**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**

-----------------------



**BÁO CÁO BÀI TẬP LỚN**

**HỌC PHẦN : CƠ SỞ DỮ LIỆU NÂNG CAO**

**ĐỀ TÀI: XÂY DỰNG HỆ THỐNG QUẢN LÝ TÀI LIỆU CHO DOANH NGHIỆP VỪA VÀ NHỎ SỬ DỤNG CSDL MONGODB**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | **TS PHẠM VĂN HÀ** |
| **Mã lớp độc lập:** | **CH HTTT K14.2** |
| **Nhóm:** | **4** |

**Học viên thực hiện: 1. Nguyễn Văn Ánh – 2024700134**

**2. Đỗ Xuân Đoài – 2024700143**

**3. Bùi Thị Thạo – 2024700116**

**4. Nguyễn Bá Hải – 2024700100**

**Hà Nội - 2024**

**LỜI CẢM ƠN**

Trước tiên, chúng tôi xin gửi lời cảm ơn chân thành đến quý thầy cô, nhà trường và bạn bè đã quan tâm và động viên chúng tôi trong suốt quá trình nghiên cứu và hoàn thiện tiểu luận này. Sự quan tâm và hỗ trợ của quý thầy cô không chỉ là nguồn động viên to lớn mà còn là yếu tố quan trọng giúp chúng tôi xác định hướng đi và hoàn thiện đề tài.

Chúng tôi đặc biệt xin bày tỏ lòng biết ơn sâu sắc đến Thầy TS. Phạm Văn Hà vì sự tận tâm và chỉ dẫn quý báu trong quá trình hướng dẫn thực hiện bài tiểu luận. Những kiến thức sâu rộng và những gợi ý của thầy đã giúp chúng tôi hiểu rõ hơn về các phương pháp phân tích dữ liệu và áp dụng chúng vào đề tài " *Xây dựng hệ thống quản lý tài liệu cho doanh nghiệp vừa và nhỏ sử dụng cơ sở dữ liệu MongoDB*". Sự kiên nhẫn và đồng hành của thầy đã giúp chúng tôi vượt qua những thử thách và hoàn thành báo cáo này một cách tốt nhất.

Chúng tôi cũng xin gửi lời cảm ơn đến tất cả những người đã hỗ trợ chúng tôi trong quá trình nghiên cứu. Những đóng góp, ý kiến và sự hỗ trợ của các bạn đã đóng vai trò quan trọng trong việc nâng cao chất lượng của báo cáo này.

Cuối cùng, chúng tôi cam kết sẽ tiếp tục nỗ lực, phấn đấu và trưởng thành hơn trong con đường nghiên cứu và học tập. Chúng tôi luôn trân trọng và biết ơn những sự giúp đỡ mà chúng tôi đã nhận được và sẽ cố gắng chia sẻ kiến thức và kinh nghiệm của mình để góp phần vào sự phát triển của cộng đồng.

***Học viên***

*Nguyễn Văn Ánh*

*Đỗ Xuân Đoài*

*Bùi Thị Thạo*

*Nguyễn Bá Hải*

**LỜI MỞ ĐẦU**

Trong kỷ nguyên số hóa hiện nay, việc quản lý thông tin và tài liệu hiệu quả là yếu tố quan trọng quyết định sự thành công của các doanh nghiệp. Đối với các doanh nghiệp vừa và nhỏ, việc tổ chức, lưu trữ, và truy xuất thông tin một cách nhanh chóng không chỉ giúp tối ưu hóa quy trình làm việc mà còn góp phần nâng cao khả năng cạnh tranh và đáp ứng kịp thời các nhu cầu của thị trường. Tuy nhiên, việc quản lý tài liệu trong môi trường doanh nghiệp thường gặp nhiều thách thức, đặc biệt khi phải xử lý khối lượng lớn dữ liệu phi cấu trúc và đảm bảo tính bảo mật cao.

Trong bối cảnh đó, cơ sở dữ liệu NoSQL, đặc biệt là MongoDB, đã nổi lên như một giải pháp hiệu quả cho việc quản lý dữ liệu phi cấu trúc với khả năng mở rộng cao và tính linh hoạt vượt trội. MongoDB không chỉ hỗ trợ việc lưu trữ và truy xuất dữ liệu một cách nhanh chóng, mà còn cung cấp các công cụ mạnh mẽ giúp doanh nghiệp dễ dàng quản lý, tìm kiếm, và phân tích tài liệu. Điều này giúp cải thiện đáng kể hiệu suất công việc và tối ưu hóa các quy trình quản lý thông tin trong doanh nghiệp.

Bài tiểu luận này sẽ đi sâu vào việc xây dựng một hệ thống quản lý tài liệu dành cho các doanh nghiệp vừa và nhỏ, sử dụng cơ sở dữ liệu MongoDB. Nghiên cứu sẽ bắt đầu bằng việc giới thiệu tổng quan về MongoDB, phân tích các ưu điểm và nhược điểm của việc sử dụng cơ sở dữ liệu này trong quản lý tài liệu. Tiếp theo, quá trình thiết kế và triển khai hệ thống sẽ được trình bày chi tiết, bao gồm việc lựa chọn kiến trúc hệ thống, thiết kế mô hình dữ liệu, và xây dựng các chức năng quản lý tài liệu.

Cuối cùng, bài tiểu luận sẽ đánh giá hiệu quả của hệ thống quản lý tài liệu được xây dựng, so sánh với các phương pháp quản lý tài liệu truyền thống, và đưa ra những khuyến nghị cho việc áp dụng rộng rãi trong các doanh nghiệp vừa và nhỏ. Qua đó, nghiên cứu hy vọng sẽ góp phần cung cấp một giải pháp công nghệ tiên tiến, giúp doanh nghiệp vượt qua các thách thức trong việc quản lý tài liệu và nắm bắt các cơ hội phát triển trong thời đại số.Chúng tôi hy vọng rằng, thông qua việc nghiên cứu và phân tích sâu sắc, đề tài này sẽ góp phần quan trọng vào việc bảo vệ các giao dịch tài chính và thúc đẩy sự phát triển bền vững của hệ thống thanh toán điện tử toàn cầu.

Đề tài này được thể hiện qua 4 chương:

Chương 1: Tổng quan về đề tài

Chương 2: Tổng quan về NoSQL và MongoDB

Chương 3: Phương pháp kỹ thuật

Chương 4: Giới thiệu chương trình

**MỤC LỤC**

[CHƯƠNG 1. TỔNG QUAN VỀ ĐỀ TÀI 9](#_Toc175332496)

[1.1. Lí do chọn đề tài 9](#_Toc175332497)

[1.2. Mục đích nghiên cứu 10](#_Toc175332498)

[1.3. Phạm vi và đối tượng nghiên cứu 10](#_Toc175332499)

[CHƯƠNG 2. TỔNG QUAN VỀ NOSQL VÀ MONGODB 12](#_Toc175332500)

[2.1. Giới thiệu về NoSQL 12](#_Toc175332501)

[2.1.1. Khái niệm về NoSQL? 12](#_Toc175332502)

[2.1.2. Đặc điểm của NoSQL? 13](#_Toc175332503)

[2.2. Giới thiệu về MongoDB 15](#_Toc175332504)

[2.2.1. Khái niệm về MongoDB ? 15](#_Toc175332505)

[2.2.2. Một số đặc điểm nổi bật của MongoDB 16](#_Toc175332506)

[2.3. Kiến trúc và mô hình dữ liệu của MongoDB 17](#_Toc175332507)

[2.3.1. Kiến Trúc của MongoDB 17](#_Toc175332508)

[2.3.2. Mô hình dữ liệu của MongoDB 18](#_Toc175332509)

[2.4. Ưu điểm của MongoDB trong quản lý tài liệu 19](#_Toc175332510)

[2.5. Ưu điểm của MongoDB trong quản lý tài liệu 20](#_Toc175332511)

[CHƯƠNG 3. PHƯƠNG PHÁP KĨ THUẬT 22](#_Toc175332512)

[3.1. Phân tích yêu cầu của hệ thống 22](#_Toc175332513)

[3.1.1. Quản lý tài liệu 22](#_Toc175332514)

[3.1.2. Quản lý người dùng 22](#_Toc175332515)

[3.1.3. Tìm kiếm và truy vấn 23](#_Toc175332516)

[3.1.4. Giao diện người dùng 23](#_Toc175332517)

[3.2. Thiết kế hệ thống quản lý tài liệu 23](#_Toc175332518)

[3.2.1. Kiến trúc tổng thể 23](#_Toc175332519)

[3.2.2. Mô hình dữ liệu và cấu trúc tài liệu 24](#_Toc175332520)

[3.3. Triển khai hệ thống 26](#_Toc175332521)

[3.3.1. Môi trường và công cụ phát triển 26](#_Toc175332522)

[CHƯƠNG 4. GIỚI THIỆU CHƯƠNG TRÌNH 28](#_Toc175332523)

[4.1. Giao diện trang đăng nhập 28](#_Toc175332524)

[4.1.1. Trang đăng nhập 29](#_Toc175332525)

[4.1.2. Trang quản lý tài liệu đến 29](#_Toc175332526)

[4.1.3. Trang quản lý tài liệu đi 30](#_Toc175332527)

[4.1.4. Trang quản lý tài liệu nội bộ 30](#_Toc175332528)

[4.1.5. Trang quản lý hợp đồng 31](#_Toc175332529)

[4.1.6. Trang quản lý tài liệu nội bộ 31](#_Toc175332530)

[4.1.7. Trang quản lý người dùng 32](#_Toc175332531)

[4.1.8. Trang thêm tài liệu 32](#_Toc175332532)

[4.1.9. Trang tìm kiếm 32](#_Toc175332533)

[4.1.10. Trang thêm người dùng 33](#_Toc175332534)

[KẾT LUẬN 34](#_Toc175332535)

[TÀI LIỆU THAM KHẢO 35](#_Toc175332536)

**DANH MỤC HÌNH ẢNH**

[Hình 2.1: Khái niệm về MongoDB 12](#_Toc175327164)

[Hình 2.2: Khái niệm về MongoDB 15](#_Toc175327165)

[Hình 2.3: Đặc điểm của MongoDB 16](#_Toc175327166)

[Hình 2.4: Kiến trúc MongoDB 17](#_Toc175327167)

[Hình 3.1: Môi trường phát triển hệ thống 27](#_Toc175327168)

[Hình 4.1: Trang đăng nhập 28](#_Toc175327169)

[Hình 4.2: Trang dashboard 29](#_Toc175327170)

[Hình 4.3: Trang quản lý tài liệu đến 29](#_Toc175327171)

[Hình 4.4: Trang quản lý tài liệu đi 30](#_Toc175327172)

[Hình 4.5: Trang quản lý tài liệu nội bộ 30](#_Toc175327173)

[Hình 4.6: Trang quản lý hợp đồng 31](#_Toc175327174)

[Hình 4.7: Trang quản lý tài liệu nội bộ 31](#_Toc175327175)

[Hình 4.8: Trang quản lý người dùng 32](#_Toc175327176)

[Hình 4.9: Trang thêm tài liệu 32](#_Toc175327177)

[Hình 4.10: Trang tìm kiếm 33](#_Toc175327178)

[Hình 4.11: Trang thêm người dùng 33](#_Toc175327179)

**DANH MỤC CHỮ VIẾT TẮT**

|  |  |
| --- | --- |
| JSON | JavaScript Object Notation. |
| XML | eXtensible Markup Language |
| SQL | Structured Query Language |

# TỔNG QUAN VỀ ĐỀ TÀI

## Lí do chọn đề tài

Trong bối cảnh kinh tế hiện nay, doanh nghiệp vừa và nhỏ (SMEs) đóng vai trò quan trọng trong việc thúc đẩy tăng trưởng kinh tế, tạo việc làm và đổi mới sáng tạo. Tuy nhiên, các doanh nghiệp này thường gặp phải nhiều thách thức trong việc quản lý tài liệu và thông tin một cách hiệu quả, đặc biệt khi khối lượng dữ liệu ngày càng gia tăng. Việc lưu trữ, quản lý, và truy xuất tài liệu một cách nhanh chóng và chính xác là yếu tố quan trọng giúp doanh nghiệp tối ưu hóa quy trình hoạt động và nâng cao năng suất làm việc.

