|  |
| --- |
| HỌC VIỆN KỸ THUẬT MẬT MÃ  **KHOA CÔNG NGHỆ THÔNG TIN**  Description: Logo HvKTMM  **Khoa Công nghệ thông tin**  **ĐỒ ÁN**  **THỰC TẬP CƠ SỞ**  **Đề tài:**  **XÂY DỰNG TRANG WEB QUẢN LÝ TRUNG TÂM DẠY HỌC**  Giảng viên hướng dẫn: ThS. LÊ ĐỨC THUẬN  Sinh viên thực hiện: HOÀNG XUÂN LONG CT030333  NGUYỄN VĂN HUYÊN CT030326  NGUYỄN MINH HIỂN CT030319    Hà Nội, 12-2021 |

**MỤC LỤC**

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 1](#_Toc91201125)

[1.1 Đặt vấn đề 1](#_Toc91201126)

[1.2 Mục đích đề tài 1](#_Toc91201127)

[1.3 Ngôn ngữ và công cụ hỗ trợ 2](#_Toc91201128)

[1.3.1 Ngôn ngữ lập trình JavaScript và thư viện ReactJS 2](#_Toc91201129)

[1.3.2 MongoDB 5](#_Toc91201130)

[1.3.3 GraphQL 7](#_Toc91201131)

[1.3.4 Visual Studio Code 10](#_Toc91201132)

[1.3.5 Figma 14](#_Toc91201133)

[CHƯƠNG 2. THIẾT KẾ VÀ XÂY DỰNG HỆ THỐNG 17](#_Toc91201134)

[2.1 Khảo sát hệ thống 17](#_Toc91201135)

[2.2 Phân tích hệ thống 17](#_Toc91201136)

[2.2.1 Phân tích hệ thống về mặt chức năng 17](#_Toc91201137)

[2.2.1.1 Biểu đồ phân rã chức năng 18](#_Toc91201138)

[2.2.2 Sơ đồ Usecase 19](#_Toc91201139)

[2.3 Cơ sở dữ liệu 20](#_Toc91201140)

[2.3.1 Xác định thực thể 20](#_Toc91201141)

[2.3.2 Mô hình ER 20](#_Toc91201142)

[2.3.3 Các bảng dữ liệu 21](#_Toc91201143)

[CHƯƠNG 3. THỰC NGHIỆM 23](#_Toc91201144)

[3.1 Tổng quan Website 23](#_Toc91201145)

[3.1.1 Giao diện trang chủ 23](#_Toc91201146)

**DANH MỤC HÌNH ẢNH**

[**Hình 1: Logo JavaScript** 2](#_Toc91201147)

[**Hình 2: Logo ReactJS** 3](#_Toc91201148)

[**Hình 3: Giao diện làm việc MongoDB** 5](#_Toc91201149)

[**Hình 4: Logo MongoDB** 6](#_Toc91201150)

[**Hình 5: GraphQL** 7](#_Toc91201151)

[**Hình 6: Cú pháp GraphQL** 8](#_Toc91201152)

[**Hình 7: Logo Visual Studio Code** 10](#_Toc91201153)

[**Hình 8: Một số tính năng Visual Studio Code** 11](#_Toc91201154)

[**Hình 9: Giao diện Visual Studio Code** 13](#_Toc91201155)

[**Hình 10: Logo Figma** 14](#_Toc91201156)

[**Hình 11: Giao diện làm việc Figma** 16](#_Toc91201157)

[**Hình 12: Sơ đồ quy trình nghiệp vụ** 17](#_Toc91201158)

[**Hình 13: Biểu đồ phân rã chức năng** 18](#_Toc91201159)

[**Hình 14: Sơ đồ Usecase** 19](#_Toc91201160)

[**Hình 15: Mô hình ER** 20](#_Toc91201161)

[**Hình 16: Giao diện trang chủ website quản lí trung tâm** 23](#_Toc91201162)

**DANH MỤC BẢNG**

[Bảng 1: Bảng dữ liệu User 21](#_Toc91200000)

[Bảng 2: Bảng dữ liệu Course 21](#_Toc91200001)

[Bảng 3: Bảng dữ liệu Diemdanh 22](#_Toc91200002)

[Bảng 4: Bảng dữ liệu Luong 22](#_Toc91200003)

**LỜI NÓI ĐẦU**

Ngày nay sự phát triển mạnh mẽ của tin học làm cho máy tính trở thành phương tiện không thể thiếu được trong mọi lĩnh vực đời sống. Hơn lúc nào hết các tính năng của máy tính được khai thác một cách triệt để. Nếu như vài ba năm trước máy tính ở nước ta máy tính được sử dụng chủ yếu như là một công cụ đẻ soạn thảo văn bản thông thường, hoặc các công việc lập trình, quản lý phức tạp, xử lý dữ liệu biểu bảng, thương mại, khoa học v.v… thì giờ đây, cùng với sự vươn xa của mạng Internet trên lãnh thổ Việt Nam máy tính còn là phương tiện bạn có thể ngồi trên bàn làm việc cá nhân tại gia đình mà trao đổi thông tin liên lạc đi khắp toàn cầu.

Nền tin học càng phát triển thì con người càng có nhiều những phương pháp mới, công cụ mới để xử lý thông tin và nắm bắt được nhiều thông tin hơn. tin học được ứng dụng trong mọi nghành nghề, mọi lĩnh vực của đời sống xã hội, Việc áp dụng Tin học vào trong quản lý, sản xuất, kinh doanh du lịch là một xu hướng tất yếu.

Kết quả của việc áp dụng tin học trong quản lý là việc hình thành các hệ thống thông tin quản lý nhằm phục vụ cho nhu cầu xử lý dữ liệu và cung cấp thông tin cho các chủ sở hữu hệ thống đó. Ở nước ta hiện nay trong những năm gần đây các ứng dụng tin học vào quản lý đang ngày một nhiều hơn, đa dạng hơn. Vì vậy nhóm chúng em thực hiện đề tài này nhằm ứng dụng công nghệ thông tin trong việc quản lí giáo dục.

Nhóm em xin gửi lời cảm ơn đến thầy **Lê Đức Thuận** đã tận tìnhhướng dẫn nhóm chúng em trong quá trình nghiên cứu và hoàn thành đồ án thực tập cơ sở này.

# TỔNG QUAN VỀ ĐỀ TÀI

## Đặt vấn đề

*(Cần bổ sung)*

## Mục đích đề tài

Tạo ra một website hỗ trợ trung tâm dạy học trong việc quản lý giảng viên, học sinh, các khóa học, giảm tải công việc cho các giảng viên.

Website sau khi hoàn thành sẽ đáp ứng những yêu cầu cơ bản :

* **Khách hàng :**

Khách hàng là những người có nhu cầu đăng kí học tại trung tâm, vậy website phải giúp khách hàng:

* Đăng kí các khóa học của trung tâm ngay tại nhà ,không cần thiết phải gặp trực tiến người quản lí trung tâm để đăng kí
* Xem lại các video của khóa học
* Có thể đóng học phí thông qua website
* **Trung tâm :**

Là người có quyền đăng nhập, quản lý và làm chủ mọi hoạt động của hệ thống website. Người quản lý có một username và password để truy cập vào hệ thống nhằm thực hiện các chức năng sau:

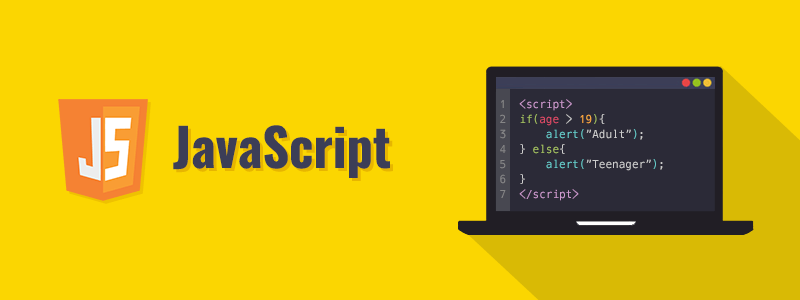
* Quản lí các khóa học một cách dễ dàng
* Thêm, sửa, xóa các khóa học vào cơ sở dữ liệu
* Điểm danh học viên các lớp học
* Thông báo đến học viên
* Quản lí giảng viên đang giảng dạy tại trung tâm ( Thông tin, tiết dạy, lương,…)

Bên cạnh những chức năng nêu trên, website phải có giao diện đẹp mắt, dễ truy cập và sử dụng cho cả khách hàng và bên quản lí trung tâm. Ngoài ra, website phải tuyệt đối bảo mật về thông tin khách hàng đăng kí học tại trung tâm. Website cũng cần có tính dễ nâng cấp, dễ bảo trì, sửa chữa khi cần bổ sung những tính năng mới.

