

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH  
HỌC KỲ I, NĂM HỌC 2024-2025

**THIẾT KẾ VÀ CÀI ĐẶT CƠ SỞ DỮ LIỆU  
CHO VIỆC KHẢO SÁT MỨC ĐỘ HÀI LÒNG  
CỦA SINH VIÊN TRONG VIỆC  
KHẢO SÁT SINH VIÊN CUỐI KHÓA  
BẰNG NOSQL**

*Giảng viên hướng dẫn:*  
ThS. Phan Thị Phương Nam

*Sinh viên thực hiện:*  
Họ tên: Nguyễn Hoài An  
MSSV: 110122029  
Lớp: DA22TTC

*Trà Vinh, tháng 12 năm 2024*

KHOA KỸ THUẬT VÀ CÔNG NGHỆ  
BỘ MÔN CÔNG NGHỆ THÔNG TIN



THỰC TẬP ĐỒ ÁN CƠ SỞ NGÀNH  
HỌC KỲ I, NĂM HỌC 2024-2025

**THIẾT KẾ VÀ CÀI ĐẶT CƠ SỞ DỮ LIỆU  
CHO VIỆC KHẢO SÁT MỨC ĐỘ HÀI LÒNG  
CỦA SINH VIÊN TRONG VIỆC  
KHẢO SÁT SINH VIÊN CUỐI KHÓA  
BẰNG NOSQL**

*Giảng viên hướng dẫn:*  
ThS. Phan Thị Phương Nam

*Sinh viên thực hiện:*  
Họ tên: Nguyễn Hoài An  
MSSV: 110122029  
Lớp: DA22TTC

*Trà Vinh, tháng 12 năm 2024*

## This image shows a full page of white paper with horizontal dotted lines. The lines are evenly spaced and run across the width of the page, providing a guide for writing. There are no margins, text, or other markings on the paper.

**Giáo viên hướng dẫn**  
(Ký tên và ghi rõ họ tên)

[illegible]

**Thành viên hội đồng**  
(Ký tên và ghi rõ họ tên)

## LỜI CẢM ƠN

Em muốn cảm ơn sâu sắc cho sự giúp đỡ và hướng dẫn, mà em đã nhận được từ giảng viên hướng dẫn trong quá trình hoàn thành đồ án cơ sở ngành trong lĩnh vực Công nghệ Thông tin.

Thời gian qua, em đã học được nhiều kinh nghiệm quý giá và kiến thức từ giảng viên hướng dẫn, và em rất biết ơn sự hỗ trợ và động viên mà giảng viên hướng dẫn đã cung cấp cho em trong suốt quá trình nghiên cứu.

Thông qua báo cáo này, em xin gửi lời cảm ơn đến cô Phan Thị Phương Nam – giảng viên bộ môn CNTT đã trang bị cho em những kiến thức quý báu thông qua đồ án cơ sở ngành với đề tài “Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL” tạo cơ sở để em thực hiện hoàn thành đồ án.

Trong quá trình nghiên cứu đề tài, do kiến thức còn hạn chế nên em vẫn còn nhiều thiếu sót trong quá trình tìm hiểu, đánh giá và trình bày đề tài. Em rất mong nhận được những đóng góp ý kiến của cô để em khắc phục và phát triển để có thể từng bước hoàn thiện hơn.

Em xin chân thành cảm ơn !

## MỤC LỤC

NHẬN XÉT CỦA GIẢNG VIÊN HƯỚNG DẪN .....	i
NHẬN XÉT CỦA THÀNH VIÊN HỘI ĐỒNG.....	ii
LỜI CẢM ƠN .....	iii
MỤC LỤC.....	iv
DANH MỤC HÌNH ẢNH .....	vi
DANH MỤC BẢNG BIỂU .....	vii
DANH MỤC TỪ VIẾT TẮT.....	viii
TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH .....	1
MỞ ĐẦU.....	3
CHƯƠNG 1 TỔNG QUAN.....	5
1.1 Lý do cho thiết kế và cài đặt cơ sở dữ liệu NoSQL cho khảo sát sự hài lòng của sinh viên cuối khóa .....	5
1.2 Những vấn đề cần giải quyết.....	5
1.3 Các nội dung nghiên cứu chính.....	6
CHƯƠNG 2 NGHIÊN CỨU LÝ THUYẾT .....	7
2.1 Quy trình nghiệp vụ cho việc thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát.....	7
2.2 Giới thiệu về NoSQL.....	8
2.2.1 Giới thiệu sơ lược về NoSQL.....	8
2.2.2 Một số khái niệm cơ bản trong NoSQL.....	9
2.2.3 Kiến trúc của NoSQL .....	10
2.2.4 Tính chất của NoSQL .....	11
2.2.5 Đặc tính của NoSQL.....	11
2.2.6 Chức năng của NoSQL.....	12
2.2.7 Các dạng cơ sở dữ liệu NoSQL .....	12
2.2.8 Một số thuật ngữ tương đương trong Relational Database và NoSQL database.....	14
2.3 Tổng quan về MongoDB.....	15
2.3.1 Tổng quan về MongoDB .....	15
2.3.2 Các lợi thế của MongoDB so với RDBMS .....	16
2.3.3 Một số khái niệm cơ bản trong MongoDB.....	16

2.3.4 Các kiểu dữ liệu trong MongoDB .....	18
2.3.5 Các thao tác dữ liệu trên MongoDB .....	20
2.3.6 Làm việc với Collection .....	21
2.3.7 Làm việc với Document .....	24
2.3.8 Mệnh đề Where trong truy vấn MongoDB (tương đương mệnh đề WHERE trong RDBMS) .....	27
2.3.9 Cập nhật Document .....	31
2.3.10 Xóa document trong MongoDB .....	32
2.3.11 Tạo backup và resrote trong MongoDB .....	33
2.3.12 Phục hồi dữ liệu trong MongoDB .....	35
CHƯƠNG 3 HIỆN THỰC HÓA NGHIÊN CỨU .....	36
3.1 Phân tích yêu cầu hệ thống .....	36
3.1.1 Yêu cầu chức năng.....	36
3.1.2 Yêu cầu phi chức năng.....	36
3.2 Công cụ được sử dụng .....	36
3.3 Thiết kế cơ sở dữ liệu .....	36
3.3.1 Cấu trúc cơ sở dữ liệu .....	36
3.3.2 Chèn dữ liệu mẫu .....	38
CHƯƠNG 4 KẾT QUẢ NGHIÊN CỨU .....	41
4.1 Kết quả đạt được.....	41
4.2 Các câu lệnh truy vấn .....	41
CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN .....	53
5.1 Kết luận .....	53
5.1.1 Kết luận chung .....	53
5.1.2 Hạn chế .....	53
5.2 Hướng phát triển.....	53
DANH MỤC TÀI LIỆU THAM KHẢO .....	55
PHỤ LỤC.....	56

## DANH MỤC HÌNH ẢNH

Hình 2.1 Cơ sở dữ liệu NoSQL.....	8
Hình 2.2 Hình ảnh về MongoDB .....	15
Hình 2.3 RDBMS so với MongoDB .....	16
Hình 2.4 Minh họa Document trong Collection .....	17
Hình 1 Mẫu phiếu khảo sát .....	56
Hình 2. Hướng dẫn cài đặt MongoDB .....	57
Hình 3. Hướng dẫn cài đặt MongoDB .....	57
Hình 4. Hướng dẫn cài đặt MongoDB .....	58
Hình 5. Hướng dẫn cài đặt MongoDB .....	59
Hình 6. Hướng dẫn cài đặt MongoDB .....	60
Hình 7. Hướng dẫn cài đặt MongoDB .....	61
Hình 8. Hướng dẫn cài đặt MongoDB .....	61
Hình 9. Hướng dẫn cài đặt MongoDB .....	62
Hình 10. Hướng dẫn cài đặt MongoDB .....	62



## **DANH MỤC BẢNG BIỂU**

Bảng 2.1 Bảng so sánh các thuật ngữ tương đương của RDBMS với NoSQL .....	10
Bảng 2.2 Một số thuật ngữ tương đương trong Relational Database và NoSQL database .....	15
Bảng 2.3 Mối quan hệ giữa các thuật ngữ của RDBMS và MongoDB .....	18
Bảng 2.4 Tham số options của Collection .....	21
Bảng 2.5 Danh sách các tùy chọn options của Collection .....	22
Bảng 2.6 Các phép toán trong điều kiện của phép chọn .....	28
Bảng 2.7 Các lệnh tùy chọn trong lệnh Backup.....	34

## DANH MỤC TỪ VIẾT TẮT

Từ viết tắt	Diễn giải đầy đủ
ACID	Atomicity: Tính nguyên tử Consistency: Tính nhất quán Isolation: Tính cô lập Durability: Tính bền vững
API	Application Programming Interface
BSON	Binary JSON (JSON nhị phân)
CMS	Content Management System
CSDL	Cơ sở dữ liệu
HTML	HyperText Markup Language
JSON	JavaScript Object Notation
NoREL	No Relational
NoSQL	Not Only SQL Structured Query Language
RDBMS	Relational Database Management System
UNIX	UNiplexed Information and Computing System
UTF-8	Unicode Transformation Format - 8-bit
XML	eXtensible Markup Language

## TÓM TẮT ĐỒ ÁN CƠ SỞ NGÀNH

### 1) Mục tiêu của đề tài

- Xây dựng cơ sở dữ liệu NoSQL: Thiết kế và triển khai cơ sở dữ liệu phù hợp để lưu trữ thông tin khảo sát mức độ hài lòng của sinh viên.
- Đảm bảo tính hiệu quả: Đáp ứng yêu cầu truy xuất dữ liệu nhanh chóng, linh hoạt và dễ mở rộng.
- Phân tích dữ liệu: Cung cấp khả năng phân tích kết quả khảo sát nhằm đưa ra các đánh giá và cải thiện chất lượng đào tạo.