Hệ quản trị cơ sở dữ liệu truyền thống như SQL thường gặp khó khăn khi xử lý dữ liệu phi cấu trúc và không thể đáp ứng được nhu cầu linh hoạt của các doanh nghiệp hiện đại. Trong khi đó, cơ sở dữ liệu NoSQL, đặc biệt là MongoDB, đã chứng minh được khả năng vượt trội trong việc quản lý dữ liệu phi cấu trúc, với tính năng linh hoạt, dễ mở rộng, và hiệu suất cao. MongoDB giúp các doanh nghiệp dễ dàng lưu trữ và quản lý các loại dữ liệu khác nhau, từ văn bản, hình ảnh, đến các tài liệu có cấu trúc phức tạp, đồng thời hỗ trợ việc tìm kiếm và truy xuất thông tin một cách nhanh chóng.

Chính vì lý do đó, việc nghiên cứu và phát triển hệ thống quản lý tài liệu sử dụng MongoDB cho các doanh nghiệp vừa và nhỏ là một đề tài có tính thực tiễn cao. Đề tài này không chỉ giúp giải quyết các vấn đề hiện hữu trong quản lý tài liệu của doanh nghiệp, mà còn cung cấp một giải pháp công nghệ hiện đại, phù hợp với xu thế phát triển của thời đại số.

Ngoài ra, việc ứng dụng MongoDB vào hệ thống quản lý tài liệu còn mở ra cơ hội nghiên cứu và phát triển thêm các tính năng nâng cao như quản lý tài liệu phi cấu trúc, tối ưu hóa quá trình tìm kiếm, và tích hợp với các hệ thống khác trong doanh nghiệp. Đây là những lý do khiến đề tài "Xây dựng hệ thống quản lý tài liệu cho doanh nghiệp vừa và nhỏ sử dụng CSDL MongoDB" trở nên đặc biệt quan trọng và đáng được nghiên cứu.

## Mục đích nghiên cứu

Mục đích của nghiên cứu này là xây dựng một hệ thống quản lý tài liệu đơn giản nhưng hiệu quả, đáp ứng nhu cầu của các doanh nghiệp vừa và nhỏ, thông qua việc sử dụng cơ sở dữ liệu MongoDB. Nghiên cứu này nhằm đạt được các mục tiêu sau:

* Phát triển một hệ thống với giao diện trực quan, dễ thao tác, giúp người dùng có thể quản lý tài liệu một cách thuận tiện mà không cần nhiều kiến thức kỹ thuật.
* Ứng dụng khả năng xử lý dữ liệu phi cấu trúc, tính linh hoạt, và khả năng mở rộng của MongoDB để xây dựng một hệ thống quản lý tài liệu phù hợp với các doanh nghiệp vừa và nhỏ.
* Tạo ra một hệ thống có hiệu suất tốt, đảm bảo việc lưu trữ, tìm kiếm, và truy xuất tài liệu một cách nhanh chóng, đồng thời dễ dàng bảo trì và mở rộng khi cần thiết.

## Phạm vi và đối tượng nghiên cứu

Nghiên cứu này tập trung vào việc thiết kế và phát triển một hệ thống quản lý tài liệu đơn giản sử dụng cơ sở dữ liệu MongoDB, phù hợp với các doanh nghiệp vừa và nhỏ. **Phạm vi nghiên cứu bao gồm:**

* Khía cạnh kỹ thuật: Nghiên cứu và ứng dụng các tính năng của MongoDB trong việc lưu trữ, quản lý, và truy xuất tài liệu. Điều này bao gồm việc thiết kế mô hình dữ liệu, phát triển các chức năng cơ bản như lưu trữ, tìm kiếm, và quản lý tài liệu.
* Phạm vi ứng dụng: Hệ thống được thiết kế để áp dụng trong các doanh nghiệp vừa và nhỏ, nơi mà yêu cầu quản lý tài liệu cần được thực hiện đơn giản, dễ sử dụng, và có khả năng mở rộng trong tương lai.
* Giới hạn nghiên cứu: Nghiên cứu sẽ không đi sâu vào các tính năng nâng cao như quản lý phiên bản tài liệu, tích hợp hệ thống với các nền tảng khác hoặc các chức năng bảo mật phức tạp. Mục tiêu là tạo ra một hệ thống đơn giản và hiệu quả, dễ dàng triển khai và sử dụng trong các môi trường doanh nghiệp nhỏ.

**Đối tượng nghiên cứu:**

Đối tượng nghiên cứu của đề tài này là hệ thống quản lý tài liệu cho doanh nghiệp vừa và nhỏ, với mục tiêu cải thiện hiệu quả quản lý thông tin và tối ưu hóa quy trình làm việc. Cụ thể:

* Công ty vừa và nhỏ: Các doanh nghiệp có quy mô nhỏ, nơi việc quản lý tài liệu thường gặp nhiều thách thức do thiếu nguồn lực và công nghệ phù hợp.
* Người dùng hệ thống: Nhân viên và quản lý trong doanh nghiệp, những người có nhu cầu sử dụng hệ thống để quản lý tài liệu hàng ngày, nhưng không cần nhiều kiến thức kỹ thuật để thao tác hệ thống.

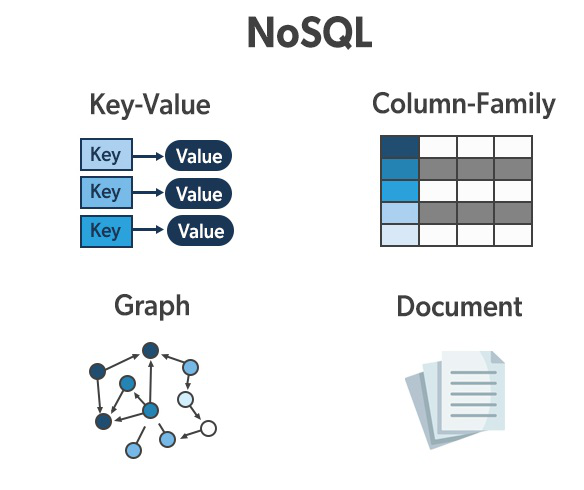
Phần "Phạm vi và đối tượng nghiên cứu" này xác định rõ ràng giới hạn và đối tượng của nghiên cứu, giúp tập trung vào các khía cạnh cần thiết và phù hợp với mục tiêu của đề tài.

# TỔNG QUAN VỀ NOSQL VÀ MONGODB

## Giới thiệu về NoSQL

### Khái niệm về NoSQL?

**NoSQL** (Not Only SQL) là một loại cơ sở dữ liệu được thiết kế để lưu trữ, quản lý, và truy xuất dữ liệu theo cách không sử dụng mô hình bảng (table) và quan hệ (relations) truyền thống như các cơ sở dữ liệu quan hệ (RDBMS). Thay vào đó, NoSQL tập trung vào việc cung cấp các phương thức lưu trữ và truy cập dữ liệu linh hoạt, phù hợp với các ứng dụng có yêu cầu cao về hiệu suất, khả năng mở rộng, và khả năng xử lý dữ liệu lớn.



Hình 2.1: Khái niệm về NoSQL

Các cơ sở dữ liệu NoSQL thường được sử dụng trong các hệ thống yêu cầu khả năng mở rộng ngang (horizontal scalability) cao, nơi mà dữ liệu có thể được phân phối trên nhiều máy chủ mà vẫn đảm bảo hiệu suất cao. NoSQL hỗ trợ nhiều kiểu lưu trữ dữ liệu khác nhau, bao gồm:

* **Key-Value Stores**: Lưu trữ dữ liệu dưới dạng cặp key-value, trong đó key là một chuỗi định danh và value là dữ liệu được lưu trữ.
* **Document Stores**: Lưu trữ dữ liệu dưới dạng tài liệu (document), thường ở định dạng JSON hoặc BSON, cho phép linh hoạt trong việc tổ chức dữ liệu.
* **Column-Family Stores**: Lưu trữ dữ liệu theo cột thay vì theo hàng, giúp tối ưu hóa việc truy xuất dữ liệu theo cột.
* **Graph Databases**: Lưu trữ dữ liệu dưới dạng đồ thị, với các nút (nodes) và các cạnh (edges) để biểu diễn mối quan hệ phức tạp giữa các đối tượng.

NoSQL đã trở nên phổ biến trong các ứng dụng web, mạng xã hội, thương mại điện tử, và các hệ thống phân tích dữ liệu lớn (big data), nơi mà các cơ sở dữ liệu quan hệ truyền thống khó có thể đáp ứng được yêu cầu về hiệu suất và khả năng mở rộng. MongoDB, Cassandra, Redis, và Couchbase là một số ví dụ phổ biến về các hệ quản trị cơ sở dữ liệu NoSQL.

