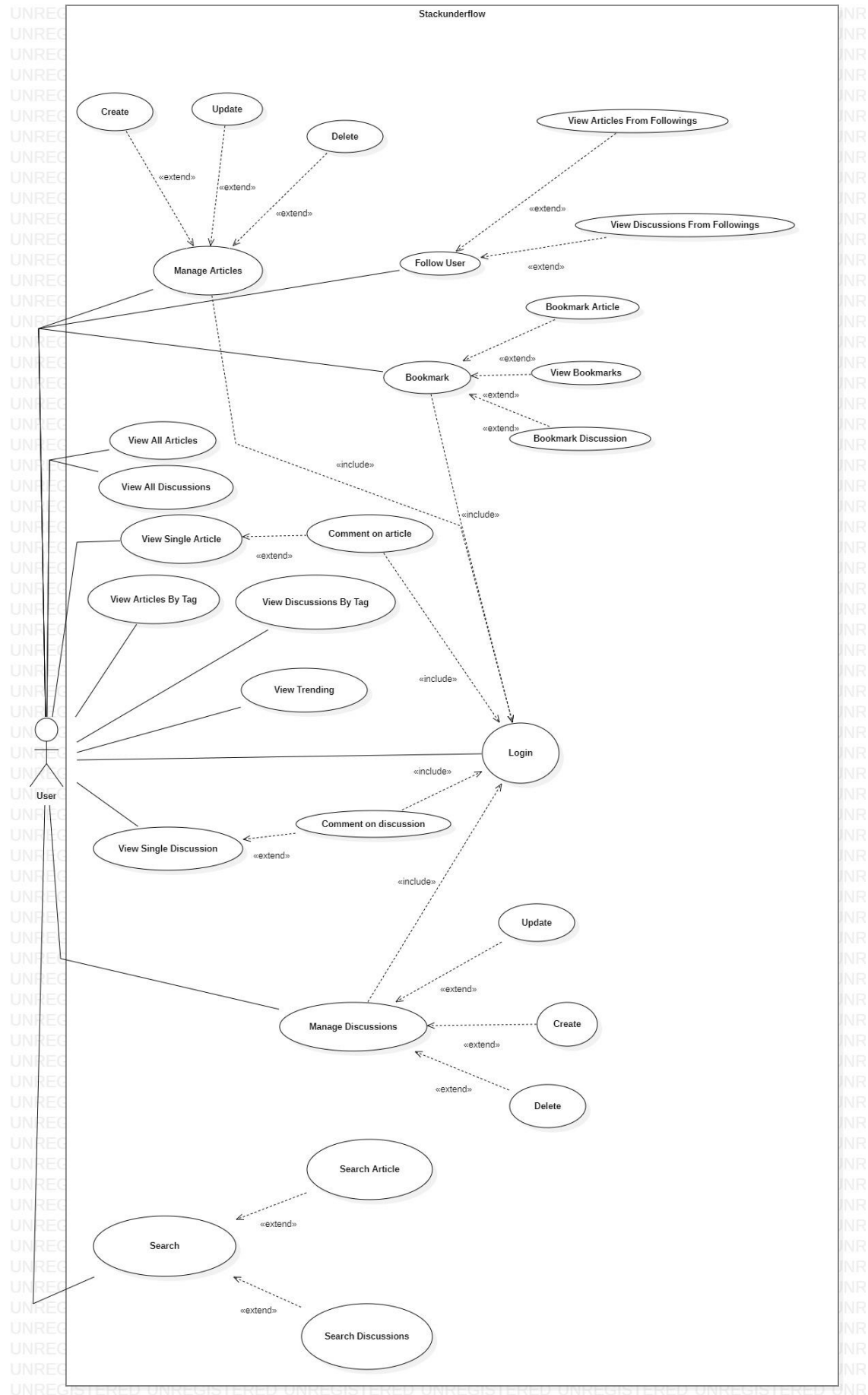
Người dùng có tài khoản:

- Đăng, bình luận, xóa, chỉnh sửa bài viết của họ.
- Quản lý bình luận trên bài viết của họ (ẩn, phản hồi, report).
- Xem, tìm kiếm bài viết.
- Thực hiện đánh dấu trang, theo dõi người dùng khác.

2.2.2. Yêu cầu phi chức năng

Bên cạnh các chức năng cơ bản, website còn đáp ứng một số yêu cầu phi chức năng khác nhằm cải thiện trải nghiệm sử dụng cho người dùng như là: ngôn ngữ (tiếng Việt), chạy được trên các trình duyệt web (Google Chrome, FireFox, Cốc Cốc, Internet Explorer). Ngoài ra, giao diện có bố cục rõ ràng, đơn giản, dễ hiểu và dễ sử dụng. Hiệu suất hoạt động của website luôn ở trạng thái đảm bảo khi có nhiều kết nối tới cơ sở dữ liệu cùng một lúc. An toàn và bảo mật thông tin của người dùng cũng là một phân ưu tiên của ứng dụng.

3. Lược đồ use-case



4. Đặc tả use-case

Tên Use-Case	Login
Mô tả	Use-case này cho phép người dùng xem đăng nhập vào hệ thống. Để đăng nhập vào hệ thống, người dùng cần nhập username và password.
Actor	Người dùng
Use-Case liên quan	Không
Tiền điều kiện	Người dùng cần có tài khoản hợp lệ
Hậu điều kiện	Hệ thống hiển thị trang chủ dưới với quyền truy cập của người dùng
Luồng sự kiện chính	<ol style="list-style-type: none">1. Người dùng nhập username và password.2. Hệ thống nhận username và password.3. Hệ thống xác thực username và password.4. Hệ thống hiển thị trang chủ dưới với quyền truy cập của người dùng.5. Kết thúc use case.
Luồng sự kiện phụ	<p>3a. Thiếu username/password</p> <ol style="list-style-type: none">1. Hệ thống yêu cầu nhập username/password.2. Use case tiếp tục ở bước 1. <p>3b. Username/password không hợp lệ</p> <ol style="list-style-type: none">1. Hệ thống hiển thị “Username hoặc password không hợp lệ”.2. Hệ thống yêu cầu nhập username/password3. Use case tiếp tục ở bước 1.

Tên Use-Case	View All Articles
Mô tả	Use-case này cho phép người dùng xem toàn bộ bài viết có trên hệ thống.
Actor	Người dùng

Use-Case liên quan	Không
Tiền điều kiện	Không
Hậu điều kiện	Hệ thống hiển thị trang chủ với các bài viết
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng truy cập trang chủ hệ thống. 2. Hệ thống lấy thông tin các bài viết từ cơ sở dữ liệu. 3. Hệ thống hiển thị trang chủ với các bài viết. 4. Kết thúc use case.
Luồng sự kiện phụ	<ol style="list-style-type: none"> 2a. Hệ thống không có bài viết trong cơ sở dữ liệu. 2b. Hệ thống bị lỗi (hoặc bảo trì).

Tên Use-Case	Xem tất cả cuộc thảo luận
Mô tả	Use-case này cho phép người dùng xem toàn bộ các cuộc thảo luận có trên hệ thống.
Actor	Người dùng
Use-Case liên quan	Không
Tiền điều kiện	Không
Hậu điều kiện	Hệ thống hiển thị trang chủ với những cuộc thảo luận
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng truy cập trang chủ hệ thống. 2. Hệ thống lấy thông tin các cuộc thảo luận từ cơ sở dữ liệu. 3. Hệ thống hiển thị trang chủ với các cuộc thảo luận. 4. Kết thúc use case.
Luồng sự kiện phụ	<ol style="list-style-type: none"> 2a. Hệ thống không có cuộc thảo luận nào trong cơ sở dữ liệu. 2b. Hệ thống bị lỗi (hoặc bảo trì).

Tên Use-Case	View Trending
Mô tả	Use-case này cho phép người dùng xem toàn bộ bài viết/cuộc thảo luận có trên hệ thống theo tiêu chí thịnh hành.
Actor	Người dùng
Use-Case liên quan	Không
Tiền điều kiện	Không
Hậu điều kiện	Hệ thống hiển thị trang mục thịnh hành
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án xem mục thịnh hành. 2. Hệ thống lấy toàn bộ các bài viết và cuộc thảo luận trong hệ thống. 3. Hệ thống sắp xếp kết quả truy vấn theo lượt xem, lượt đánh dấu trang và lượt bình luận. 4. Hệ thống hiển thị trang mục thịnh hành. 5. Kết thúc use case.
Luồng sự kiện phụ	<ol style="list-style-type: none"> 2a. Hệ thống không có bài viết/cuộc thảo luận nào trong cơ sở dữ liệu. 2b. Hệ thống bị lỗi (hoặc bảo trì).

Tên Use-Case	View Articles By Tag
Mô tả	Use-case này cho phép người dùng xem toàn bộ bài viết có tag được lựa chọn
Actor	Người dùng
Use-Case liên quan	Không
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng phải ở những trang web hiển thị danh sách bài viết. 2. Bài viết phải có tag.

Hậu điều kiện	Hệ thống hiển thị trang web với các bài viết có tag được lựa chọn
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng lựa chọn một trong những tag được hiển thị. 2. Hệ thống lấy tất cả các bài viết có gắn tag được lựa chọn từ cơ sở dữ liệu. 3. Hệ thống hiển thị trang web với các bài viết có tag được lựa chọn. 4. Kết thúc use case.
Luồng sự kiện phụ	<ol style="list-style-type: none"> 1a. Bài viết không có tag. 2a. Hệ thống bị lỗi (hoặc bảo trì).