### 2) Phạm vi nghiên cứu

- Đối tượng khảo sát: Sinh viên cuối khóa tại Trường đại học Trà Vinh.
- Nội dung khảo sát: Tập trung vào mức độ hài lòng với các yếu tố như chất lượng giảng dạy, cơ sở vật chất, hỗ trợ học tập.
- Công cụ sử dụng: Sử dụng cơ sở dữ liệu NoSQL và hệ quản trị cơ sở dữ liệu MongoDB để lưu trữ dữ liệu.

### 3) Thiết kế cơ sở dữ liệu

#### Mô hình cơ sở dữ liệu:

- Document-based database: Lựa chọn MongoDB để lưu trữ thông tin khảo sát dưới dạng các tài liệu JSON.
- Cấu trúc dữ liệu:
  - o Thông tin sinh viên (ID, họ tên, lớp, khoa).
  - o Kết quả khảo sát (câu hỏi, câu trả lời, mức độ hài lòng).
  - o Thời gian và trạng thái khảo sát.

#### Luồng hoạt động:

- Nhập dữ liệu sinh viên.
- Tạo và quản lý các bộ câu hỏi khảo sát.
- Thu thập và lưu trữ kết quả từ sinh viên.
- Truy xuất và phân tích dữ liệu báo cáo.

#### 4) **Kiểm tra và triển khai**

- Kiểm tra chức năng: Đảm bảo tất cả các tính năng như lưu trữ, truy xuất, và phân tích dữ liệu hoạt động chính xác.
- Kiểm tra hiệu năng: Đánh giá tốc độ truy xuất dữ liệu khi số lượng câu hỏi tăng.
- Kiểm tra bảo mật: Đảm bảo thông tin sinh viên và kết quả khảo sát được bảo mật, tránh rò rỉ dữ liệu.
- Kiểm tra tính mở rộng: Thử nghiệm khả năng mở rộng khi hệ thống cần tích hợp thêm tính năng hoặc tăng lượng dữ liệu.
- Thực hiện

#### 5) **Ý nghĩa của đề tài**

- Đối với nhà trường: Hỗ trợ đánh giá chất lượng giảng dạy và cơ sở vật chất, từ đó cải thiện môi trường học tập cho sinh viên.
- Đối với sinh viên: Tạo cơ hội để phản hồi và đóng góp ý kiến về chương trình đào tạo.
- Đối với nghiên cứu: Đề tài góp phần khám phá tiềm năng của NoSQL trong lĩnh vực giáo dục, mở ra hướng đi mới trong việc quản lý dữ liệu phi cấu trúc.

## MỞ ĐẦU

### 1. Mô tả:

Đề tài tập trung vào việc xây dựng một cơ sở dữ liệu hiệu quả và linh hoạt nhằm thu thập, lưu trữ và phân tích dữ liệu từ các cuộc khảo sát đánh giá mức độ hài lòng của sinh viên về quá trình học tập cuối khóa. Thay vì sử dụng các cơ sở dữ liệu quan hệ truyền thống, đề tài này sẽ ứng dụng cơ sở dữ liệu NoSQL để khai thác tối đa các ưu điểm về khả năng mở rộng, linh hoạt trong cấu trúc dữ liệu và khả năng xử lý dữ liệu lớn. Việc thiết kế cơ sở dữ liệu cho việc khảo sát trực tuyến sẽ mang lại nhiều lợi ích và cấp thiết như:

#### 1) Tính linh hoạt và mở rộng của NoSQL:

- Dữ liệu đa dạng: Dữ liệu thu thập từ các cuộc khảo sát thường có cấu trúc không đồng nhất, bao gồm cả dữ liệu định lượng và định tính. NoSQL với khả năng xử lý dữ liệu không cấu trúc và bán cấu trúc một cách hiệu quả sẽ là lựa chọn phù hợp.
- Mở rộng dễ dàng: Khi số lượng sinh viên và câu hỏi khảo sát tăng lên, NoSQL có thể dễ dàng mở rộng quy mô để đáp ứng nhu cầu lưu trữ và xử lý dữ liệu.

#### 2) Cải thiện chất lượng đào tạo:

- Phản hồi nhanh: Việc sử dụng NoSQL giúp phân tích dữ liệu nhanh chóng, từ đó đưa ra những phản hồi kịp thời về chất lượng đào tạo, các vấn đề mà sinh viên gặp phải.
- Quyết định chính xác: Dựa trên dữ liệu khảo sát, các nhà quản lý giáo dục có thể đưa ra những quyết định chính xác để cải thiện chương trình đào tạo, nâng cao chất lượng giảng dạy.

#### 3) Đáp ứng yêu cầu của thời đại:

- Công nghệ mới: Việc áp dụng NoSQL vào quá trình khảo sát và phân tích dữ liệu thể hiện sự cập nhật với công nghệ mới nhất trong lĩnh vực quản lý giáo dục.
- Tối ưu hóa quy trình: Sử dụng NoSQL giúp tự động hóa nhiều công đoạn trong quá trình xử lý dữ liệu, giảm thiểu lỗi và tiết kiệm thời gian.

### 2. Phương pháp thực hiện:

- **Mục tiêu nghiên cứu:**

- Xây dựng một cơ sở dữ liệu cho việc khảo sát và quản lý dữ liệu khảo sát dựa trên NoSQL.
- **Nội dung nghiên cứu:**
  - Xây dựng cơ sở dữ liệu về khảo sát sinh viên cuối khóa bằng NoSQL;
  - Cài đặt công cụ MongoDB;
  - Cài đặt cơ sở dữ liệu;
  - Chọn dữ liệu mẫu thử;
  - Nhập mẫu thử;
  - Truy vấn dữ liệu bằng NoSQL.
- **Đối tượng nghiên cứu:**
  - Cơ sở dữ liệu cho việc khảo sát dựa trên NoSQL.
  - Nghiên cứu dữ liệu mẫu khảo sát.
- **Phương pháp nghiên cứu:**
  - Nghiên cứu tài liệu về NoSQL, MongoDB.
  - Nghiên cứu mẫu khảo sát, phân tích dữ liệu.
  - Nghiên cứu tài liệu về công cụ thực hiện: MongoDB.
  - Thực nghiệm: cài đặt và kiểm thử.
- **Phạm vi thực hiện đề tài:**

Thực hiện thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa.

### 3. Kết quả đạt được:

Hoàn thành cơ sở dữ liệu NoSQL cho việc khảo sát mức độ hài lòng của sinh viên cuối khóa, có thể lưu trữ dữ liệu khảo sát.

## CHƯƠNG 1 TỔNG QUAN

### 1.1 Lý do cho thiết kế và cài đặt cơ sở dữ liệu NoSQL cho khảo sát sự hài lòng của sinh viên cuối khóa

Việc khảo sát sự hài lòng của sinh viên cuối khóa là một hoạt động thường xuyên và quan trọng trong các cơ sở giáo dục. Tuy nhiên, để thu thập, lưu trữ và phân tích một lượng lớn dữ liệu khảo sát, các phương pháp truyền thống thường gặp phải những hạn chế nhất định. Cơ sở dữ liệu NoSQL, với khả năng linh hoạt, mở rộng và xử lý dữ liệu phi cấu trúc hiệu quả, mang đến những lợi ích vượt trội cho việc như:

**Dữ liệu đa dạng:** Khảo sát sinh viên thường thu thập nhiều loại dữ liệu khác nhau (văn bản, số, ngày tháng, dữ liệu định vị...), NoSQL có khả năng lưu trữ linh hoạt các loại dữ liệu này.

**Khả năng mở rộng:** Khi số lượng sinh viên và câu hỏi khảo sát tăng lên, NoSQL có thể dễ dàng mở rộng quy mô cơ sở dữ liệu để đáp ứng nhu cầu.

**Hiệu suất cao:** NoSQL thường có hiệu suất truy vấn và xử lý dữ liệu nhanh hơn so với các cơ sở dữ liệu quan hệ truyền thống, đặc biệt khi xử lý với lượng dữ liệu lớn.

**Linh hoạt:** Cấu trúc dữ liệu trong NoSQL có thể thay đổi dễ dàng, giúp thích ứng với các thay đổi trong thiết kế khảo sát.

### 1.2 Những vấn đề cần giải quyết

Thu thập và tích hợp dữ liệu:

Xây dựng hệ thống thu thập dữ liệu khảo sát từ nhiều nguồn khác nhau (form online, giấy, app...).