### Đặc điểm của NoSQL?

Cơ sở dữ liệu NoSQL có những đặc điểm nổi bật giúp chúng phù hợp với các yêu cầu cụ thể của các hệ thống hiện đại, đặc biệt là các hệ thống có quy mô lớn và yêu cầu xử lý dữ liệu phức tạp. Dưới đây là những đặc điểm chính của NoSQL:

1. **Khả năng mở rộng ngang (Horizontal Scalability)**:
   * NoSQL cho phép mở rộng hệ thống bằng cách thêm nhiều máy chủ mới vào mạng lưới (cluster) thay vì chỉ tăng cường năng lực của một máy chủ duy nhất. Điều này giúp cải thiện hiệu suất và khả năng xử lý dữ liệu mà không gây gián đoạn hệ thống.
2. **Linh hoạt trong mô hình dữ liệu (Flexible Data Models)**:
   * NoSQL hỗ trợ nhiều kiểu mô hình dữ liệu như key-value, document, column-family, và graph, cho phép lưu trữ và truy cập dữ liệu mà không cần tuân thủ một cấu trúc cố định như trong cơ sở dữ liệu quan hệ. Điều này giúp dễ dàng quản lý các loại dữ liệu không đồng nhất hoặc có cấu trúc phức tạp.
3. **Hiệu suất cao (High Performance)**:
   * NoSQL được tối ưu hóa để xử lý khối lượng lớn dữ liệu và yêu cầu truy vấn nhanh chóng. Các hệ thống NoSQL thường được thiết kế để hoạt động hiệu quả với các tác vụ đọc/ghi dữ liệu lớn và liên tục.
4. **Khả năng chịu lỗi và tính sẵn sàng cao (Fault Tolerance and High Availability)**:
   * Các cơ sở dữ liệu NoSQL thường được thiết kế để hoạt động liên tục ngay cả khi một hoặc nhiều nút trong hệ thống bị lỗi. Dữ liệu có thể được sao chép và phân phối trên nhiều nút để đảm bảo tính sẵn sàng cao và khả năng phục hồi nhanh chóng.
5. **Không cần cấu trúc dữ liệu cứng nhắc (Schema-less)**:
   * Các hệ thống NoSQL không yêu cầu phải định nghĩa trước cấu trúc dữ liệu (schema) như trong RDBMS. Điều này cho phép dễ dàng thay đổi hoặc mở rộng cấu trúc dữ liệu mà không cần chỉnh sửa hoặc gián đoạn hoạt động của hệ thống.
6. **Hỗ trợ xử lý dữ liệu phân tán (Distributed Data Handling)**:
   * NoSQL hỗ trợ khả năng xử lý dữ liệu trên nhiều máy chủ hoặc vùng địa lý khác nhau, điều này giúp giảm tải cho hệ thống trung tâm và cải thiện khả năng truy cập dữ liệu từ các vị trí khác nhau.
7. **Tương thích với dữ liệu lớn (Big Data Compatibility)**:
   * NoSQL thường được sử dụng trong các ứng dụng cần xử lý và lưu trữ dữ liệu lớn (big data), chẳng hạn như phân tích dữ liệu thời gian thực, lưu trữ dữ liệu từ mạng xã hội, hoặc quản lý các giao dịch thương mại điện tử với quy mô lớn.

Với những đặc điểm trên, NoSQL đã trở thành lựa chọn ưu tiên cho nhiều ứng dụng hiện đại, nơi mà khả năng mở rộng, hiệu suất cao, và tính linh hoạt của mô hình dữ liệu là những yếu tố then chốt.

## Giới thiệu về MongoDB

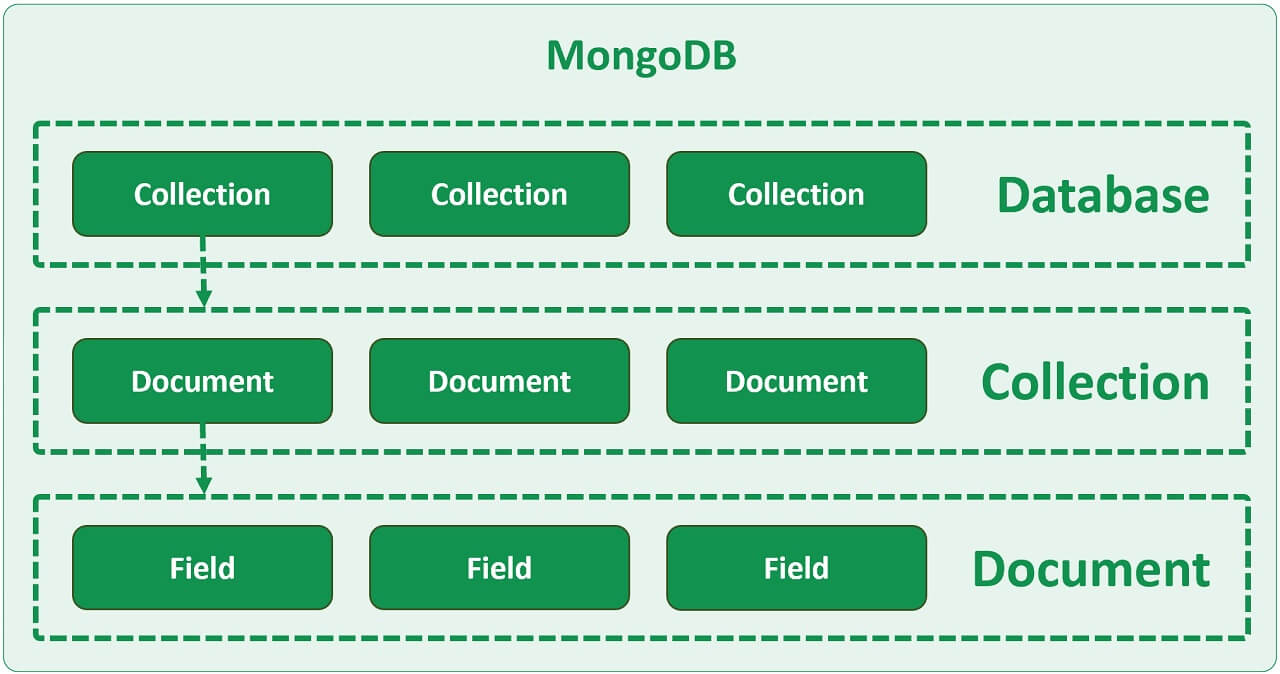
### Khái niệm về MongoDB ?



Hình 2.2: Khái niệm về MongoDB

MongoDB là một hệ quản trị cơ sở dữ liệu NoSQL mã nguồn mở, được phát triển nhằm đáp ứng nhu cầu xử lý và quản lý dữ liệu phi cấu trúc trong các ứng dụng hiện đại. Khác với các hệ quản trị cơ sở dữ liệu quan hệ truyền thống như MySQL, PostgreSQL, nơi dữ liệu được lưu trữ dưới dạng bảng với các dòng và cột, MongoDB lưu trữ dữ liệu dưới dạng tài liệu dưới cấu trúc, hay còn gọi là BSON. Điều này mang lại cho MongoDB khả năng linh hoạt cao trong việc quản lý dữ liệu phi cấu trúc và hỗ trợ tốt cho các ứng dụng cần xử lý lượng lớn dữ liệu không đồng nhất.

### Một số đặc điểm nổi bật của MongoDB



Hình 2.3: Đặc điểm của MongoDB

 **Linh hoạt trong cấu trúc dữ liệu:** MongoDB không yêu cầu một lược đồ cố định (schema-less), cho phép mỗi tài liệu trong cùng một tập hợp (collection) có thể có các trường dữ liệu khác nhau. Điều này rất hữu ích trong việc xử lý các loại dữ liệu đa dạng và thay đổi liên tục, phù hợp với các yêu cầu của các doanh nghiệp hiện đại.

 **Khả năng mở rộng:** MongoDB hỗ trợ phân mảnh dữ liệu (sharding), cho phép mở rộng cơ sở dữ liệu theo chiều ngang bằng cách phân phối dữ liệu trên nhiều máy chủ khác nhau. Điều này đảm bảo rằng hệ thống có thể xử lý tốt các khối lượng dữ liệu lớn và tăng trưởng nhanh chóng.

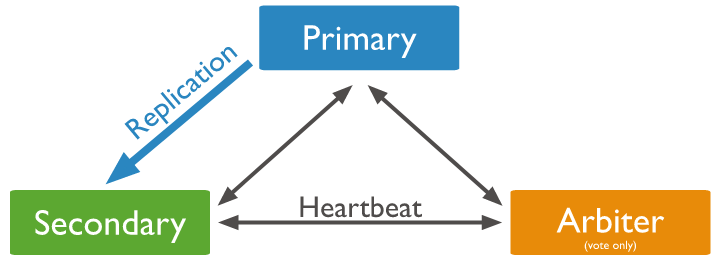
 **Hiệu suất cao:** MongoDB cung cấp các tính năng như chỉ mục (indexing) và tích hợp bộ nhớ đệm (in-memory) giúp tăng tốc độ truy vấn và xử lý dữ liệu. Điều này đặc biệt quan trọng trong các ứng dụng yêu cầu tốc độ truy xuất dữ liệu nhanh.

 **Khả năng tương tác với nhiều ngôn ngữ lập trình:** MongoDB hỗ trợ tích hợp với nhiều ngôn ngữ lập trình phổ biến như Python, Java, PHP, Node.js, giúp các nhà phát triển dễ dàng sử dụng và tích hợp vào các ứng dụng hiện có.

## Kiến trúc và mô hình dữ liệu của MongoDB

### Kiến Trúc của MongoDB

MongoDB được thiết kế với kiến trúc phân tán, cho phép nó mở rộng dễ dàng và xử lý khối lượng dữ liệu lớn. Kiến trúc cơ bản của MongoDB bao gồm các thành phần chính sau:



Hình 2.4: Kiến trúc MongoDB

* **Database (Cơ sở dữ liệu):** Là đơn vị quản lý dữ liệu cao nhất trong MongoDB. Mỗi cơ sở dữ liệu có thể chứa nhiều tập hợp dữ liệu (collections). Một cơ sở dữ liệu có thể được tạo ra khi cần thiết và có thể chứa các tập hợp dữ liệu có liên quan đến nhau.
* **Collection (Tập hợp dữ liệu):** Là đơn vị lưu trữ dữ liệu trong MongoDB, tương tự như bảng trong các hệ quản trị cơ sở dữ liệu quan hệ. Một tập hợp dữ liệu chứa nhiều tài liệu (documents) và không yêu cầu có một lược đồ cố định, cho phép lưu trữ các tài liệu với cấu trúc khác nhau.
* **Document (Tài liệu):** Là đơn vị cơ bản của dữ liệu trong MongoDB, lưu trữ dữ liệu dưới định dạng BSON (Binary JSON). Mỗi tài liệu là một đối tượng JSON với các trường và giá trị tương ứng. Tài liệu có thể chứa các trường dữ liệu cơ bản (chuỗi, số, ngày giờ) cũng như các cấu trúc dữ liệu phức tạp (mảng, tài liệu lồng ghép).
* **Replica Set (Tập hợp bản sao):** Là một nhóm các máy chủ MongoDB cùng quản lý một cơ sở dữ liệu. Replica Set cung cấp tính sẵn sàng cao (high availability) bằng cách sao lưu dữ liệu trên nhiều máy chủ, đảm bảo rằng dữ liệu vẫn có thể truy cập ngay cả khi một hoặc nhiều máy chủ gặp sự cố.
* **Sharding (Phân mảnh dữ liệu):** Là một cơ chế mở rộng theo chiều ngang, cho phép phân phối dữ liệu trên nhiều máy chủ hoặc nút. Sharding giúp MongoDB xử lý khối lượng dữ liệu lớn bằng cách chia nhỏ dữ liệu và lưu trữ chúng trên nhiều máy chủ khác nhau, giảm tải cho các máy chủ đơn lẻ và cải thiện hiệu suất.

