

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC KINH TẾ - TÀI CHÍNH TP. HỒ CHÍ MINH



ĐỒ ÁN CHUYÊN NGÀNH

TÊN ĐỀ TÀI:

XÂY DỰNG WEBSITE QUẢN LÝ SỰ KIỆN, HOẠT ĐỘNG TẠI UEF

Ngành : Công nghệ thông tin

Chuyên ngành : Công nghệ phần mềm

Giảng viên hướng dẫn : TS. Văn Thị Thiên Trang

Sinh viên thực hiện :

Nguyễn Thu Hà

185050757

18D1TH-PM01

TP. Hồ Chí Minh, năm 2022

BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC KINH TẾ - TÀI CHÍNH TP. HỒ CHÍ MINH

ĐỒ ÁN CHUYÊN NGÀNH

TÊN ĐỀ TÀI:
XÂY DỰNG WEBSITE QUẢN LÝ SỰ KIỆN, HOẠT ĐỘNG TẠI UEF

Ngành : Công nghệ thông tin

Chuyên ngành : Công nghệ phần mềm

Giảng viên hướng dẫn : TS. Văn Thị Thiên Trang

Sinh viên thực hiện :

Nguyễn Thu Hà

185050757

18D1TH-PM01

TP. Hồ Chí Minh, năm 2022

LỜI CẢM ƠN

Để hoàn thành bài báo cáo môn đồ án này, chúng em đã nhận được sự giúp đỡ và góp ý nhiệt tình của quý thầy cô và các bạn trường Đại học Kinh Tế - Tài Chính.

Trước hết em xin chân thành cảm ơn quý thầy cô bạn trường Đại học Kinh Tế - Tài Chính, đặc biệt là những thầy cô đã tận tình dạy bảo chúng em suốt quãng thời gian học tập tại trường.

Em xin gửi lời cảm ơn sâu sắc đến cô Văn Thị Thiên Trang đã giành thời gian và tâm huyết hướng dẫn chúng em hoàn thành báo cáo môn đồ án.

Mặc dù chúng em đã có nhiều cố gắng để hoàn thành báo cáo môn đồ án, tuy nhiên không thể tránh khỏi những thiếu sót, rất mong nhận được sự đóng góp của quý thầy cô và các bạn!

TP. HCM, Tháng 1 năm 2022

Sinh viên thực hiện

Nguyễn Thu Hà

Mục lục

Danh mục các bảng	1
Danh mục các hình vẽ	2
LỜI MỞ ĐẦU	3
Chương 1 . TỔNG QUAN	4
1.1 Giới thiệu đề tài.....	4
1.2 Mục tiêu đề tài	4
1.3 Bố cục đề tài.....	6
Chương 2 . CƠ SỞ LÝ THUYẾT.....	7
2.1 Node JS	7
2.1.1 Khái niệm về Node JS	7
2.1.2 NPM và Package	7
2.2 React JS.....	8
2.2.1 Giới thiệu về React JS	8
2.2.2 Các khái niệm cơ bản về React JS.....	9
2.3 Hệ quản trị Cơ sở dữ liệu MySQL.....	11
2.3.1 Giới thiệu về MySQL	11
2.3.2 Cách thức hoạt động của MySQL	12
2.3.3 Tính năng cốt lõi của MySQL	13
2.4 Các phần mềm hỗ trợ khác	14
2.4.1 Xampp (Phần mềm test trang web trên local host)	14

2.4.2	Figma	14
2.4.3	Draw.io	14
Chương 3 . PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG		15
3.1	Phân tích.....	15
3.1.1	Phân tích thực thể	15
3.1.2	Lược đồ quan hệ	16
3.1.3	Mô hình quan niệm dữ liệu ERD (Mô hình thực thể kết hợp)	17
3.2	Ràng buộc dữ liệu	17
3.2.1	Ràng buộc khóa chính	17
3.2.2	Ràng buộc khóa phụ	17
3.3	Phân tích hệ thống.....	18
3.3.1	Sơ đồ Use-Case.....	18
3.3.2	Đặc tả một số Use-Case.....	19
3.3.3	Sơ đồ tuần tự (Sơ đồ tương tác đối tượng)	26
Chương 4 . THIẾT KẾ GIAO DIỆN VÀ CHỨC NĂNG		29
4.1	Mô hình chức năng	29
4.2	Giao diện của chương trình.....	29
4.2.1	Form hệ thống (cả người dùng và quản lý)	29
4.2.2	Form hệ thống (dùng cho quản lý)	32
4.2.3	Form người dùng (dùng cho sinh viên)	35
Chương 5 . KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN		38
5.1	Kết luận.....	38
5.1.1	Kết quả đạt được.....	38

5.1.2	Những hạn chế.....	38
5.2	Hướng phát triển	39
TÀI LIỆU THAM KHẢO.....		40

Danh mục các bảng

Bảng 3-1: Đặc tả Use-Case: Tạo sự kiện	19
Bảng 3-2: Đặc tả Use-Case: Sửa thông tin sự kiện (Cập nhật sự kiện)	20
Bảng 3-3: Đặc tả Use-Case: Xóa sự kiện.....	21
Bảng 3-4: Đặc tả Use-Case: Kết thúc sự kiện.....	22
Bảng 3-5: Đặc tả Use-Case: Tìm kiếm sự kiện.....	23
Bảng 3-6: Đặc tả Use-Case: Thêm sự kiện yêu thích	24
Bảng 3-7: Đặc tả Use-Case: Xóa sự kiện yêu thích	25

Danh mục các hình vẽ

Hình 2-1: Sự khác nhau về cách hoạt động của Virtual Dom và DOM	10
Hình 3-1: Thực thể sự kiện	15
Hình 3-2: Thực thể Người dùng.....	16
Hình 3-3: Lược đồ quan hệ	16
Hình 3-4: Mô hình thực thể kết hợp.....	17
Hình 3-5: Sơ đồ Use-Case.....	18
Hình 3-6: Sơ đồ tuần tự: Tạo sự kiện.....	26
Hình 3-7: Sơ đồ tuần tự: Chỉnh sửa sự kiện.....	27
Hình 3-8: Sơ đồ tuần tự: Xóa sự kiện	27
Hình 3-9: Sơ đồ tuần tự: Kết thúc sự kiện	28
Hình 3-10: Sơ đồ tuần tự: Thêm và xóa sự kiện yêu thích	28
Hình 4-1: Giao diện đăng nhập	30
Hình 4-2: Giao diện trang chủ.....	31
Hình 4-3: Giao diện liên hệ.....	31
Hình 4-4: Giao diện tạo sự kiện	32
Hình 4-5: Giao diện chỉnh sửa thông tin sự kiện	33
Hình 4-6: Giao diện dropdown của sự kiện	34
Hình 4-7: Giao diện xác nhận lại của chức năng xóa sự kiện.....	34
Hình 4-8: Giao diện quản lý các quản trị viên khoa	35
Hình 4-9: Giao diện đăng ký tài khoản	36
Hình 4-10: Giao diện quản lý tài khoản.....	36
Hình 4-11: Giao diện chỉnh sửa thông tin tài khoản sinh viên	37

LỜI MỞ ĐẦU

Những năm gần đây, thuật ngữ “Công nghệ thông tin” hay “IT” đều không còn là những từ ngữ xa lạ với mọi người. Bởi vì, lợi ích mà công nghệ thông tin đem lại cho đời sống rất nhiều như là khả năng quản lý thông tin vô cùng chính xác, nhanh chóng và mức độ loan truyền thông tin cũng rất nhanh. Chúng ta có thể kể đến sự phát triển khác của ngành công nghệ thông tin, đó là việc sử dụng Internet ngày càng phổ biến. Nhờ Internet mà chỉ cần một cú nhấp chuột, con người có thể truy cập các websites và tiếp cận thông tin một cách dễ dàng và tiện lợi hơn bao giờ hết.

Như chúng ta đã biết, sự kiện và hoạt động ngoại khóa là một phần quen thuộc của các bạn sinh viên. Để giúp sinh viên có thể dễ dàng tiếp cận với các thông tin về sự kiện, hoạt động tại trường Đại học, giúp sinh viên nắm được các thông tin quan trọng thì một website tin tức là rất cần thiết. Website tin tức là những trang web trang báo điện tử, được thiết kế chuyên biệt dùng để truyền tải những thông tin mới nhất, nóng hổi nhất ở một lĩnh vực cụ thể nào đó hoặc đa dạng lĩnh vực trong cuộc sống hiện nay. Các độc giả sẽ dễ dàng tiếp cận được những tin tức mới nhất trên website tin tức thông qua chiếc điện thoại thông minh hoặc là laptop, máy tính,...khi có kết nối Internet. Nhờ có website tin tức mà tốc độ lĩnh hội được đa dạng kiến thức cũng như các tin tức về đời sống, xã hội của con người ngày càng nhanh và hiệu quả. Vì vậy có thể thấy website tin tức là một trong những lĩnh vực website không thể thiếu trong nền tảng Internet hiện nay.

Chúng em rất mong được sự chỉ dẫn quý báu của thầy cô, cùng các bạn sinh viên tạo điều kiện cho chúng em có thể nâng cấp, hoàn chỉnh hệ thống ngày càng tốt hơn nhằm mang lại một hiệu quả thực tiễn nhất.

Chương 1 . TỔNG QUAN

1.1 Giới thiệu đề tài

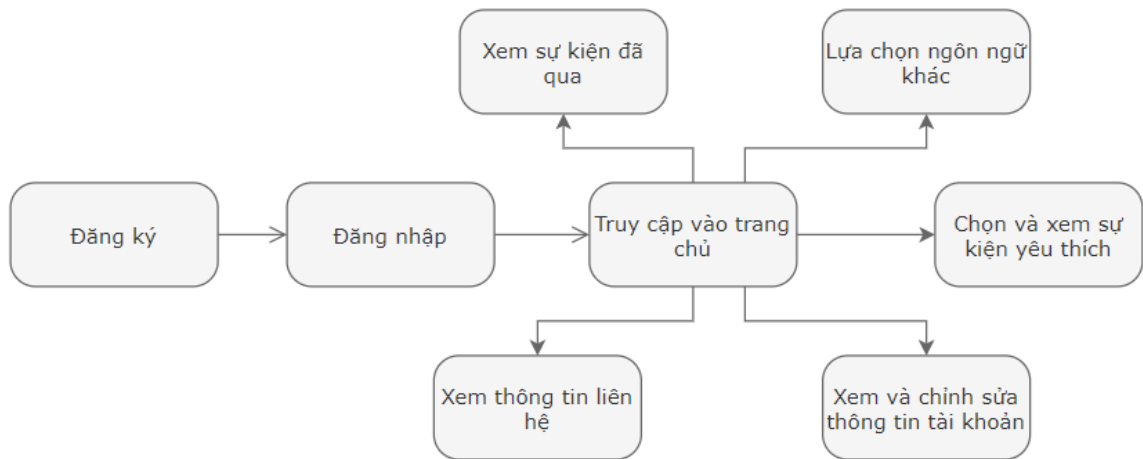
Sự kiện và hoạt động là một phần không thể thiếu của trường Đại học Kinh tế - Tài chính (UEF), hằng năm rất nhiều hoạt động bổ ích, có ý nghĩa được tổ chức. Tuy nhiên cùng với sự phát triển và mở rộng của nhà trường, các sự kiện và hoạt động dần đa dạng và phân khoa nhiều hơn. Điều này làm cho các bạn sinh viên UEF gặp khó khăn trong việc cập nhật đầy đủ thông tin về sự kiện, hoạt động được tổ chức. Để giải quyết vấn đề này nhóm chúng em đã xây dựng Website quản lý các sự kiện, hoạt động tại UEF. Website này đóng vai trò hỗ trợ cả về phía nhà trường và phía học sinh, giúp nhà trường quản lý sự kiện một cách logic và đơn giản đồng thời cung cấp thông tin về sự kiện cho sinh viên một cách sinh động qua giao diện gần gũi người dùng.