Tiến hành làm sạch và chuẩn hóa dữ liệu để đảm bảo tính chính xác và thống nhất.

Thiết kế cấu trúc cơ sở dữ liệu:

Xác định các thực thể và thuộc tính cần lưu trữ (sinh viên, câu hỏi, câu trả lời,...).

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

---

Lựa chọn mô hình dữ liệu phù hợp (Document, Key-Value, Column Family...) cho từng loại dữ liệu.

Xây dựng các truy vấn và báo cáo:

Thiết kế các truy vấn để trích xuất thông tin từ cơ sở dữ liệu phục vụ cho việc phân tích.

Tạo các báo cáo trực quan hóa kết quả khảo sát để dễ dàng hiểu và sử dụng.

Bảo mật và bảo vệ dữ liệu:

Đảm bảo tính bảo mật của dữ liệu sinh viên, tuân thủ các quy định về bảo vệ thông tin cá nhân.

Xây dựng các cơ chế sao lưu và phục hồi dữ liệu để tránh mất mát thông tin.

### 1.3 Các nội dung nghiên cứu chính

**Khảo sát các giải pháp NoSQL:** Nghiên cứu các hệ quản trị cơ sở dữ liệu NoSQL phổ biến hiện nay (MongoDB) để lựa chọn giải pháp phù hợp nhất cho đề tài.

**Thiết kế mô hình dữ liệu:** Xây dựng mô hình dữ liệu chi tiết cho cơ sở dữ liệu khảo sát, bao gồm các collection, document và các index cần thiết.

**Phát triển ứng dụng khảo sát:** Phát triển một ứng dụng web hoặc mobile để thu thập dữ liệu khảo sát từ sinh viên.

**Đánh giá hiệu năng:** Đánh giá hiệu năng của cơ sở dữ liệu khi xử lý một lượng lớn dữ liệu khảo sát.



## CHƯƠNG 2 NGHIÊN CỨU LÝ THUYẾT

### 2.1 Quy trình nghiệp vụ cho việc thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát

**Mục tiêu:** Xây dựng một cơ sở dữ liệu khảo sát hiệu quả, linh hoạt và dễ sử dụng để thu thập, lưu trữ dữ liệu về sự hài lòng của sinh viên.

**Đối tượng:** Sinh viên năm cuối của các ngành học khác nhau trong trường đại học Trà Vinh.

**Nội dung khảo sát:** Đánh giá sự hài lòng của sinh viên về các khía cạnh như chất lượng giảng dạy, cơ sở vật chất, hoạt động ngoại khóa, dịch vụ hỗ trợ, v.v.

#### Các nghiệp vụ chính:

**Tạo khảo sát:** Tạo một khảo sát mới bao gồm tên, mô tả và các câu hỏi.

#### Các loại phiếu khảo sát:

##### 1) Khảo sát sinh viên

- **Khảo sát môn học:** Đánh giá chất lượng giảng dạy, nội dung môn học và phương pháp giảng dạy của giảng viên.
- **Khảo sát hoạt động giảng dạy và chuẩn đầu ra học phần:** Tập trung vào mức độ đạt được chuẩn đầu ra và hiệu quả của hoạt động giảng dạy.
- **Khảo sát cuối khóa:** Phản hồi về toàn bộ trải nghiệm học tập và dịch vụ trong thời gian học.
- **Khảo sát cố vấn học tập:** Đánh giá sự hỗ trợ và hướng dẫn của cố vấn học tập đối với sinh viên.

##### 2) Khảo sát chương trình đào tạo

- **Khảo sát chương trình đào tạo:** Lấy ý kiến đóng góp từ sinh viên và giảng viên để cải thiện chương trình học, đảm bảo tính phù hợp và thực tiễn.
- **Khảo sát cải tiến chương trình đào tạo và chuẩn đầu ra:** Tập trung vào việc cải tiến các tiêu chuẩn và nội dung đào tạo theo nhu cầu thực tế.

**Thu thập dữ liệu:** Lưu trữ dữ liệu câu trả lời của sinh viên vào cơ sở dữ liệu MongoDB.

## 2.2 Giới thiệu về NoSQL

### 2.2.1 Giới thiệu sơ lược về NoSQL

Cơ sở dữ liệu NoSQL là một Hệ thống quản lý dữ liệu không quan hệ (Nonrelational Data Management System) có lược đồ (schema) linh hoạt, dễ mở rộng. Mục đích chính của việc sử dụng cơ sở dữ liệu NoSQL là dành cho các kho dữ liệu phân tán với nhu cầu lưu trữ dữ liệu lớn. NoSQL được sử dụng cho dữ liệu lớn và ứng dụng web thời gian thực. Chẳng hạn các công ty như Twitter, Facebook và Google thu thập hàng terabyte dữ liệu người dùng mỗi ngày.

Cơ sở dữ liệu NoSQL là viết tắt của “Not Only SQL” hoặc “Not SQL”. Hệ thống quản lý CSDL quan hệ SQL, ta sử dụng cú pháp SQL để lưu trữ và truy xuất dữ liệu. Đối với hệ thống cơ sở dữ liệu NoSQL, nó bao gồm một loạt các công nghệ cơ sở dữ liệu có thể lưu trữ dữ liệu có cấu trúc, bán cấu trúc, phi cấu trúc và đa hình[1].



*Hình 2.1 Cơ sở dữ liệu NoSQL [2]*

Thuật ngữ NoSQL được sử dụng bởi Carlo Strozzi vào năm 1998 để đặt tên cho cơ sở dữ liệu quan hệ mã nguồn mở Strozzi NoSQL nhỏ gọn, mà không tiết lộ giao diện SQL tiêu chuẩn, nhưng là vẫn còn là kiểu quan hệ. RDBMS của ông khác với khái niệm chung về cơ sở dữ liệu NoSQL được định nghĩa trong năm 2009. Strozzi gợi ý rằng, vì phong trào NoSQL hiện thời đi mất từ mô hình kiểu quan hệ

cùng với nhau; vì thế nên được gọi cho phù hợp hơn đó là “NoREL”, ám chỉ tới “No Relational”.

Johan Oskarsson của Last.fm giới thiệu lại thuật ngữ NoSQL vào đầu năm 2009 khi tổ chức một sự kiện thảo luận về "các cơ sở dữ liệu phân tán, không quan hệ nguồn mở". Tên gọi này cố gắng để đánh dấu sự xuất hiện ngày càng nhiều các kho lưu trữ dữ liệu phân tán, không quan hệ, bao gồm các nhân bản mã nguồn mở BigTable/MapReduce của Google và Dynamo của Amazon. Hầu hết các hệ thống NoSQL đầu tiên đã không cố gắng cung cấp các bảo đảm tính nguyên tử, nhất quán, tách biệt và bền vững, trái với ưu thế thực tế trong các hệ thống cơ sở dữ liệu kiểu quan hệ[3].

### 2.2.2 Một số khái niệm cơ bản trong NoSQL

- Fields: tương đương với khái niệm Columns trong SQL;
- Document: thay thế khái niệm row trong SQL. Đây cũng chính là khái niệm làm nên sự khác biệt giữa NoSQL và SQL, một document chứa số cột (fields) không cố định trong khi một row thì số cột (columns) là định sẵn trước;
- Collection: tương đương với khái niệm table trong SQL. Một collection là tập hợp các document. Điều đặc biệt là một collection có thể chứa các document hoàn toàn khác nhau;
- Key-value: cặp khóa - giá trị được dùng để lưu trữ dữ liệu trong NoSQL;
- Cursor: tạm dịch là con trỏ. Sử dụng cursor để lấy dữ liệu từ database.
- Trong các hệ cơ sở dữ liệu quan hệ, các cột được định nghĩa theo bảng còn với hệ cơ sở dữ liệu không ràng buộc, các cột được định nghĩa ở mỗi document. Bởi thế, các document quản lý gần như tất cả, các collection không cần quản lý chặt chẽ những gì đang xảy ra trong nó nữa.

*Bảng 2.1 Bảng so sánh các thuật ngữ tương đương của RDBMS với NoSQL [4]*

<b>RDBMS</b>	<b>NoSQL</b>
Database	Database
Table	Collection
Row	Document
Column	Field
Index	Index
Join	\$lookup, embedded documents

### 2.2.3 Kiến trúc của NoSQL

Kiến trúc của NoSQL khác với kiến trúc của SQL, bao gồm các thành phần sau:

- Các đối tượng dữ liệu: NoSQL không sử dụng bảng, hàng và cột như trong RDBMS. Thay vào đó, nó sử dụng các đối tượng dữ liệu, chẳng hạn như tài liệu (document), cặp khóa-giá trị (key-value), hoặc các đồ thị (graph) để lưu trữ dữ liệu;
- Không sử dụng SQL: NoSQL không sử dụng SQL để truy vấn dữ liệu, thay vào đó, nó cung cấp các API để truy vấn và truy cập dữ liệu;
- Mở rộng dễ dàng: NoSQL được thiết kế để mở rộng dễ dàng và có thể được triển khai trên nhiều máy chủ để xử lý lưu lượng truy cập cao và dữ liệu lớn;
- Hỗ trợ dữ liệu phi cấu trúc: NoSQL cho phép lưu trữ dữ liệu phi cấu trúc hoặc có cấu trúc thay đổi theo thời gian, giúp cho việc lưu trữ và truy vấn dữ liệu linh hoạt hơn;
- Tính nhất quán thấp: Trong NoSQL, tính nhất quán của dữ liệu không được đảm bảo tuyệt đối như trong RDBMS. Thay vào đó, NoSQL có thể cung cấp mức độ nhất quán ở mức độ thấp hơn, tùy thuộc vào yêu cầu và tính chất của dữ liệu;

- Tính linh hoạt: NoSQL có thể được sử dụng cho nhiều mục đích khác nhau, bao gồm lưu trữ dữ liệu, truy vấn dữ liệu nhanh, lưu trữ các dữ liệu lớn và phân tích dữ liệu.