### Mô hình dữ liệu của MongoDB

Mô hình dữ liệu của MongoDB được thiết kế để xử lý dữ liệu phi cấu trúc và linh hoạt. Các thành phần chính trong mô hình dữ liệu của MongoDB bao gồm:

* **Tài liệu (Document):** Tài liệu là đơn vị lưu trữ dữ liệu cơ bản trong MongoDB. Tài liệu được biểu diễn dưới dạng JSON hoặc BSON, với cấu trúc dạng khóa-giá trị (key-value). Các trường trong tài liệu có thể chứa dữ liệu đơn giản hoặc các cấu trúc phức tạp như mảng và tài liệu lồng ghép.
* **Tập hợp dữ liệu (Collection):** Tập hợp dữ liệu là một nhóm các tài liệu liên quan, tương tự như bảng trong cơ sở dữ liệu quan hệ. Tập hợp dữ liệu không yêu cầu lược đồ cố định, cho phép lưu trữ các tài liệu với cấu trúc khác nhau trong cùng một tập hợp.
* **Cơ sở dữ liệu (Database):** Một cơ sở dữ liệu chứa nhiều tập hợp dữ liệu. Mỗi cơ sở dữ liệu có thể được phân chia và quản lý độc lập, giúp tổ chức dữ liệu theo các ứng dụng hoặc dự án khác nhau.
* **Lược đồ linh hoạt (Flexible Schema):** MongoDB cho phép lưu trữ các tài liệu với cấu trúc khác nhau trong cùng một tập hợp, giúp dễ dàng thích ứng với các thay đổi trong yêu cầu và cấu trúc dữ liệu mà không cần phải thay đổi toàn bộ lược đồ của cơ sở dữ liệu.

## Ưu điểm của MongoDB trong quản lý tài liệu

MongoDB mang lại nhiều ưu điểm nổi bật khi được sử dụng trong quản lý tài liệu, đặc biệt là trong môi trường của các doanh nghiệp vừa và nhỏ. Dưới đây là các lợi ích chính của MongoDB trong việc quản lý tài liệu:

1. MongoDB lưu trữ dữ liệu dưới dạng tài liệu JSON hoặc BSON, cho phép quản lý dữ liệu phi cấu trúc và bán cấu trúc một cách linh hoạt. Điều này đặc biệt hữu ích khi làm việc với các loại tài liệu có cấu trúc không đồng nhất hoặc thay đổi theo thời gian.
2. MongoDB hỗ trợ phân mảnh dữ liệu (sharding), giúp mở rộng hệ thống theo chiều ngang bằng cách phân phối dữ liệu trên nhiều máy chủ hoặc nút. Điều này cho phép hệ thống quản lý tài liệu có thể xử lý khối lượng dữ liệu lớn và tăng trưởng nhanh chóng mà không ảnh hưởng đến hiệu suất.
3. MongoDB cung cấp các tính năng tối ưu hóa hiệu suất như chỉ mục (indexing) và tích hợp bộ nhớ đệm (in-memory). Điều này giúp cải thiện tốc độ truy vấn và xử lý dữ liệu, đáp ứng nhanh chóng các yêu cầu của người dùng trong quá trình quản lý tài liệu.
4. MongoDB cung cấp các công cụ tìm kiếm và truy xuất dữ liệu mạnh mẽ, bao gồm các khả năng tìm kiếm văn bản toàn văn, tìm kiếm gần đúng, và các truy vấn phức tạp. Điều này giúp người dùng dễ dàng tìm kiếm và truy xuất các tài liệu cần thiết một cách nhanh chóng và chính xác.
5. MongoDB sử dụng cơ chế Replica Set để sao lưu dữ liệu và đảm bảo tính sẵn sàng cao. Các bản sao của cơ sở dữ liệu được duy trì trên nhiều máy chủ, đảm bảo rằng dữ liệu vẫn có thể truy cập ngay cả khi một hoặc nhiều máy chủ gặp sự cố.
6. MongoDB hỗ trợ tích hợp với nhiều ngôn ngữ lập trình và công nghệ khác nhau như Python, Java, Node.js, và các công cụ phân tích dữ liệu. Điều này giúp dễ dàng tích hợp MongoDB vào các ứng dụng quản lý tài liệu hiện có và mở rộng hệ thống khi cần thiết.
7. MongoDB cho phép lưu trữ và quản lý các loại dữ liệu đa dạng như văn bản, hình ảnh, tài liệu có cấu trúc phức tạp, và dữ liệu liên quan. Điều này giúp hệ thống quản lý tài liệu có thể xử lý nhiều loại tài liệu khác nhau mà không cần phải chuyển đổi dữ liệu hoặc thay đổi cấu trúc cơ sở dữ liệu.

## Ưu điểm của MongoDB trong quản lý tài liệu

MongoDB đã chứng minh được giá trị của nó trong nhiều lĩnh vực và ứng dụng khác nhau, đặc biệt là trong các doanh nghiệp vừa và nhỏ (SMEs). Dưới đây là các ứng dụng tiêu biểu của MongoDB trong các doanh nghiệp vừa và nhỏ:

1. MongoDB cung cấp một giải pháp linh hoạt và hiệu quả để quản lý các loại tài liệu phi cấu trúc và bán cấu trúc, chẳng hạn như báo cáo, hợp đồng, tài liệu kỹ thuật, và thông tin khách hàng. Nhờ vào mô hình dữ liệu tài liệu, doanh nghiệp có thể lưu trữ và quản lý các tài liệu với cấu trúc khác nhau trong cùng một tập hợp dữ liệu, dễ dàng truy xuất và xử lý thông tin khi cần.
2. Trong các hệ thống quản lý khách hàng (CRM), MongoDB cho phép lưu trữ và phân tích thông tin khách hàng, lịch sử giao dịch, và các tương tác khác một cách linh hoạt. Khả năng tìm kiếm mạnh mẽ và mô hình dữ liệu linh hoạt giúp doanh nghiệp dễ dàng theo dõi và phân tích dữ liệu khách hàng để cải thiện dịch vụ và phát triển chiến lược kinh doanh.
3. Doanh nghiệp có thể sử dụng MongoDB để quản lý thông tin về sản phẩm, dịch vụ, và kho hàng. Với khả năng mở rộng và lưu trữ dữ liệu đa dạng, MongoDB giúp doanh nghiệp quản lý danh mục sản phẩm, theo dõi mức tồn kho, và xử lý đơn hàng một cách hiệu quả.
4. MongoDB hỗ trợ các công cụ phân tích dữ liệu mạnh mẽ, giúp doanh nghiệp thực hiện các phân tích dữ liệu phức tạp và tạo báo cáo chi tiết. Khả năng lưu trữ và truy xuất dữ liệu nhanh chóng giúp doanh nghiệp nhanh chóng nhận được thông tin quan trọng để ra quyết định và đánh giá hiệu quả kinh doanh.
5. MongoDB dễ dàng tích hợp với nhiều công nghệ và công cụ khác như hệ thống phân tích dữ liệu, công cụ báo cáo, và các nền tảng phát triển ứng dụng. Điều này cho phép doanh nghiệp xây dựng các giải pháp tùy chỉnh và mở rộng hệ thống quản lý tài liệu để đáp ứng các nhu cầu cụ thể của mình.
6. Với khả năng xử lý dữ liệu nhanh chóng và mở rộng dễ dàng, MongoDB là sự lựa chọn lý tưởng cho các ứng dụng web và dịch vụ trực tuyến. Doanh nghiệp có thể xây dựng các ứng dụng web mạnh mẽ để phục vụ khách hàng, quản lý các quy trình kinh doanh, và cung cấp các dịch vụ trực tuyến hiệu quả.
7. Doanh nghiệp có thể sử dụng MongoDB để lưu trữ và phân tích dữ liệu lịch sử, giúp dự đoán xu hướng và xác định các cơ hội kinh doanh. Khả năng phân tích dữ liệu lớn và phức tạp giúp doanh nghiệp đưa ra các quyết định dựa trên thông tin chính xác và kịp thời.

Sự linh hoạt, khả năng mở rộng, và hiệu suất cao của MongoDB làm cho nó trở thành một công cụ hữu ích cho các doanh nghiệp vừa và nhỏ trong việc quản lý dữ liệu và tài liệu.

# PHƯƠNG PHÁP KĨ THUẬT

## Phân tích yêu cầu của hệ thống

Phân tích yêu cầu là bước quan trọng đầu tiên trong quá trình thiết kế hệ thống, giúp xác định các nhu cầu và yêu cầu cụ thể để đảm bảo rằng hệ thống quản lý tài liệu đáp ứng đúng mục tiêu và yêu cầu của người dùng. Dưới đây là các yêu cầu chính của hệ thống:

### Quản lý tài liệu

Hệ thống phải hỗ trợ toàn diện việc lưu trữ, quản lý, và xử lý các loại tài liệu khác nhau. Các yêu cầu cụ thể bao gồm:

* **Lưu trữ tài liệu:** Hệ thống cần lưu trữ tài liệu dưới dạng số hóa trong cơ sở dữ liệu, đảm bảo tính toàn vẹn và an toàn của dữ liệu. Tài liệu có thể bao gồm văn bản, hợp đồng, báo cáo, tài liệu kỹ thuật, và các định dạng khác.
* **Tìm kiếm tài liệu:** Cung cấp chức năng tìm kiếm mạnh mẽ cho phép người dùng tìm tài liệu theo từ khóa, tiêu đề, ngày tạo, hoặc các thuộc tính khác. Hệ thống nên hỗ trợ tìm kiếm nâng cao với khả năng lọc và phân loại kết quả.
* **Phân loại tài liệu:** Hệ thống cần phân loại tài liệu theo các danh mục và thẻ để dễ dàng quản lý và truy xuất. Ví dụ, tài liệu có thể được phân loại theo loại, ngày tháng, hoặc dự án.