1.2 Mục tiêu đề tài

Mục tiêu của luận văn là tạo ra một chương trình để hỗ trợ quản lý các sự kiện, hoạt động tại trường UEF. Giúp cho việc cập nhật thông tin diễn ra nhanh chóng và dễ dàng hơn.

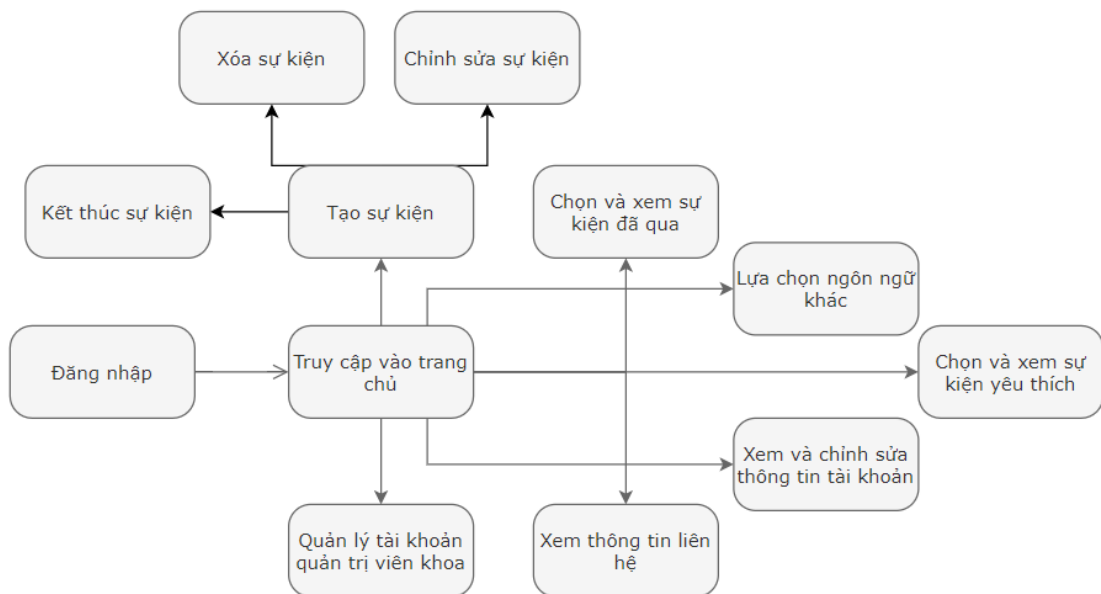
Sau khi phân tích các yêu cầu nghiệp vụ và tìm hiểu thực tế, tài liệu mô tả. Các chức năng và trình tự chức năng của chương trình sẽ được thực theo mô hình sau.

- Phần của sinh viên:



Hình 1: Sơ đồ chức năng của sinh viên

- Phần chức năng của quản trị viên tổng:



Hình 2: Sơ đồ chức năng của quản trị viên tổng.

1.3 Bố cục đề tài

- Chương 1: Tổng quan
 - Giới thiệu đề tài: Bài toán cụ thể, vấn đề cần giải quyết.
 - Mục tiêu đề tài: Ý nghĩa khoa học và thực tiễn của đề tài hướng tới.
 - Bố cục đề tài: Cấu trúc của đồ án gồm các chương và tóm tắt từng chương.
- Chương 2: Cơ sở lý thuyết
 - Giới thiệu, giải thích các phương pháp – lý thuyết – và công nghệ sử dụng để giải quyết nhiệm vụ của đồ án.
- Chương 3: Phân tích và thiết kế hệ thống.
 - Nêu chi tiết vấn đề cần giải quyết như mô hình hệ thống, cấu trúc chương trình, các sơ đồ UML.
 - Nêu các tính chất ràng buộc của bài toán.
 - Nêu các bước thực hiện.
 - Giải thích từng hình, từng bảng.
- Chương 4: Thiết kế giao diện và chức năng
 - Trình bày các chức năng của sản phẩm thông qua sơ đồ chức năng.
 - Trình bày các giao diện chính của website và giải thích ngắn gọn ở mỗi giao diện.
- Chương 5: Kết luận
 - Nêu những kết quả mà nhóm đạt được thông qua thời gian hai tháng.
 - Nêu ra những vấn đề chưa được giải quyết của sản phẩm đồ án.
 - Đưa ra hướng phát triển cho sản phẩm trong tương lai.

Chương 2 . CƠ SỞ LÝ THUYẾT

2.1 Node JS

2.1.1 *Khái niệm về Node JS*

Nodejs là một nền tảng (Platform) phát triển độc lập được xây dựng ở trên Javascript Runtime của Chrome mà chúng ta có thể xây dựng được các ứng dụng mạng một cách nhanh chóng và dễ dàng mở rộng. Phần Core bên dưới của Nodejs được viết hầu hết bằng C++ nên cho tốc độ xử lý và hiệu năng khá cao. Nodejs tạo ra được các ứng dụng có tốc độ xử lý nhanh, realtime thời gian thực nên được áp dụng cho các sản phẩm có lượng truy cập lớn, cần mở rộng nhanh, cần đổi mới công nghệ, hoặc tạo ra các dự án Startup nhanh nhất có thể.

2.1.2 *NPM và Package*

2.1.2.1 *NPM*

NPM (**N**ode **P**ackage **M**anager) là một công cụ tạo và quản lý các thư viện lập trình Javascript cho NodeJS. Trong cộng đồng Javascript, các lập trình viên chia sẻ hàng trăm nghìn các thư viện với các đoạn code đã thực hiện sẵn một chức năng nào đó. Nó giúp cho các dự án mới tránh phải viết lại các thành phần cơ bản, các thư viện lập trình hay thậm chí cả các framework.

2.1.2.2 *Package*

Dựa theo chức năng mà ta chia package ra làm hai loại, đó là Simple dependencies (hay Dependencies) và Development dependencies.

Dependencies là những package bắt buộc phải có trong quá trình chạy sản phẩm, kiểu như một thư viện cung cấp các hàm mà code của bạn cần. Khi cài đặt dependencies, NPM sẽ tự động cài đặt tất cả các dependencies cần thiết.

Development dependencies là những package bắt buộc khi phát triển cũng như phát hành sản phẩm. Ví dụ như các trình biên dịch giúp biên dịch đoạn code của bạn về javascript, các framework phục vụ cho việc kiểm thử,... Khi cài đặt Development Dependencies, NPM sẽ chỉ cài đặt các dependencies mà cần thiết.

2.1.2.3 Express Framework

Express là một framework giành cho Node JS. Nó cung cấp rất nhiều tính năng mạnh mẽ trên nền tảng Web cũng như trên các ứng dụng di động. Express hỗ trợ các phương thức HTTP và middleware tạo ra một API vô cùng mạnh mẽ và dễ sử dụng.

Một số chức năng chính của Express như sau:

- Thiết lập các lớp trung gian để trả về các HTTP request.
- Định nghĩa router cho phép sử dụng với các hành động khác nhau dựa trên phương thức HTTP và URL.
- Cho phép trả về các trang HTML dựa vào các tham số.

2.2 React JS

2.2.1 Giới thiệu về React JS

ReactJS là một thư viện JavaScript có tính hiệu quả và linh hoạt để xây dựng các thành phần giao diện người dùng (UI) có thể sử dụng lại. ReactJS giúp phân chia các UI phức tạp thành các thành phần nhỏ (được gọi là component) giúp chúng ta dễ dàng quản lý, mở rộng hệ thống. React JS luôn giữ các components ở trạng thái stateless (nhiều nhất có thể) khiến ta dễ dàng quản lý bởi nó chẳng khác gì một trang static HTML. Bản thân các components này không có trạng thái nhận đầu vào từ bên ngoài và chỉ hiển thị ra dựa vào các đầu vào đó, điều này lý giải tại sao nó lại mang tính tái sử dụng và dễ dàng cho việc test. React JS là một framework hiển thị view chú ý đến hiệu năng (performanceminded). Rất nhiều đối thủ nặng ký về framework MVVM (Model-View-ViewModel) mất một thời gian lớn để hiển thị

những lượng data lớn, như trong trường hợp những danh sách (list) và tương tự. Nhưng với React đó không còn là vấn đề, vì nó chỉ hiển thị những gì thay đổi.

React JS được tạo ra bởi Jordan Walke, một kỹ sư phần mềm tại Facebook. **ReactJS** ban đầu được phát triển và duy trì bởi Facebook và sau đó được sử dụng trong các sản phẩm của mình như **WhatsApp & Instagram**.

ReactJS được dùng để xây dựng các ứng dụng [*single page application*] (SPA). Một trong những điểm hấp dẫn của ReactJS là nó không chỉ được xây dựng bên phía clients mà còn sử dụng được bên phía server.

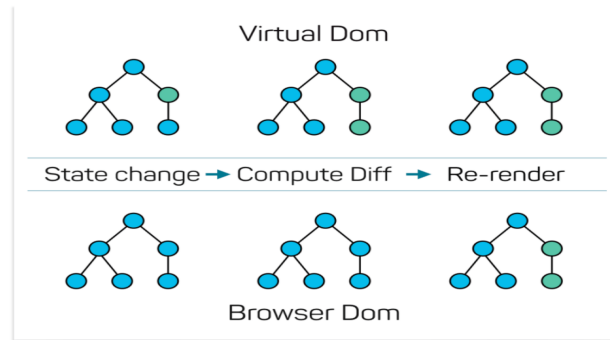
2.2.2 Các khái niệm cơ bản về React JS

2.2.2.1 Virtual DOM

Virtual DOM là một đại diện ảo của DOM (Document Object Model). Mỗi khi trạng thái ứng dụng thay đổi, DOM ảo được cập nhật thay vì DOM thực.

Khi các phần tử mới được thêm vào giao diện người dùng, một DOM ảo, được biểu thị dưới dạng cây sẽ được tạo. Mỗi phần tử là một nút trên cây này. Nếu trạng thái của bất kỳ phần tử nào trong số này thay đổi. Cây này sau đó được so sánh hoặc "diffing" với cây DOM ảo trước đó.

Khi điều này được thực hiện xong, DOM ảo sẽ tính toán phương pháp tốt nhất có thể để thực hiện những thay đổi này đối với DOM thực. Điều này đảm bảo rằng có các hoạt động tối thiểu trên DOM thực. Do đó, giảm chi phí hiệu suất của việc cập nhật DOM thực.



Hình 2-1: Sự khác nhau về cách hoạt động của Virtual Dom và DOM

2.2.2.2 JSX

JSX (Javascript XML), là cấu trúc XML được viết bên trong cấu trúc Javascript, cho phép viết các đoạn mã HTML trong React một cách dễ dàng và có cấu trúc hơn.