## 2.2.4 Tính chất của NoSQL

Dưới đây là một số tính chất của NoSQL:

- Độ linh hoạt cao: NoSQL được thiết kế để linh hoạt và có khả năng mở rộng dễ dàng, giúp nó phù hợp với các ứng dụng phân tán và các hệ thống lớn;
- Khả năng xử lý dữ liệu lớn: NoSQL được thiết kế để xử lý các khối dữ liệu lớn mà không yêu cầu phải thực hiện các hoạt động truy vấn phức tạp;
- Khả năng phân tán: NoSQL cho phép dữ liệu được phân tán trên nhiều máy chủ, giúp nó đảm bảo tính sẵn sàng cao và giảm thiểu rủi ro mất dữ liệu;
- Khả năng xử lý dữ liệu đa dạng: NoSQL hỗ trợ nhiều loại dữ liệu khác nhau bao gồm dữ liệu cấu trúc, bán cấu trúc và phi cấu trúc, cho phép nó lưu trữ và xử lý các dạng dữ liệu khác nhau;
- Tốc độ xử lý nhanh: NoSQL được thiết kế để tối ưu hóa tốc độ xử lý, giúp nó đáp ứng các yêu cầu về xử lý dữ liệu trực tiếp và thời gian thực;
- Thiết kế cho các ứng dụng web: NoSQL được thiết kế để phù hợp với các ứng dụng web hiện đại, giúp nó hỗ trợ khả năng lưu trữ và xử lý các tài liệu HTML, JSON, XML;
- Không yêu cầu kỹ năng SQL: NoSQL không sử dụng ngôn ngữ truy vấn SQL, do đó không yêu cầu người dùng phải có kỹ năng SQL để sử dụng nó.

## 2.2.5 Đặc tính của NoSQL

Cơ sở dữ liệu NoSQL có các đặc tính sau:

- Mô hình cơ sở dữ liệu là phi quan hệ;
- Tập trung cho khả năng phân tán và mở rộng thuộc tính theo chiều rộng;
- Quy tắc cho lược đồ thì yếu hoặc không có;
- Sao chép dữ liệu dễ dàng;
- Dễ dàng truy cập được cung cấp thông qua một API;

- Mô hình nhất quán không phải là ACID.

Cơ sở dữ liệu quan hệ RDBMS (Relational Database Management System) sử dụng cú pháp SQL để lưu trữ và truy xuất dữ liệu. Đối với hệ thống cơ sở dữ liệu NoSQL, bao gồm một loạt các công nghệ cơ sở dữ liệu có thể lưu trữ dữ liệu có cấu trúc, bán cấu trúc, phi cấu trúc và đa hình.

### 2.2.6 Chức năng của NoSQL

Các chức năng chính của NoSQL bao gồm:

- Lưu trữ và xử lý dữ liệu phi cấu trúc hoặc có cấu trúc thay đổi: NoSQL cho phép lưu trữ dữ liệu phi cấu trúc hoặc có cấu trúc thay đổi theo thời gian, giúp cho việc lưu trữ và truy vấn dữ liệu linh hoạt hơn. Ví dụ, hệ thống NoSQL có thể lưu trữ các tài liệu (document), cặp khóa-giá trị (key-value), hoặc các đồ thị (graph);
- Mở rộng dễ dàng: NoSQL được thiết kế để mở rộng dễ dàng và có thể được triển khai trên nhiều máy chủ để xử lý lưu lượng truy cập cao và dữ liệu lớn;
- Truy vấn và truy cập dữ liệu nhanh chóng: NoSQL cung cấp các API để truy vấn và truy cập dữ liệu nhanh chóng, đặc biệt là khi làm việc với dữ liệu lớn;
- Tính linh hoạt cao: NoSQL có thể được sử dụng cho nhiều mục đích khác nhau, bao gồm lưu trữ dữ liệu, truy vấn dữ liệu nhanh, lưu trữ các dữ liệu lớn và phân tích dữ liệu;
- Tính sẵn sàng cao: NoSQL hỗ trợ các tính năng như sao lưu dự phòng và khả năng chịu lỗi, giúp giảm thiểu thời gian chết và đảm bảo sẵn sàng dữ liệu;
- Hỗ trợ phân tán dữ liệu: NoSQL có thể lưu trữ và quản lý dữ liệu trên nhiều máy chủ khác nhau, giúp giảm thiểu tải cho một máy chủ duy nhất và đảm bảo tính khả dụng của dữ liệu;

Những chức năng này giúp NoSQL trở thành lựa chọn phù hợp trong các ứng dụng web, phân tích dữ liệu và các hệ thống lưu trữ và quản lý dữ liệu lớn.

### 2.2.7 Các dạng cơ sở dữ liệu NoSQL

Có nhiều hệ thống cơ sở dữ liệu NoSQL như: MongoDB, RavenDB, Redis, Neo4j,... Xong có thể chia NoSQL thành 4 loại:

- Key-Value stores;
- Document stores;
- Column-Family Databases;
- Graph Databases;

#### 2.2.7.1 Key-Value stores

Cơ sở dữ liệu **Key-Value Stores** khi nó có các đặc tính như sau:

- Có một tập các đối tượng dữ liệu nhận dạng, các khóa;
- Đối với mỗi khóa, có chính xác một đối tượng dữ liệu mô tả có liên quan, giá trị cho khóa đó;
- Chỉ định một khóa cho phép truy vấn giá trị liên quan trong cơ sở dữ liệu.

Để truy vấn dữ liệu trong CSDL, ta dựa vào key để lấy dữ liệu ra. Các CSDL dạng này có tốc độ truy vấn rất nhanh.

**Database tiêu biểu:** Riak, Redis, MemCache, Project Voldemort, CouchBase

**Ứng dụng:** Do tốc độ truy xuất nhanh, key-value stores thường được dùng để làm cache cho ứng dụng (Tiêu biểu là Redis và MemCache). Ngoài ra, nó còn được dùng để lưu thông tin trong sessions, profiles/preferences của user...

#### 2.2.7.2 Document stores

Mỗi đối tượng sẽ được lưu trữ trong CSDL dưới dạng một document. Dữ liệu sẽ được lưu trữ dưới dạng BSON/JSON/XML dưới database. Dữ liệu không có lược đồ cứng như SQL. Do đó ta **có thể thêm/sửa các thuộc tính, thay đổi table,...** rất nhanh và đơn giản. Cơ sở dữ liệu dạng này có tốc độ truy vấn nhanh, có thể thực hiện các câu truy vấn phức tạp, dễ mở rộng (scalability). Mỗi cơ sở dữ liệu có **một kiểu truy vấn riêng**.

**Database tiêu biểu:** MongoDB, RavenDB, CouchDB, TerraStone, OrientDB.

**Ứng dụng:** Do nhanh và linh động, document database thường đóng vai trò làm **database cho các ứng dụng prototype, big data**, e-commerce, CMS. Ngoài ra, ta còn dùng nó để lưu log hoặc history.

#### 2.2.7.3 Column-Family Database

**Giới thiệu:** Dữ liệu được lưu trong CSDL dưới dạng các cột, thay vì các hàng như SQL. Mỗi hàng sẽ có một key/id riêng. Điểm đặc biệt là **các hàng trong một bảng sẽ có số lượng cột khác nhau**. Câu lệnh truy vấn của nó khá giống SQL.

**Database tiêu biểu:** Cassandra (Phát triển bởi Facebook), HyperTable, Apache Hbase.

**Ứng dụng:** Column-Family Database được sử dụng khi ta cần **ghi một số lượng lớn dữ liệu, big data**. Nó còn được ứng dụng trong 1 số CMS và ứng dụng e-commerce.

#### 2.2.7.4 Graph Database

Dữ liệu trong graph database được lưu dưới dạng các node. Mỗi node sẽ có 1 label, 1 số properties như một row trong SQL. Các node này được kết nối với nhau bằng các relationship. Graph database tập trung nhiều vào relationship giữa các node, áp dụng nhiều thuật toán duyệt node để tăng tốc độ.