## Ngôn ngữ và công cụ hỗ trợ

### Ngôn ngữ lập trình JavaScript và thư viện ReactJS

* JavaScript: Ngôn ngữ này được dùng rộng rãi cho các trang web (phía người dùng) cũng như phía máy chủ (với Nodejs).
* JavaScript được sử dụng rộng rãi trong việc kết hợp với HTML/CSS để thiết kế web động.



**Hình 1: Logo JavaScript**

**Lịch sử hình thành**

Sun Microsystems đã phát minh thành một ngôn ngữ phức tạp và mạnh mẽ mà chúng ta đã biết đó là ngôn ngữ Java. Mặc dù Java có tính khả dụng cao nhưng nó lại phù hợp nhất đối với các nhà lập trình có kinh nghiệm và cho các công việc phức tạp hơn.

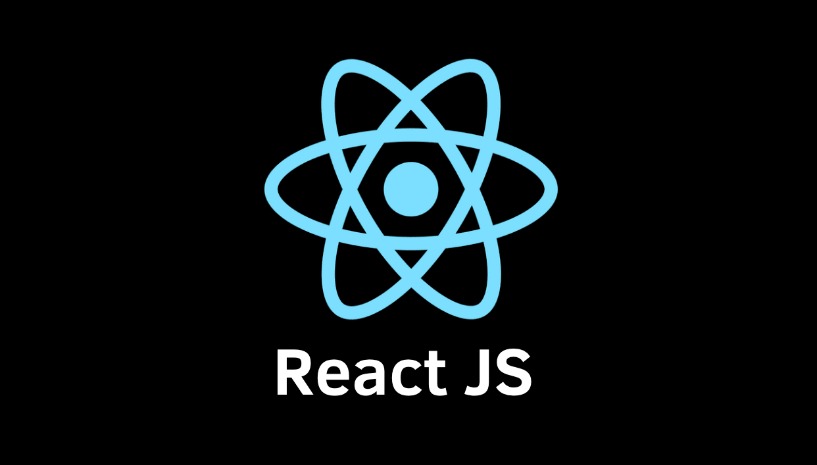
Netscape Communications đã thấy được nhu cầu đối với một ngôn ngữ ‘nửa nọ nửa kia’ – một ngôn ngữ sẽ cho phép các cá nhân thiết kế các trang Web có khả năng tương tác với người dùng hoặc với Java applets nhưng sẽ đòi hỏi phải có kinh nghiệm lập trình tối thiểu.

LiveScript chỉ là một ngôn ngữ phát thảo hơn là thực tế, nhưng nó lại khiến cho người ta tin rằng nó ‘bắt cầu’ cho ý tưởng tốt hơn. LiveScript được thiết kế theo tinh thần của nhiều ngôn ngữ script đơn giản nhưng nó lại có tính khả dụng cao được thiết kế đặc biệt để xây dựng các trang Web (chẳng hạn như HTML và các form tương tác).

Để giúp ‘bán chạy’ ngôn ngữ mới này, Netscape hợp tác với Sun cho ra đời ngôn ngữ JavaScript. Trên thực tế, Microsoft là người tiên phong thực thi phiên bản của JavaScript (còn có tên là Jscript), nhưng họ vẫn chưa tham chiếu đến những đặc tả chính thức của JavaScript.

Mục tiêu của JavaScript là nhằm cung cấp cho các nhà phát triển trang Web một số khả năng và sự điều khiển trên chức năng của một trang Web. Mã JavaScript có khả năng nhúng trong tài liệu HTML để điều khiển nội dung của trang Web và xác nhận tính hợp lý của dữ liệu mà người dùng nhập vào. Khi một trang hiển thị trong trình duyệt, các câu lệnh được trình duyệt thông dịch và thực thi.

**Thư viện ReactJS**



**Hình 2: Logo ReactJS**

ReactJS là một thư viện JavaScript mã nguồn mở được thiết kế bởi Facebook để tạo ra những ứng dụng web hấp dẫn, nhanh và hiệu quả với mã hóa tối thiểu. Mục đích cốt lõi của ReactJS không chỉ khiến cho trang web phải thật mượt mà còn phải nhanh, khả năng mở rộng cao và đơn giản.

Sức mạnh của nó xuất phát từ việc tập trung vào các thành phần riêng lẻ. Chính vì vậy, thay vì làm việc trên toàn bộ ứng dụng web, ReactJS cho phép một developer có thể phá vỡ giao diện người dùng phức tạp thành các thành phần đơn giản hơn.

Ngày nay, [ReactJS](https://topdev.vn/viec-lam-it/reactjs-kt214) đã trở nên rất phổ biến bởi những tính năng linh hoạt và đơn giản với hơn 1,300 developer và hơn 94,000 trang web đang sử dụng ReactJS. Nhiều người ám chỉ rằng ReactJS sẽ là tương lai của việc phát triển web.

**Đặc trưng của ReactJS:**

* **JSX**: Trong React, thay vì thường xuyên sử dụng JavaScript để thiết kế bố cục trang web thì sẽ dùng JSX. JSX được đánh giá là sử dụng đơn giản hơn JavaScript và cho phép trích dẫn HTML cũng như việc sử dụng các cú pháp thẻ HTML để render các subcomponent. JSX tối ưu hóa code khi biên soạn, vì vậy nó chạy nhanh hơn so với code JavaScript tương đương.
* **Redux**: **Redux** là một predictable state management tool cho các ứng dụng Javascript. Nó giúp bạn viết các ứng dụng hoạt động một cách nhất quán, chạy trong các môi trường khác nhau (client, server, and native) và dễ dàng để test. **Redux** ra đời lấy cảm hứng từ tư tưởng của ngôn ngữ **Elm** và kiến trúc **Flux**của Facebook. Do vậy Redux thường dùng kết hợp với React.
* **Single-way data flow (Luồng dữ liệu một chiều):** ReactJS không có những module chuyên dụng để xử lý data, vì vậy ReactJS chia nhỏ view thành các component nhỏ có mỗi quan hệ chặt chẽ với nhau. Tại sao chúng ta phải quan tâm tới cấu trúc và mối quan hệ giữa các component trong ReactJS? Câu trả lời chính là luồng truyền dữ liệu trong ReactJS: Luồng dữ liệu một chiều từ cha xuống con. Việc ReactJS sử dụng one-way data flow có thể gây ra một chút khó khăn cho những người muốn tìm hiểu và ứng dụng vào trong các dự án. Tuy nhiên, cơ chế này sẽ phát huy được ưu điểm của mình khi cấu trúc cũng như chức năng của view trở nên phức tạp thì ReactJS sẽ phát huy được vai trò của mình.
* **Virtual DOM**: Những Framework sử dụng Virtual-DOM như ReactJS khi Virtual-DOM thay đổi, chúng ta không cần thao tác trực tiếp với DOM trên View mà vẫn phản ánh được sự thay đổi đó. Do Virtual-DOM vừa đóng vai trò là Model, vừa đóng vai trò là View nên mọi sự thay đổi trên Model đã kéo theo sự thay đổi trên View và ngược lại. Có nghĩa là mặc dù chúng ta không tác động trực tiếp vào các phần tử DOM ở View nhưng vẫn thực hiện được cơ chế Data-binding. Điều này làm cho tốc độ ứng dụng tăng lên đáng kể – môt lợi thế không thể tuyệt vời hơn khi sử dụng Virtula-DOM.