Đặc điểm của JSX:

- Faster (Nhanh hơn): JSX thực hiện tối ưu hóa trong khi biên dịch sang mã Javascript. Các mã này cho thời gian thực hiện nhanh hơn nhiều so với một mã tương đương viết trực tiếp bằng Javascript.
- Safer (An toàn hơn): Ngược với Javascript, JSX là kiểu statically-typed , nghĩa là nó được biên dịch trước khi chạy, giống như Java, C++. Vì thế các lỗi sẽ được phát hiện ngay trong quá trình biên dịch.
- Easier (Dễ dàng hơn): JSX kế thừa dựa trên Javascript, vì vậy rất dễ dàng để cho các lập trình viên Javascripts có thể sử dụng.

2.2.2.3 Thành phần (Component)

Trong React, các component hoạt động giống như các hàm trả về các thành phần HTML

Các component là các thành phần độc lập và có thể sử dụng lại, các component thực hiện công việc giống như các functions trong JavaScript nhưng

chúng độc lập và nhiệm vụ chính là trả về HTML thông qua hàm `render`. Có 2 loại component là Function Component và Class Component.

2.2.2.4 Props và State

Props là một tham số được chuyển qua lại giữa các React Components, các props này được truyền qua các component với cú pháp giống như là HTML attributes.

State là một object mà lưu trữ giá trị của các thuộc tính bên trong components và chỉ tồn tại trong phạm vi của component đó. Mỗi khi bạn thay đổi giá trị của một state thì component đó sẽ được render lại.

2.2.2.5 React Lifecycle

React Lifecycle là một vòng đời của component, khi chúng ta tiến hành render một component thì **ReactJS** thực hiện nhiều tiến trình khác nhau, các tiến trình này được lặp đi lặp lại đối với các component.

Component có thể được viết theo kiểu của vòng đời này như sau:

- Tạo vòng đời (Mount trên DOM)
- Cập nhật vòng đời (Updating)
- Kết thúc vòng đời (Unmount từ DOM)

2.3 Hệ quản trị Cơ sở dữ liệu MySQL

2.3.1 Giới thiệu về MySQL

SQL là ngôn ngữ phổ biến nhất để thêm, truy cập và quản lý nội dung trong cơ sở dữ liệu. Nó được chú ý nhất vì khả năng xử lý nhanh, độ tin cậy đã được chứng minh, dễ sử dụng và linh hoạt. MySQL là một phần thiết yếu của hầu hết mọi ứng dụng PHP mã nguồn mở. Các ví dụ điển hình cho các tập lệnh dựa trên PHP và MySQL là WordPress, Joomla, Magento và Drupal. MySQL đang trở nên phổ biến vì nhiều lý do tốt:

- MySQL được phát hành theo giấy phép nguồn mở. Vì vậy, bạn không phải trả tiền để sử dụng nó.
- MySQL là một chương trình rất mạnh theo đúng nghĩa của nó. Nó xử lý một tập hợp lớn các chức năng của các gói cơ sở dữ liệu mạnh mẽ và đắt tiền nhất.
- MySQL sử dụng một dạng chuẩn của ngôn ngữ dữ liệu SQL nổi tiếng.
- MySQL hoạt động trên nhiều hệ điều hành và với nhiều ngôn ngữ bao gồm PHP, PERL, C, C ++, JAVA, v.v.
- MySQL hoạt động rất nhanh và hoạt động tốt ngay cả với các tập dữ liệu lớn.
- MySQL rất thân thiện với PHP, ngôn ngữ được đánh giá cao nhất để phát triển web.
- MySQL hỗ trợ cơ sở dữ liệu lớn, lên tới 50 triệu hàng hoặc nhiều hơn trong một bảng. Giới hạn kích thước tệp mặc định cho một bảng là 4GB, nhưng bạn có thể tăng mức này (nếu hệ điều hành của bạn có thể xử lý nó) đến giới hạn lý thuyết là 8 triệu terabyte (TB).
- MySQL là tùy biến. Giấy phép GPL mã nguồn mở cho phép các lập trình viên sửa đổi phần mềm MySQL để phù hợp với môi trường cụ thể của riêng họ.

2.3.2 Cách thức hoạt động của MySQL

MySQL dựa trên mô hình client-server. Cốt lõi của MySQL là máy chủ MySQL, xử lý tất cả các hướng dẫn cơ sở dữ liệu (hoặc các lệnh). Máy chủ MySQL có sẵn như là một chương trình riêng biệt để sử dụng trong môi trường mạng client-server và như một thư viện có thể được nhúng (hoặc liên kết) vào các ứng dụng riêng biệt.

MySQL hoạt động cùng với một số chương trình tiện ích hỗ trợ quản trị cơ sở dữ liệu MySQL. Các lệnh được gửi đến MySQLServer thông qua máy khách MySQL, được cài đặt trên máy tính.

MySQL ban đầu được phát triển để xử lý cơ sở dữ liệu lớn một cách nhanh chóng. Mặc dù MySQL thường chỉ được cài đặt trên một máy, nhưng nó có thể gửi cơ sở dữ liệu đến nhiều vị trí, vì người dùng có thể truy cập thông qua các giao diện máy khách MySQL khác nhau. Các giao diện này gửi các câu lệnh SQL đến máy chủ và sau đó hiển thị kết quả.

2.3.3 Tính năng cốt lõi của MySQL

MySQL cho phép dữ liệu được lưu trữ và truy cập trên nhiều công cụ lưu trữ, bao gồm InnoDB, CSV và NDB. MySQL cũng có khả năng sao chép dữ liệu và phân vùng bảng để có hiệu suất và độ bền tốt hơn. Người dùng MySQL không bắt buộc phải học các lệnh mới; họ có thể truy cập dữ liệu của mình bằng các lệnh SQL tiêu chuẩn.

MySQL được viết bằng C và C++ và có thể truy cập và có sẵn trên hơn 20 nền tảng, bao gồm Mac, Windows, Linux và Unix. RDBMS hỗ trợ cơ sở dữ liệu lớn với hàng triệu bản ghi và hỗ trợ nhiều loại dữ liệu bao gồm các số nguyên có chữ ký hoặc không dấu có độ dài 1, 2, 3, 4 và 8 byte(s); FLOAT; DOUBLE; CHAR; VARCHAR; BINARY; VARBINARY; TEXT; BLOB; DATE; TIME; DATETIME; TIMESTAMP; YEAR; SET; ENUM; và các kiểu OpenGIS. Các loại chuỗi có độ dài cố định và biến đổi cũng được hỗ trợ.

Để bảo mật, MySQL sử dụng một đặc quyền truy cập và hệ thống mật khẩu được mã hóa cho phép xác minh dựa trên máy chủ. Các máy khách MySQL có thể kết nối với Máy chủ MySQL bằng một số giao thức, bao gồm cả giao thức TCP/IP trên bất kỳ nền tảng nào. MySQL cũng hỗ trợ một số chương trình máy khách và tiện ích, chương trình dòng lệnh và công cụ quản trị như MySQL Workbench.

2.4 Các phần mềm hỗ trợ khác

2.4.1 Xampp (*Phần mềm test trang web trên local host*)

Xampp là chương trình tạo máy chủ Web được tích hợp sẵn Apache, PHP, MySQL, FTP Server, Mail Server, phpMyAdmin và các công cụ như phpMyAdmin. Không như Appserv, Xampp có chương trình quản lý khá tiện lợi, cho phép chủ động bật tắt hoặc khởi động lại các dịch vụ máy chủ bất kỳ lúc nào.

2.4.2 Figma

Figma là phần mềm dựa trên nền tảng web (web - base app) chuyên dụng dùng để thiết kế giao diện người dùng UI/UX cho cả web và app. Figma có thể làm việc trên mọi hệ điều hành, hoạt động nhanh và bao gồm các tính năng cộng tác làm việc nhóm trong thời gian thực. Figma giúp các nhóm phát triển tạo, thử nghiệm và chuyển giao các dự án thiết kế tốt hơn từ đầu cho đến khi hoàn thành.

2.4.3 Draw.io

Draw.io là một ứng dụng vẽ biểu đồ miễn phí cho phép người dùng tạo và chia sẻ các sơ đồ trên trình duyệt web. Công cụ trực tuyến này hoạt động với G Suite / Google Drive và Dropbox. Người dùng cũng có thể làm việc trên sơ đồ ngoại tuyến và lưu trữ cục bộ bằng ứng dụng draw.io dành cho macOS, Windows và Linux. Draw.io cung cấp giao diện trực quan với tính năng kéo và thả, các mẫu sơ đồ có thể tùy chỉnh và thư viện hình khối mở rộng. Người dùng có thể tạo và chỉnh sửa một loạt các sơ đồ, biểu đồ bao gồm biểu đồ luồng, biểu đồ quy trình, sơ đồ tổ chức, sơ đồ ER, UML, sơ đồ mạng, ...

Chương 3 . PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

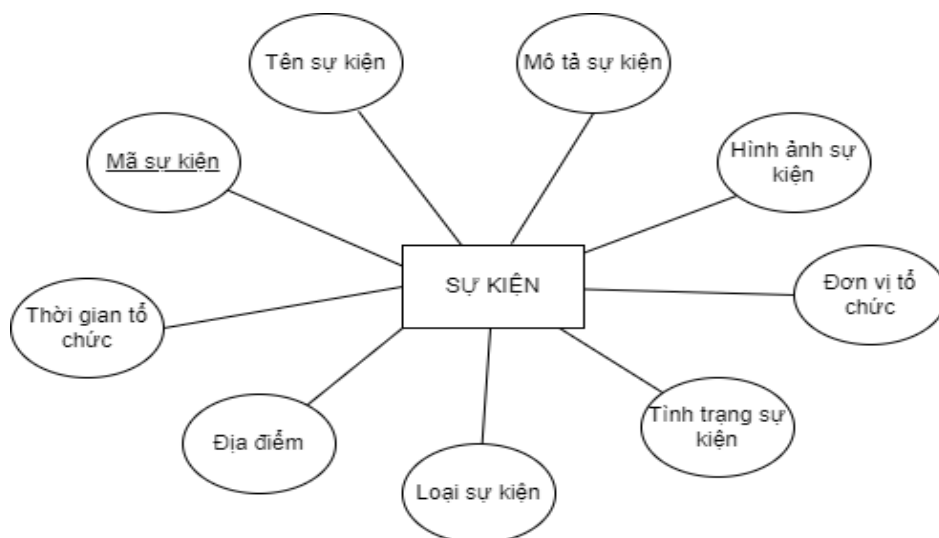
3.1 Phân tích

3.1.1 Phân tích thực thể

Website quản lý sự kiện, hoạt động tại UEF có hai thực thể là "Sự kiện" và "Người dùng".