**Database tiêu biểu:** Neo4j, InfiniteGraph, OrientDB, HYPERGRAPHDB

**Ứng dụng:** Khi cần truy vấn các mối quan hệ, graph database **truy vấn nhanh và dễ hơn nhiều** so với database. Nó được dùng trong các hệ thống: mạng nơ ron, chuyển tiền bạc, mạng xã hội (tìm bạn bè), **giới thiệu sản phẩm** (dựa theo sở thích/ lịch sử mua sắm của người dùng)... Neo4j là một database free, lại có một cộng đồng rất lớn, với vô số bài hướng dẫn, các bạn nên học thử.

#### 2.2.8 Một số thuật ngữ tương đương trong Relational Database và NoSQL database



*Bảng 2.2 Một số thuật ngữ tương đương trong Relational Database và NoSQL database [5]*

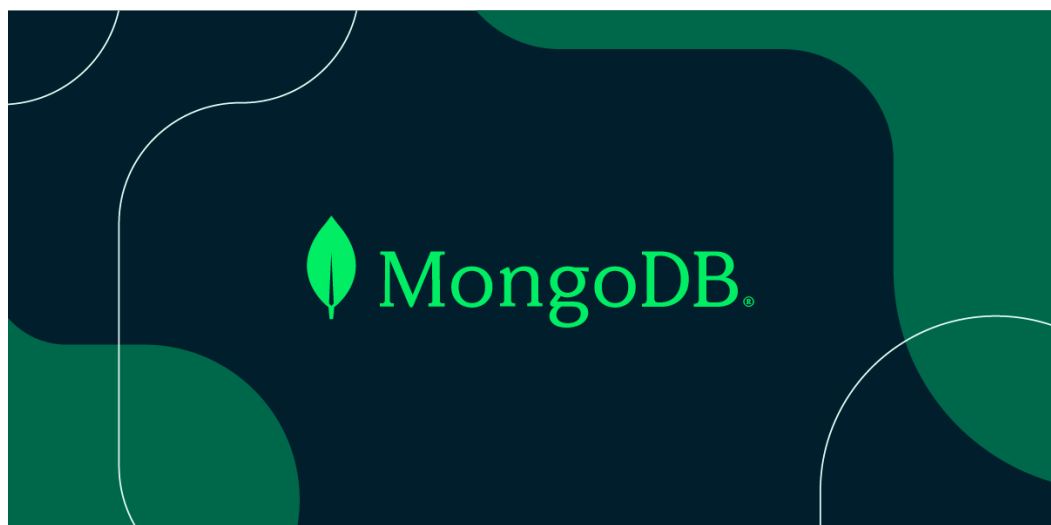
<b>RELATIONAL</b>	<b>KEY-VALUE (RIAK)</b>	<b>DOCUMENT (MONGODB)</b>	<b>COLUMN- FAMILY (CASSANDRA)</b>	<b>PRAPH (NEO4J)</b>
instance	cluster	mongod	cluster	Instance
table	bucket	collection	column-family	label
row	key-value	document	row	node
row-id	key	_id		
schema		database		schema

Bảng này chỉ có tính tương đối. Các node trong Graph DB không có key, mà các properties của nó sẽ đóng vai trò thay thế key.

## 2.3 Tổng quan về MongoDB

### 2.3.1 Tổng quan về MongoDB

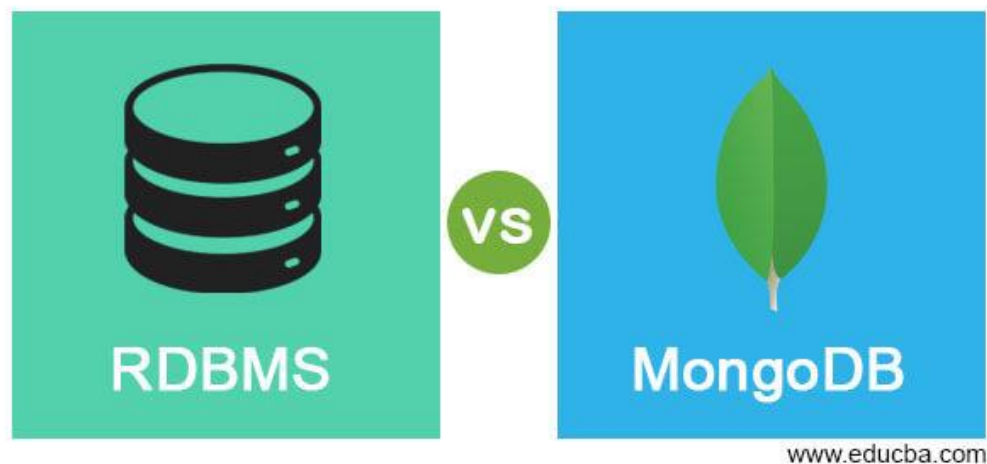
MongoDB là một cơ sở dữ liệu đa nền tảng, hoạt động trên các khái niệm Collection và Document, nó cung cấp hiệu suất cao, tính khả dụng cao và khả năng mở rộng dễ dàng.



*Hình 2.2 Hình ảnh về MongoDB [6]*

### 2.3.2 Các lợi thế của MongoDB so với RDBMS

- Ít Schema hơn: MongoDB là một cơ sở dữ liệu dựa trên Document, trong đó một Collection giữ các Document khác nhau. Số trường, nội dung và kích cỡ của Document này có thể khác với Document khác.
- Cấu trúc của một đối tượng là rõ ràng.
- Không có các Join phức tạp.
- Khả năng truy vấn sâu hơn. MongoDB hỗ trợ các truy vấn động trên các Document bởi sử dụng một ngôn ngữ truy vấn dựa trên Document mạnh mẽ như SQL.
- Tuning; MongoDB là dễ dàng để mở rộng.
- Việc chuyển đổi/ánh xạ của các đối tượng ứng dụng đến các đối tượng cơ sở dữ liệu là không cần thiết.
- Sử dụng bộ nhớ nội tại để lưu giữ phần công việc, giúp truy cập dữ liệu nhanh hơn.



Hình 2.3 RDBMS so với MongoDB [7]

### 2.3.3 Một số khái niệm cơ bản trong MongoDB

#### 2.3.3.1 Khái niệm Database

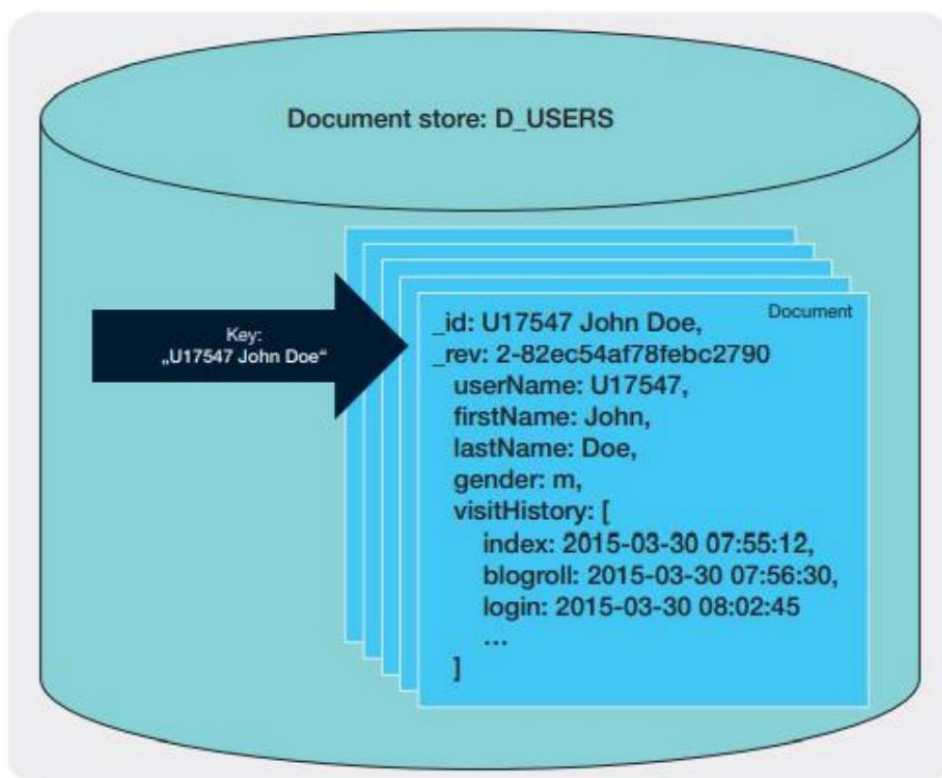
Database là một nơi chứa vật lý cho các Collection. Mỗi Database lấy tập hợp các file riêng của nó trên hệ thống file. Mỗi MongoDB Server có thể có nhiều cơ sở dữ liệu.

### 2.3.3.2 Khái niệm Collection

Collection là một nhóm các Document trong MongoDB. Collection tương đương như một Table trong RDBMS – Relational Database Management System. Do đó, một Collection tồn tại bên trong một cơ sở dữ liệu duy nhất. Các Collection không có ràng buộc (Relationship) như các hệ quản trị cơ sở dữ liệu quan hệ nên việc truy xuất rất nhanh. Do đó mỗi collection có thể chứa nhiều thể loại khác nhau, không giống như table trong hệ quản trị cơ sở dữ liệu quan hệ có các field cố định. Các Document bên trong một Collection có thể có nhiều trường khác nhau. Đặc biệt, tất cả các Document trong một Collection là tương tự nhau hoặc với cùng mục đích liên quan.