### NodeJS và Express

#### Node JS

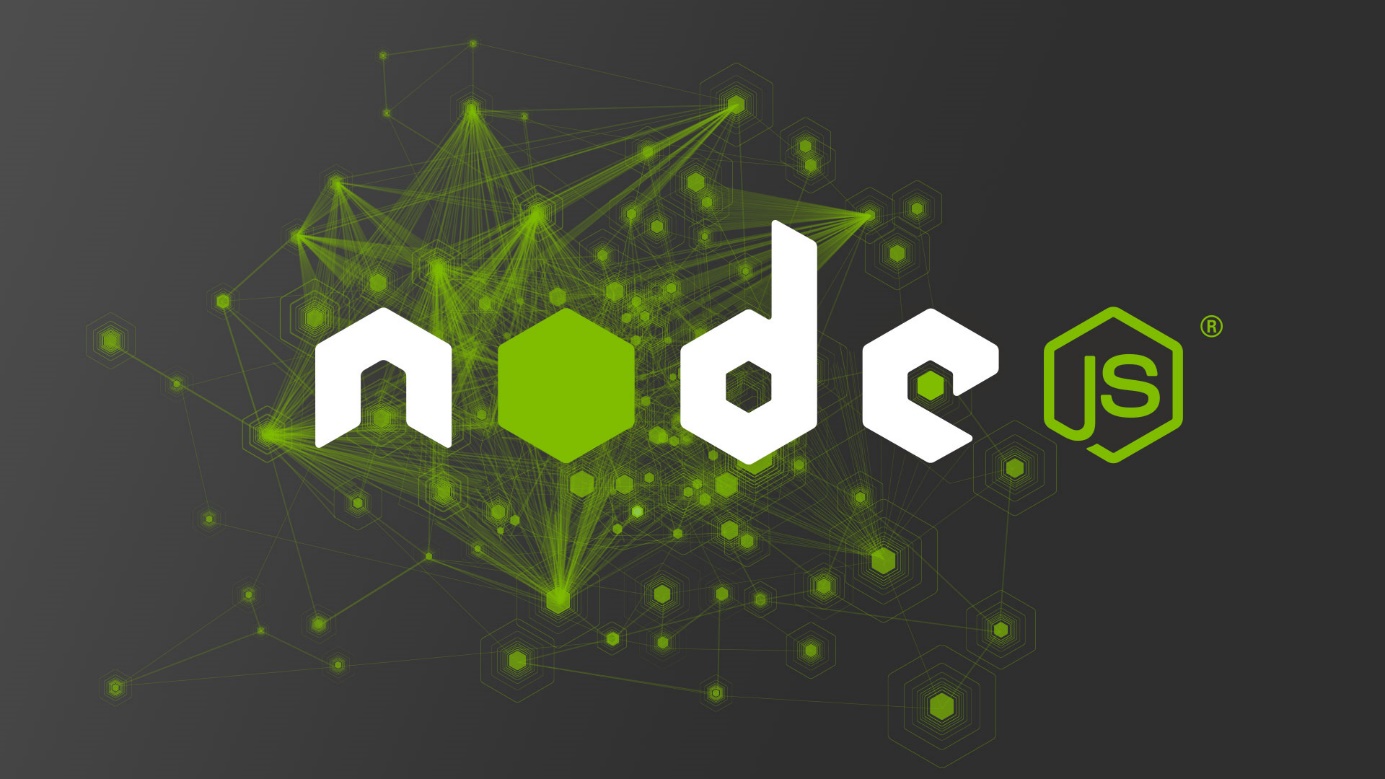
Nodejs là một **nền tảng** (Platform) phát triển độc lập được xây dựng ở trên Javascript Runtime của Chrome mà chúng ta có thể xây dựng được các ứng dụng mạng một cách nhanh chóng và dễ dàng mở rộng.

Nodejs được xây dựng và phát triển từ năm 2009, bảo trợ bởi công ty Joyent, trụ sở tại California, Hoa Kỳ. Dù sao thì chúng ta cũng nên biết qua một chút chút lịch sử của thứ mà chúng ta đang học một chút chứ nhỉ? =))

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.

Nodejs 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.



**Hình 3: Logo Node JS**

Các ứng dụng Nodejs được viết bằng **javascript**, ngôn ngữ này là một ngôn ngữ khá thông dụng. Theo tác giả của ngôn ngữ Javascript, Ryan Dahl: “Javascript có những đặc tính mà làm cho nó rất khác biệt so với các ngôn ngữ lập trình động còn lại, cụ thể là nó không có khái niệm về đa luồng, tất cả là đơn luồng và hướng sự kiện.”

Nodejs chạy đa nền tảng phía Server, sử dụng kiến trúc hướng sự kiện Event-driven, cơ chế non-blocking I/O làm cho nó nhẹ và hiệu quả.

Có thể chạy ứng dụng Nodejs ở bất kỳ đâu trên máy Mac – Window – Linux, hơn nữa cộng đồng Nodejs rất lớn và hoàn toàn miễn phí. Các bạn có thể thấy cộng đồng Nodejs lớn như thế nào tại đây, các package đều hoàn toàn free: <https://www.npmjs.com/>

Các ứng dụng NodeJS đáp ứng tốt thời gian thực và chạy đa nền tảng, đa thiết bị.

#### Express

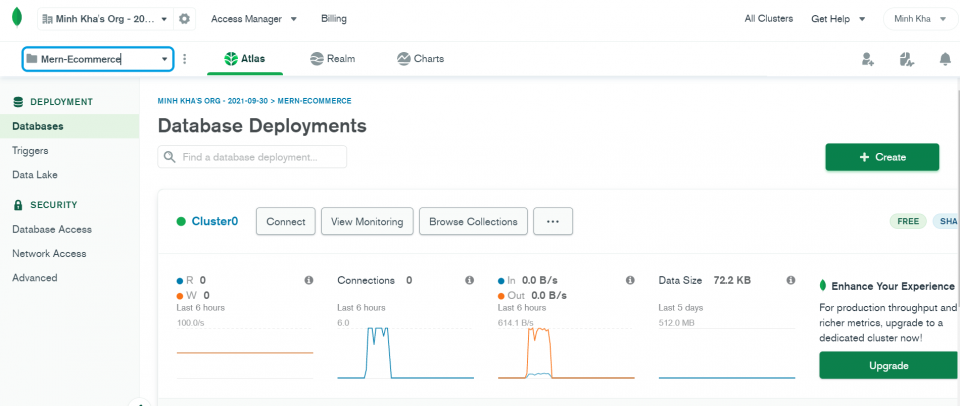
Express là một framework giành cho nodejs. Nó cung cấp cho chúng ta rất nhiều tính năng mạnh mẽ trên nền tảng web cũng như trên các ứng dụng di động. Express hỗ rợ các phương thức HTTP và midleware tạo ra môt API vô cùng mạnh mẽ và dễ sử dụng. Có thể tổng hợp một số chức năng chính của express như sau:

* Phát triển máy chủ nhanh chóng: Expressjs cung cấp nhiều tính năng dưới dạng các hàm để dễ dàng sử dụng ở bất kỳ đâu trong chương trình. Điều này đã loại bỏ nhu cầu viết mã từ đó tiết kiệm được thời gian.
* Phần mềm trung gian Middleware: Đây là phần mềm trung gian có quyền truy cập vào cơ sở dữ liệu, yêu cầu của khách hàng và những phần mềm trung gian khác. Phần mềm Middleware này chịu trách nhiệm chính cho việc tổ chức có hệ thống các chức năng của Express.js.
* Định tuyến - Routing: Express js cung cấp cơ chế định tuyến giúp duy trì trạng thái của website với sự trợ giúp của URL.
* Tạo mẫu - Templating: Các công cụ tạo khuôn mẫu được Express.js cung cấp cho phép các nhà xây dựng nội dung động trên các website bằng cách tạo dựng các mẫu HTML ở phía máy chủ.
* Gỡ lỗi - Debugging: Để phát triển thành công các ứng dụng web không thể thiết đi việc gỡ lỗi. Giờ đây với Expressjs việc gỡ lỗi đã trở nên dễ dàng hơn nhờ khả năng xác định chính xác các phần ứng dụng web có lỗi.



**Hình 4: Logo Express**

### MongoDB



**Hình 5: Giao diện làm việc MongoDB**

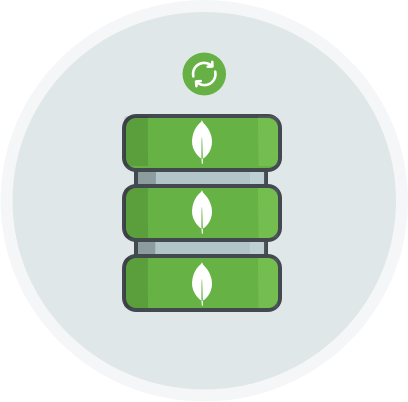
**MongoDB**là một database hướng tài liệu (document), một dạng NoSQL database. Vì thế, MongoDB sẽ tránh cấu trúc table-based của relational database để thích ứng với các tài liệu như JSON có một schema rất linh hoạt gọi là BSON. [**MongoDB**](https://topdev.vn/viec-lam-it/mongodb-kt90) sử dụng lưu trữ dữ liệu dưới dạng Document JSON nên mỗi một collection sẽ các các kích cỡ và các document khác nhau. Các dữ liệu được lưu trữ trong document kiểu JSON nên truy vấn sẽ rất nhanh.