Tên Use-Case	View Discussion By Tag
Mô tả	Use-case này cho phép người dùng xem toàn bộ các cuộc thảo luận có tag được lựa chọn
Actor	Người dùng
Use-Case liên quan	Không
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng phải ở những trang web hiển thị danh sách các cuộc thảo luận. 2. Cuộc thảo luận phải có tag.
Hậu điều kiện	Hệ thống hiển thị trang web với các cuộc thảo luận có tag được lựa chọn
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng lựa chọn một trong những tag được hiển thị. 2. Hệ thống lấy tất cả các cuộc thảo luận có gắn tag được lựa chọn từ cơ sở dữ liệu. 3. Hệ thống hiển thị trang web với các cuộc thảo luận có tag được lựa chọn. 4. Kết thúc use case.
Luồng sự kiện phụ	<ol style="list-style-type: none"> 1a. Bài viết không có tag. 2a. Hệ thống bị lỗi (hoặc bảo trì).

Tên Use-Case	View Single Article
Mô tả	Use-case này cho phép người dùng xem nội dung một bài viết theo lựa chọn của người dùng.
Actor	Người dùng
Use-Case liên quan	Comment
Tiền điều kiện	1. Người dùng phải ở những trang web hiển thị danh sách các bài viết.
Hậu điều kiện	Hệ thống hiển thị trang web với thông tin của bài viết được chọn.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng lựa chọn bài viết trên danh sách các bài viết được hiển thị. 2. Hệ thống nhận input (articleID) của người dùng. 3. Hệ thống lấy thông tin của bài viết tương ứng. 4. Hệ thống hiển thị trang web với thông tin của bài viết được chọn. 5. Kết thúc use case.
Luồng sự kiện phụ	<ol style="list-style-type: none"> 1a. Không có bài viết nào trên hệ thống. 2a. Hệ thống không nhận được articleID từ input người dùng.

Tên Use-Case	View Single Discussion
Mô tả	Use-case này cho phép người dùng xem nội dung một cuộc thảo luận theo lựa chọn của người dùng.
Actor	Người dùng
Use-Case liên quan	Reply
Tiền điều kiện	1. Người dùng phải ở những trang web hiển thị danh sách các cuộc thảo luận.

Hậu điều kiện	Hệ thống hiển thị trang web với thông tin của cuộc thảo luận được chọn.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng lựa chọn cuộc thảo luận trên danh sách các bài viết được hiển thị. 2. Hệ thống nhận input (discussionID) của người dùng. 3. Hệ thống lấy thông tin của cuộc thảo luận tương ứng. 4. Hệ thống hiển thị trang web với thông tin của cuộc thảo luận được chọn. 5. Kết thúc use case.
Luồng sự kiện phụ	<ol style="list-style-type: none"> 1a. Không có cuộc thảo luận nào trên hệ thống. 2a. Hệ thống không nhận được discussionID từ input người dùng.

Tên Use Case	Comment on article
Mô tả	Use-case này cho phép người dùng đăng bình luận lên bài viết. Người dùng phải nhập bình luận vào text editor trên bài viết để có thể đăng bình luận,
Actor	Người dùng
Use-case liên quan	Xem bài viết riêng lẻ
Tiền điều kiện	1. Người dùng phải ở trên trang hiển thị thông tin bài viết.
Hậu điều kiện	Bài viết hiển thị bình luận được đăng lên bởi người dùng.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng nhập bình luận. 2. Hệ thống kích hoạt use case login. 3. Người dùng đăng bình luận. 4. Hệ thống nhận bình luận từ bài viết của người dùng đang bình luận. 5. Hệ thống lưu bình luận. 6. Hệ thống hiển thị bài viết kèm theo bình luận của người dùng.

	7. Kết thúc use case.
Luồng sự kiện phụ	Hệ thống bị lỗi hoặc bảo trì.

Tên Use case	Comment on discussion
Mô tả	Use case này cho phép người dùng đăng phản hồi cho cuộc thảo luận. Người dùng cần phải nhập phản hồi vào text editor để có thể đăng phản hồi.
Actor	Người dùng
Use-case liên quan	Không
Tiền điều kiện	1. Người dùng phải ở trang hiển thị thông tin cuộc thảo luận.
Hậu điều kiện	Cuộc thảo luận hiển thị phản hồi được đăng lên bởi người dùng.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng nhập phản hồi. 2. Hệ thống kích hoạt use case login. 3. Người dùng đăng phản hồi. 4. Hệ thống phản hồi luận từ bài viết của người dùng đang phản hồi. 5. Hệ thống lưu phản hồi. 6. Hệ thống hiển thị bài thảo luận kèm theo phản hồi của người dùng. 7. Kết thúc use case.
Luồng sự kiện phụ	Hệ thống bị lỗi hay bảo trì

Tên Use case	Search Article
Mô tả	Use case này cho phép người dùng tìm kiếm bài viết theo keyword.
Actor	Người dùng

Use-case liên quan	Không
Tiền điều kiện	Không
Hậu điều kiện	Hệ thống hiện trang web liệt kê kết quả tìm kiếm.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng nhập từ khóa tìm kiếm. 2. Người dùng đăng từ khóa tìm kiếm. 3. Hệ thống nhận từ khóa tìm kiếm. 4. Hệ thống truy xuất dữ liệu, lấy danh sách các bài viết theo từ khóa. 5. Hệ thống hiện trang web liệt kê kết quả tìm kiếm. 6. Kết thúc use case.
Luồng sự kiện phụ	1a. Người dùng không nhập từ khóa. <ol style="list-style-type: none"> 1. Hệ thống hỏi từ khóa. 2. Use case tiếp tục ở bước 1.

Tên Use case	Search Discussion
Mô tả	Use case này cho phép người dùng tìm kiếm thảo luận theo keyword.
Actor	Người dùng
Use-case liên quan	Không
Tiền điều kiện	Không
Hậu điều kiện	Hệ thống hiện trang web liệt kê kết quả tìm kiếm.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng nhập từ khóa tìm kiếm. 2. Người dùng đăng từ khóa tìm kiếm. 3. Hệ thống nhận từ khóa tìm kiếm. 4. Hệ thống truy xuất dữ liệu, lấy danh sách các bài viết theo từ khóa. 5. Hệ thống hiện trang web liệt kê kết quả tìm kiếm. 6. Kết thúc use case.
Luồng sự kiện phụ	Người dùng không nhập từ khóa. <ol style="list-style-type: none"> 1. Hệ thống hỏi từ khóa.

	2. Use case tiếp tục ở bước 1.
--	--------------------------------

Tên Use case	Create Article
Mô tả	Use case này cho phép người dùng đăng bài viết lên hệ thống.
Actor	Người dùng
Use-case liên quan	Login
Tiền điều kiện	1. Người dùng phải có tài khoản hợp lệ trên website. 2. Người dùng phải đăng nhập thành công hệ thống.
Hậu điều kiện	Hệ thống lưu bài viết của người dùng.
Luồng sự kiện chính	1. Người dùng chọn phương án đăng bài viết. 2. Hệ thống kích hoạt use case login. 3. Hệ thống hiện thị trang web mục đăng bài viết. 4. Người dùng nhập thông tin của bài viết. 5. Hệ thống lưu bài viết của người dùng. 6. Kết thúc use case.
Luồng sự kiện phụ	Hệ thống bị lỗi hoặc bảo trì.

Tên Use case	Đăng Cuộc Thảo Luận
Mô tả	Use case này cho phép người dùng đăng cuộc thảo luận lên hệ thống.
Actor	Người dùng
Use-case liên quan	Login
Tiền điều kiện	1. Người dùng phải có tài khoản hợp lệ trên website. 2. Người dùng phải đăng nhập thành công hệ thống.
Hậu điều kiện	Hệ thống lưu bài viết của người dùng.

Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án đăng cuộc thảo luận. 2. Hệ thống kích hoạt use case login. 3. Hệ thống hiển thị trang web mục đăng cuộc thảo luận. 4. Người dùng nhập thông tin của đăng cuộc thảo. 5. Hệ thống lưu cuộc thảo luận của người dùng. 6. Kết thúc use case.
Luồng sự kiện phụ	Hệ thống bị lỗi hoặc bảo trì.

Tên Use case	Update Article
Mô tả	Use case này cho phép người dùng cập nhật thông tin của bài viết.
Actor	Người dùng
Use-case liên quan	Login
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người cần có tài khoản hợp lệ trên website. 2. Người dùng phải đăng nhập trên hệ thống. 3. Bài viết được cập nhật phải do quyền sở hữu của người dùng.
Hậu điều kiện	Hệ thống cập nhật bài viết cho người dùng.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án cập nhật bài viết. 2. Hệ thống kích hoạt use case login. 3. Người dùng nhập thông tin cập nhật bài viết. 4. Hệ thống kiểm chứng thông tin người dùng. 5. Hệ thống cập nhật bài viết cho người dùng. 6. Kết thúc usecase.
Luồng sự kiện phụ	<p>4a. Người dùng không phải là chủ sở hữu của bài viết.</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi không phải chủ sở hữu. 2. Kết thúc use case.

Tên Use case	Delete Article
Mô tả	Use case này cho phép người dùng xóa thông tin của bài viết.
Actor	Người dùng
Use-case liên quan	Login
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng cần có tài khoản hợp lệ trên website. 2. Người dùng phải đăng nhập trên hệ thống. 3. Bài viết được xóa phải do quyền sở hữu của người dùng.
Hậu điều kiện	Hệ thống xóa bài viết cho người dùng.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án xóa bài viết. 2. Hệ thống kích hoạt use case login. 3. Hệ thống kiểm chứng thông tin người dùng. 4. Hệ thống xóa bài viết cho người dùng. 5. Kết thúc usecase.
Luồng sự kiện phụ	<p>Người dùng không phải là chủ sở hữu của bài viết.</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi không phải chủ sở hữu. 2. Kết thúc use case.