### Quản lý người dùng

Các chức năng quản lý người dùng là cần thiết để đảm bảo rằng hệ thống có thể điều phối và kiểm soát quyền truy cập của người dùng:

* **Tạo và quản lý tài khoản:** Hệ thống phải hỗ trợ việc tạo mới, chỉnh sửa, và xóa tài khoản người dùng. Mỗi tài khoản người dùng sẽ có các thông tin cá nhân cơ bản và mật khẩu bảo mật.
* **Phân quyền truy cập:** Hệ thống cần cung cấp cơ chế phân quyền để quản lý quyền truy cập của người dùng đối với các tài liệu và chức năng hệ thống. Ví dụ, người dùng có thể được phân quyền đọc, chỉnh sửa, hoặc xóa tài liệu.

### Tìm kiếm và truy vấn

Chức năng tìm kiếm và truy vấn giúp người dùng dễ dàng tìm ra tài liệu cần thiết:

* **Tìm kiếm cơ bản:** Hệ thống cần cung cấp chức năng tìm kiếm nâng cao cho phép người dùng tìm tài liệu theo nhiều tiêu chí như từ khóa, ngày tháng, loại tài liệu, và thẻ phân loại.
* **Truy vấn linh hoạt:** Cung cấp các khả năng truy vấn linh hoạt để lọc và phân loại tài liệu dựa trên các thuộc tính và tiêu chí cụ thể.

### Giao diện người dùng

Giao diện người dùng là phần tiếp xúc chính của người dùng với hệ thống và cần đáp ứng các yêu cầu sau:

* **Thân thiện và dễ sử dụng:** Thiết kế giao diện người dùng trực quan, dễ sử dụng, và phù hợp với nhu cầu của người dùng. Đảm bảo rằng người dùng có thể dễ dàng thực hiện các chức năng như thêm mới, chỉnh sửa, và xóa tài liệu.
* **Tính năng và tương tác:** Cung cấp các tính năng đầy đủ như giao diện cho việc tạo, quản lý và tìm kiếm tài liệu. Các chức năng cần có bao gồm các biểu mẫu nhập liệu, nút điều hướng, và các thông báo hệ thống.

## Thiết kế hệ thống quản lý tài liệu

### Kiến trúc tổng thể

* **Giao diện người dùng:**
  + Giao diện người dùng được xây dựng bằng HTML, CSS, và JavaScript. Để cải thiện trải nghiệm người dùng và đảm bảo giao diện tương thích trên nhiều thiết bị, các framework như Bootstrap hoặc Tailwind CSS sẽ được sử dụng.
  + Giao diện người dùng cung cấp các tính năng như thêm mới tài liệu, chỉnh sửa, xóa tài liệu, và tìm kiếm tài liệu. Mục tiêu là tạo ra một giao diện trực quan, dễ sử dụng, và dễ duy trì.
* **Phần mềm máy chủ:**
  + Phần mềm máy chủ được phát triển bằng Python, sử dụng Django. Django cung cấp một khung làm việc mạnh mẽ với nhiều tính năng tích hợp sẵn cho các ứng dụng lớn hơn.
  + Backend xử lý các yêu cầu từ giao diện người dùng, thực hiện các thao tác như lưu trữ, truy xuất, cập nhật và xóa dữ liệu từ cơ sở dữ liệu MongoDB.
* **Cơ sở dữ liệu:**
  + MongoDB được chọn làm cơ sở dữ liệu chính để lưu trữ và quản lý các tài liệu và dữ liệu liên quan. MongoDB cung cấp khả năng mở rộng và hiệu suất cao nhờ cấu trúc dữ liệu dạng document-oriented.
  + MongoDB lưu trữ dữ liệu dưới dạng các tập hợp collections và tài liệu documents. Tập hợp dữ liệu sẽ chứa các tài liệu theo các mô hình dữ liệu được thiết kế, giúp tổ chức và quản lý dữ liệu một cách hiệu quả và linh hoạt.

### Mô hình dữ liệu và cấu trúc tài liệu

Mô hình dữ liệu của hệ thống được thiết kế để tổ chức và quản lý dữ liệu một cách hiệu quả. Các tập hợp dữ liệu chính bao gồm:

#### Collection contracts (Hợp đồng)

* **Mô tả:** Lưu trữ thông tin về các hợp đồng của doanh nghiệp.
* **Các trường chính:**
  + \_id: ObjectId tự động được tạo bởi MongoDB.
  + id: Mã định danh duy nhất cho hợp đồng.
  + title: Tiêu đề của hợp đồng.
  + start\_date: Ngày bắt đầu của hợp đồng.
  + end\_date: Ngày kết thúc của hợp đồng.
  + document\_id: Mã định danh của tài liệu liên quan.
  + responsible\_user\_id: Mã định danh của người chịu trách nhiệm.
  + file\_url: Đường dẫn tới tệp tin hợp đồng.

#### Collection: counters (Bộ đếm)

* **Mô tả:** Theo dõi giá trị của các bộ đếm tự động cho các loại tài liệu khác nhau.
* **Các trường chính:**
  + \_id: ObjectId tự động được tạo bởi MongoDB.
  + sequence\_value: Giá trị hiện tại của bộ đếm cho loại tài liệu.

#### Collection: documents (Tài liệu)

* **Mô tả:** Lưu trữ các tài liệu chính trong hệ thống.
* **Các trường chính:**
  + \_id: ObjectId tự động được tạo bởi MongoDB.
  + title: Tiêu đề của tài liệu.
  + date: Ngày tạo hoặc ngày của tài liệu.
  + file\_url: Đường dẫn tới tệp tin tài liệu.
  + description: Mô tả về tài liệu.

#### Collection: incoming\_documents (Tài liệu đến)

* **Mô tả:** Lưu trữ thông tin về các tài liệu đến doanh nghiệp.
* **Các trường chính:**
  + \_id: ObjectId tự động được tạo bởi MongoDB.
  + id: Mã định danh duy nhất cho tài liệu đến.
  + title: Tiêu đề của tài liệu.
  + file\_url: Đường dẫn tới tệp tin tài liệu.
  + date\_received: Ngày nhận tài liệu.
  + sender: Người gửi tài liệu.

#### Collection: outgoing\_documents (Tài liệu đi)

* **Mô tả:** Lưu trữ thông tin về các tài liệu đi từ doanh nghiệp.
* **Các trường chính:**
  + \_id: ObjectId tự động được tạo bởi MongoDB.
  + id: Mã định danh duy nhất cho tài liệu đi.
  + title: Tiêu đề của tài liệu.
  + file\_url: Đường dẫn tới tệp tin tài liệu.
  + date\_sent: Ngày gửi tài liệu.
  + receiver: Người nhận tài liệu.

#### Collection: users (Người dùng)

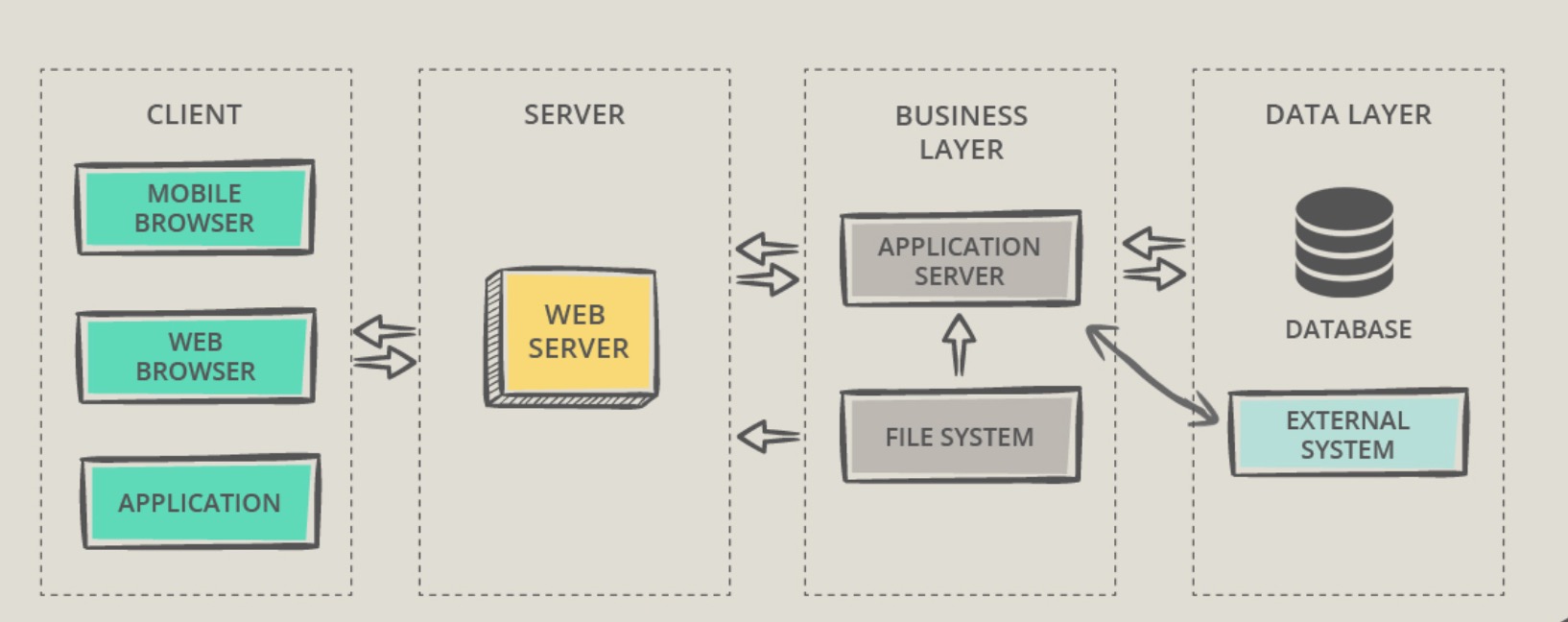
* **Mô tả:** Lưu trữ thông tin về người dùng trong hệ thống.
* **Các trường chính:**
  + \_id: ObjectId tự động được tạo bởi MongoDB.
  + username: Tên đăng nhập của người dùng.
  + email: Địa chỉ email của người dùng.
  + password: Mật khẩu của người dùng (đã mã hóa).
  + role: Vai trò của người dùng trong hệ thống.
  + status: Trạng thái của tài khoản người dùng.
  + id: Mã định danh duy nhất cho người dùng.