MongoDB lần đầu ra đời bởi MongoDB Inc., tại thời điểm đó là thế hệ 10, vào tháng Mười năm 2007, nó là một phần của sản phẩm PaaS (Platform as a Service) tương tự như Windows Azure và Google App Engine. Sau đó nó đã được chuyển thành nguồn mở từ năm 2009.

MongoDB đã trở thành một trong những NoSQL database nổi trội nhất bấy giờ, được dùng làm backend cho rất nhiều website như eBay, SourceForge và The New York Times.

Các feature của MongoDB gồm có:

* **Các ad hoc query:** hỗ trợ search bằng field, các phép search thông thường, regular expression searches, và range queries.
* **Indexing:** bất kì field nào trong BSON document cũng có thể được index.
* **Replication:** có ý nghĩa là “nhân bản”, là có một phiên bản giống hệt phiên bản đang tồn tại, đang sử dụng. Với cơ sở dữ liệu, nhu cầu lưu trữ lớn, đòi hỏi cơ sở dữ liệu toàn vẹn, không bị mất mát trước những sự cố ngoài dự đoán là rất cao. Vì vậy, người ta nghĩ ra khái niệm “nhân bản”, tạo một phiên bản cơ sở dữ liệu giống hệt cơ sở dữ liệu đang tồn tại, và lưu trữ ở một nơi khác, đề phòng có sự cố.
* **Aggregation:** Các Aggregation operation xử lý các bản ghi dữ liệu và trả về kết quả đã được tính toán. Các phép toán tập hợp nhóm các giá trị từ nhiều Document lại với nhau, và có thể thực hiện nhiều phép toán đa dạng trên dữ liệu đã được nhóm đó để trả về một kết quả duy nhất. Trong SQL, count(\*) và GROUP BY là tương đương với Aggregation trong MongoDB.
* **Lưu trữ file:** MongoDB được dùng như một hệ thống file tận dụng những function trên và hoạt động như một cách phân phối qua sharding.



**Hình 6: Logo MongoDB**

**Ưu điểm của Mo**ngoDB:

* Dữ liệu lưu trữ phi cấu trúc, không có tính ràng buộc, toàn vẹn nên tính sẵn sàng cao, hiệu suất lớn và dễ dàng mở rộng lưu trữ.
* Dữ liệu được caching (ghi đệm) lên RAM, hạn chế truy cập vào ổ cứng nên tốc độ đọc và ghi cao.

**Nhược điểm của MongoDB:**

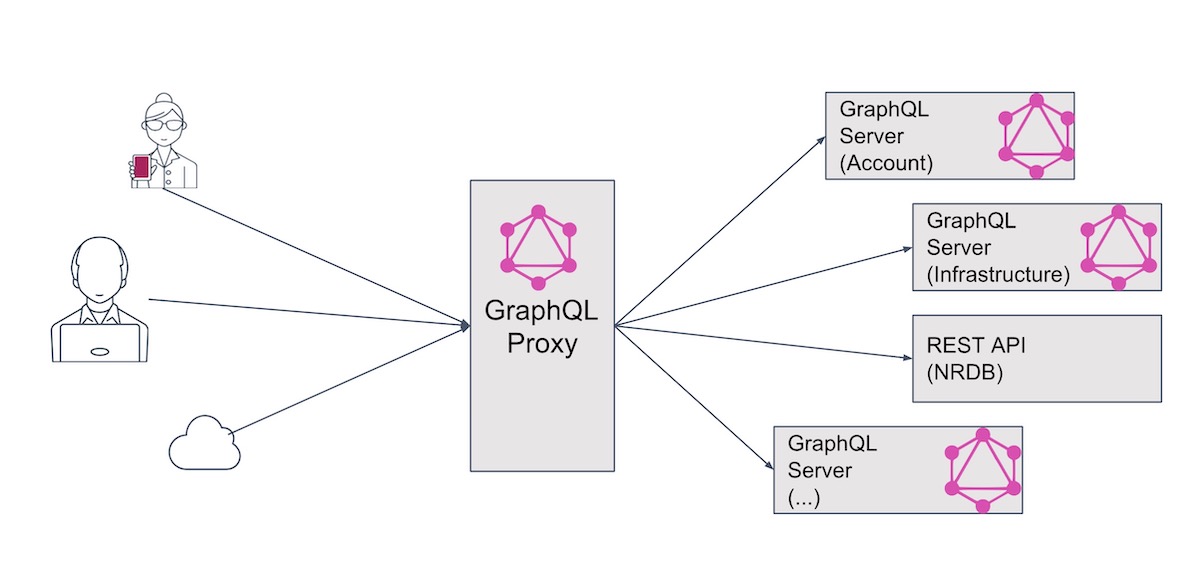
* Không ứng dụng được cho các mô hình giao dịch nào có yêu cầu độ chính xác cao do không có ràng buộc.
* Không có cơ chế transaction (giao dịch) để phục vụ các ứng dụng ngân hàng.
* Dữ liệu lấy RAM làm trọng tâm hoạt động vì vậy khi hoạt động yêu cầu một bộ nhớ RAM lớn.
* Mọi thay đổi về dữ liệu mặc định đều chưa được ghi xuống ổ cứng ngay lập tức vì vậy khả năng bị mất dữ liệu từ nguyên nhân mất điện đột xuất là rất cao.

### GraphQL

GraphQL là ngôn ngữ thao tác và truy vấn dữ liệu nguồn mở cho API, cung cấp cho client 1 cách thức dễ dàng để request chính xác những gì họ cần, giúp việc phát triển API dễ dàng hơn theo thời gian. GraphQL được Facebook phát triển nội bộ vào năm 2012 trước khi phát hành công khai vào năm 2015.

GraphQL bao gồm 3 điểm đặc trưng bao gồm:

* Cho phép client xác định chính xác những dữ liệu gì họ cần
* GraphQL làm cho việc tổng hợp dữ liệu từ nhiều nguồn dễ dàng hơn
* Sử dụng một type system để khai báo dữ liệu.



**Hình 7: GraphQL**

**Trả về chính xác những gì bạn gửi request**

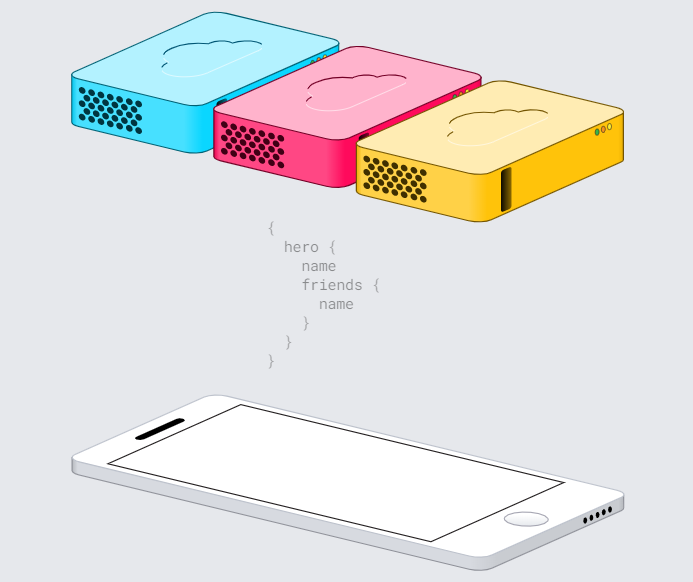
* Khi bạn gửi 1 request GraphQL đến API của bạn, bạn sẽ nhận được chính xác những gì bạn yêu cầu trong request, không hơn không kém.
* Các truy vấn GraphQL luôn trả về kết quả có thể dự đoán được.
* Các ứng dụng sử dụng GraphQL rất nhanh và ổn định vì GraphQL kiểm soát dữ liệu mà nó nhận được chứ không phải máy chủ.