Tên Use case	Update Discussion
Mô tả	Use case này cho phép người dùng cập nhật thông tin của cuộc thảo luận.
Actor	Người dùng
Use-case liên quan	Login
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng cần có tài khoản hợp lệ trên website. 2. Người dùng phải đăng nhập trên hệ thống.

	3. Cuộc thảo luận được cập nhật phải do quyền sở hữu của người dùng.
Hậu điều kiện	Hệ thống cập nhật cuộc thảo luận cho người dùng.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án cập nhật cuộc thảo luận. 2. Hệ thống kích hoạt use case login. 3. Người dùng nhập thông tin cập nhật cuộc thảo luận. 4. Hệ thống kiểm chứng thông tin người dùng. 5. Hệ thống cập nhật cuộc thảo luận cho người dùng. 6. Kết thúc usecase.
Luồng sự kiện phụ	<p>Người dùng không phải là chủ sở hữu của cuộc thảo luận.</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi không phải chủ sở hữu. 2. Kết thúc use case.

Tên Use case	Delete Discussion
Mô tả	Use case này cho phép người dùng xóa thông tin của cuộc thảo luận.
Actor	Người dùng
Use-case liên quan	Login
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng cần có tài khoản hợp lệ trên website. 2. Người dùng phải đăng nhập trên hệ thống. 3. Cuộc thảo luận được xóa phải do quyền sở hữu của người dùng.
Hậu điều kiện	Hệ thống xóa cuộc thảo luận cho người dùng.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án xóa cuộc thảo luận. 2. Hệ thống kích hoạt use case login. 3. Hệ thống kiểm chứng thông tin người dùng. 4. Hệ thống xóa cuộc thảo luận cho người dùng. 5. Kết thúc usecase.
Luồng sự kiện phụ	<p>Người dùng không phải là chủ sở hữu của cuộc thảo luận.</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi không phải chủ sở hữu.

	2. Kết thúc use case.
--	-----------------------

Tên Use case	Thêm tag vào bài viết
Mô tả	Use case này cho phép người dùng thêm tag vào một bài viết, người dùng nhập tên tag vào textbox, tách mỗi tag bằng một dấu “,”.
Actor	Người dùng
Use-case liên quan	Tạo bài viết, cập nhật bài viết
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng cần phải đăng nhập vào hệ thống. 2. Người dùng phải ở trang tạo bài viết hoặc cập nhật bài viết.
Hậu điều kiện	Hệ thống thêm tag vào bài viết
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng nhập các tag vào trường tạo tag. 2. Hệ thống kiểm chứng thông tin người dùng. 3. Hệ thống nhận tên các tag. 4. Hệ thống thêm các tag vào bài viết. 5. Kết thúc usecase.
Luồng sự kiện phụ	<p>Người dùng không phải là chủ sở hữu của bài viết:</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi không phải chủ sở hữu. 2. Kết thúc use case. <p>Người dùng nhập tag đã có sẵn cho bài viết:</p> <ol style="list-style-type: none"> 1. Hệ thống bỏ tag đã có. 2. Tiếp tục use case ở bước 4.

Tên Use case	Thêm tag vào cuộc thảo luận
Mô tả	Use case này cho phép người dùng thêm tag vào một cuộc thảo luận, người dùng nhập tên tag vào textbox, tách mỗi tag bằng một dấu “,”.
Actor	Người dùng
Use-case liên quan	Tạo cuộc thảo luận, cập nhật cuộc thảo luận

Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng cần phải đăng nhập vào hệ thống. 2. Người dùng phải ở trang tạo bài viết hoặc cập nhật bài viết.
Hậu điều kiện	Hệ thống thêm tag vào cuộc thảo luận.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng nhập các tag vào trường tạo tag. 2. Hệ thống kiểm chứng thông tin người dùng. 3. Hệ thống nhận tên các tag. 4. Hệ thống thêm các tag vào cuộc thảo luận. 5. Kết thúc usecase.
Luồng sự kiện phụ	<p>Người dùng không phải là chủ sở hữu của cuộc thảo luận:</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi không phải chủ sở hữu. 2. Kết thúc use case. <p>Người dùng nhập tag đã có sẵn cho cuộc thảo luận:</p> <ol style="list-style-type: none"> 3. Hệ thống bỏ tag đã có. 4. Tiếp tục use case ở bước 4.

Tên Use case	Xóa tag cho bài viết
Mô tả	Use case này cho phép người dùng xóa tag trong bài viết.
Actor	Người dùng
Use-case liên quan	Cập nhật bài viết
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng cần phải đăng nhập vào hệ thống. 2. Người dùng phải ở trang cập nhật bài viết.
Hậu điều kiện	Hệ thống xóa tag trong bài viết.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn tag để xóa. 2. Hệ thống kiểm chứng thông tin người dùng. 3. Hệ thống xóa tag trong bài viết. 4. Kết thúc usecase.
Luồng sự kiện phụ	<p>Người dùng không phải là chủ sở hữu của bài viết.</p> <ol style="list-style-type: none"> 1. Hệ thống báo lỗi không phải chủ sở hữu. 2. Kết thúc use case.

Tên Use case	Xóa tag cho cuộc thảo luận
Mô tả	Use case này cho phép người dùng xóa tag trong cuộc thảo luận.
Actor	Người dùng
Use-case liên quan	Cập nhật cuộc thảo luận
Tiền điều kiện	1. Người dùng cần phải đăng nhập vào hệ thống. 2. Người dùng phải ở trang cập nhật cuộc thảo luận.
Hậu điều kiện	Hệ thống xóa tag trong bài viết
Luồng sự kiện chính	1. Người dùng chọn tag để xóa. 2. Hệ thống kiểm chứng thông tin người dùng. 3. Hệ thống xóa tag trong cuộc thảo luận. 4. Kết thúc usecase.
Luồng sự kiện phụ	Người dùng không phải là chủ sở hữu của cuộc thảo luận. 1. Hệ thống báo lỗi không phải chủ sở hữu. 2. Kết thúc use case.

Tên Use case	Bookmark Article
Mô tả	Use case này cho phép người dùng đánh dấu trang bài viết.
Actor	Người dùng
Use-case liên quan	Login, xem danh sách đánh dấu
Tiền điều kiện	1. Người dùng cần phải đăng nhập vào hệ thống.
Hậu điều kiện	Hệ thống thêm bài viết vào danh sách bài viết được đánh dấu.
Luồng sự kiện chính	1. Người dùng chọn phương án đánh dấu bài viết. 2. Hệ thống kích hoạt use case login.

	3. Hệ thống kiểm tra thông tin bài viết. 4. Hệ thống thêm bài viết vào danh sách bài viết được đánh dấu. 5. Kết thúc usecase.
Luồng sự kiện phụ	3a. Bài viết đã được đánh dấu sẵn. 1. Hệ thống đổi phương án đánh dấu sang hủy đánh dấu. 2. Kết thúc use case.

Tên Use case	Bookmark Discussion
Mô tả	Use case này cho phép người dùng đánh dấu trang cuộc thảo luận.
Actor	Người dùng
Use-case liên quan	Login, xem danh sách đánh dấu
Tiền điều kiện	1. Người dùng cần phải đăng nhập vào hệ thống.
Hậu điều kiện	Hệ thống thêm cuộc thảo luận vào danh sách cuộc thảo luận được đánh dấu.
Luồng sự kiện chính	1. Người dùng chọn phương án đánh dấu cuộc thảo luận. 2. Hệ thống kích hoạt use case login. 3. Hệ thống kiểm tra thông tin cuộc thảo luận. 4. Hệ thống thêm cuộc thảo luận vào danh sách cuộc thảo luận được đánh dấu. 5. Kết thúc usecase.
Luồng sự kiện phụ	3a. Cuộc thảo luận đã được đánh dấu sẵn. 1. Hệ thống đổi phương án đánh dấu sang hủy đánh dấu. 2. Kết thúc use case.

Tên Use case	View Bookmarks
--------------	----------------

Mô tả	Use case này cho phép người dùng xem danh sách các bài viết và cuộc thảo luận đã được đánh dấu.
Actor	Người dùng
Use-case liên quan	Login
Tiền điều kiện	1. Người dùng cần phải đăng nhập vào hệ thống.
Hậu điều kiện	Hệ thống hiển thị danh sách các bài viết/cuộc thảo luận đã được đánh dấu bởi người dùng.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án xem danh sách đánh dấu. 2. Hệ thống kích hoạt use case login. 3. Hệ thống kiểm tra thông tin người dùng. 4. Hệ thống hiển thị danh sách các bài viết/cuộc thảo luận đã được đánh dấu bởi người dùng. 5. Kết thúc usecase.
Luồng sự kiện phụ	<p>4a. Người dùng chưa có đánh dấu trang</p> <ol style="list-style-type: none"> 1. Hệ thống hiển thị báo lỗi “chưa có mục được đánh dấu”. 2. Kết thúc use case.