## Triển khai hệ thống

### Môi trường và công cụ phát triển

Trong quá trình phát triển hệ thống quản lý tài liệu, việc lựa chọn môi trường và công cụ phát triển phù hợp là rất quan trọng. Các công cụ và môi trường phát triển chính được sử dụng bao gồm:

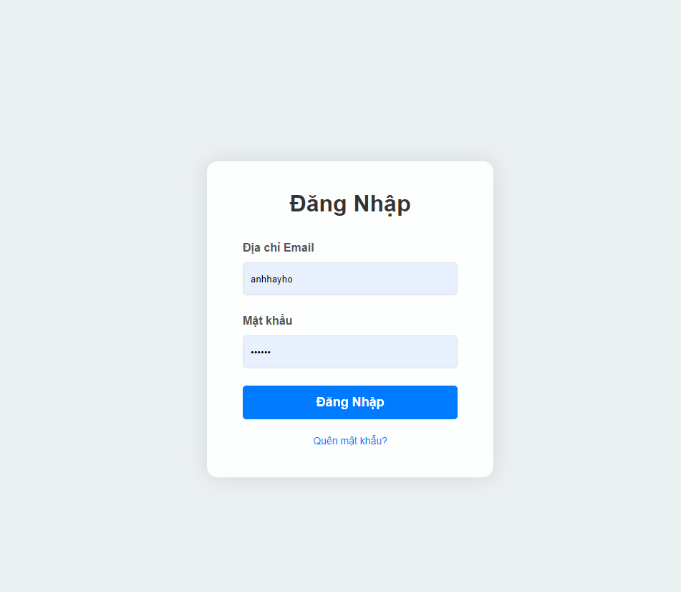
* **Hệ điều hành:** Windows, hệ điều hành phổ biến.
* **Ngôn ngữ lập trình:** Python 3.8 là ngôn ngữ chính được sử dụng, kết hợp với framework Flask hoặc Django để xây dựng phần mềm máy chủ.
* **Cơ sở dữ liệu:** MongoDB được sử dụng để lưu trữ dữ liệu. MongoDB là cơ sở dữ liệu NoSQL mạnh mẽ, hỗ trợ lưu trữ dữ liệu dưới dạng JSON và có khả năng mở rộng tốt.
* **Công cụ quản lý mã nguồn:**  Git được sử dụng để quản lý mã nguồn, kết hợp với GitHub để lưu trữ và chia sẻ mã nguồn với các thành viên trong nhóm.
* **Trình soạn thảo mã nguồn:**  Visual Studio Code (VSCode) là trình soạn thảo mã nguồn chính được sử dụng, với các plugin hỗ trợ như Python, MongoDB, và Git.
* **Trình Duyệt Web (Web Browser):** Google Chrome được sử dụng để kiểm tra và thử nghiệm giao diện người dùng.



Hình 3.1: Môi trường phát triển hệ thống

# GIỚI THIỆU CHƯƠNG TRÌNH

## Giao diện trang đăng nhập



Hình 4.1: Trang đăng nhập

Hình ảnh minh họa giao diện đăng nhập trực tuyến với thiết kế đơn giản và tinh tế, sử dụng ngôn ngữ tiếng Việt. Phần tiêu đề nổi bật "Đăng Nhập" ngay phía trên, hướng dẫn người dùng bước vào hệ thống.

Giao diện bao gồm hai trường thông tin quan trọng:

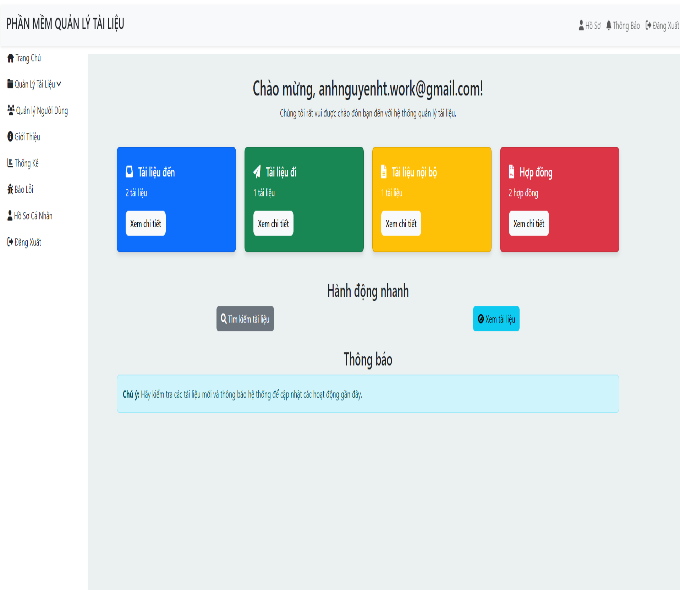
* Trường "Địa chỉ Email" dành cho người dùng nhập địa chỉ email cá nhân.
* Trường "Mật khẩu" yêu cầu người dùng cung cấp thông tin mật khẩu để xác minh danh tính.

Bên dưới là nút "Đăng Nhập" màu xanh dương đậm, dễ nhìn và thu hút, mời gọi người dùng thực hiện thao tác để tiến vào hệ thống.

Ngoài ra, một liên kết nhỏ "Quên mật khẩu?" được bố trí tinh tế phía dưới, giúp người dùng dễ dàng khôi phục tài khoản nếu gặp khó khăn trong việc nhớ mật khẩu. Toàn bộ giao diện mang phong cách trực quan, đơn giản nhưng hiệu quả, phù hợp với người dùng ở mọi trình độ kỹ thuật.Top of Form