**Hình 8: Cú pháp GraphQL**

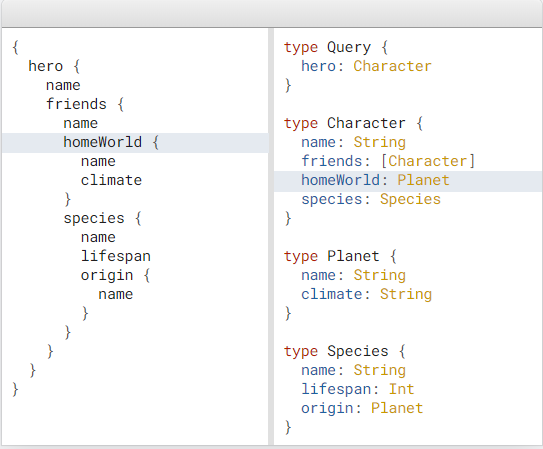
**Nhận nhiều dữ liệu trong một request duy nhất**

* Các câu query GraphQL không chỉ có thể truy xuất các thuộc tính của một dữ liệu mà còn làm việc trơn tru với các đối tượng khác.
* Trong khi các API REST chúng ta hay dùng phải yêu cầu request từ nhiều URL thì API GraphQL lại có thể lấy tất cả dữ liệu mà ứng dụng của bạn cần trong một request duy nhất.
* Các ứng dụng sử dụng GraphQL có tốc độ xử lý rất nhanh ngay cả trên các kết nối chậm



**Mô tả những gì có thể với một type system**

* API GraphQL được tổ chức theo types và fields, truy cập toàn bộ dữ liệu của bạn từ một endpoint duy nhất.
* GraphQL sử dụng các types để đảm bảo ứng dụng của bạn chỉ request những gì có thể, ngoài ra GraphQL còn giải thích các lỗi 1 cách rất dễ hiểu.
* Ứng dụng của bạn có thể sử dụng các *types* để tránh viết code phân tích cú pháp thủ công.



### Visual Studio Code

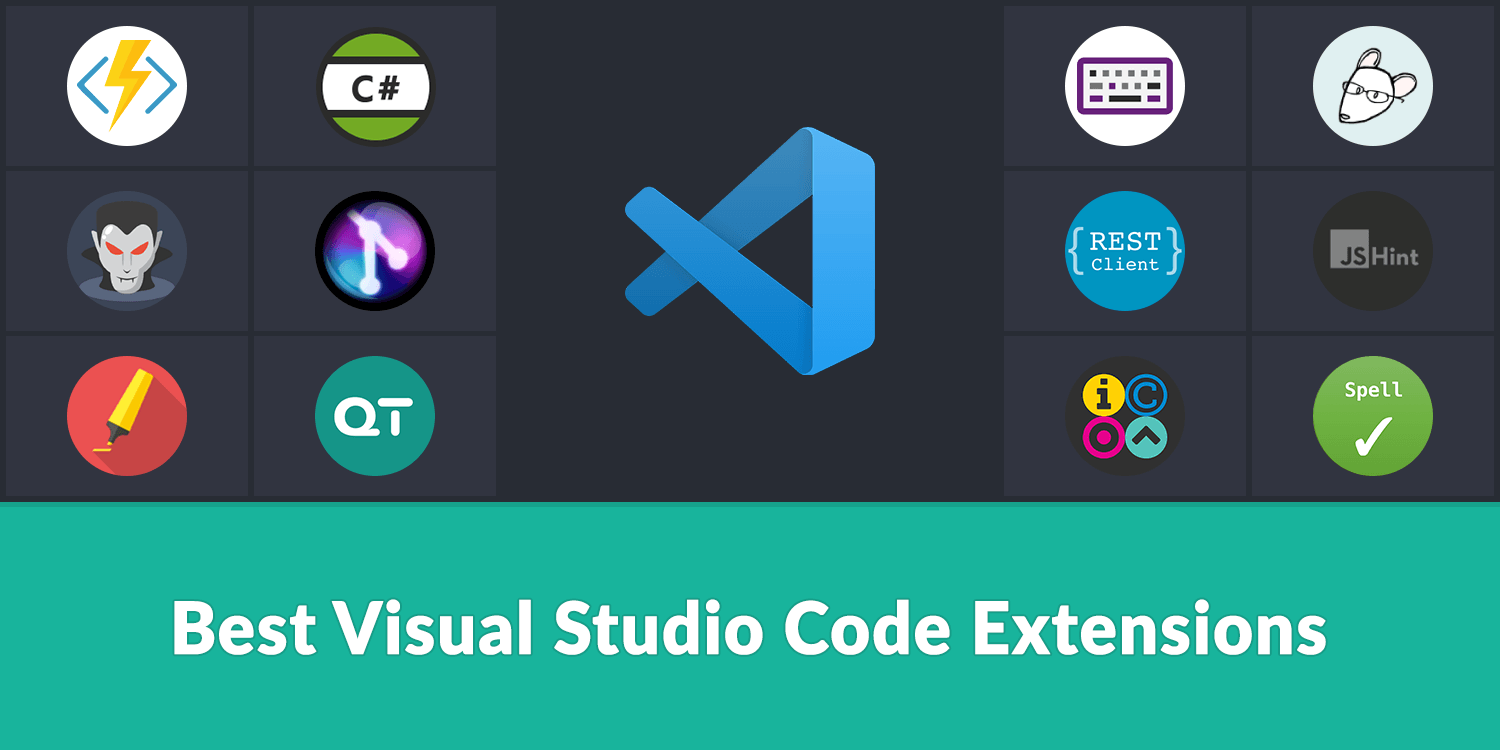


**Hình 9: Logo Visual Studio Code**

Visual Studio Code là một trình biên tập lập trình code miễn phí dành cho Windows, Linux và macOS, Visual Studio Code được phát triển bởi Microsoft. Nó được xem là một sự kết hợp hoàn hảo giữa IDE và Code Editor.

Visual Studio Code hỗ trợ chức năng debug, đi kèm với Git, có syntax highlighting, tự hoàn thành mã thông minh, snippets, và cải tiến mã nguồn. Nhờ tính năng tùy chỉnh, Visual Studio Code cũng cho phép người dùng thay đổi theme, phím tắt, và các tùy chọn khác.

**Một số tính năng của Visual Studio Code**



**Hình 10: Một số tính năng Visual Studio Code**

**Hỗ trợ nhiều ngôn ngữ lập trình**

Visual Studio Code hỗ trợ nhiều ngôn ngữ lập trình như C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript, … Vì vậy, nó dễ dàng phát hiện và đưa ra thông báo nếu chương chương trình có lỗi.

**Hỗ trợ đa nền tảng**

Các trình viết code thông thường chỉ được sử dụng hoặc cho Windows hoặc Linux hoặc Mac Systems. Nhưng Visual Studio Code có thể hoạt động tốt trên cả ba nền tảng trên.

**Cung cấp kho tiện ích mở rộng**

Trong trường hợp lập trình viên muốn sử dụng một ngôn ngữ lập trình không nằm trong số các ngôn ngữ Visual Studio hỗ trợ, họ có thể tải xuống tiện ích mở rộng. Điều này vẫn sẽ không làm giảm hiệu năng của phần mềm, bởi vì phần mở rộng này hoạt động như một chương trình độc lập.

**Kho lưu trữ an toàn**

Đi kèm với sự phát triển của lập trình là nhu cầu về lưu trữ an toàn. Với Visual Studio Code, người dùng có thể hoàn toàn yên tâm vì nó dễ dàng kết nối với Git hoặc bất kỳ kho lưu trữ hiện có nào.

**Hỗ trợ web**

Visual Studio Code hỗ trợ nhiều ứng dụng web. Ngoài ra, nó cũng có một trình soạn thảo và thiết kế website.

**Lưu trữ dữ liệu dạng phân cấp**

Phần lớn tệp lưu trữ đoạn mã đều được đặt trong các thư mục tương tự nhau. Ngoài ra, Visual Studio Code còn cung cấp các thư mục cho một số tệp đặc biệt quan trọng.

**Hỗ trợ viết Code**

Một số đoạn code có thể thay đổi chút ít để thuận tiện cho người dùng. Visual Studio Code sẽ đề xuất cho lập trình viên các tùy chọn thay thế nếu có.