Tên Use case	Follow User
Mô tả	Use case này cho phép người dùng theo dõi một người dùng khác trên hệ thống. Người dùng không thể tự theo dõi chính bản thân mình.
Actor	Người dùng
Use-case liên quan	Login, xem danh sách đang theo dõi
Tiền điều kiện	<ol style="list-style-type: none"> 1. Người dùng cần đăng nhập vào hệ thống. 2. Người dùng phải ở trên các trang web hiển thị thông tin bài viết/cuộc thảo luận riêng lẻ .
Hậu điều kiện	Hệ thống thêm đối tượng theo dõi vào danh sách đang theo dõi.

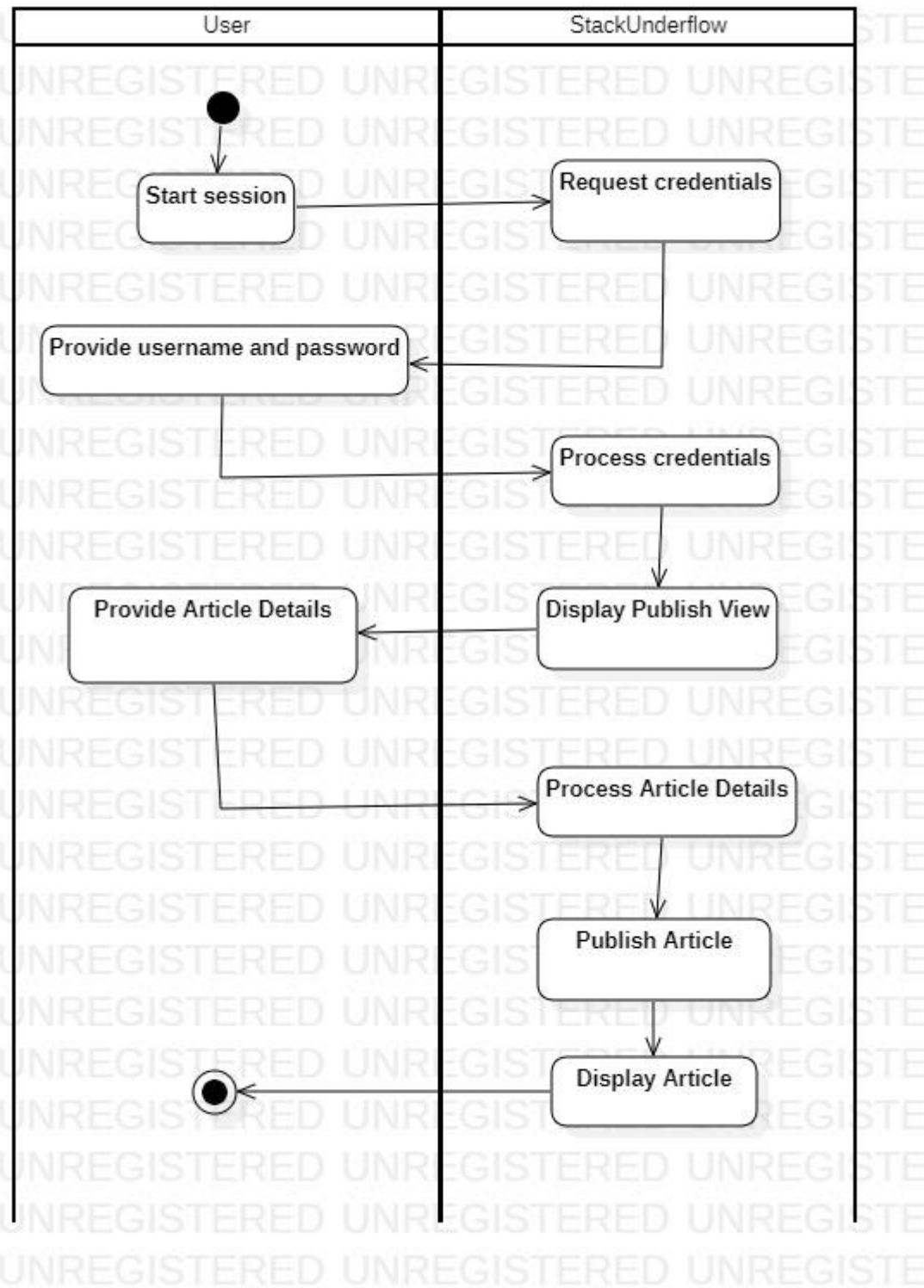
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án theo dõi. 2. Hệ thống kích hoạt use case login. 3. Hệ thống kiểm tra thông tin đối tượng được chọn để theo dõi. 4. Hệ thống thêm đối tượng theo dõi vào danh sách đang theo dõi. 5. Kết thúc usecase.
Luồng sự kiện phụ	3a. Đối tượng theo dõi chính là người dùng <ol style="list-style-type: none"> 1. Hệ thống giấu phương án theo dõi. 2. Kết thúc use case.

Tên Use case	View All Follows
Mô tả	Use case này cho phép người dùng xem các bài viết/cuộc thảo luận từ những đối tượng được theo dõi.
Actor	Người dùng
Use-case liên quan	Login
Tiền điều kiện	1. Người dùng cần đăng nhập vào hệ thống.
Hậu điều kiện	Hệ thống hiển thị các bài viết/cuộc thảo luận từ những đối tượng theo dõi.
Luồng sự kiện chính	<ol style="list-style-type: none"> 1. Người dùng chọn phương án xem danh sách theo dõi. 2. Hệ thống kích hoạt use case login. 3. Hệ thống kiểm tra thông tin người dùng. 4. Hệ thống kiểm tra danh sách đối tượng theo dõi. 5. Hệ thống hiển thị danh sách tên các đối tượng được theo dõi và các bài viết/cuộc thảo luận từ danh sách theo dõi. 6. Kết thúc usecase.
Luồng sự kiện phụ	3a. Người dùng chưa có đối tượng theo dõi. <ol style="list-style-type: none"> 1. Hệ thống hiển thị báo lỗi “chưa có đối tượng theo dõi”. 2. Kết thúc use case.

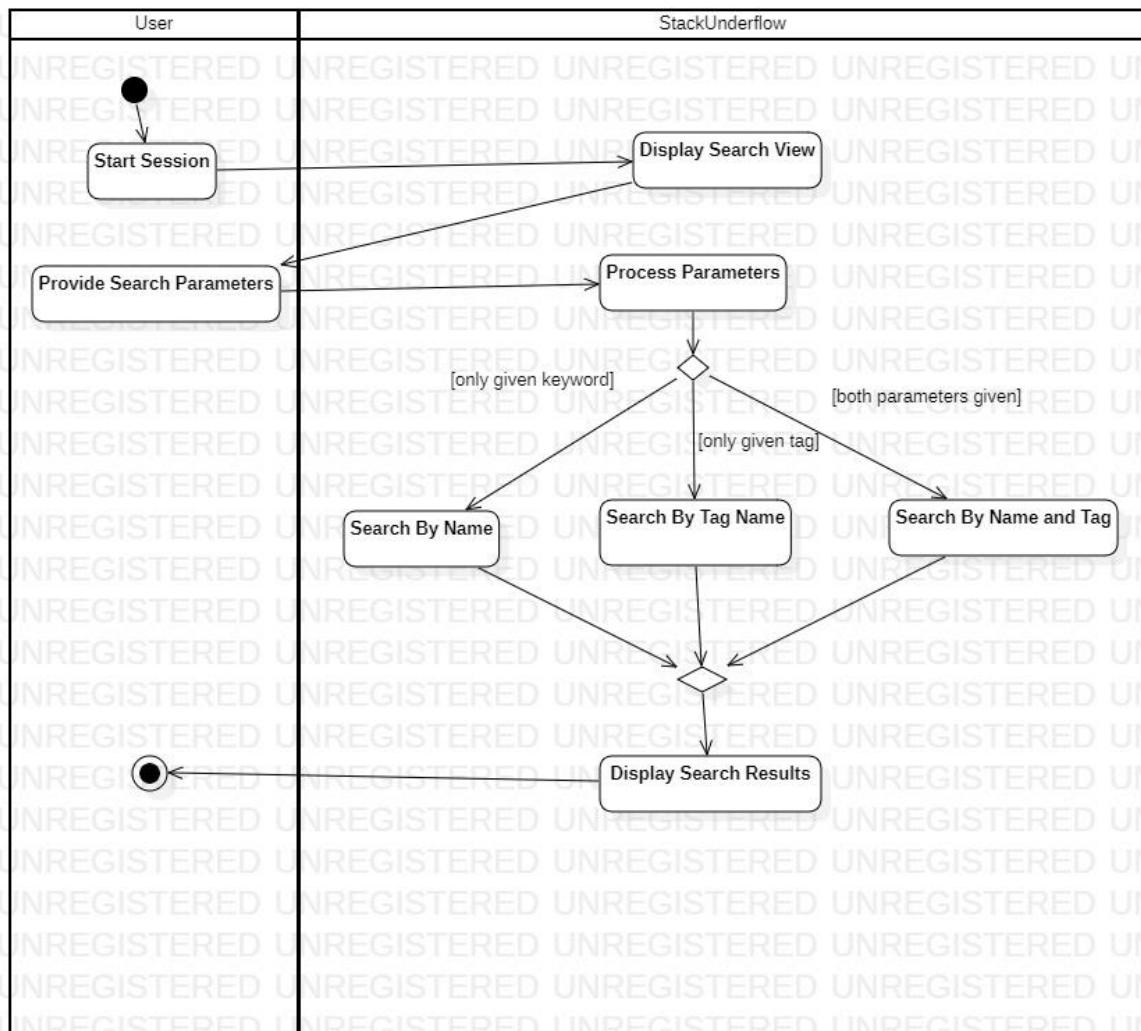
5. Activity Diagram

Do không đủ nhân lực nên với báo cáo này, tôi chỉ có thể vẽ activity diagram cho 3 chức năng quan trọng nhất