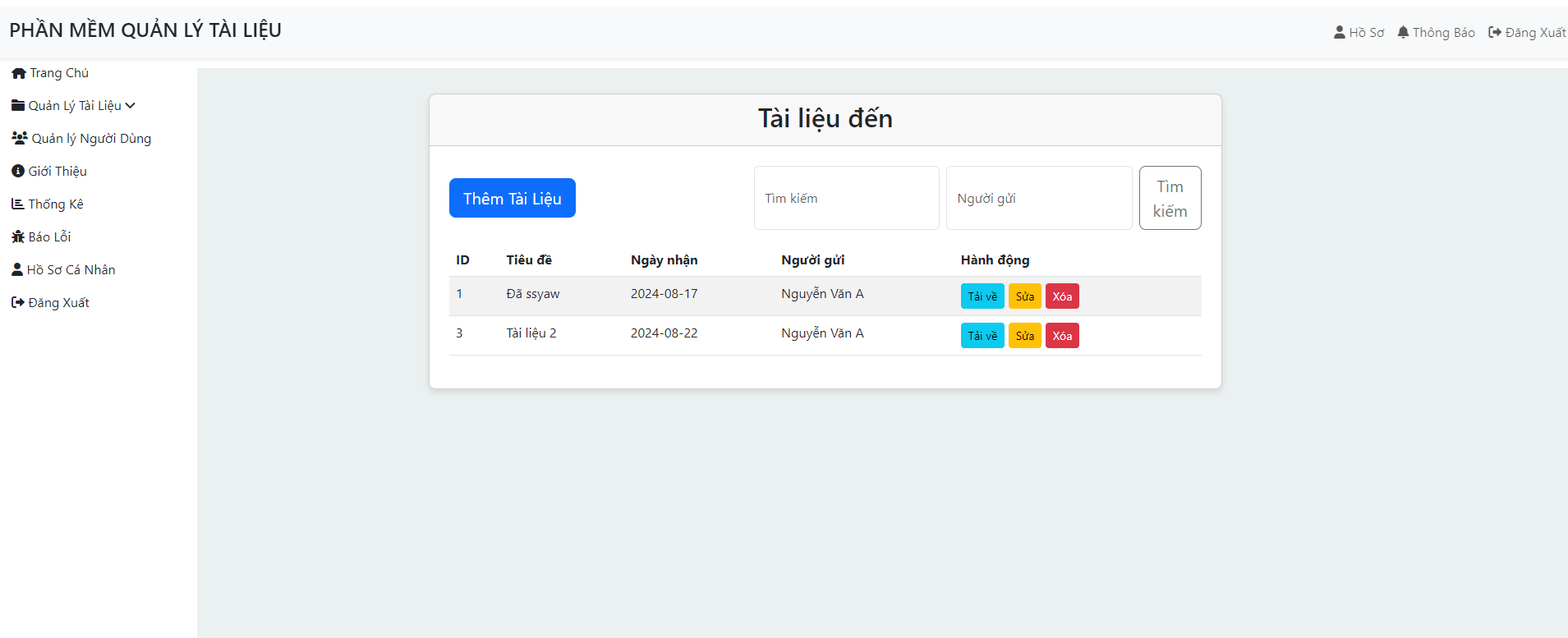
Bottom of Form

### Trang đăng nhập



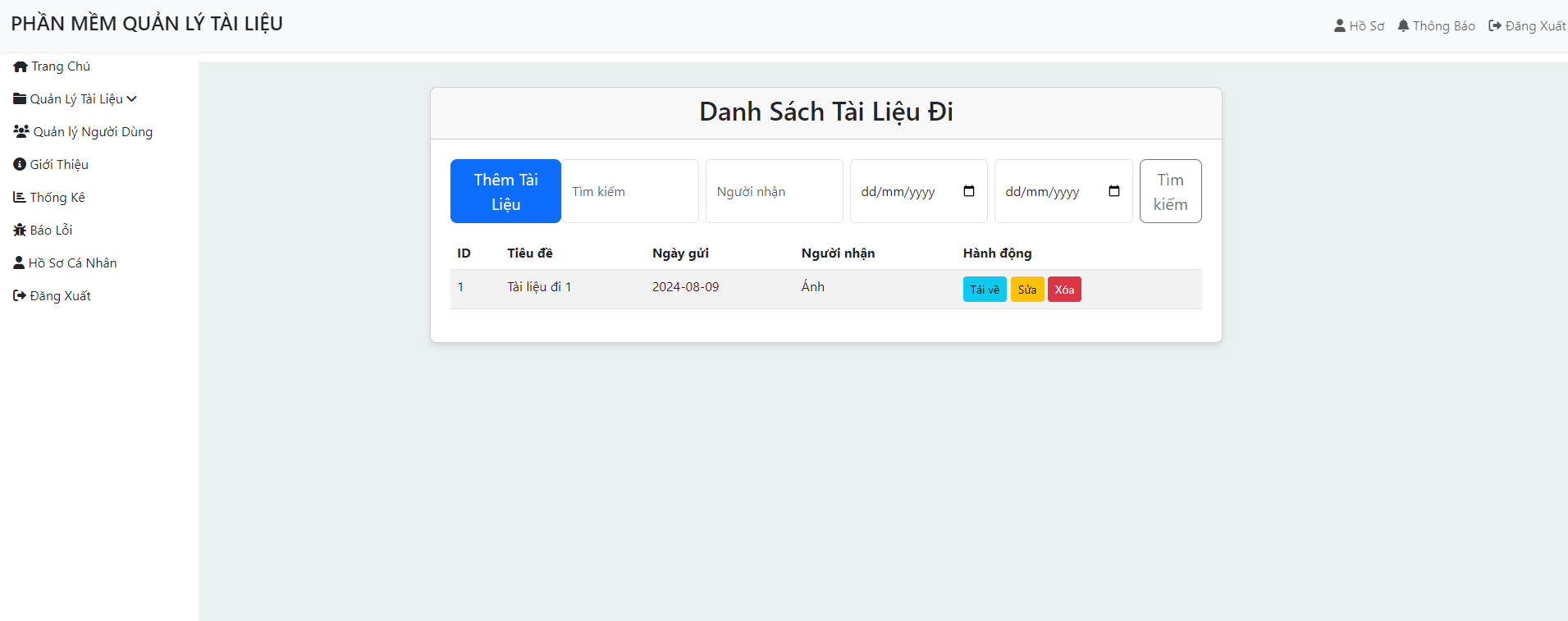
Hình 4.2: Trang dashboard

### Trang quản lý tài liệu đến



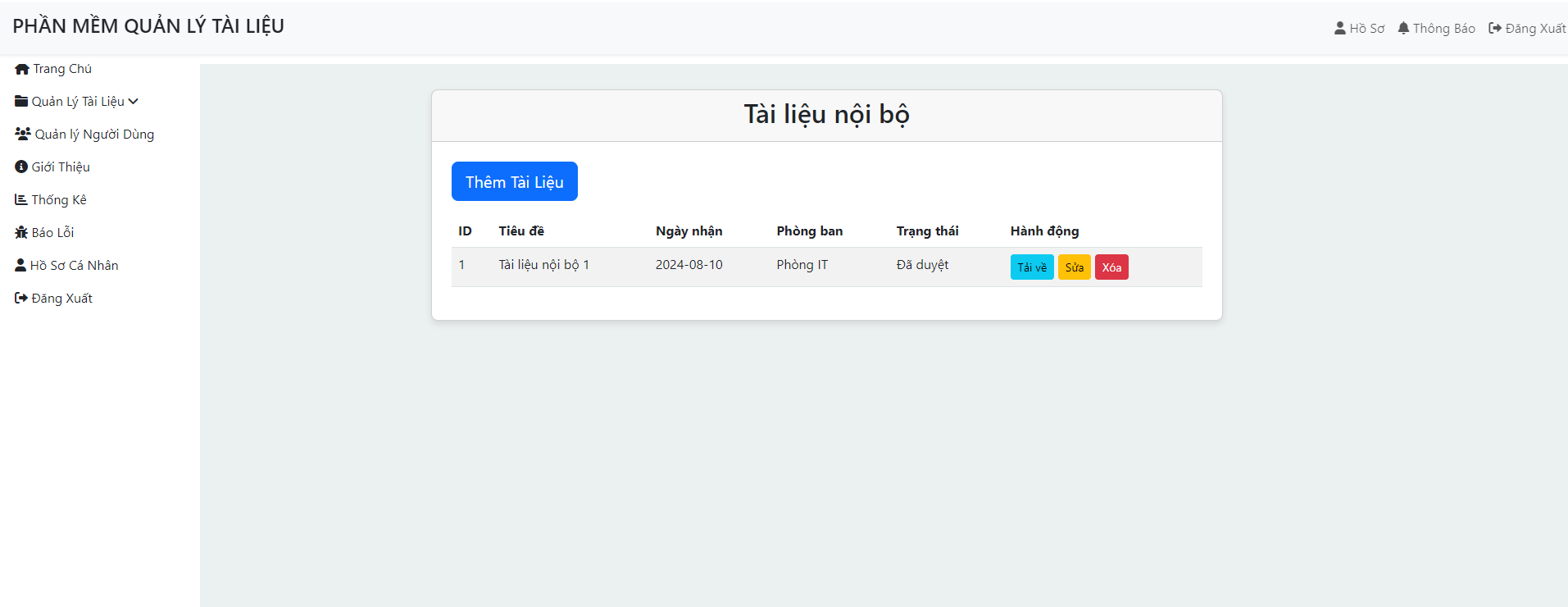
Hình 4.3: Trang quản lý tài liệu đến

### Trang quản lý tài liệu đi



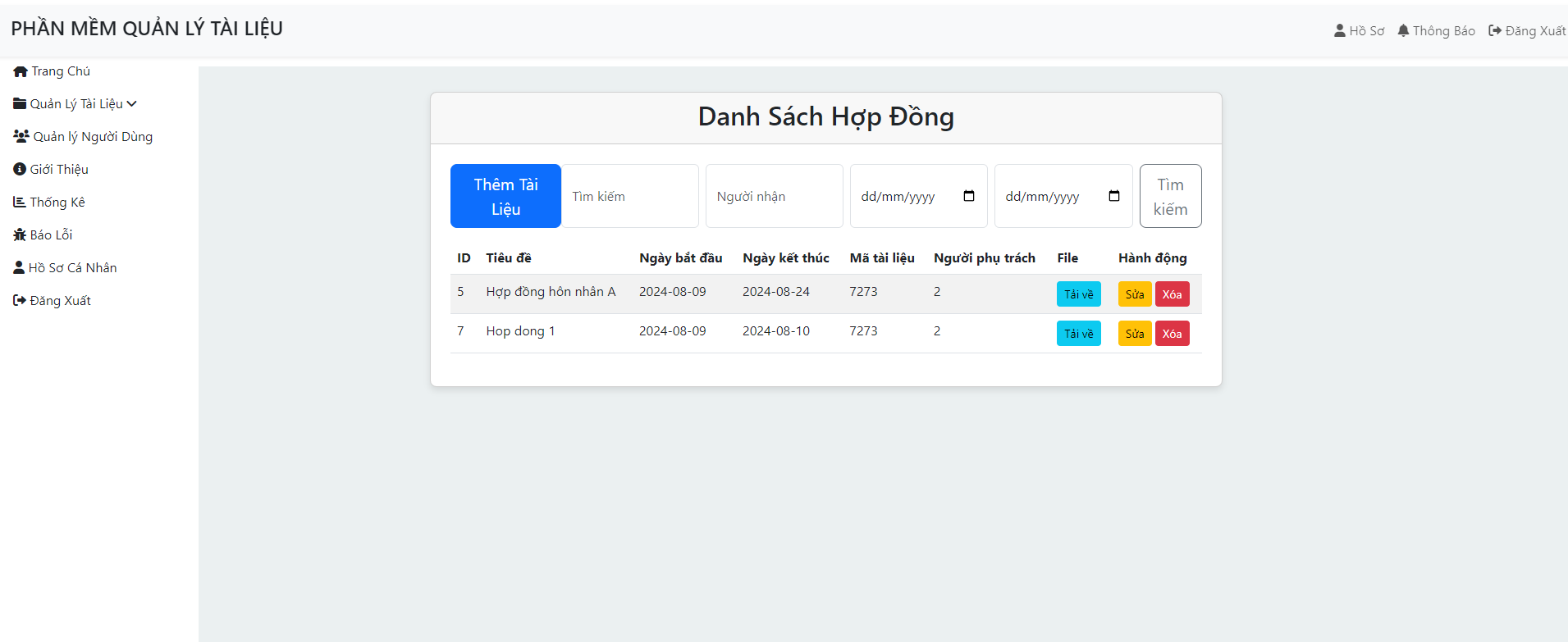
Hình 4.4: Trang quản lý tài liệu đi

### Trang quản lý tài liệu nội bộ



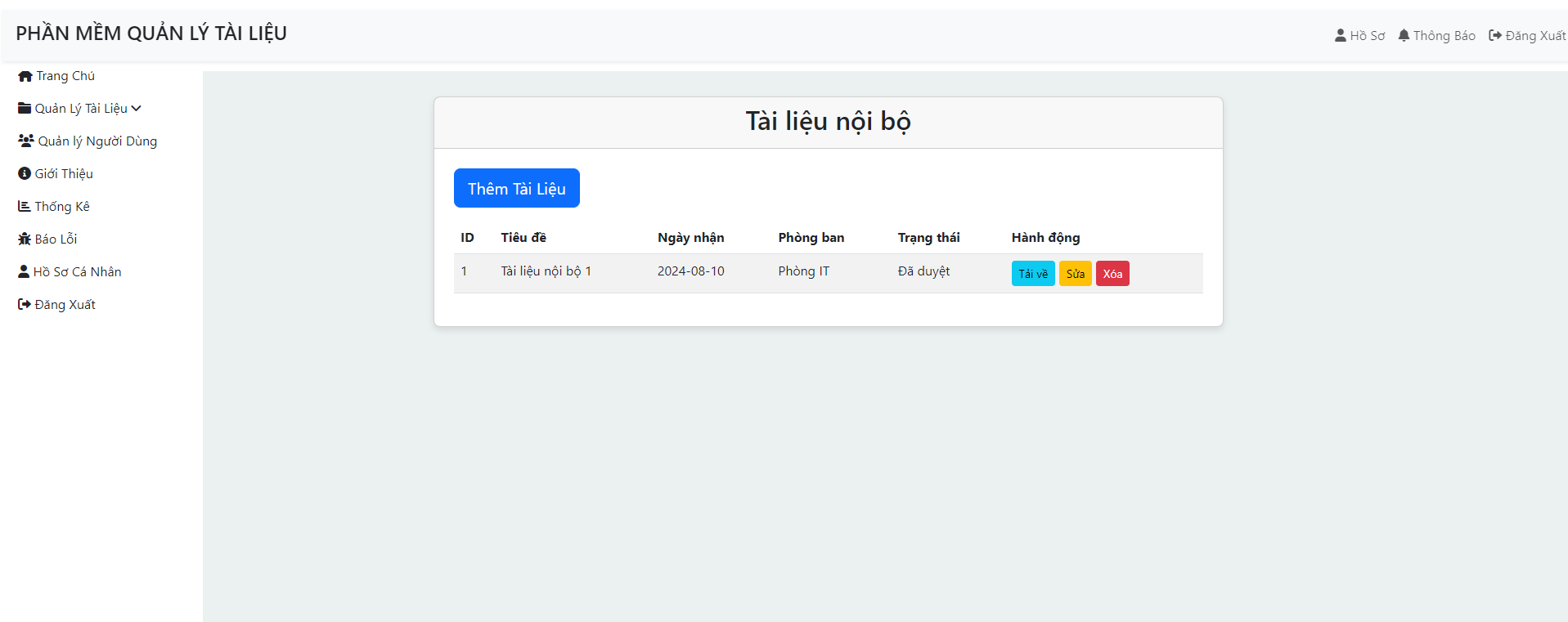
Hình 4.5: Trang quản lý tài liệu nội bộ

### Trang quản lý hợp đồng



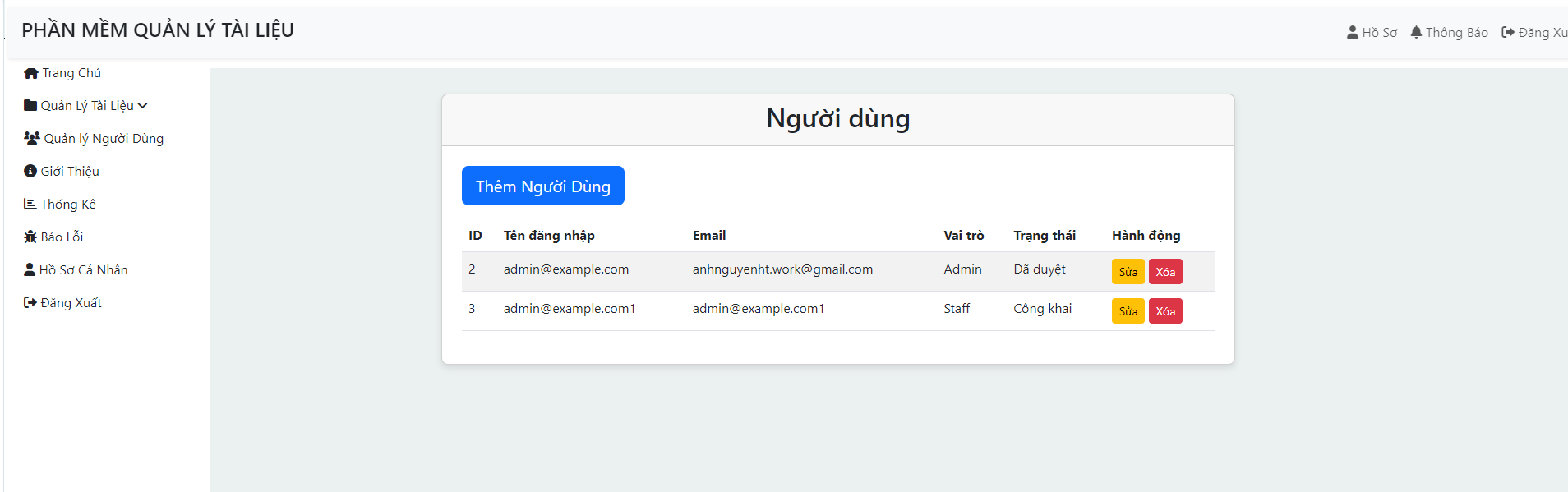
Hình 4.6: Trang quản lý hợp đồng

### Trang quản lý tài liệu nội bộ



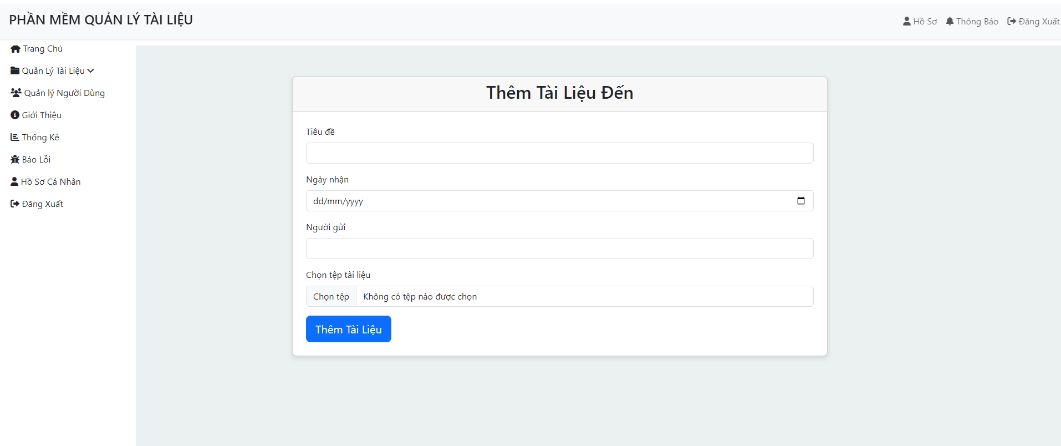
Hình 4.7: Trang quản lý tài liệu nội bộ

### Trang quản lý người dùng



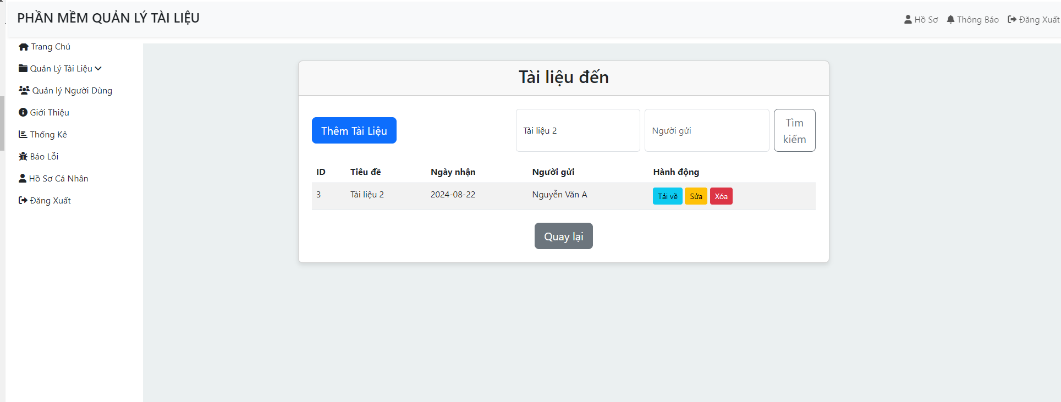
Hình 4.8: Trang quản lý người dùng

### Trang thêm tài liệu



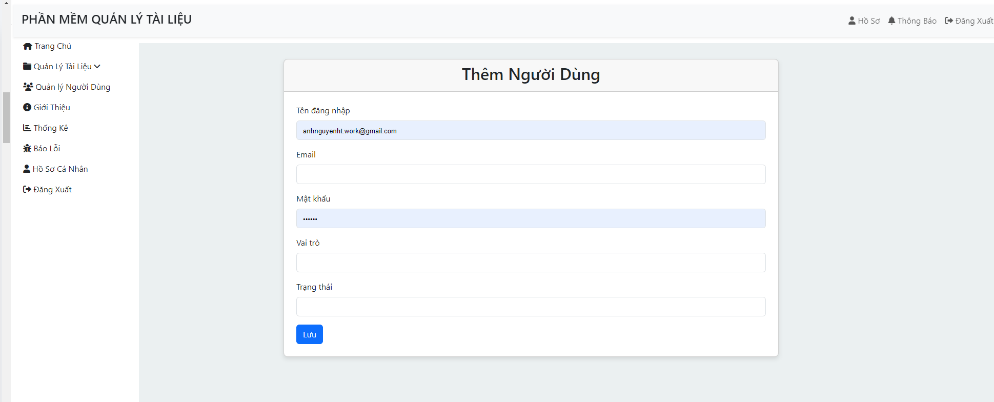
Hình 4.9: Trang thêm tài liệu

### Trang tìm kiếm



Hình 4.10: Trang tìm kiếm

### Trang thêm người dùng



Hình 4.11: Trang thêm người dùng

# KẾT LUẬN

Bài tiểu luận này đã trình bày quá trình xây dựng và triển khai hệ thống quản lý tài liệu cho doanh nghiệp vừa và nhỏ, sử dụng công nghệ Django và MongoDB. Hệ thống được triển khai trên nền tảng web, cho phép người dùng quản lý tài liệu đơn giản.

**Những gì đã làm được:**

* **Xây dựng hệ thống:** Hệ thống đã được xây dựng và triển khai thành công trên web với Django, sử dụng MongoDB để lưu trữ dữ liệu. Cơ sở dữ liệu bao gồm 7 bảng (collections), đáp ứng đầy đủ các yêu cầu cơ bản về quản lý tài liệu, bao gồm lưu trữ, tìm kiếm, phân loại và quản lý các loại tài liệu khác nhau.
* **Tính năng quản lý tài liệu:** Các chức năng cơ bản như tạo, chỉnh sửa, xóa tài liệu, quản lý người dùng và phân quyền đã được tích hợp và hoạt động ổn định. Hệ thống cũng hỗ trợ tìm kiếm và truy vấn tài liệu hiệu quả.