**Hỗ trợ thiết bị đầu cuối**

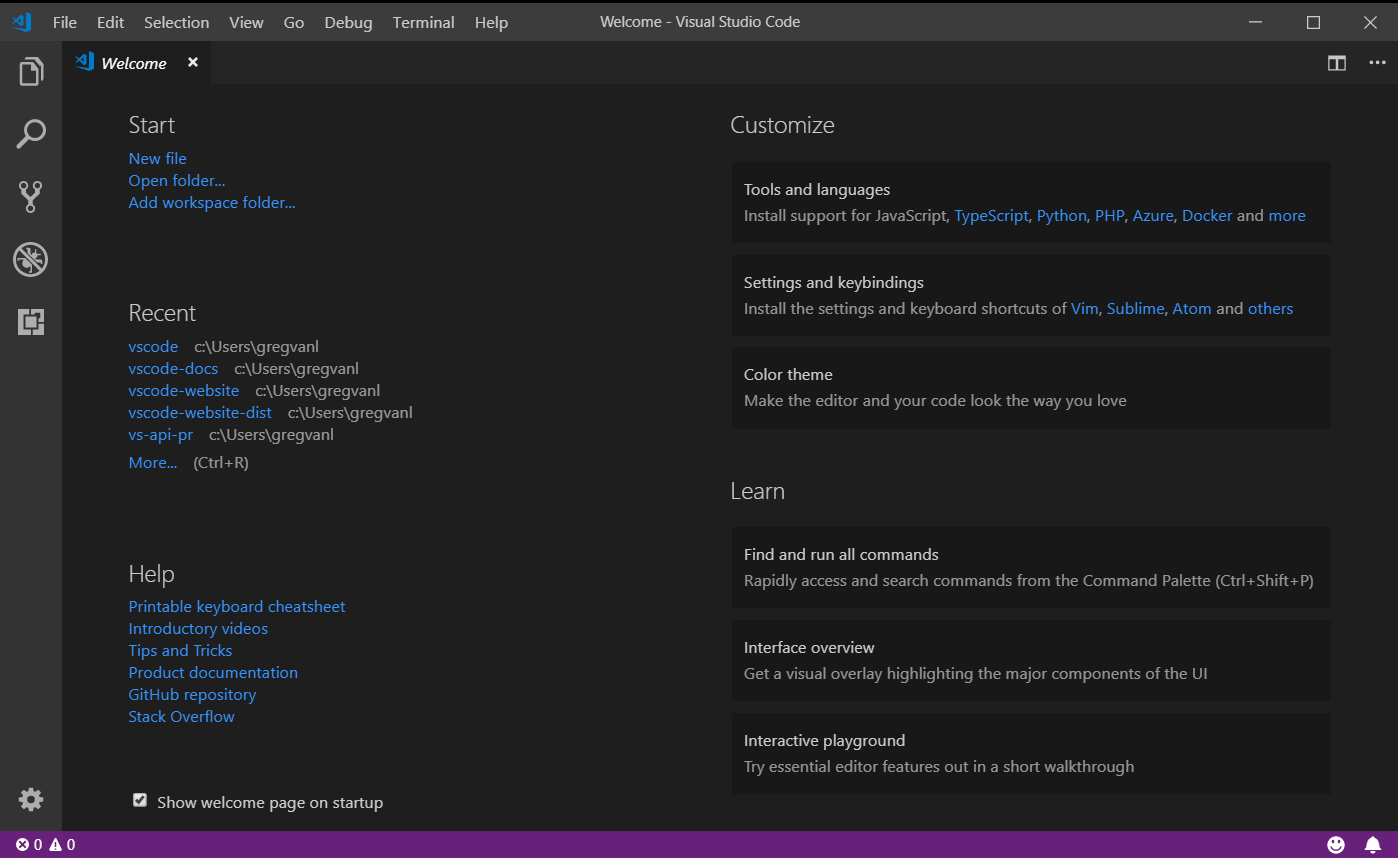
Visual Studio Code có tích hợp thiết bị đầu cuối, giúp người dùng khỏi phải chuyển đổi giữa hai màn hình hoặc trở về thư mục gốc khi thực hiện các thao tác.

**Màn hình đa nhiệm**

Người dùng Visual Studio Code có thể mở cùng lúc nhiều tệp tin và thư mục – mặc dù chúng không hề liên quan với nhau.

**Intellisense**

Hầu hết các trình viết mã đều có tính năng nhắc mã Intellisense, nhưng ít chương trình nào chuyên nghiệp bằng Visual Studio Code. Nó có thể phát hiện nếu bất kỳ đoạn mã nào không đầy đủ. Thậm chí, khi lập trình viên quên không khai báo biến, Intellisense sẽ tự động giúp họ bổ sung các cú pháp còn thiếu.



**Hình 11: Giao diện Visual Studio Code**

**Hỗ trợ Git**

Visual Studio Code hỗ trợ kéo hoặc sao chép mã trực tiếp từ GitHub. Mã này sau đó có thể được thay đổi và lưu lại trên phần mềm.

**Bình luận**

Việc để lại nhận xét giúp người dùng dễ dàng nhớ công việc cần hoàn thành.

**Ưu điểm**

* Hỗ trợ đa nền tảng: Windows, Linux, Mac
* Hỗ trợ đa ngôn ngữ: C/C++, C#, F#, Visual Basic, HTML, CSS, JavaScript, JSON
* Ít dung lượng
* Tính năng mạnh mẽ
* Intellisense chuyên nghiệp
* Giao diện thân thiện

### Figma

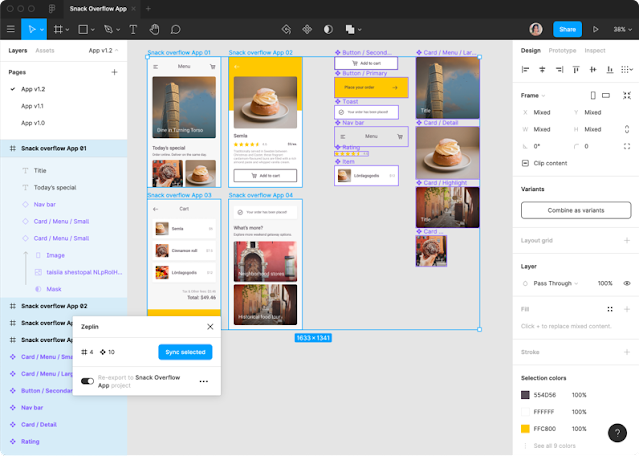
Figma là công cụ được ra mắt vào năm 2016, với giao diện thân thiện và tính dễ sử dụng, Figma đã nhanh chóng nổi lên và trở thành một công cụ thiết kế giao diện người dùng phổ biến trong cộng đồng công nghệ toàn cầu. Một số thương hiệu lớn sử dụng Figma cho tới thời điểm hiện tại có thể kể đến như Microsoft, Twitter, GitHub, Dropbox…Khác với những công cụ thiết kế trước đây, Figma được thiết kế trên nền tảng đám mây. Đây là công cụ có tính năng tương tự như Sketch, tuy nhiên nó hỗ trợ làm việc nhóm tốt hơn.



**Hình 12: Logo Figma**

**Những tính năng nổi bật của Figma**

* **Tính tương thích:** Figma có thể hoạt động tương thích trên tất cả các hệ điều hành có trình duyệt web, từ Windows, Linux, Macs cho đến Chromebooks đều có thể dùng công cụ này. Tính tới thời điểm này, đây là công cụ thiết kế duy nhất có thể làm được điều này. Tất cả người dùng có thể chia sẻ, mở hay chỉnh sửa một file Figma dễ dàng dù đang dùng bất kỳ hệ điều hành nào.
* Hỗ trợ làm việc nhóm hiệu quả: Là một công cụ browser-based, vậy nên mọi người trong team có thể dễ dàng làm việc với nhau tương tự như làm việc trên Google Docs. Mỗi thành viên trong nhóm đang xem hoặc chỉnh sửa file, Figma sẽ hiển thị đại diện bằng một avatar tròn trên thanh công cụ với tên riêng.
* **Chia sẻ file đơn giản**: Ngoài những tính năng trên, Figma còn thu hút người dùng bởi tính năng chia sẻ file đơn giản. Bạn có thể chia sẻ file Figma bằng việc gửi link, người nhận link sẽ có khả năng truy cập vào file đó. Nếu bạn lựa chọn Link to selected frame người nhận sẽ nhìn thấy ngay frame bạn đang chọn lúc chia sẻ. Đây là tính năng được rất nhiều designer sử dụng.
* **Tính năng feedback thuận tiện:**Figma có hỗ trợ bình luận ngay trên app, có cả ở trong chế độ design và prototyping. Ngoài ra những bình luận này còn có thể được theo dõi qua Slack hoặc email. Bạn có thể sử dụng tính năng này để ghi chú lại những thảo luận và sửa file ngay lập tức trong các cuộc họp. Với chức năng bình luận ngay trong file thiết kế như thế này, Figma giúp bạn tiết kiệm được rất nhiều thời gian chờ hỗ trợ.
* **Tất cả mọi thứ đều được lưu trữ online:** Bạn đã bao giờ gặp phải tính trạng tốn rất nhiều thời gian để thiết kế, file sắp hoàn thành mà vì một lý do nào đó nó không được lưu hay chưa? Cảm giác lúc đó sẽ như thế nào nhỉ? Với Figma thì mọi thứ đã được lưu trên cloud. Bạn sẽ không còn cảm giác lo sợ làm mất file, hỏng máy hay hàng tá các vấn đề trời ơi đất hỡi khác nữa. Bạn có thể tạo và sắp xếp file theo project, mỗi file chia ra làm nhiều page tùy theo nhu cầu để dễ dàng quản lý.
* **Kho Plugin phong phú:** Thời gian đầu khi mới ra mắt, các plugin của Figma không được phong phú như Sketch. Tuy nhiên vào những năm 2019, Figma đã tung ra kho plugin cho riêng mình.
* **File versioning:** Chức năng file versioning được tích hợp sẵn trong Figma. Chức năng này cho phép người dùng lưu trữ lại trạng thái của file design bất kỳ lúc nào.
* **Prototype tích hợp sẵn:** Prototype là công cụ mà Figma được tích hợp trước cả Sketch. Khi sử dụng Figma, bạn không cần phải nhờ đến các công cụ hỗ trợ prototype như Marvel hay Invision, hay phải export ra file hình để chuyển cho các bộ phận liên quan như trước đây.