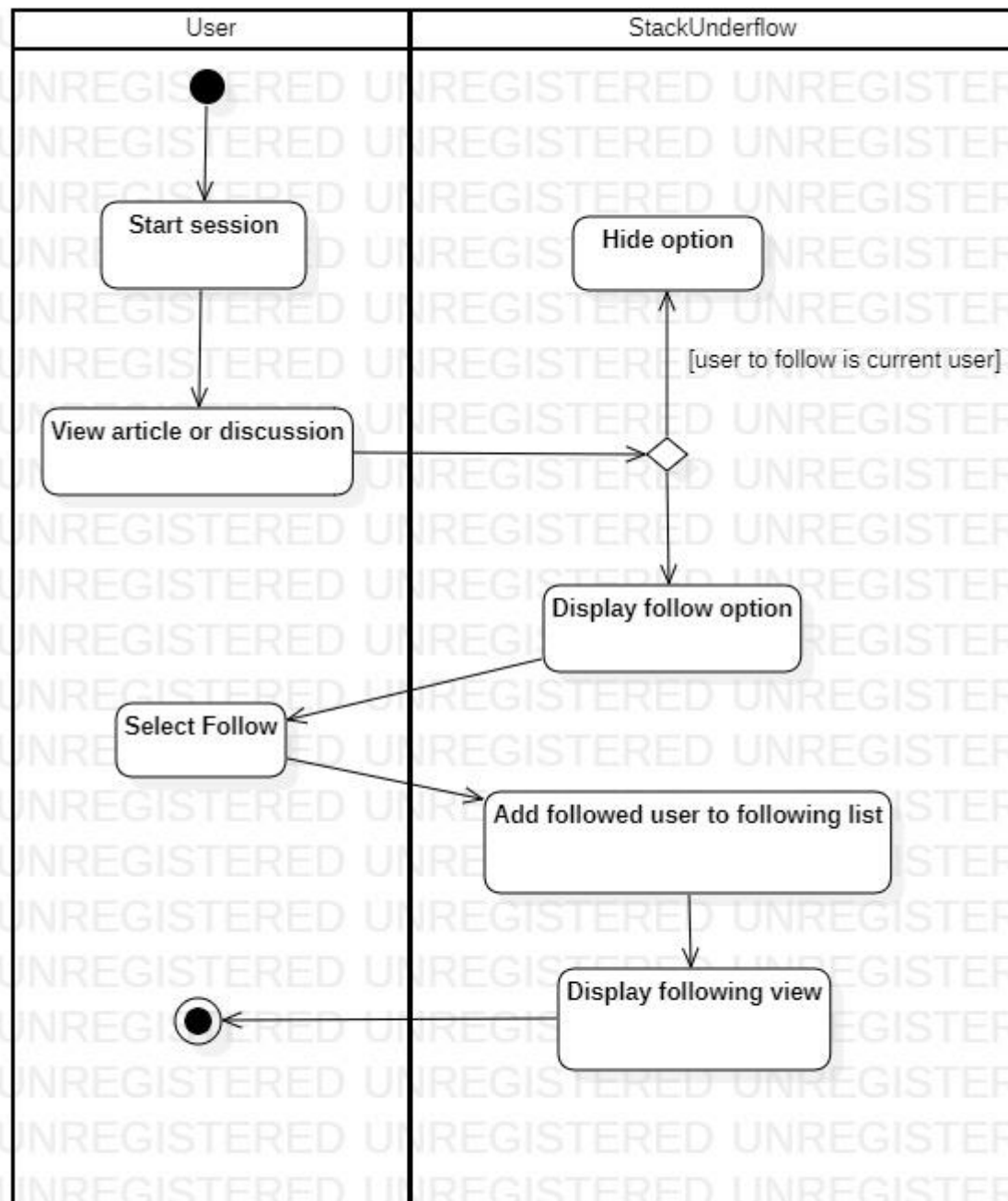
Use-case create article



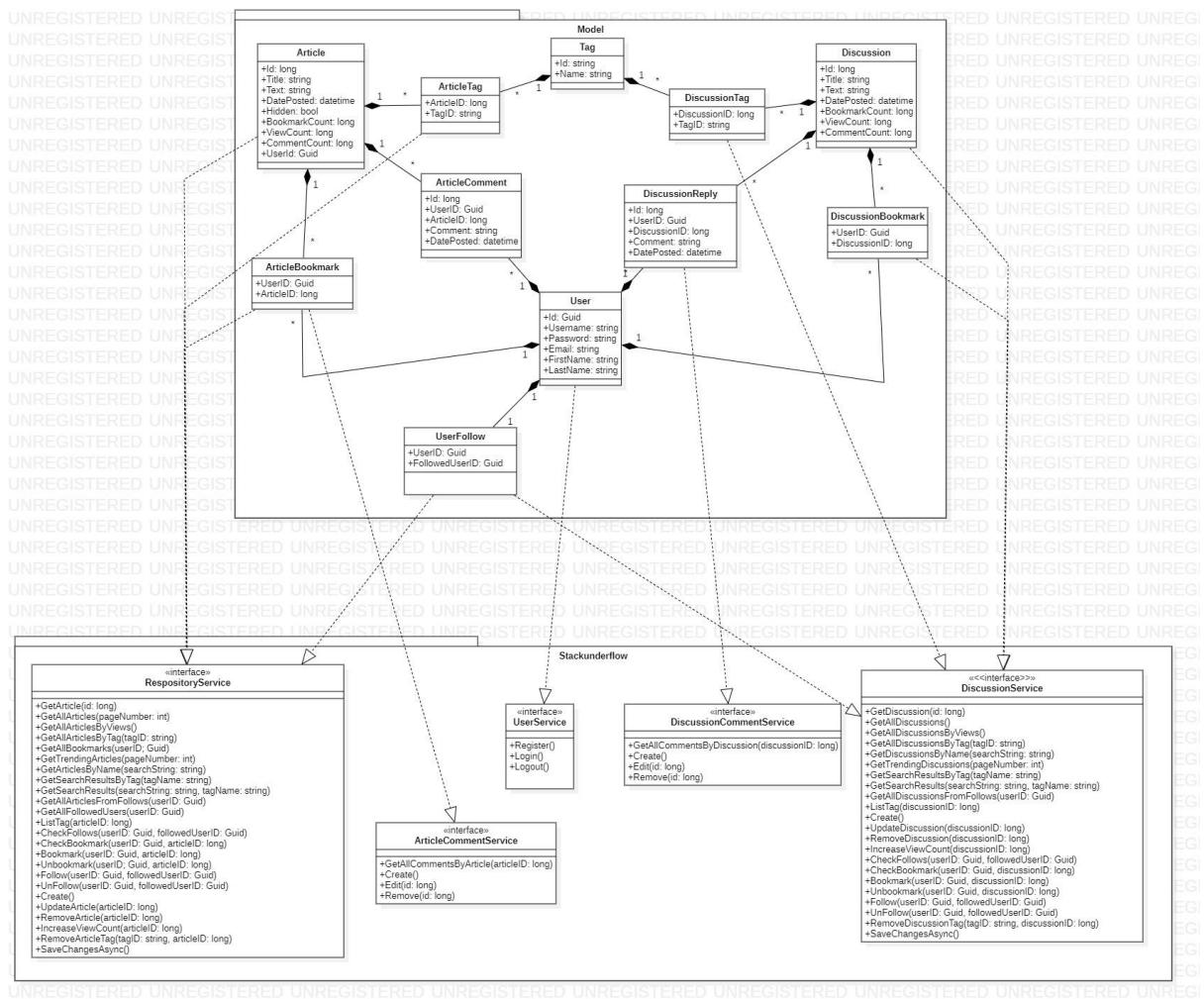
Use case Search



Use case Follow User



6. Class diagram



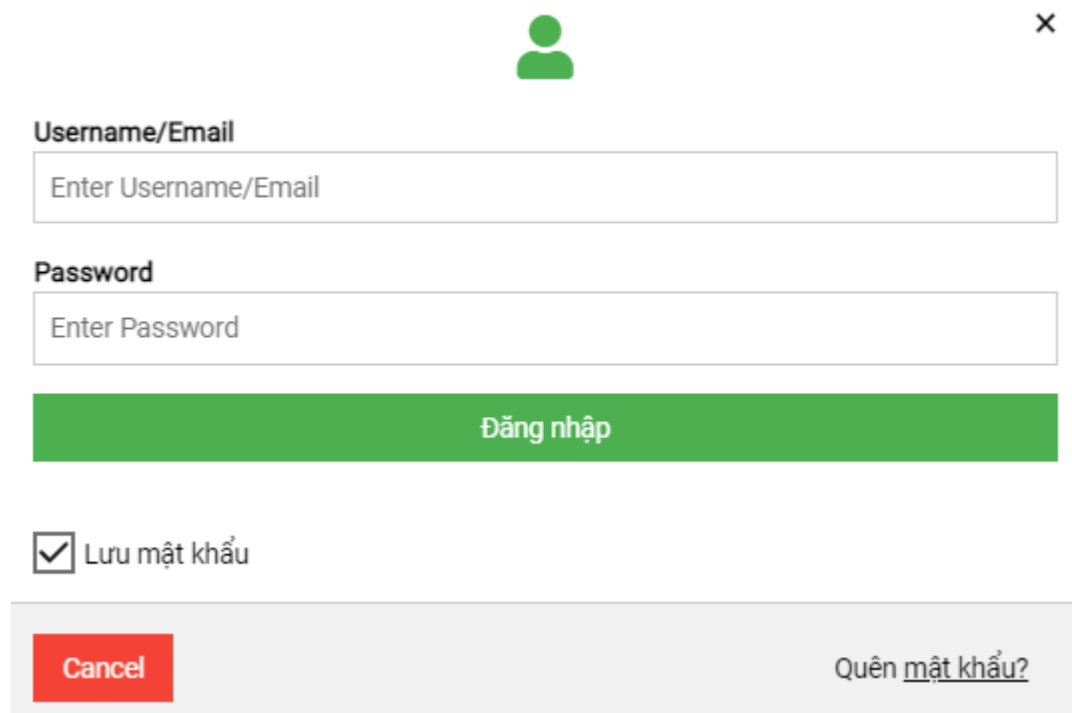
7. Thiết kế database



8. Thiết kế giao diện

Giao diện đăng nhập:

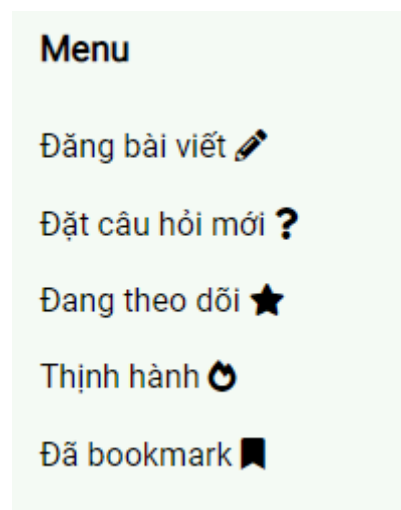
Khi mở phần đăng nhập của website lên, chúng ta sẽ được chuyển đến màn hình đăng nhập



The login form is displayed within a modal window. At the top center is a green circular user icon, and at the top right is a close button (X). Below these are two input fields: the first is labeled 'Username/Email' and contains the placeholder text 'Enter Username/Email'; the second is labeled 'Password' and contains the placeholder text 'Enter Password'. A green button labeled 'Đăng nhập' is positioned below the password field. Underneath the button is a checkbox labeled 'Lưu mật khẩu'. At the bottom of the modal, there is a red 'Cancel' button on the left and a link 'Quên mật khẩu?' on the right.

Hình 7: Giao diện đăng nhập tài khoản

Giao diện chức năng:



Hình 8: Giao diện chức năng

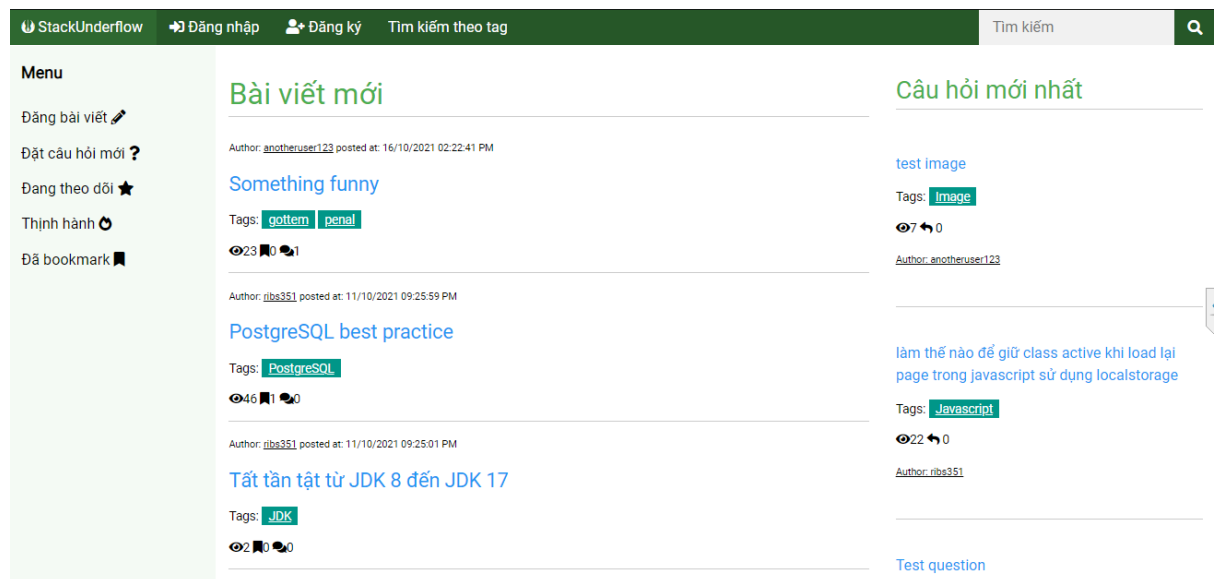
Về phía website, user có thể chọn và sử dụng các chức năng trong mục menu (với điều kiện là đã có tài khoản) ở phía bên trái của trang web:

- + Đăng bài viết
- + Đặt câu hỏi mới
- + Đang theo dõi
- + Thịnh hành
- + Đã bookmark
- + Tìm kiếm

Về phía website, user có thể chọn và sử dụng chức năng trong mục menu (với điều kiện là chưa có tài khoản) ở phía bên trái của trang web:

- + Thịnh hành
- + Tìm kiếm

Giao diện trang chủ



Hình 9: Giao diện trang chủ

Khi truy cập vào website (có tài khoản hoặc không có tài khoản), giao diện trang chủ sẽ hiển thị, gồm các mục:

- + Menu (Đăng bài viết, Đặt câu hỏi mới, Đang theo dõi, Thịnh hành, Đã bookmark)
- + Bài viết mới
- + Câu hỏi mới nhất
- + Đăng nhập
- + Đăng ký
- + Thanh tìm kiếm
- + Tìm kiếm theo tag

Giao diện bài viết

Tất tần tật từ JDK 8 đến JDK 17

Author: [ribs351](#) posted at: 11/10/2021 09:25:01 PM

Tags: [JDK](#)

Ngày 15/09 vừa qua, các iFan đã được "thỏa mãn" với màn ra mắt của iPhone 13. Hòa chung không khí đó, các tín đồ Java cũng được "sung sướng" khi Oracle chính thức release JDK 17 LTS sau 3 năm thống trị của JDK 11 LTS release vào 09/2018.

Khi JDK 8 LTS release vào 03/2014, nó là một cuộc cách mạng siêu to khổng lồ khi có rất nhiều sự khác biệt với phiên bản tiền nhiệm JDK 7. Rất nhiều features nổi bật như stream API, lambda expression, method reference, functional interface, interface default method, Optional... Phải nói là cực kì đồ sộ, chẳng thể mà người người, nhà nhà, các công ty, các dự án update ầm ầm.

4 năm sau, Oracle release JDK 11 LTS nhưng không thực sự có sự dịch chuyển mạnh mẽ như JDK 8 đã làm được trước đó.

Đến thời điểm hiện tại cũng còn khá các công ty và project sử dụng JDK 8 hoặc kết hợp cả 2 JDK 8 và JDK 11 (theo nguồn

khảo sát tự tìm hiểu ).

Oracle vẫn support JDK 8 cho đến hết 03/2022. Tức là chỉ còn còn vài tháng ngắn ngủi (tính từ thời điểm viết bài này). Thực ra cũng không cần quan tâm lắm vì đa số sử dụng OpenJDK vì OracleJDK không còn free nếu sử dụng cho mục đích thương mại từ lâu.


Nhưng...

Với JDK 17, mọi thứ hoàn toàn free bao gồm cả mục đích thương mại.

JDK 11 vẫn ổn, thậm chí JDK 8 vẫn đang phổ biến. Vậy JDK 17 có những feature gì nổi bật, đáng để chuyển đổi hay không?












Hình 10: Giao diện một bài viết

- Khi đã truy cập vào trang web thành công, người dùng (có tài khoản hoặc không có tài khoản) có thể chọn xem bài viết.
- Người dùng có thể xem tên bài, người đăng, mốc thời gian đăng, các tag liên quan và nội dung bài viết, xem các bình luận.
- Về chức năng đăng bình luận bài viết:
 - + Đối với người dùng chưa có tài khoản mà muốn bình luận, hệ thống sẽ chuyển qua giao diện đăng nhập
 - + Đối với người dùng đã có tài khoản, có thể đăng bình luận:



Bình luận

Paragraph

B *I*           

Viết phản hồi...