**Những gì chưa làm được:**

* **Tính năng nâng cao:** Một số tính năng nâng cao như tích hợp hệ thống với các nền tảng khác hoặc hỗ trợ mở rộng trong tương lai chưa được triển khai.
* **Kiểm thử toàn diện:** Mặc dù hệ thống đã hoạt động tốt với các chức năng cơ bản, nhưng vẫn cần thực hiện thêm các bước kiểm thử toàn diện để đảm bảo tính ổn định và bảo mật ở mức độ cao hơn.

Tổng kết lại, hệ thống đã đạt được các mục tiêu cơ bản đề ra ban đầu, tạo nền tảng vững chắc cho việc mở rộng và nâng cấp trong tương lai. Những kinh nghiệm và kết quả đạt được từ quá trình này sẽ là nền tảng quý báu cho các dự án tiếp theo.

# TÀI LIỆU THAM KHẢO

1. MongoDB, Inc. (2024). *MongoDB Documentation*. Truy cập từ <https://www.mongodb.com/docs/>.
2. Django Software Foundation. (2024). *Django Documentation*. Truy cập từ <https://docs.djangoproject.com/>.
3. Robson, M. (2021). *Django for Beginners: Build Websites with Python and Django*. Independently published.
4. Chodorow, K. (2019). *MongoDB: The Definitive Guide* (3rd ed.). O'Reilly Media.
5. Welling, L., & Thomson, L. (2021). *PHP and MySQL Web Development* (5th ed.). Addison-Wesley.
6. Hartl, M. (2022). *Ruby on Rails Tutorial: Learn Web Development with Rails* (7th ed.). Addison-Wesley Professional.
7. Zandstra, M. (2022). *PHP Objects, Patterns, and Practice* (6th ed.). Apress.
8. Nguyen, T. A., & Tran, V. D. (2023). *Phát triển hệ thống quản lý tài liệu cho doanh nghiệp vừa và nhỏ*. Tạp chí Công Nghệ Thông Tin Việt Nam, 12(4), 101-115.
9. Watts, A. (2020). *Learn Web Development with Python: Get Hands-on with HTML, CSS, and JavaScript*. Packt Publishing.
10. Freeman, E., & Robson, E. (2018). *Head First JavaScript Programming*. O'Reilly Media.
11. Nguyen, T. A., & Pham, M. Q. (2022). *Phát triển hệ thống quản lý tài liệu bằng Django và MongoDB*. Kỷ yếu Hội nghị Khoa học và Công nghệ, Trường Đại học Bách Khoa Hà Nội.