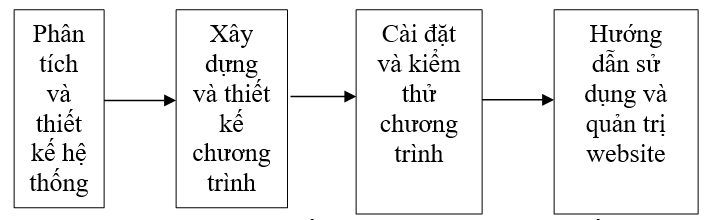


**Hình 13: Giao diện làm việc Figma**

# THIẾT KẾ VÀ XÂY DỰNG HỆ THỐNG

## Khảo sát hệ thống

Sơ đồ quy trình nghiệp vụ



**Hình 14: Sơ đồ quy trình nghiệp vụ**

## Phân tích hệ thống

### Phân tích hệ thống về mặt chức năng

**Khách hàng:**

Đặng nhập (học viên): Học viên đăng nhập bằng tài khoản đã đăng kí vào website

Đăng kí tài khoản

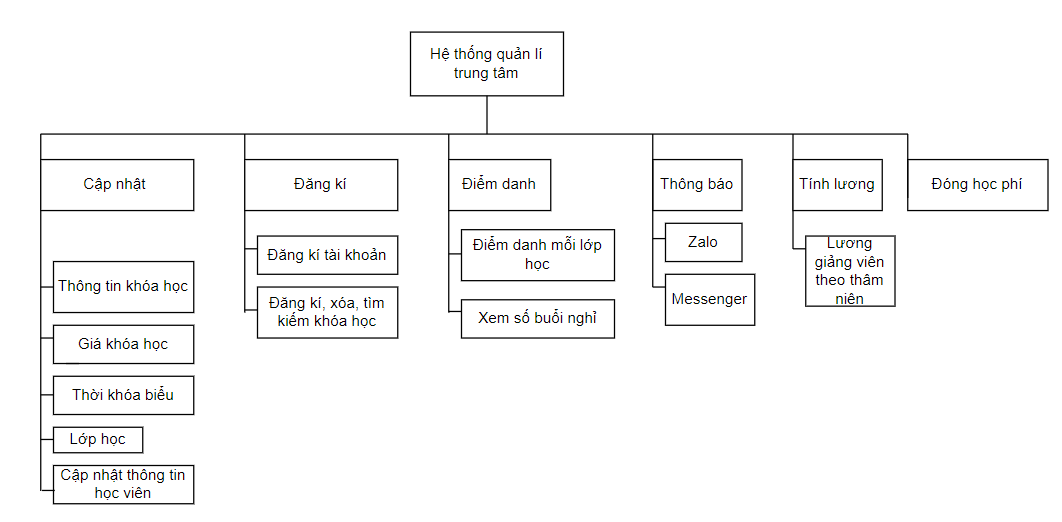
Đăng kí khóa học

Đóng tiền học phí

**Trung tâm:**

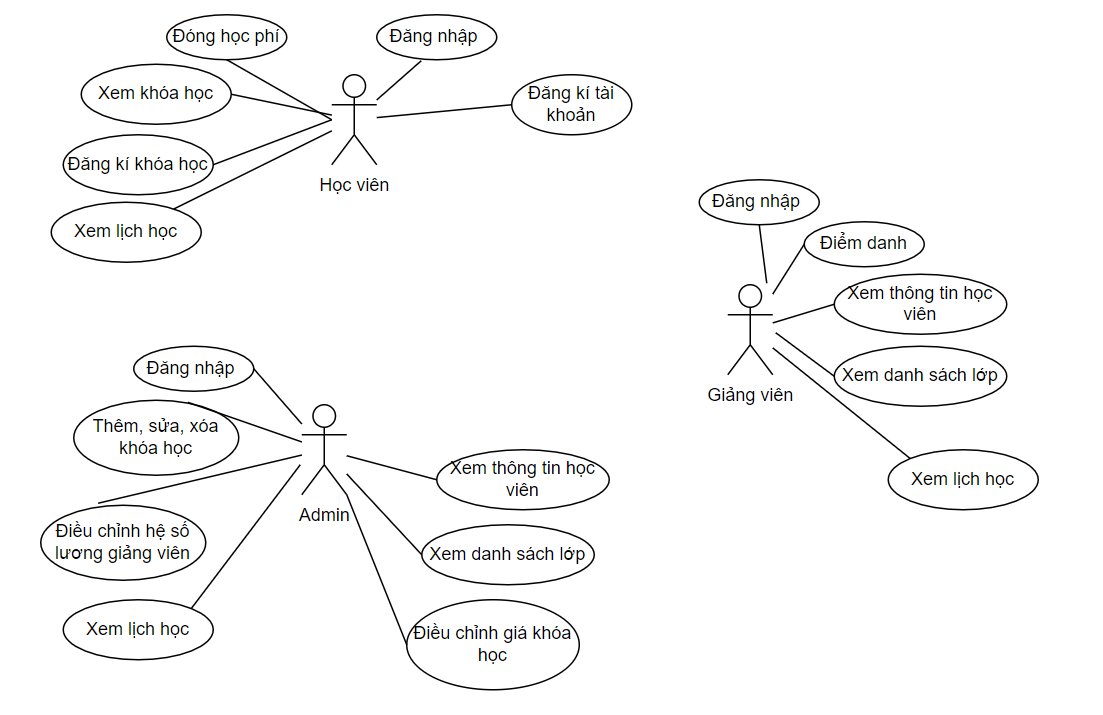
* **Thiết lập thời khoá biểu cho từng lớp trực quan**: Có thể nhìn thấy thời khoá biểu của lớp theo tuần, số buổi đã học của lớp đó.
* **Thiết lập tiền lương trả cho giáo viên**: Tính lương cho giảng viên theo buổi cho tuỳ giáo viên, mỗi giáo viên có mức lương theo buổi khác nhau) hoặc theo tháng cho 1 vài giảng viên cơ hữu.
* **Thiết lập giá khóa học:** Mỗi khóa học sẽ có giá khác nhau
* **Điểm danh cho từng lớp học theo từng buổi**: Giảng viên có thể điểm danh, và có thể nhìn được tổng số buổi học và buổi nghỉ của từng học viên
* **Thông báo nghỉ hoặc học bù cho học viên:** Thông báo qua zalo cho từng lớp theo số điện thoại hoặc messenger của facebook