- Thực thể Sự kiện (Event):

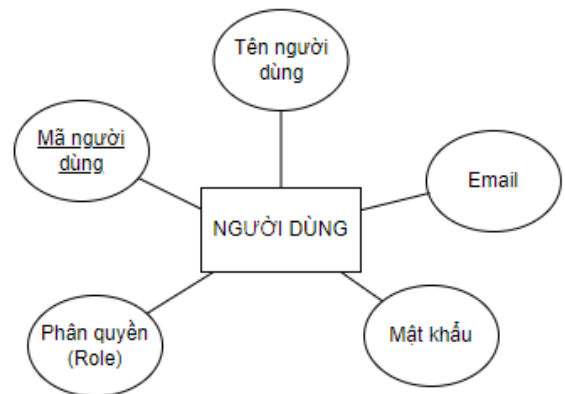
- Mã sự kiện
- Tên sự kiện
- Mô tả sự kiện
- Hình ảnh sự kiện
- Đơn vị tổ chức
- Loại sự kiện
- Địa điểm
- Thời gian tổ chức
- Tình trạng sự kiện



Hình 3-1: Thực thể sự kiện

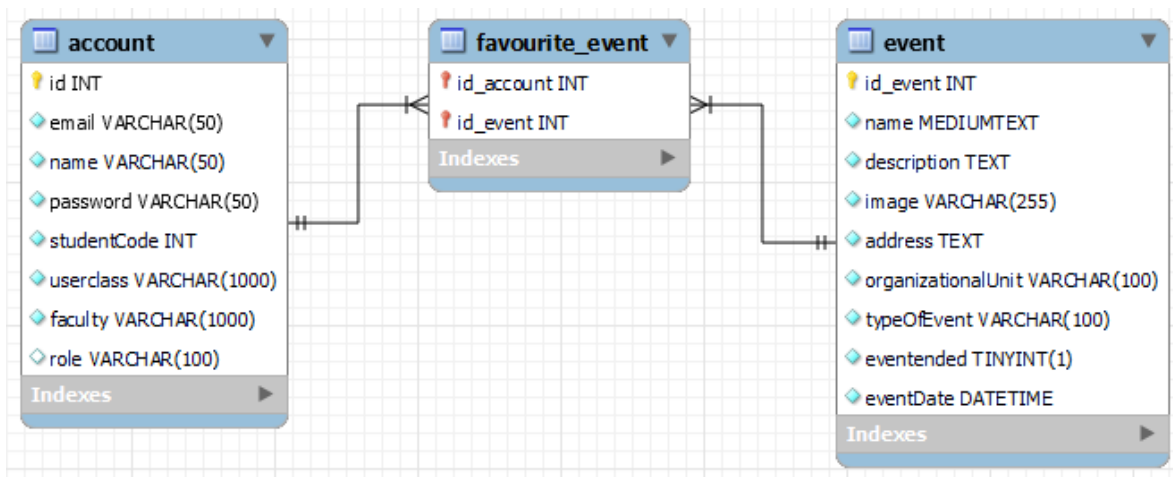
- Thực thể Người dùng (User)

- Mã người dùng
- Tên người dùng
- Email
- Mật khẩu
- Phân quyền (Role)



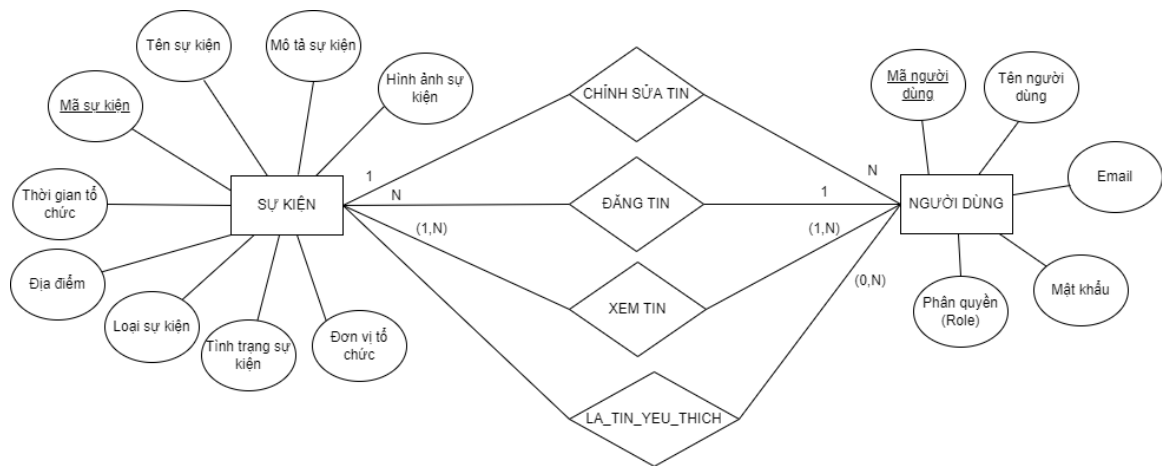
Hình 3-2: Thực thể Người dùng

3.1.2 Lược đồ quan hệ



Hình 3-3: Lược đồ quan hệ

3.1.3 Mô hình quan niệm dữ liệu ERD (Mô hình thực thể kết hợp)



Hình 3-4: Mô hình thực thể kết hợp

3.2 Ràng buộc dữ liệu

3.2.1 Ràng buộc khóa chính

- Ràng buộc R1: **id** là khóa chính của bảng **account**
- Ràng buộc R2: **id_event** là khóa chính của bảng **event**

3.2.2 Ràng buộc khóa phụ

- Ràng buộc R3:

Bối cảnh: Trên quan hệ **account** và **favourite_event**

Nội dung: **id_account** là khóa ngoại của bảng **account** tham chiếu **account (id)**

$$\text{favourite_event}[\text{id_account}] \subset \text{account}[\text{id}]$$

Bảng tầm ảnh hưởng:

Quan hệ	Thêm	Xóa	Sửa
Account	-	+	+(id)
Favourite_event	+	+	+(id_account)

– Ràng buộc R4:

Bối cảnh: Trên quan hệ **event** và **favourite_event**

Nội dung: **id_event** là khóa ngoại của bảng **event** tham chiếu **event (id_event)**

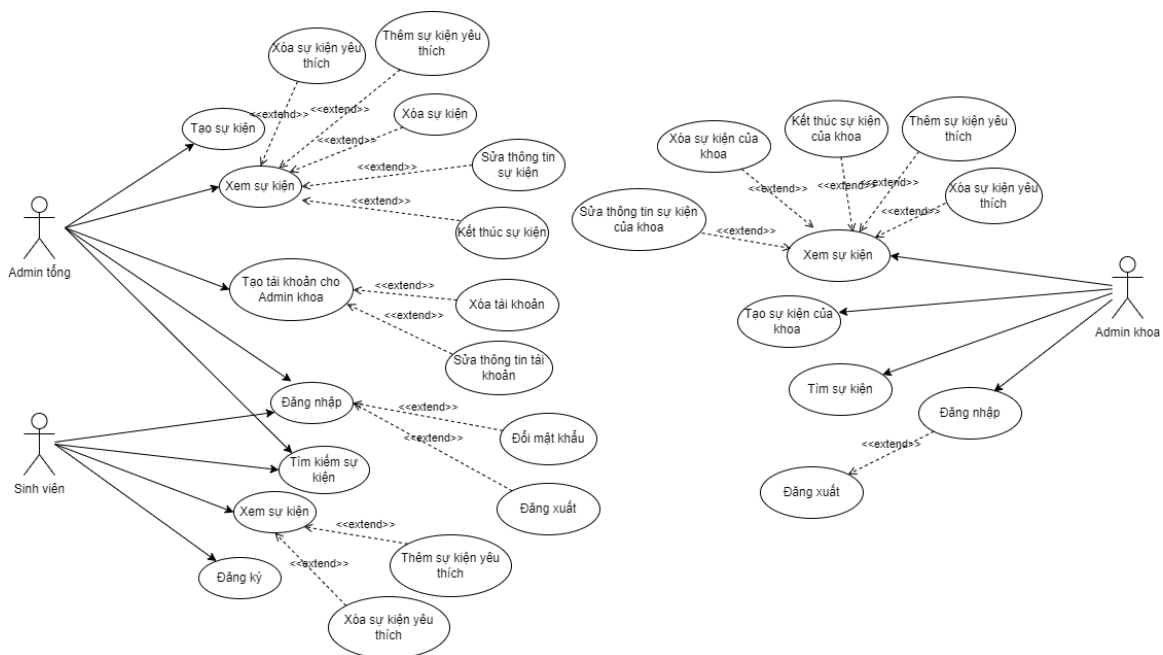
$\text{favourite_event}[\text{id_event}] \subset \text{event}[\text{id_event}]$

Bảng tầm ảnh hưởng:

Quan hệ	Thêm	Xóa	Sửa
Event	-	+	+(id_event)
Favourite_event	+	+	+(id_event)

3.3 Phân tích hệ thống

3.3.1 Sơ đồ Use-Case



Hình 3-5: Sơ đồ Use-Case

3.3.2 Đặc tả một số Use-Case

– Đặc tả Use-Case Tạo sự kiện:

Bảng 3-1: Đặc tả Use-Case: Tạo sự kiện

Use Case ID	1	
Tên Use Case	UC Tạo sự kiện	
Mô tả	Quản trị viên đăng sự kiện lên trang chủ	
Tác nhân (Actor)	Quản trị viên	
Pre-conditions	Đã đăng nhập và trang web xác thực là quản trị viên	
Dòng sự kiện (Flow)	Chính (Basic)	<ol style="list-style-type: none"> 1. Chọn biểu tượng (+) để đăng sự kiện 2. Giao diện tạo sự kiện mới hiện lên. Quản trị viên nhập các thông tin gồm tên sự kiện, mô tả sự kiện, địa điểm, hình ảnh, ngày giờ tổ chức, đơn vị tổ chức và loại sự kiện. 3. Sau khi nhập đầy đủ thông tin thì nhấn Đăng sự kiện 4. Hoàn tất đăng tin. Tin sẽ được tải lên trang chủ web.
	Thay thế (Alternative)	2-a. Quản trị viên ghi thiếu thông tin mà trang web bắt buộc phải ghi. Web hiện thông báo để Quản trị viên bổ sung thông tin.
	Ngoại lệ (Exception)	Quản trị viên thoát khỏi form tạo sự kiện mới, kết thúc việc đăng tin.

		Use case kết thúc.
Post-conditions	Bài đăng của quản trị viên được lưu vào database của website, bài đăng xuất hiện trên trang chủ.	
Mở rộng		

– Đặc tả *Use-Case* Sửa thông tin sự kiện (Cập nhật sự kiện):

Bảng 3-2: Đặc tả Use-Case: Sửa thông tin sự kiện (Cập nhật sự kiện)

Use Case ID	2	
Tên Use Case	UC Sửa thông tin sự kiện	
Mô tả	Quản trị viên cập nhật lại thông tin của một sự kiện đã đăng	
Tác nhân (Actor)	Quản trị viên	
Pre-conditions	Đã đăng nhập, trang web xác thực là quản trị viên, đã có sự kiện sẵn	
Dòng sự kiện (Flow)	Chính (Basic)	<ol style="list-style-type: none"> 1. Chọn biểu tượng (...) để mở thanh Dropdown, chọn biểu tượng Chỉnh sửa. 2. Giao diện chỉnh sửa sự kiện hiện lên. Quản trị viên cập nhật các thông tin muốn thay đổi (tên sự kiện, mô tả sự kiện, địa điểm, hình ảnh, ngày giờ tổ chức, đơn vị tổ chức và loại sự kiện). 3. Sau khi cập nhập đủ thông tin muốn chỉnh sửa thì nhấn Cập nhật. 4. Hoàn tất cập nhật tin. Tin sẽ được tải lên

		trang chủ web.
	Thay thế (Alternative)	
	Ngoại lệ (Exception)	Quản trị viên thoát khỏi giao diện chỉnh sửa sự kiện, kết thúc việc cập nhật sự kiện Use case kết thúc.
Post-conditions	Bài đăng sau khi được cập nhật được lưu vào database của website, bài đăng xuất hiện trên trang chủ.	
Mở rộng		

– Đặc tả *Use-Case* Xóa sự kiện :