### 2.3.3.3 Khái niệm Document

Một Document trong MongoDB, có cấu trúc tương tự như kiểu dữ liệu JSON, là một tập hợp các cặp key-value. Các Document có lược đồ động, nghĩa là Document trong cùng một Collection không cần thiết phải có cùng một tập hợp các trường hoặc cấu trúc giống nhau, và các trường chung trong Document của một Collection có thể giữ các kiểu dữ liệu khác nhau.



Hình 2.4 Minh họa Document trong Collection [8]

Bảng dưới đây trình bày mối quan hệ giữa các thuật ngữ của hệ quản trị cơ sở dữ liệu quan hệ và MongoDB.

*Bảng 2.3 Mối quan hệ giữa các thuật ngữ của RDBMS và MongoDB [9]*

<b>RDBMS</b>	<b>MongoDB</b>
Database	Database
Table	Collection
Tuple/Row	Document
column	Field
Table Join	Embedded Documents
Primary Key	Primary Key (Giá trị mặc định là '_id' được cung cấp bởi chính MongoDB)

Ví dụ dưới đây minh họa cấu trúc của một Document đơn giản của một Blog site với một cặp key-value phân biệt bởi dấu phẩy (,).

```
{
  _id: ObjectId('5816baa0a2b7a9f009f2f2b7'),
  title: 'MongoDB Overview',
  description: 'MongoDB is no sql database',
  by: 'tutorials point',
  tags: ['mongodb', 'database', 'NoSQL'],
  likes: 100
}
```

Ở đây, **\_id** là một số thập lục phân để đảm bảo tính duy nhất của mỗi Document. Người dùng có thể cung cấp **\_id** trong khi chèn vào Document. Nếu người dùng không cung cấp **\_id**, thì MongoDB sẽ cung cấp một id duy nhất cho mỗi Document.

#### **2.3.4 Các kiểu dữ liệu trong MongoDB**

- Chuỗi: Đây là kiểu dữ liệu được sử dụng phổ biến nhất để lưu giữ dữ liệu. Chuỗi trong MongoDB phải là UTF-8 hợp lệ.

- Số nguyên: Kiểu dữ liệu này được sử dụng để lưu một giá trị số. Số nguyên có thể là 32 bit hoặc 64 bit phụ thuộc vào Server của bạn.
- Boolean: Kiểu dữ liệu này được sử dụng để lưu giữ một giá trị Boolean (true/false).
- Double: Kiểu dữ liệu này được sử dụng để lưu các giá trị số thực dấu chấm động.
- Min/ Max keys: Kiểu dữ liệu này được sử dụng để so sánh một giá trị với các phần tử BSON thấp nhất và cao nhất.
- Mảng: Kiểu dữ liệu này được sử dụng để lưu giữ các mảng hoặc danh sách hoặc nhiều giá trị vào trong một key.
- Timestamp: Giúp thuận tiện cho việc ghi chép hoặc đánh dấu thời điểm một Document được sửa đổi hoặc được thêm vào.
- Object: Kiểu dữ liệu này được sử dụng cho các Document được nhúng vào.
- Null: Kiểu dữ liệu này được sử dụng để lưu một giá trị Null.
- Symbol: Kiểu dữ liệu này được sử dụng giống như một chuỗi, tuy nhiên, nói chung nó được dành riêng cho các ngôn ngữ mà sử dụng kiểu symbol cụ thể.
- Code: Kiểu dữ liệu này được sử dụng để lưu giữ JavaScript code vào trong Document.
- Date : Kiểu dữ liệu này được sử dụng để lưu giữ date và time hiện tại trong định dạng UNIX time. Bạn có thể xác định date time riêng cho bạn bằng việc tạo đối tượng Date và truyền ngày, tháng, năm vào trong đó.
- Object ID: Kiểu dữ liệu này được sử dụng để lưu giữ ID của Document.
- Binary data: Kiểu dữ liệu này được sử dụng để lưu giữ dữ liệu nhị phân.
- Regular expression: Kiểu dữ liệu này được sử dụng để lưu giữ Regular Expression.

## 2.3.5 Các thao tác dữ liệu trên MongoDB

### 2.3.5.1 Tạo cơ sở dữ liệu

Sử dụng lệnh `use` để tạo cơ sở dữ liệu, các cơ sở dữ liệu được xác định bởi tên của nó. Lệnh `use` sẽ tạo một cơ sở dữ liệu mới nếu CSDL chưa tồn tại, nếu CSDL đã tồn tại thì kết quả sẽ trả về cơ sở dữ liệu đang tồn tại.

Cú pháp cơ bản

```
use < DATABASE_NAME >
```

Trong đó quy tắc đặt tên cho < DATABASE\_NAME > như sau

- Gồm chữ và số;
- Phân biệt chữ hoa hay chữ thường;
- Không chứa chuỗi rỗng ("");
- Không chứa các ký tự sau: /, \, ., ", \*, , :, |, ?, \$, (khoảng trắng), or \0 (the null character)
- Tên cơ sở dữ liệu giới hạn tối đa 64 bytes.

Ví dụ: Tạo cơ sở dữ liệu tên `nosql_db`

```
>use MyNoSQL  
<'switched to db MyNoSQL'
```

### 2.3.5.2 Kiểm tra cơ sở dữ liệu hiện hành

Sử dụng lệnh `db` với cú pháp như sau:

```
>db  
<MyNoSQL
```

### 2.3.5.3 Kiểm tra danh sách các cơ sở dữ liệu

Sử dụng lệnh `show dbs`.

```
>show dbs
```

Cơ sở dữ liệu mặc định trong MongoDB là `test`, nếu người dùng không tạo thêm bất kỳ cơ sở dữ liệu nào thì các Collection sẽ được lưu trữ trong `test`. Khi một cơ sở dữ liệu được tạo mà chưa có bất kỳ Collection nào thì tên cơ sở dữ liệu đó sẽ không hiện lên trong danh sách các cơ sở dữ liệu khi sử dụng lệnh **show dbs**.

#### 2.3.5.4 Xóa cơ sở dữ liệu

Sử dụng lệnh `db.dropDatabase()`

Lệnh này sẽ xóa các cơ sở dữ liệu đã chọn. Nếu không chọn cơ sở dữ liệu nào thì sẽ xóa cơ sở dữ liệu mặc định test.

Trước khi xóa thì dùng lệnh `show dbs` để kiểm tra danh sách các cơ sở dữ liệu hiện có sau đó chọn cơ sở dữ liệu cần xóa.

```
>show dbs
```

(hiển thị danh sách các db hiện có)

```
>use MyNoSQL
<'switched to db MyNoSQL'
>db.dropDatabase()
>{"dropped" : "Tên cơ sở dữ liệu cần xóa", "ok" : 1 }
```

Kiểm tra lại xem cơ sở dữ liệu đã xóa chưa

```
>show dbs
```

(Hiển thị danh sách các db hiện có, kiểm tra xem db đã xóa còn tồn tại hay không)

### 2.3.6 Làm việc với Collection

#### 2.3.6.1 Tạo Collection

Sử dụng phương thức `db.createCollection(name, options)` để tạo Collection.

Trong đó name là tên của Collection. Options là một Document và được sử dụng để xác định cấu hình cho Collection.

Bảng 2.4 Tham số options của Collection [10]

Tham số	Kiểu	Miêu tả
Name	Chuỗi	Tên của Collection
Options	Document	(Tùy chọn) Xác định các tùy chọn về kích thước bộ nhớ và việc lập chỉ mục

Tham số options là tùy chọn, người dùng cần xác định tên của Collection.

Dưới đây là danh sách các tùy chọn cho options người dùng có thể sử dụng:

*Bảng 2.5 Danh sách các tùy chọn options của Collection [10]*

Trường	Kiểu	Miêu tả
capped	Boolean	(Tùy chọn) Nếu true, kích hoạt một Capped Collection. Đây là một Collection có kích thước cố định mà tự động ghi đè các entry cũ nhất khi nó tiếp cận kích cỡ tối đa. Nếu bạn xác định là true, thì cần xác định tham số size
Size	Số	(Tùy chọn) Xác định kích thước tối đa (giá trị byte) cho một Capped Collection. Nếu tham số capped là true, thì cần xác định trường này
autoIndexID	Boolean	(Tùy chọn) Nếu true, tự động tạo chỉ mục trên các trường _id. Giá trị mặc định là false
Max	Số	(Tùy chọn) Xác định số Document tối đa được cho phép trong một Capped Collection

Trong khi thực hiện việc chèn dữ liệu vào Document, đầu tiên MongoDB kiểm tra trường size của Capped Collection, sau đó nó kiểm tra trường max.