#### Biểu đồ phân rã chức năng

****

**Hình 15: Biểu đồ phân rã chức năng**

### Sơ đồ Usecase



**Hình 16: Sơ đồ Usecase**

## Cơ sở dữ liệu

### Xác định thực thể

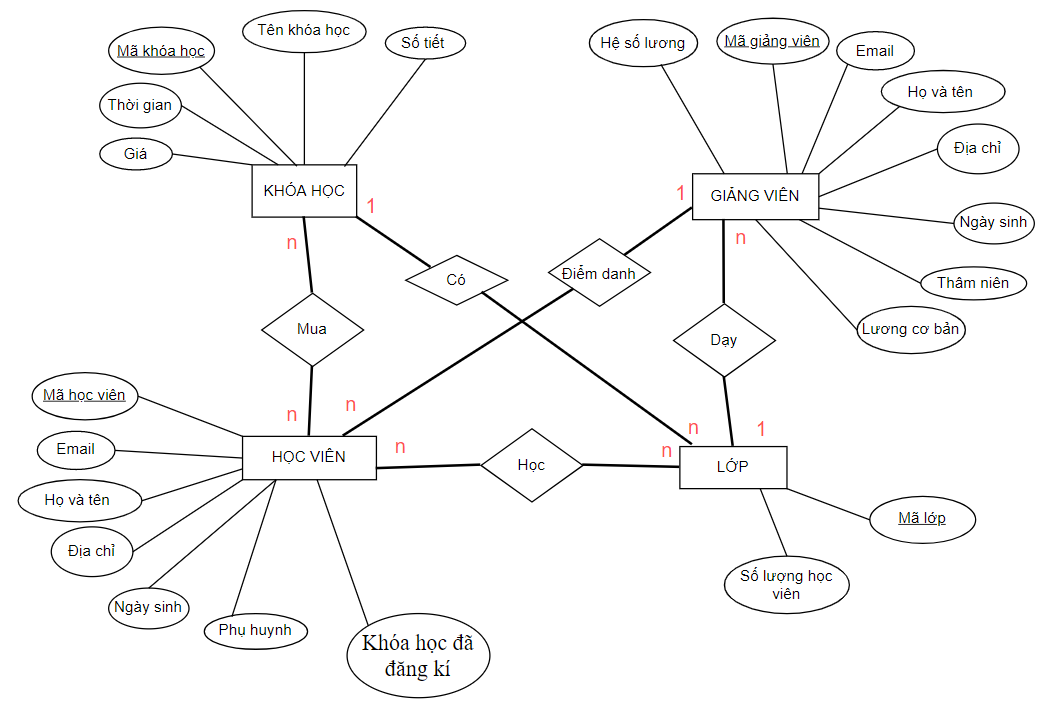
**Học viên (Mã hv**, Họ và tên, ngày sinh, địa chỉ, điện thoại, khóa học đã đăng kí, phụ huynh**)**

**Khóa học (Mã môn học,** tên khóa học, số tiết, thời gian, giá**)**

**Lớp (Mã lớp**, số lượng học viên**)**

**Giảng viên (Mã gv,** họ và tên, ngày sinh, giới tính, địa chỉ, điện thoại,thâm niên, hệ số lương, lương cơ bản ghi chú)

### Mô hình ER (bỏ lớp )



**Hình 17: Mô hình ER**

### Các bảng dữ liệu

Bảng **User** để lưu thông tin người dùng bao gồm admin, học viên, giảng viên

**Bảng 1: Bảng dữ liệu User**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ghi chú** |
| iduser | Char | ID người dùng |
| email | Char | Email người dùng(dùng để đăng nhập) |
| dia chi | Char | Địa chỉ người dùng |
| ngay sinh | Datetime | Ngày sinh người dùng |
| password | Char | Mật khẩu để đăng nhập |
| role(1,2,3) | Number | Phân quyền người dùng: 1:admin ; 2:giảng viên ; 3:học viên |
| Luong | Number | Lương của giảng viên |
| Tham nien | Char | Thâm niên |
| Phu huynh | Char | Thông tin phụ huynh |

Bảng **Course** để lưu thông tin khóa học

**Bảng 2: Bảng dữ liệu Course**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ghi chú** |
| idcourse | Char | Mã khóa học |
| name | Char | Tên khóa học |
| iduser | Char | Mã |
| gia | Char | Giá khóa học |
| tongsobuoi | Number | Số buổi học của khóa học |
| batdau | Datetime | Ngày bắt đầu khóa học |
| ketthuc | Datetime | Ngày kết thức khóa học |
| thoikhoabieu | String | Thời khóa biểu |
| joincode | Char | Mã để vào lớp |

Bảng **Diemdanh** để lưu thông tin điểm danh buổi học

**Bảng 3: Bảng dữ liệu Diemdanh**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ghi chú** |
| ID | Char |  |
| idclass | Char | Id lớp học |
| iduser | Char | Id học viên |
| Date | datetime | Ngày học |

Bảng **Luong** để quản lí lương giảng viên

**Bảng 4: Bảng dữ liệu Luong**

|  |  |  |
| --- | --- | --- |
| **Tên trường** | **Kiểu dữ liệu** | **Ghi chú** |
| iduser | Char | Id giảng viên |
| teachedday | Number | Số ngày đã lên lớp |
| salary | Number | Hệ số lương |

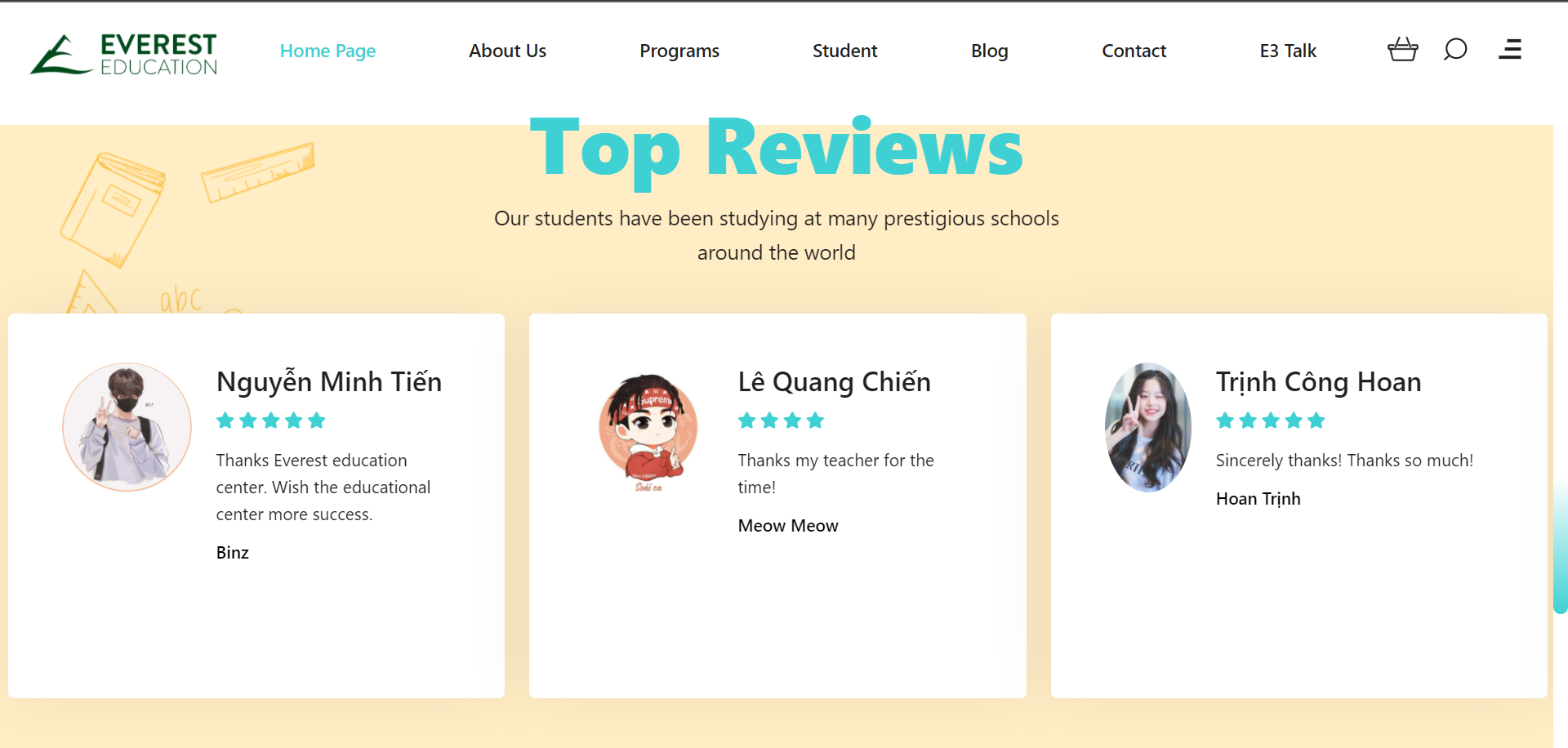
### Kết nối dữ liệu với DataBase

*(cần bổ sung)*

# THỰC NGHIỆM

## Tổng quan Website

### Giao diện trang chủ



**Hình 18: Giao diện trang chủ website quản lí trung tâm**