Bảng 3-3: Đặc tả Use-Case: Xóa sự kiện

Use Case ID	3	
Tên Use Case	UC Xóa sự kiện	
Mô tả	Quản trị viên xóa sự kiện đã đăng	
Tác nhân (Actor)	Quản trị viên	
Pre-conditions	Đã đăng nhập, trang web xác thực là quản trị viên, đã có sự kiện sẵn	
Dòng sự kiện (Flow)	Chính (Basic)	<ol style="list-style-type: none"> 1. Chọn biểu tượng (...) để mở thanh Dropdown, chọn biểu tượng Xóa 2. Aalertbox hiện lên, thông báo liệu quản trị viên có muốn xóa hay không

		<p>3. Chọn button “Có”, tin được xóa khỏi website</p> <p>4. Hoàn tất xóa tin.</p>
	Thay thế (Alternative)	3-a. Chọn nút “ Không”, không xóa sự kiện khỏi website
	Ngoại lệ (Exception)	
Post-conditions	Bài đăng xóa khỏi database và website	
Mở rộng		

– Đặc tả *Use-Case* Kết thúc sự kiện:

Bảng 3-4: Đặc tả Use-Case: Kết thúc sự kiện

Use Case ID	4	
Tên Use Case	UC Kết thúc sự kiện	
Mô tả	Quản trị viên kết thúc sự kiện	
Tác nhân (Actor)	Quản trị viên	
Pre-conditions	Đã đăng nhập, trang web xác thực là quản trị viên, đã có sự kiện sẵn	
Dòng sự kiện (Flow)	Chính (Basic)	<p>1. Chọn biểu tượng (...) để mở thanh Dropdown, chọn biểu tượng Kết thúc sự kiện.</p>

		2. Alertbox hiện lên, thông báo liệu quản trị viên có muốn kết thúc sự kiện hay không 3. Chọn button “Có”, sự kiện được chuyển vào trang Sự kiện đã qua 4. Hoàn tất kết thúc sự kiện.
	Thay thế (Alternative)	3-a. Chọn button “ Không”, sự kiện không kết thúc và vẫn hiện bên trang chủ
	Ngoại lệ (Exception)	
Post-conditions	Cập nhật database, bài đăng được chuyển sang trang Sự kiện đã qua.	
Mở rộng		

Mô tả use case chức năng Tạo, Xóa, Sửa, Kết thúc sự kiện khoa của quản trị viên khoa giống quản trị viên tổng nhưng phạm vi chỉ bao gồm khoa mà quản trị viên khoa quản lý.

– Đặc tả *Use-Case* Tìm kiếm sự kiện:

Bảng 3-5: Đặc tả Use-Case: Tìm kiếm sự kiện

Use Case ID	5
Tên Use Case	UC Tìm kiếm sự kiện
Mô tả	Người dùng tìm kiếm sự kiện
Tác nhân (Actor)	Quản trị viên tổng, quản trị viên khoa và sinh viên
Pre-conditions	Đã đăng nhập thành công

Dòng sự kiện (Flow)	Chính (Basic)	<ol style="list-style-type: none"> 1. Nhập từ khóa vào ô tìm kiếm 2. Nhấn button biểu tượng kính lúp 3. Trang web lọc và hiện các sự kiện gần giống với từ khóa tìm kiếm 4. Hoàn tất tìm kiếm sự kiện
	Thay thế (Alternative)	1-a. Nếu muốn tìm theo khoa, chọn “Khoa tổ chức” ở Navbar. Trang web sẽ hiện ra sự kiện theo Khoa người dùng chọn
	Ngoại lệ (Exception)	2-a. Người dùng không nhấn button kính lúp, không lọc được sự kiện. Use case tìm kiếm kết thúc
Post-conditions	Hiện thị các sự kiện giống với từ khóa tìm kiếm.	
Mở rộng		

– Đặc tả *Use-Case* Thêm sự kiện yêu thích:

Bảng 3-6: Đặc tả Use-Case: Thêm sự kiện yêu thích

Use Case ID	6
Tên Use Case	UC Thêm sự kiện yêu thích
Mô tả	Người dùng chọn thêm sự kiện yêu thích để lưu vào trang sự kiện yêu thích
Tác nhân (Actor)	Admin tổng, admin khoa và sinh viên
Pre-conditions	Đã đăng nhập, đã có sự kiện sẵn

Dòng sự kiện (Flow)	Chính (Basic)	<ol style="list-style-type: none"> 1. Chọn nút “Yêu thích” 2. Alertbox hiện lên, thông báo đã thêm sự kiện yêu thích thành công 3. Hoàn tất thêm tin yêu thích.
	Thay thế (Alternative)	
	Ngoại lệ (Exception)	
Post-conditions	Bài đăng được thêm vào trang sự kiện yêu thích.	
Mở rộng		

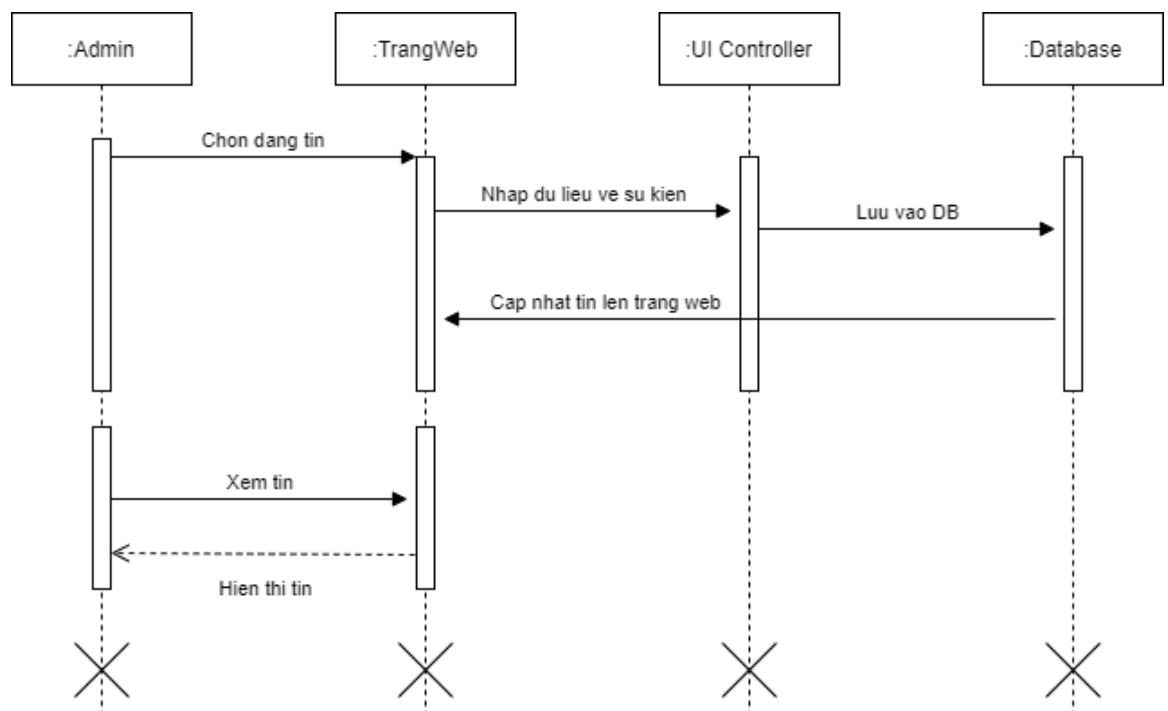
– Đặc tả *Use-Case* Xóa sự kiện yêu thích:

Bảng 3-7: Đặc tả Use-Case: Xóa sự kiện yêu thích

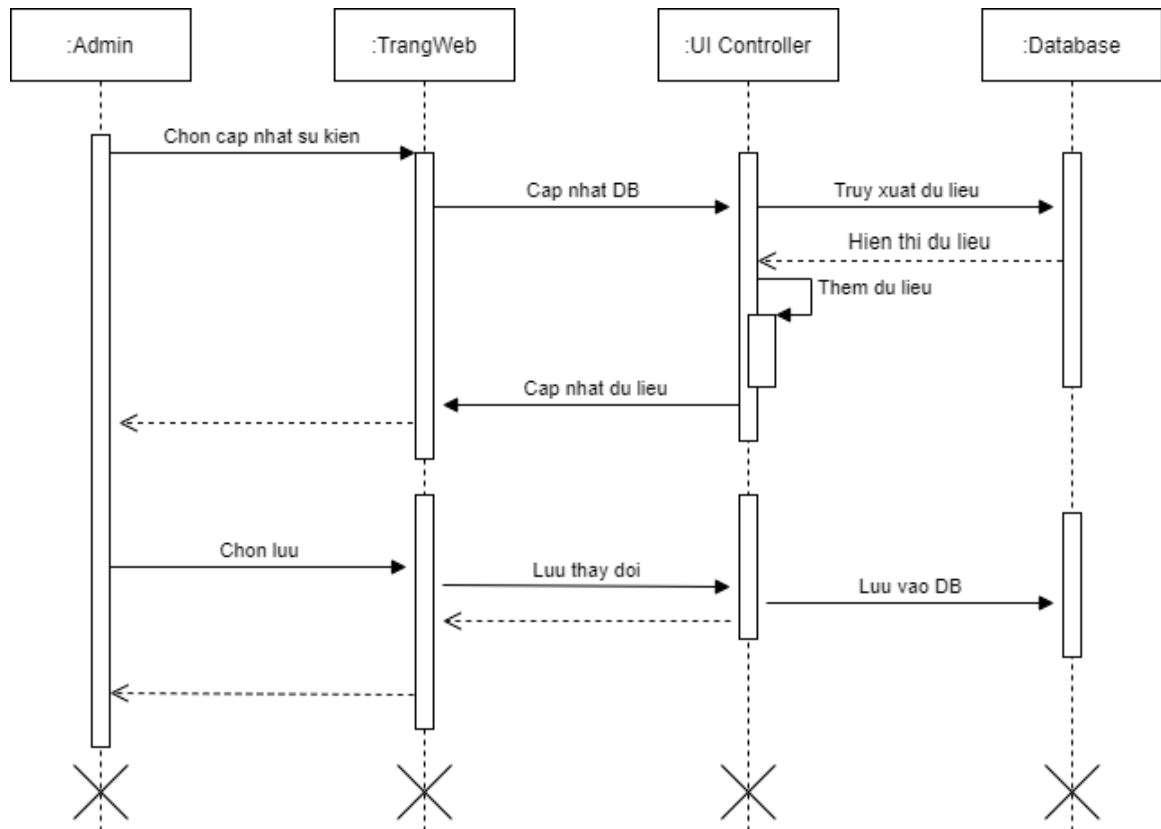
Use Case ID	7	
Tên Use Case	UC Xóa sự kiện yêu thích	
Mô tả	Người dùng chọn xóa sự kiện yêu thích.	
Tác nhân (Actor)	Admin tổng, admin khoa và sinh viên	
Pre-conditions	Đã đăng nhập, đã có sự kiện sẵn trong trang sự kiện yêu thích	
Dòng sự kiện (Flow)	Chính (Basic)	<ol style="list-style-type: none"> 1. Chọn nút “Xóa yêu thích” 2. Hoàn tất xóa tin yêu thích.
	Thay thế (Alternative)	

	Ngoại lệ (Exception)	
Post-conditions	Bài đăng được xóa khỏi trang sự kiện yêu thích.	
Mở rộng		