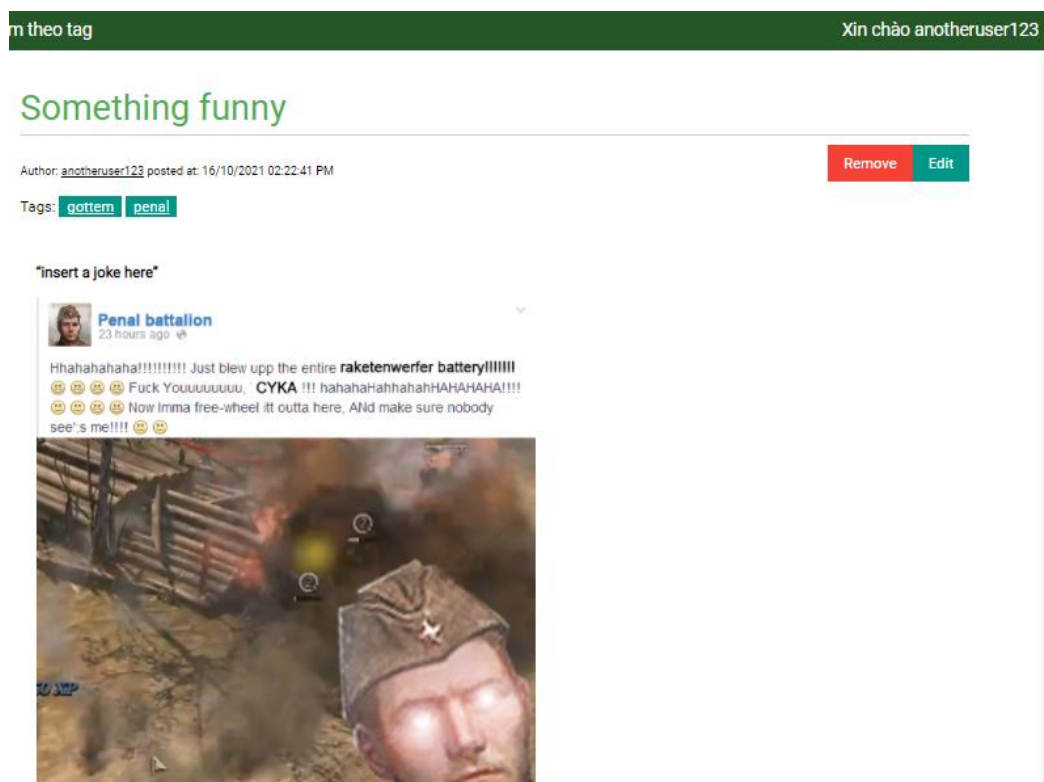
Post Comment

Giao diện đăng bài viết

Hình 12: Giao diện đăng bài viết

- Sau khi hoàn thành nội dung bài viết, người dùng có thể nhấn vào nút “post bài” để đăng bài viết lên hệ thống.

Giao diện sửa bài viết

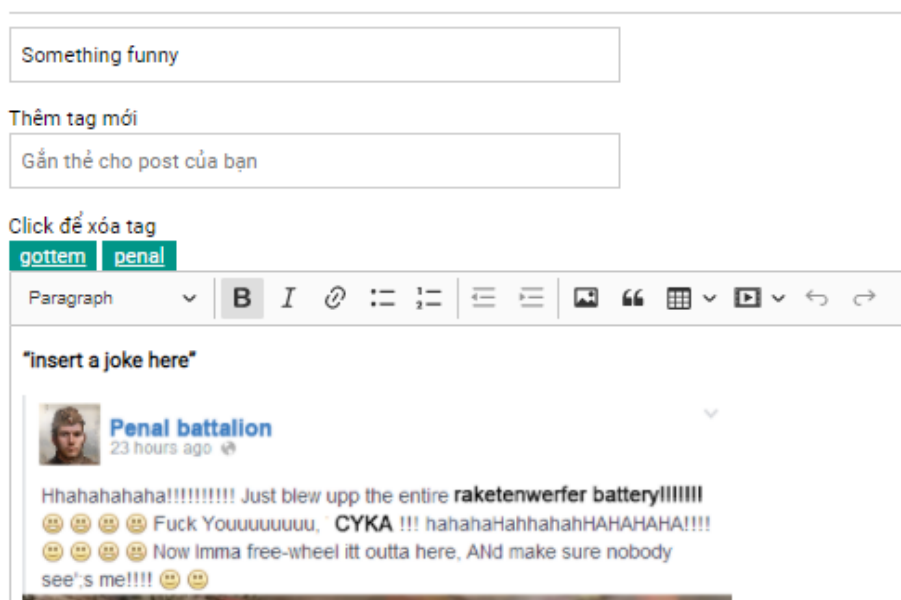


Hình 13: Chỉnh sửa, gỡ bỏ bài viết

Về chức năng chỉnh sửa hoặc gỡ bỏ bài viết, có hai trường hợp:

- + Trường hợp 1: bài viết này thuộc sở hữu của người xem. Nên người dùng có quyền chỉnh sửa hoặc gỡ bỏ bài viết.

Edit



Hình 14: Giao diện chỉnh sửa bài viết

+ Trường hợp 2: bài viết này không thuộc sở hữu người xem. Nên người dùng không được phép chỉnh sửa và gỡ bỏ bài viết.

Giao diện xem câu hỏi

làm thế nào để giữ class active khi load lại page trong javascript sử dụng localStorage

Author: [ribs351](#) posted at: 28/09/2021 03:18:52 PM

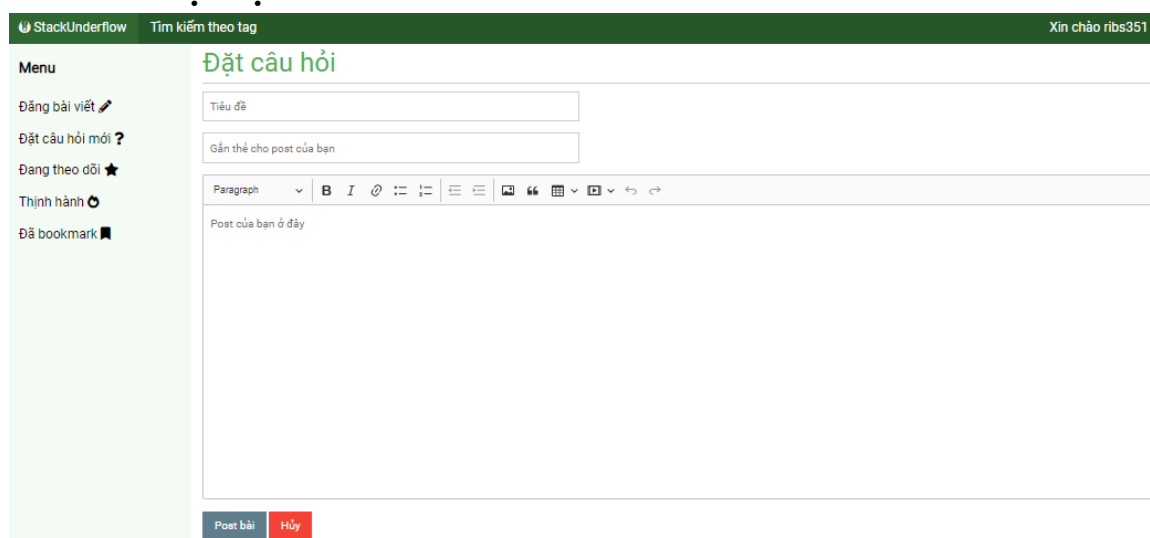
Tags: [Javascript](#)

A/C nào có thể hướng dẫn e giữ class active menu khi mình load lại trang không ạ. Có ví dụ thì tốt quá. E xin cảm ơn ạ!

Hình 15: Giao diện một câu hỏi

- Khi đã truy cập vào trang web thành công, người dùng (có tài khoản hoặc không có tài khoản) có thể chọn xem câu hỏi.
- Người dùng có thể xem tên bài, người đăng, mốc thời gian đăng, các tag liên quan và nội dung câu hỏi, xem các bình luận.
- Về chức năng đăng bình luận:
 - + Đối với người dùng chưa có tài khoản mà muốn bình luận, hệ thống sẽ chuyển qua giao diện đăng nhập
 - + Đối với người dùng đã có tài khoản, có thể đăng bình luận:

Giao diện đặt câu hỏi

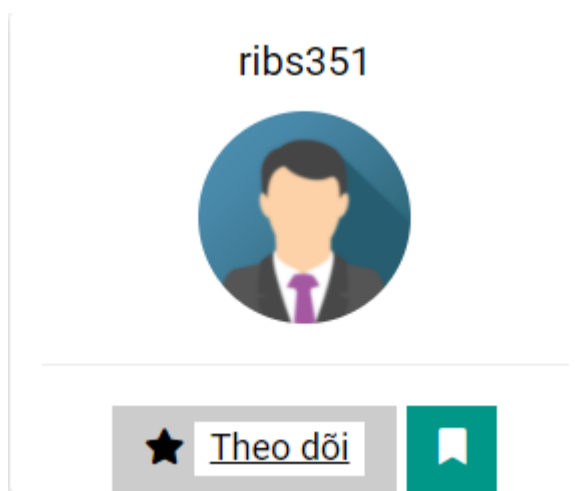


Hình 16: Giao diện đặt câu hỏi

- Khi đăng nhập thành công, người dùng có thể sử dụng chức năng đặt câu hỏi.
- Khi ở mục đặt câu hỏi, người dùng có thể thấy các trường đặt câu hỏi.

- Người dùng có thể nhập thông tin, nội dung của câu hỏi thông qua tiêu đề, thẻ tag, nội dung theo ý muốn.
- Sau khi hoàn thành nội dung câu hỏi, người dùng có thể nhấn vào nút “post bài” để đặt câu hỏi lên hệ thống.