Ví dụ:

Cú pháp cơ bản của phương thức createCollection() không có tham số options như sau:

```
>use MyNoSQL
<'switched to db MyNoSQL'
>db.createCollection("mycollection")
<{ "ok" : 1 }
```

Để kiểm tra Collection đã được tạo hay chưa ta dùng lệnh Show Collection để kiểm tra như sau:



Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
>show collections
<mycollection
```

Sử dụng phương thức createCollection() với một số option

```
>db.createCollection("mycollection",{capped:true,
autoIndexId:true, size : 6142800, max : 10000 } )
<{ ok: 1 }
```

MongoDB có thể tự động tạo Collection khi người dùng insert một Document

### 2.3.6.2 Xóa Collection

Phương thức db.collection.drop() trong MongoDB được sử dụng để xóa một Collection từ cơ sở dữ liệu.

Cú pháp cơ bản của lệnh drop() là như sau:

Đầu tiên, bạn kiểm tra các Collection có sẵn bên trong cơ sở dữ liệu MyNoSQL.

```
>use MyNoSQL
<'switched to db MyNoSQL'
>show collections
mycollection
mycollection_1
mycollection_2
MyCollection2
>
```

Xóa Collection với tên MyCollection2 như sau:

```
>db.MyCollection2.drop()
true
```

Sau đó kiểm tra lần nữa các collection bên trong cơ sở dữ liệu:

```
>show collection
mycollection
mycollection_1
mycollection_2
```

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

---

Phương thức `drop()` sẽ trả về `true`, nếu Collection đã chọn bị xóa thành công, nếu không, nó sẽ trả về `false`.

### 2.3.7 Làm việc với Document

#### 2.3.7.1 Thêm mới document

Sử dụng phương thức `insert()` hoặc `save()` để thêm Document vào Collection

Cú pháp cơ bản của lệnh `insert()` như sau:

**`db.COLLECTION_NAME.insert(document)`**

Ví dụ:

```
>db.mycollection.insert
(
{
  _id: ObjectId('5816baa0a2b7a9f009f2f2b7'),
  title: 'MongoDB Overview',
  description: 'MongoDB is no sql database',
  by: 'tutorials point',
  tags: ['mongodb', 'database', 'NoSQL'],
  likes: 100
}
```

Chèn thêm một document khác, khi không chỉ định `_id` cho document

```
>db.mycollection.insert
```

Trong đó, **COLLECTION\_NAME** là tên của collection đã được tạo trước đó, nếu collection chưa được tồn tại trong cơ sở dữ liệu thì MongoDB sẽ tạo collection và sau đó chèn document vào collection.

Trong document được chèn, nếu không xác định tham số `_id`, thì MongoDB gán một `ObjectId` duy nhất cho document này.

Để chèn nhiều document trong một truy vấn đơn, có thể truyền một mảng các document trong lệnh `insert()`.

```
(
{
  title: 'NoSQL DataBase',
  description: 'NoSQL database doesn't have tables',
  by: 'tutorials point',
```

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
url: 'http://mongodb.com',  
tags: ['use', 'database'],  
}  
)
```

Ví dụ:

```
>db.mycollection.insert ([  
{  
  title: 'MongoDB Overview',  
  description: 'MongoDB is no sql database',  
  by: 'tutorials point',  
  tags: ['mongodb', 'database', 'NoSQL'],  
  likes: 100,  
}, {  
  title: 'NoSQL DataBase',  
  description: 'NoSQL database doesn't have tables',  
  by: 'tutorials point',  
  url: 'http://mongodb.com',  
  tags: ['use', 'database'],  
}, {  
  title: 'Collection in Database',  
  description: ' Collections like tables in RDBMS',  
  by: 'tutorials point',  
  url: 'http://mongodb.com',  
  tags: ['Collection', 'database'],  
}  
]  
)
```

### 2.3.7.2 Truy vấn Document

Để truy vấn dữ liệu từ collection trong MongoDB, cần sử dụng phương thức find() trong MongoDB.

Cú pháp cơ bản của phương thức find() là như sau:

```
db.COLLECTION_NAME.find()  
db.COLLECTION_NAME.findOne()
```

Ví dụ:

```
>db.mycollection.find()
```

---

```
{ _id: ObjectId("5816baa0a2b7a9f009f2f2b7"),  
  title: 'MongoDB Overview',  
  description: 'MongoDB is no sql database',  
  by: 'tutorials point',  
  tags: [ 'mongodb', 'database', 'NoSQL' ],  
  likes: 100 }
```

---

```
{ _id: ObjectId("633b8b32b6f331dd2c6d3bdc"),  
  title: 'MongoDB Overview',  
  description: 'MongoDB is no sql database',  
  by: 'tutorials point',  
  tags: [ 'mongodb', 'database', 'NoSQL' ],  
  likes: 100 }
```

---

```
{ _id: ObjectId("633b8b32b6f331dd2c6d3bdd"),  
  title: 'NoSQL DataBase',  
  description: 'NoSQL database doesn't have tables',  
  by: 'tutorials point',  
  url: 'http://mongodb.com',  
  tags: [ 'use', 'database' ] }
```

---

```
{ _id: ObjectId("633b8b32b6f331dd2c6d3bde"),  
  title: 'Collection in Database',  
  description: ' Collections like tables in RDBMS',
```

---

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

---

by: 'tutorials point',

url: 'http://mongodb.com',

tags: [ 'Collection', 'database' ] }

---

Phương thức `find()` sẽ hiển thị tất cả document ở dạng không có cấu trúc (hiển thị không theo cấu trúc nào).

Phương thức `pretty()` trong MongoDB: để hiển thị các kết quả theo một cách đã được định dạng, có thể sử dụng phương thức `pretty()`.

```
>db.mycollection.findOne()  
{  
  _id: ObjectId("5816baa0a2b7a9f009f2f2b7"),  
  title: 'MongoDB Overview',  
  description: 'MongoDB is no sql database',  
  by: 'tutorials point',  
  tags: [ 'mongodb', 'database', 'NoSQL' ],  
  likes: 100  
}
```

Ngoài phương thức `find()`, trong MongoDB còn có phương thức `findOne()` sẽ chỉ trả về một document.

### 2.3.8 Mệnh đề Where trong truy vấn MongoDB (tương đương mệnh đề WHERE trong RDBMS)

#### 2.3.8.1 Sử dụng truy vấn với điều kiện đơn

Để truy vấn document dựa trên một số điều kiện nào đó, người dùng có thể sử dụng cú pháp:

```
>db.< collectionname >.find(<fieldname> : {$ <operator>:<value>})
```

với các phép toán như trong bảng.

*Bảng 2.6 Các phép toán trong điều kiện của phép chọn [10]*

<b>Phép toán</b>	<b>Cú pháp</b>	<b>Ví dụ</b>	<b>Mệnh đề WHERE tương đương</b>
Equality	{<key>:<value>}	db.mycollection.find({"by": "tutorials point"}).pretty()	where by = 'tutorialspoint'
Less Than	{<key>:{<\$lt:<value>}}	db.mycollection.find({"likes": {\$lt:50}}).pretty()	where likes < 50
Less Than Equals	{<key>:{<\$lte:<value>}}	db.mycollection.find({"likes": {\$lte:50}}).pretty()	where likes <= 50
Greater Than	{<key>:{<\$gt:<value>}}	db.mycollection.find({"likes": {\$gt:50}}).pretty()	where likes > 50
Greater Than Equals	{<key>:{<\$gte:<value>}}	db.mycollection.find({"likes": {\$gte:50}}).pretty()	where likes >= 50
Not Equals	{<key>:{<\$ne:<value>}}	db.mycollection.find({"likes": {\$ne:50}}).pretty()	where likes != 50

Ví dụ: Tìm document có title là "**Collection in Database**" có lệnh như sau và được kết quả

```
>db.mycollection.find({"title":"Collection in Database"}).pretty()
<{ _id: ObjectId("633b8b32b6f331dd2c6d3bde"),
  title: 'Collection in Database',
  description: ' Collections like tables in RDBMS',
  by: 'tutorials point',
  url: 'http://mongodb.com',
  tags: [ 'Collection', 'database' ]
}
```

### 2.3.8.2 Sử dụng truy vấn với điều kiện kép

#### - Điều kiện AND trong MongoDB

Cú pháp Trong phương thức find(), khi truyền nhiều key bằng cách phân biệt chúng bởi dấu phẩy (,), thì MongoDB xem nó như là điều kiện AND.

Cú pháp cơ bản của AND trong MongoDB như sau:

```
>db.collectionname.find({key1:value1, key2:value2}).pretty()
```

Ví dụ: Ví dụ sau hiển thị tất cả loạt bài hướng dẫn (tutorials) được viết bởi 'tutorials point' có title là 'MongoDB Overview'

```
>db.mycollection.find({"by":"tutorials point","title":"MongoDB Overview"}).pretty()
```

```
{ _id: ObjectId("5816baa0a2b7a9f009f2f2b7"),
  title: 'MongoDB Overview',
  description: 'MongoDB is no sql database',
  by: 'tutorials point',
  tags: [ 'mongodb', 'database', 'NoSQL' ],
  likes: 100 }
```

```
{ _id: ObjectId("633b8b32b6f331dd2c6d3bdc"),
  title: 'MongoDB Overview',
  description: 'MongoDB is no sql database',
  by: 'tutorials point',
  tags: [ 'mongodb', 'database', 'NoSQL' ],
  likes: 100 }
```

Mệnh đề WHERE tương đương trong ví dụ trên sẽ là **'where by='tutorials point' AND title = 'MongoDB Overview'** .Người dùng có thể truyền bất kỳ số cặp key- value nào trong mệnh đề find.