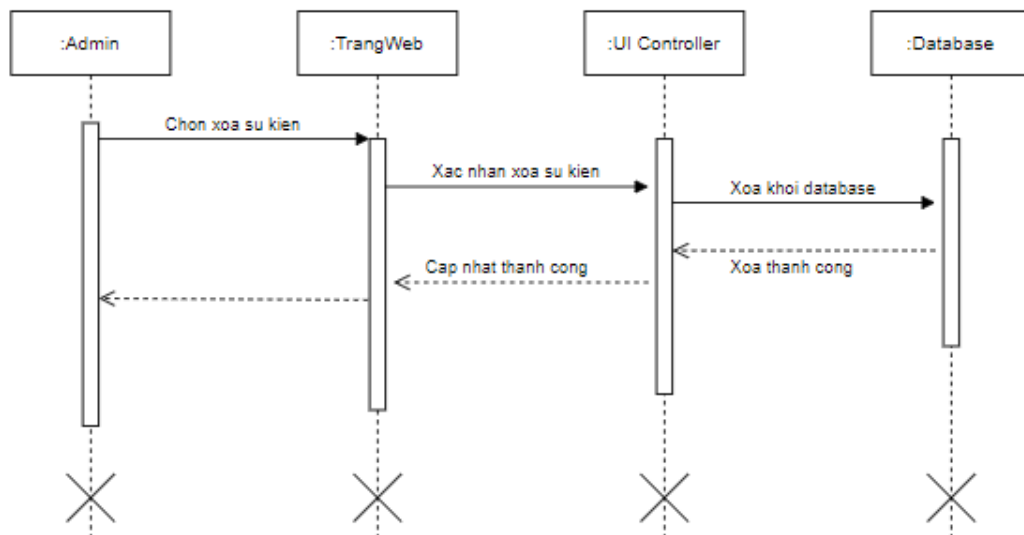
3.3.3 Sơ đồ tuần tự (Sơ đồ tương tác đối tượng)



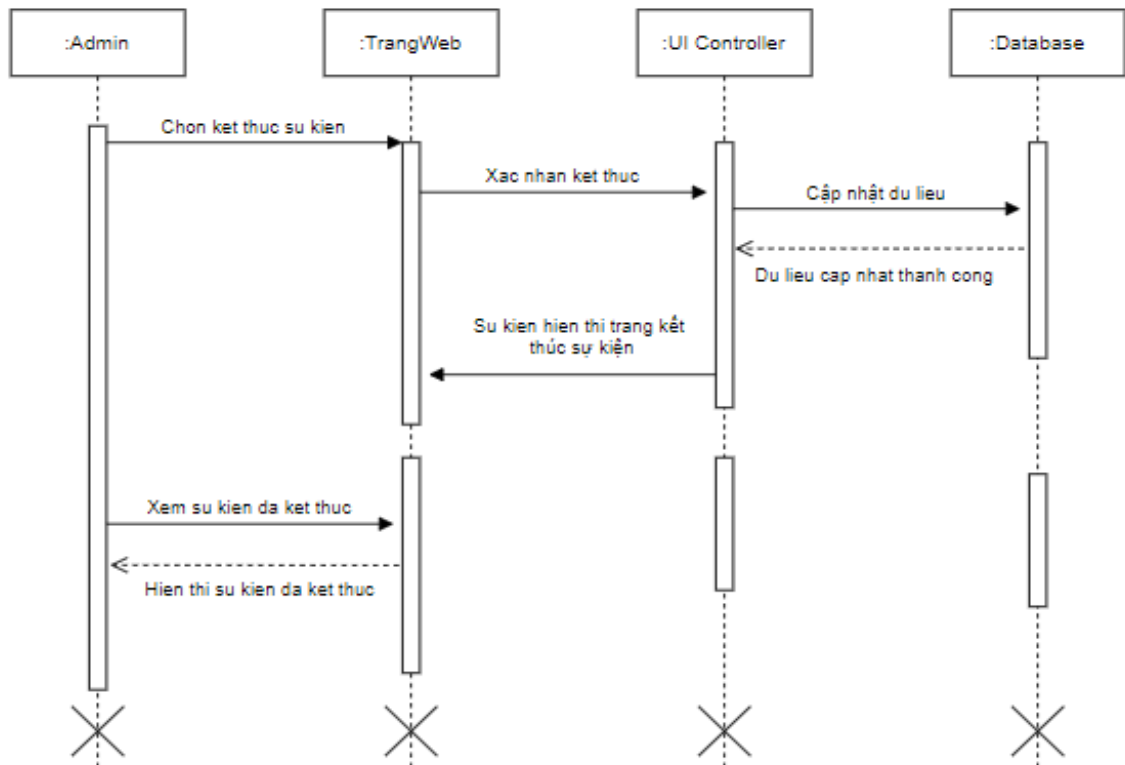
Hình 3-6: Sơ đồ tuần tự: Tạo sự kiện



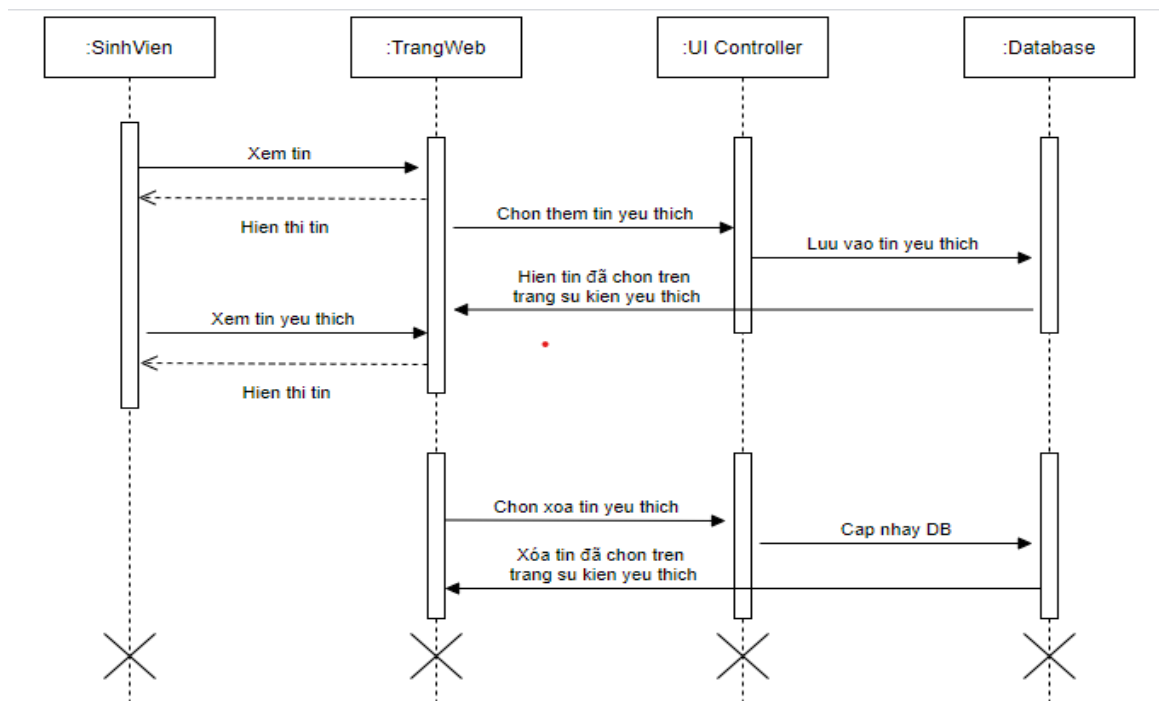
Hình 3-7: Sơ đồ tuần tự: Chỉnh sửa sự kiện



Hình 3-8: Sơ đồ tuần tự: Xóa sự kiện



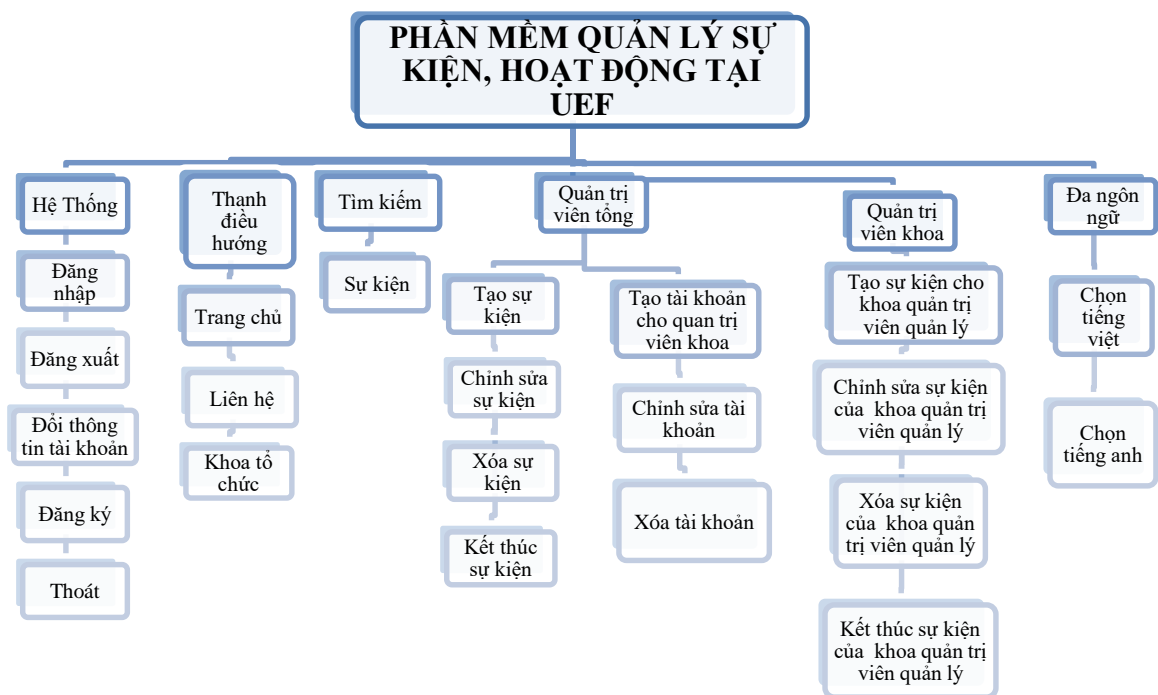
Hình 3-9: Sơ đồ tuần tự: Kết thúc sự kiện