Giao diện đang theo dõi

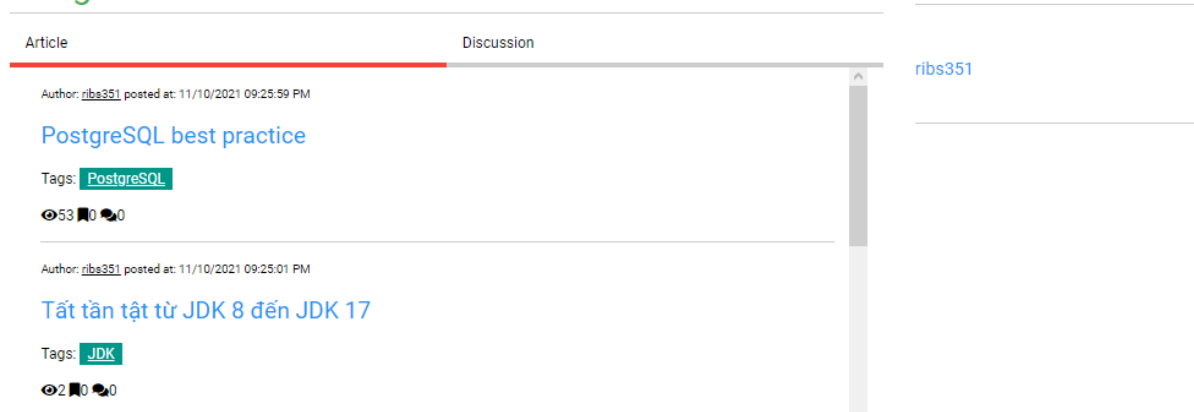


Hình 17: thông tin người dùng khác

Trước khi có thể xem giao diện theo dõi người dùng, cần chọn đối tượng để theo dõi.

Đang theo dõi

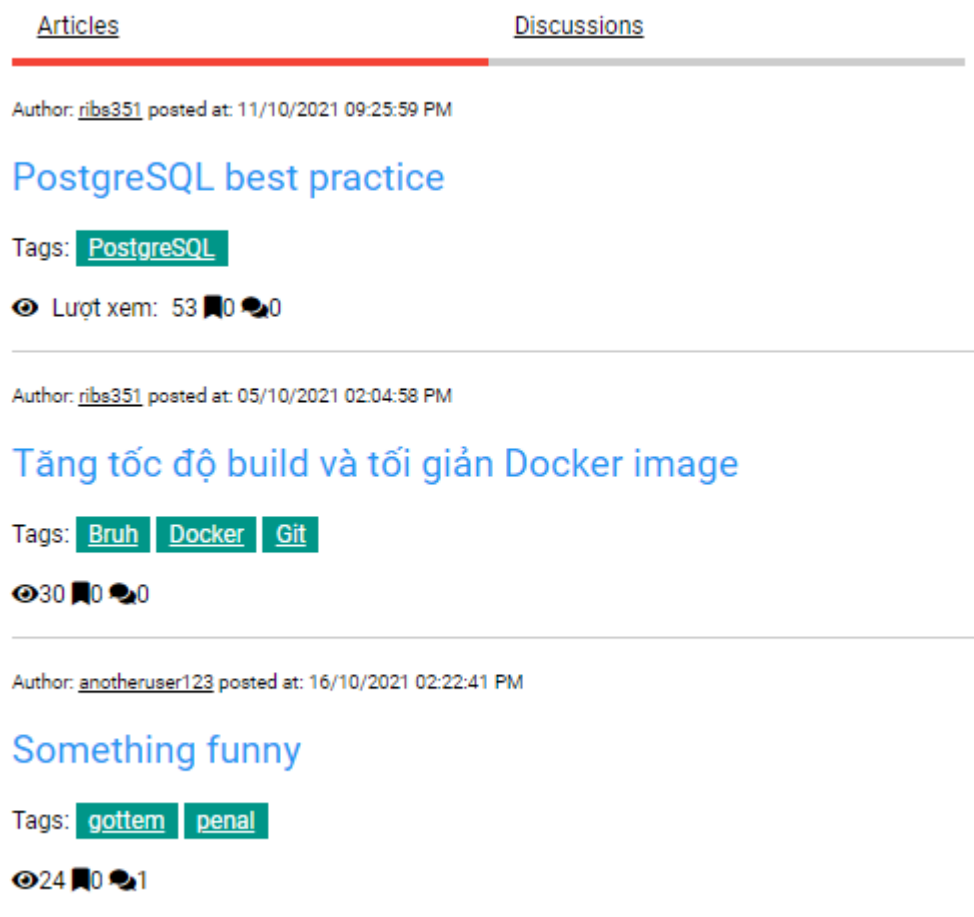
Danh sách theo dõi



Hình 18: Giao diện chức năng theo dõi

Sau khi đã có đối tượng theo dõi thì giao diện chức năng theo dõi sẽ hiện thị danh sách những tài khoản khác đang được theo dõi bởi người dùng, kèm theo danh sách các bài viết và câu hỏi được tạo bởi các tài khoản được theo dõi.

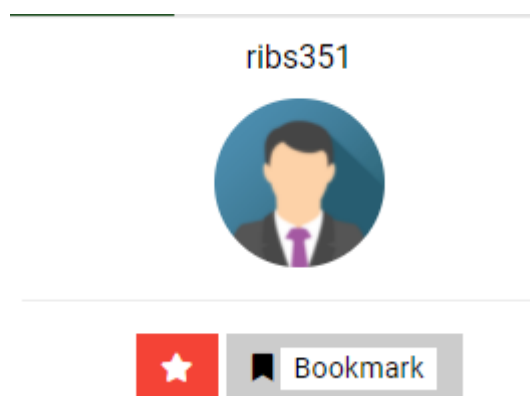
Giao diện thịnh hành



Hình 19: Giao diện thịnh hành

Có hai hạng mục: bài viết và các cuộc thảo luận
Để một bài viết/cuộc thảo luận lên trang thịnh hành cần có ba tiêu chí: lượt xem, lượt bookmark và lượt comment.

Giao diện bookmark



Hình 20: Thông tin người dùng khác

Trước khi có thể xem giao diện bookmark, cần chọn bài viết để đánh dấu trang.

Bookmark của bạn

Article

Discussion

Author: [anotheruser123](#) posted at: 16/10/2021 02:22:41 PM

[Something funny](#)

Tags: [gottem](#) [penal](#)

👁️28 🗨️1 🗳️0

Author: [ribs351](#) posted at: 11/10/2021 09:25:01 PM

[Tất tần tật từ JDK 8 đến JDK 17](#)

Tags: [JDK](#)

👁️5 🗨️1 🗳️0

Hình 21: Giao diện Bookmark

Giao diện sẽ hiển thị các bài viết/cuộc thảo luận đã được đánh dấu trang vào danh sách đã bookmark.

Giao diện chức năng tìm kiếm theo từ khóa hoặc theo tag

Tìm kiếm theo tag

Article

Discussion

Discussion Search

Từ khóa

Chọn tag tìm kiếm

Search

Hình 22: Giao diện phân loại tìm kiếm

Có hai hạng mục tìm kiếm bài viết và cuộc thảo luận:

- + Tìm kiếm từ khóa (Vd: cách sử dụng EF Core)
- + Từ kiếm theo tag (Vd: EFCore)

Người dùng có thể nhập vào cả hai trường để tìm kiếm trả về kết quả mong muốn:

Article Search

Từ khóa

Chọn tag tìm kiếm

Search

Kết quả tìm kiếm

Author: [ribs351](#) posted at: 11/10/2021 01:45:58 AM

Hello World

Tags: [C#](#)

👁️ 0 📌 0 💬 0

Hình 23: Giao diện trả kết quả do người dùng tìm kiếm

Giao diện lọc theo tag:

Bài viết với tag là Java

Author: [ribs351](#) posted at: 11/10/2021 09:24:06 PM

Kafka Consumer client API với Java

Tags: [Java](#) [Kafka](#)

👁️ 3 📌 0 💬 0

Author: [ribs351](#) posted at: 11/10/2021 09:22:08 PM

Java Object-Oriented Programming với bài toán cụ thể

Tags: [Java](#)

👁️ 2 📌 0 💬 0

Hình 24: Giao diện lọc theo tag

Khi người dùng nhấn vào tag có sẵn từ bài viết. Hệ thống sẽ hiển thị các bài viết có chứa tag người dùng đã chọn.

9. Kết luận

Sau khi thực hiện đồ án lần này, tôi đã học được thêm rất nhiều về việc tự học, cũng như việc hết cách thì phải lên tiếng để giảng viên biết và chỉ hướng đi một cách phù hợp. Và cuối cùng, nhờ việc cố gắng thực hiện đồ án này thì tôi cảm thấy rằng mình còn thiếu quá nhiều kiến thức để giải quyết các vấn đề và sau khi học xong đồ án này tôi cũng đã cải thiện lấp chỗ trống vào kiến thức của chính bản thân mình.

Tôi đã đạt được các mục tiêu mà mình mong muốn đạt được ban đầu đưa ra như sau:

- Tìm được các giải pháp cho vấn đề cũng như đề tài.
- Tự học, tìm kiếm tài liệu để hiểu thuật toán.
- Áp dụng được những kiến thức lập trình trong suốt nhiều năm học tại trường.
- Đã đạt được một số kiến thức về lập trình C# cũng như các ngôn ngữ khác như HTML, CSS và đặc biệt là framework ORM như EF Core.

Hoàn thành các mục tiêu xây dựng đề tài như:

- Xây dựng được giao diện website và các tính năng khác.
- Hoàn thành được cách lấy cơ sở dữ liệu.
- Hoàn thành được việc xử lý và quản lý tag cho từng bài viết.
- Hoàn thành tích hợp CKEditor vào ứng dụng web.

Tôi chưa đạt được các mục tiêu như chưa có phần quản lý chuyên dụng cho admin, moderator, chưa có chức năng tạo chuỗi series bài viết.

Hướng phát triển của website

- Hoàn thiện tốt hơn về mặt giao diện phần mềm.
- Mở rộng các chức năng chưa phát triển.
- Thiết kế API để website có thể trở thành ứng dụng đa nền tảng.

NHẬN XÉT CỦA GIẢNG VIÊN

[illegible]