Hình 3-10: Sơ đồ tuần tự: Thêm và xóa sự kiện yêu thích

Chương 4 . THIẾT KẾ GIAO DIỆN VÀ CHỨC NĂNG

4.1 Mô hình chức năng



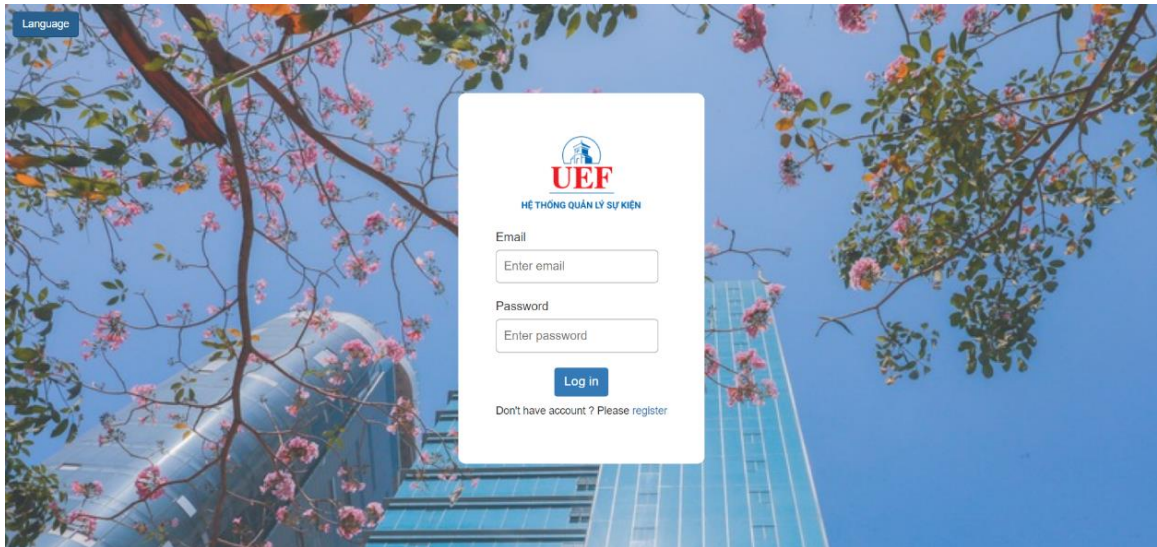
4.2 Giao diện của chương trình

4.2.1 Form hệ thống (cả người dùng và quản lý)

4.2.1.1 Giao diện đăng nhập

- Quản lý đăng nhập bằng tài khoản đã được cung cấp trước đó (Nhập 2 giá trị là email và mật khẩu)
- Trang đăng nhập được tích hợp cả 2 ngôn ngữ (Dưới hình là trang đăng nhập đang ở ngôn ngữ Tiếng Anh)

- Để lựa chọn ngôn ngữ khác (Tiếng Việt), người dùng bấm vào nút “Language” ở góc trái trên cùng màn hình.

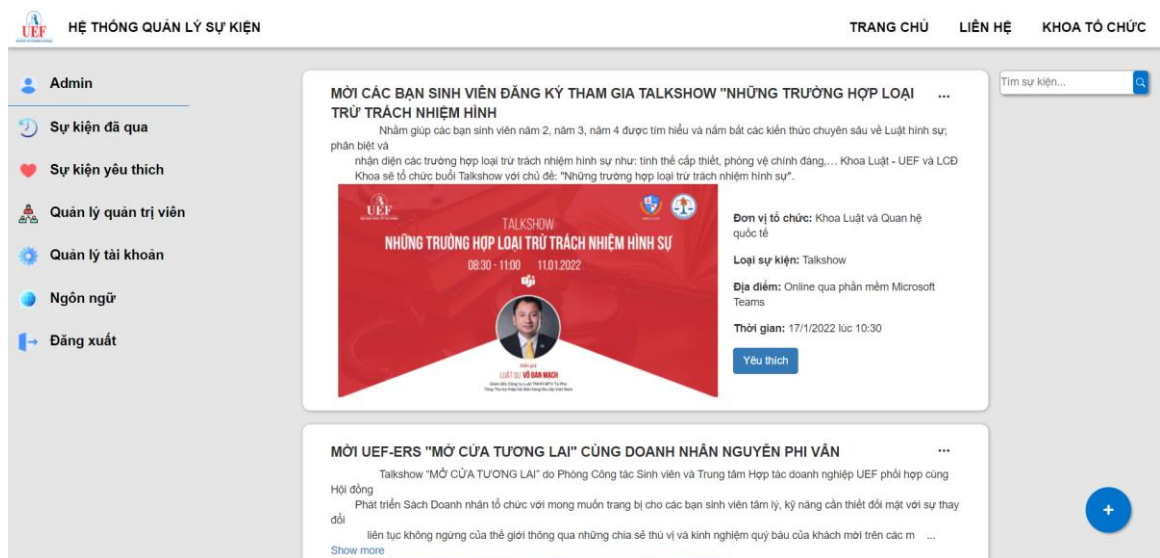


Hình 4-1: Giao diện đăng nhập

4.2.1.2 Giao diện trang chủ

Trang chủ có các chức năng:

- Xem thông tin các sự kiện.
- Xem thông tin sự kiện đã qua.
- Xem thông tin liên hệ.
- Xem các sự kiện sau khi được phân loại theo khoa tổ chức hoặc sau khi được tìm kiếm bằng thanh tìm kiếm ở góc bên phải trên cùng của giao diện.
- Thêm vào sự kiện yêu thích bằng cách nhấn nút “Yêu thích” để lưu các thông tin sự kiện mà sinh viên muốn để xem lại.

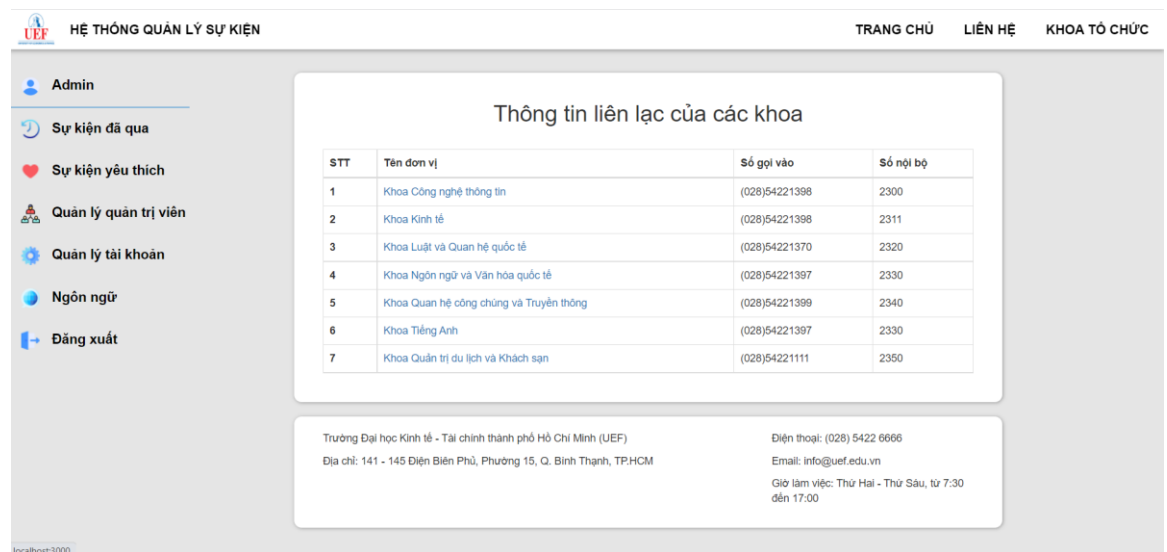


Hình 4-2: Giao diện trang chủ

4.2.1.3 Giao diện liên hệ

Trang liên hệ gồm có:

- Các thông tin (tên, số điện thoại) của các đơn vị khoa.
- Địa chỉ, điện thoại, email và giờ làm việc của trường Đại Học Kinh Tế - Tài Chính.



Hình 4-3: Giao diện liên hệ

4.2.2 Form hệ thống (dùng cho quản lý)

4.2.2.1 Giao diện tạo sự kiện

- Quản trị viên nhập các thông tin: Tên sự kiện, mô tả, địa điểm tổ chức; quản trị viên chọn các thông tin: Hình ảnh minh họa, ngày, giờ tổ chức, đơn vị tổ chức và loại sự kiện. Sau khi điền đầy đủ thông tin, quản trị viên nhấn nút “Đăng sự kiện” để thông tin sự kiện được trình bày trên giao diện chính.

The screenshot shows a web application interface for managing events. A modal window titled "Tạo sự kiện mới" (Create new event) is open, allowing an administrator to input details for a new event. The form includes the following fields:

- Tên sự kiện**: A text input field for the event name.
- Mô tả**: A larger text area for the event description.
- Địa điểm**: A text input field for the location.
- Chọn hình ảnh**: A section with a "Chọn tệp" (Choose file) button and a note "Không có tệp nào được chọn" (No files selected).
- Ngày tổ chức**: A date picker set to "dd/mm/yyyy".
- Thời gian**: A time picker set to "--:--".
- Đơn vị tổ chức**: A dropdown menu currently showing "Chọn đơn vị tổ chức" (Choose organizing unit).
- Loại sự kiện**: A dropdown menu currently showing "Chọn loại sự kiện" (Choose event type).

A blue button labeled "Đăng sự kiện" (Post event) is located at the bottom right of the modal. The background shows the main dashboard with a sidebar menu and a list of events.

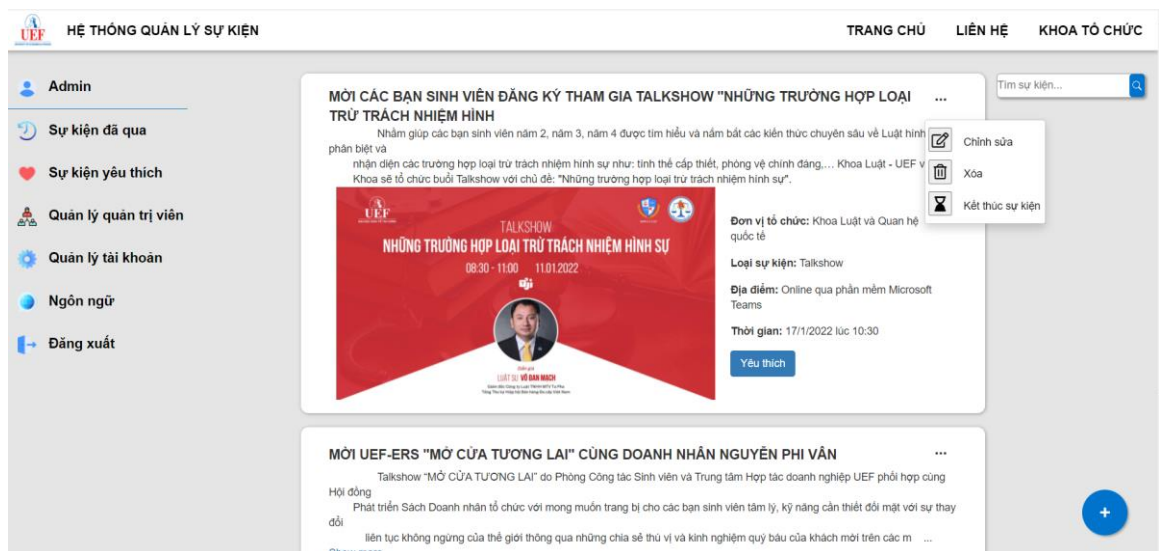
Hình 4-4: Giao diện tạo sự kiện

4.2.2.2 Giao diện chỉnh sửa thông tin sự kiện

Hình 4-5: Giao diện chỉnh sửa thông tin sự kiện

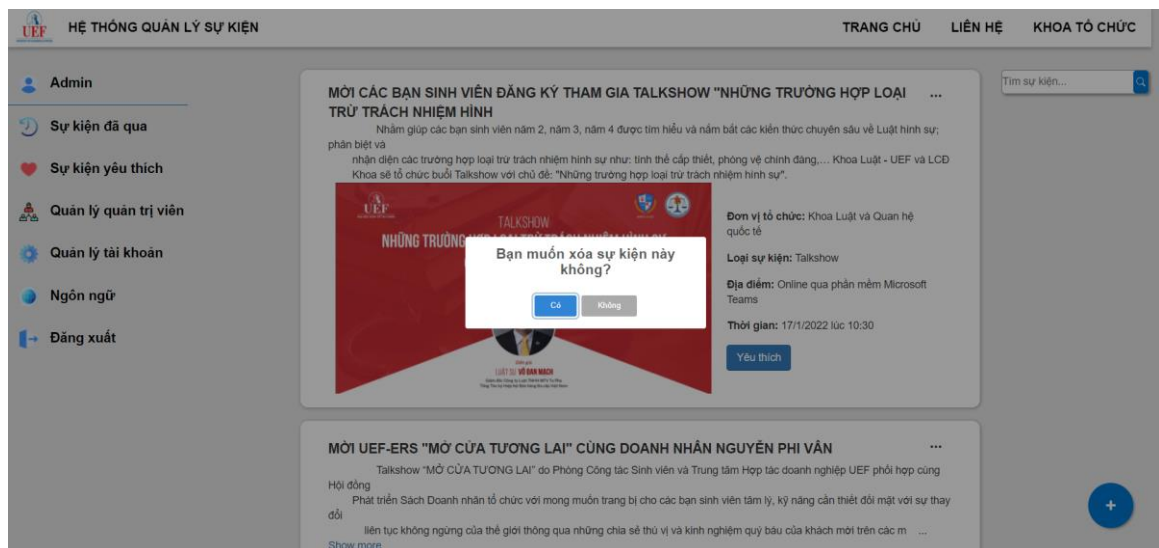
4.2.2.3 Giao diện xóa sự kiện

- Quản trị viên nhấp chuột vào biểu tượng (⋮) để dropdown hiện ra.
Dropdown gồm:
 - Chỉnh sửa
 - Xóa
 - Kết thúc sự kiện
- Quản trị viên chọn chức năng “Xóa” để xóa sự kiện.



Hình 4-6: Giao diện dropdown của sự kiện

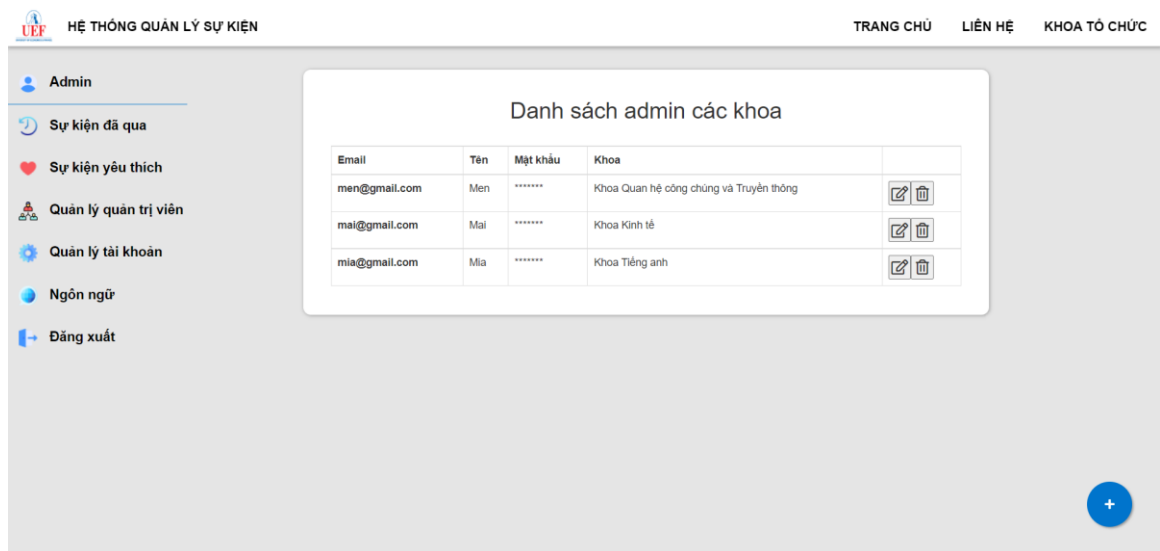
- Sau khi nhấp chuột chọn chức năng “Xóa”. Giao diện hiện ra một “pop up” để xác nhận lại quyết định của quản trị viên.



Hình 4-7: Giao diện xác nhận lại của chức năng xóa sự kiện

4.2.2.4 Giao diện quản lý các quản trị viên khoa

- Quản lý tổng có quyền tạo, chỉnh sửa và xóa thông tin tài khoản của các quản trị viên khoa.



Hình 4-8: Giao diện quản lý các quản trị viên khoa

4.2.3 Form người dùng (dùng cho sinh viên)

4.2.3.1 Giao diện đăng ký tài khoản

- Sinh viên đăng ký với 6 thông tin chính: Tên đầy đủ, email, mã số sinh viên, mật khẩu tài khoản, mã lớp và khoa.
- Sau khi đăng ký tài khoản, sinh viên nhấn vào “log in” (ở giao diện tiếng Anh) hoặc “đăng nhập” (ở giao diện tiếng Việt).
- Chức năng đa ngôn ngữ tương tự như ở trang đăng nhập.

Hình 4-9: Giao diện đăng ký tài khoản

4.2.3.2 Giao diện quản lý tài khoản

- Người dùng (sinh viên) có thể chỉnh sửa các thông tin liên quan đến bản thân thông qua chức năng này. Cụ thể, sinh viên có thể thay đổi mật khẩu hay sửa đổi tên hiển thị và các thông tin liên quan bằng cách nhấp chuột vào nút “Chỉnh sửa”.

Thông tin người dùng			
Tên:	Nguyen Thu Ha	Email:	ha@gmail.com
MSSV:	185050757	Mật khẩu:	*****
Lớp:	18D1TH-PM011	Khoa:	Khoa Công nghệ thông tin

Hình 4-10: Giao diện quản lý tài khoản

4.2.3.3 Giao diện chỉnh sửa thông tin tài khoản

- Sinh viên có thể thay đổi các thông tin một cách nhanh chóng.

The screenshot displays a web application interface for a university system. A modal window titled "Cập nhật thông tin" (Update Information) is centered on the screen. The modal contains the following fields:

- Tên (Name): Nguyen Thu Ha
- Email: ha@gmail.com
- Mật khẩu (Password): [masked with asterisks]
- MSSV: 185050757
- Lớp (Class): 18D1TH-PM011
- Khoa (Faculty): Khoa Công nghệ thông tin

A blue button labeled "Cập nhật" (Update) is located at the bottom right of the modal. The background interface includes a sidebar on the left with a user profile section (Nguyen Thu) and navigation links (Sự kiện đã qua, Sự kiện yêu thích, Quản lý tài khoản, Ngôn ngữ, Đăng xuất). The top header shows the system name "HỆ THỐNG QUẢN LÝ SỰ KIỆN" and user roles "TRANG CHỦ", "LIÊN HỆ", and "KHOA TỜ CHỨC".

Hình 4-11: Giao diện chỉnh sửa thông tin tài khoản sinh viên

Chương 5 . KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Kết luận

5.1.1 *Kết quả đạt được*

- Nhóm xây dựng được nền tảng cơ bản về quản lý sự kiện, hoạt động tại UEF.
- Xây dựng được mô hình với giao diện hướng sáng tạo (khá quen thuộc với các bạn sinh viên). Giúp các bạn sinh viên nhận được thông tin các sự kiện của trường một cách nhanh chóng và đa dạng về chuyên ngành hơn.
- Giao diện chính của website là các thông tin sự kiện được trình bày theo dạng bản tin ngắn gọn, đầy đủ thông tin. Các thông tin cần thiết như địa điểm tổ chức, ngày giờ tổ chức,... được trình bày dễ thấy hơn.
- Với chức năng đa ngôn ngữ, sản phẩm nhóm rất phù hợp với trường Đại Học Kinh Tế - Tài Chính là ngôi trường quốc tế và đa văn hóa.
- Bên cạnh đó, sản phẩm của nhóm có chức năng phân loại các sự kiện theo Khoa tổ chức, giúp các bạn sinh viên có thể tìm kiếm thông tin sự kiện mà khoa của các bạn tổ chức để có thể nắm bắt thông tin tốt hơn.
- Website có chức năng phân quyền: Quản trị viên tổng có thể thực hiện tất cả các chức năng, quản trị khoa có quyền tạo sự kiện cho chính khoa họ, sinh viên có quyền chỉnh sửa thông tin tài khoản để tiện lợi cho việc đăng ký.

5.1.2 *Những hạn chế*

- Chức năng đăng nhập chưa có đăng nhập qua tài khoản google, gmail trường của sinh viên.

- Website chưa thực hiện được chức năng đổi giao diện (thông thường có giao diện tối và giao diện sáng).
- Các thông tin của người dùng chưa có hình ảnh đại diện do người dùng tải lên.
- Chưa có chức năng cho quản trị viên xét duyệt các thay đổi thông tin do sinh viên chỉnh sửa thông tin tài khoản

5.2 Hướng phát triển

Trong tương lai, website có thể được hoàn thiện theo các hướng sau:

- Chỉnh sửa lại tư duy đăng nhập (đăng nhập thông qua tài khoản thứ ba như tài khoản google) để người dùng có thể dễ dàng sử dụng hơn.
- Thêm các ngôn ngữ vào chức năng đa ngôn ngữ (ví dụ thêm ngôn ngữ Hàn, Lào, Nhật Bản, Nga, ...).
- Cập nhật thêm giao diện tối để giúp người dùng có trải nghiệm tốt hơn.
- Sử dụng thêm các kiến thức vào chức năng bảo mật để bảo vệ thông tin người dùng (tên, mail, mật khẩu) một cách tốt nhất.
- Tạo thêm chức năng xét duyệt cho quản trị viên có thể duyệt các thông tin thay đổi dữ liệu do người dùng chỉnh sửa.

TÀI LIỆU THAM KHẢO

- [1] Phan Van Hung (19/2/2021), “Tìm hiểu về Virtual DOM trong React”, VIBLO, truy cập tại <https://viblo.asia/p/tim-hieu-ve-virtual-dom-trong-react-bJzKm4kO59N> vào ngày 9/1/2022.
- [2] “ReactJS là gì? Các khái niệm cần biết trước khi học React JS”, FREETUTS, truy cập tại <https://freetuts.net/reactjs-la-gi-cac-khai-niem-co-ban-2343.html> vào ngày 9/1/2022.
- [3] Trung Quân (23/4/2018), “Một cái nhìn tổng quan nhất về Nodejs”, VIBLO, truy cập tại <https://viblo.asia/p/mot-cai-nhin-tong-quan-nhat-ve-nodejs-Ljy5VeJ3lra> vào ngày 9/1/2022.
- [4] Rikkikudo (21/1/2020), “Cùng tìm hiểu về Npm và package”, CODELEARN, truy cập tại <https://codelearn.io/sharing/cung-tim-hieu-ve-npm-va-package> vào ngày 9/1/2022.
- [5] “HỆ QUẢN TRỊ CSDL MYSQL – MySQL cơ bản”, HIỆP SĨ IT, truy cập tại <http://hiepsiit.net/detail/mysql/hequantricsdlmysql/mysqlcoban> vào ngày 9/1/2022.
- [6] “Draw.io – Trang web Vẽ sơ đồ online chuyên nghiệp nhất, miễn phí, không giới hạn”, truy cập tại <http://thuthuatphanmem.vn/draw-io-trang-web-ve-so-do-online-chuyen-nghiep-nhat-mien-phi-khong-gioi-han/> vào ngày 9/1/2022.
- [7] “Figma”, truy cập tại <https://licensesoft.vn/figma.htm> vào ngày 9/1/2022.