## - Điều kiện OR trong MongoDB

Cú pháp:

Để truy vấn document dựa trên điều kiện OR, sử dụng từ khóa \$or. Cú pháp cơ bản của OR trong MongoDB như sau:

```
>db.collectionname.find(  
  {  
    $or: [  
      {key1: value1}, {key2:value2}  
    ]  
  }  
) .pretty()
```

Ví dụ:

Ví dụ sau sẽ hiển thị tất cả loạt bài có tags là [ 'Collection', 'database' ] hoặc có title là 'MongoDB Overview'

```
>db.mycollection.find({$or:[{"tags":["Collection", 'database']},  
{"title": "MongoDB Overview"}]}).pretty()
```

---

```
{ _id: ObjectId("5816baa0a2b7a9f009f2f2b7"),  
  title: 'MongoDB Overview',  
  description: 'MongoDB is no sql database',  
  by: 'tutorials point',  
  tags: [ 'mongodb', 'database', 'NoSQL' ],  
  likes: 100 }
```

---

```
{ _id: ObjectId("633b8b32b6f331dd2c6d3bdc"),  
  title: 'MongoDB Overview',  
  description: 'MongoDB is no sql database',  
  by: 'tutorials point',
```

---



Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

---

```
tags: [ 'mongodb', 'database', 'NoSQL' ],
```

```
likes: 100 }
```

---

```
{ _id: ObjectId("633b8b32b6f331dd2c6d3bde"),
```

```
title: 'Collection in Database',
```

```
description: ' Collections like tables in RDBMS',
```

```
by: 'tutorials point',
```

```
url: 'http://mongodb.com',
```

```
tags: [ 'Collection', 'database' ] }
```

---

### 2.3.9 Cập nhật Document

Phương thức `update()` hoặc `save()` trong MongoDB được sử dụng để cập nhật document vào trong một collection. Phương thức `update()` cập nhật các giá trị trong document đang tồn tại trong khi phương thức `save()` thay thế document đang tồn tại với document đã truyền trong phương thức `save()` đó.

Phương thức `update()` cập nhật các giá trị trong document đang tồn tại.

Cú pháp:

Cú pháp cơ bản của phương thức `update()` là như sau:

```
>db..update(SELECTIOIN_CRITERIA, UPDATED_DATA)
```

Ví dụ: Sử dụng lệnh tìm kiếm sau tìm document với "title": "MongoDB Overview"

```
>db.mycollection.find({"title": "MongoDB Overview"}).pretty()
```

Được kết quả như sau:

---

```
{ _id: ObjectId("5816baa0a2b7a9f009f2f2b7"),
```

```
title: 'MongoDB Overview',
```

```
description: 'MongoDB is no sql database',
```

```
by: 'tutorials point',
```

---

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

---

```
tags: [ 'mongodb', 'database', 'NoSQL' ],
```

```
likes: 100 }
```

---

```
{ _id: ObjectId("633b8b32b6f331dd2c6d3bdc"),
```

```
title: 'MongoDB Overview',
```

```
description: 'MongoDB is no sql database',
```

```
by: 'tutorials point',
```

```
tags: [ 'mongodb', 'database', 'NoSQL' ],
```

```
likes: 100 }
```

---

Ví dụ: Thiết lập tiêu đề mới 'How to reach MongoDB' của document với tiêu đề cũ là 'MongoDB Overview':

```
>db.mycollection.update({'title':'MongoDB Overview'},{$set:{'title':'How to reach MongoDB'}})
```

Kết quả sau khi update như sau:

```
> db.mycollection.find({"title": "How to reach MongoDB"}).pretty()
{
  _id: ObjectId("5816baa0a2b7a9f009f2f2b7"),
  title: 'How to reach MongoDB',
  description: 'MongoDB is no sql database',
  by: 'tutorials point',
  tags: [ 'mongodb', 'database', 'NoSQL' ],
  likes: 100
}
```

### 2.3.10 Xóa document trong MongoDB

- Phương thức `remove()` được sử dụng để xóa document từ collection.
- Phương thức `remove()` nhận hai tham số. Tham số đầu tiên deletion criteria xác định document để xóa, và tham số thứ hai là `justOne`.
  - o deletion criteria: (Tùy ý) Xác định document để xóa.

- justOne : (Tùy ý) Nếu được thiết lập là true (hoặc 1), thì chỉ xóa một document.

Cú pháp:

Cú pháp cơ bản của phương thức remove() là như sau:

```
>db.COLLECTION_NAME.remove(DELETION_CRITTERIA)
```

Ví dụ:

Dùng lệnh truy vấn để tìm document có title là MongoDB Overview sau và có kết quả:

```
> db.mycollection.find({"title": "MongoDB Overview"}).pretty()
{
  _id: ObjectId("633b8b32b6f331dd2c6d3bdc"),
  title: 'MongoDB Overview',
  description: 'MongoDB is no sql database',
  by: 'Tutorials',
  tags: [ 'mongodb', 'database', 'NoSQL' ],
  likes: 100
}
```

Để xóa tất cả các document có title là 'MongoDB Overview', thực hiện lệnh như sau:

```
>db. mycollection.remove({'title':'MongoDB Overview'})
```

Kiểm chứng kết quả xóa bằng lệnh sau:

```
>db.mycollection.find({"title": "MongoDB Overview"}).pretty()
```

Không tìm thấy dữ liệu vì đã bị xóa.

## 2.3.11 Tạo backup và resrote trong MongoDB

### 2.3.11.1 Tạo backup

Lệnh mongodump trong MongoDB Để tạo Backup của cơ sở dữ liệu trong MongoDB, sử dụng lệnh **mongodump**. Lệnh này sẽ dump tất cả dữ liệu của Server vào trong dump directory. Có nhiều tùy chọn có sẵn từ đó người dùng có thể giới hạn lượng dữ liệu hoặc tạo backup của Remote Server.

Cú pháp:

Cú pháp cơ bản của lệnh mongodump trong MongoDB là:

```
>mongodump
```

Ví dụ:

Bắt đầu Mongod Server. Giả sử Server của bạn đang chạy trên localhost và cổng 27017. Bây giờ, mở một dòng nhắc lệnh và tới thư mục bin của Mongod Instance, sau đó soạn lệnh mongodump.

Bạn theo dõi Collection có tên mycol có dữ liệu sau:

```
>mongodump
```

Lệnh này sẽ kết nối với Server đang chạy tại 127.0.0.1 và cổng 27017 và backup toàn bộ dữ liệu của Server tới thư mục /bin/dump. Kết quả của lệnh này như sau:

Dưới đây là một số tùy chọn có sẵn có thể được sử dụng với lệnh mongodump.

*Bảng 2.7 Các lệnh tùy chọn trong lệnh Backup [10]*

Cú pháp	Miêu tả	Ví dụ
mongodump --host HOST_NAME --port PORT_NUMBER	Lệnh này sẽ sao lưu tất cả cơ sở dữ liệu của mongod instance đã xác định	mongodump -- host tutorialspoint.com -- port 27017
mongodump --dbpath DB_PATH --out BACKUP_DIRECTORY	Lệnh này chỉ sao lưu cơ sở dữ liệu đã xác định tại path đã cho	mongodump -- dbpath /data/db/ - -out /data/backup/
mongodump --collection COLLECTION --db DB_NAME	Lệnh này chỉ sao lưu Collection đã xác định của cơ sở dữ liệu đã cho	mongodump -- collection mycol - -db test

### 2.3.12 Phục hồi dữ liệu trong MongoDB

Để phục hồi dữ liệu đã sao lưu trong MongoDB, bạn sử dụng lệnh `mongorestore`. Lệnh này phục hồi tất cả dữ liệu từ thư mục sao lưu.

Cú pháp:

Cú pháp cơ bản của lệnh `mongorestore` là:

```
>mongorestore
```

## CHƯƠNG 3 HIỆN THỰC HÓA NGHIÊN CỨU

### 3.1 Phân tích yêu cầu hệ thống

#### 3.1.1 Yêu cầu chức năng

Cơ sở dữ liệu khảo sát cần cho phép:

- Quản lý thông tin sinh viên, bao gồm mã số sinh viên, họ tên, ngành, khoa, thông tin liên hệ.
- Ghi nhận các câu trả lời của sinh viên đối với từng phần khảo sát.
- Tích hợp các thang đo mức độ hài lòng (theo thang điểm từ 1 đến 5) cho từng mục trong phiếu khảo sát.
- Lưu trữ thông tin về các kiến nghị của sinh viên (nếu có).
- Truy xuất dữ liệu để phân tích và đánh giá mức độ hài lòng.

#### 3.1.2 Yêu cầu phi chức năng

Dữ liệu cần được lưu trữ trong cơ sở dữ liệu NoSQL để đảm bảo:

- Khả năng mở rộng: Hỗ trợ số lượng lớn sinh viên và câu trả lời.
- Hiệu năng cao: Tăng tốc độ truy vấn và xử lý dữ liệu.
- Tính linh hoạt: Dễ dàng thêm hoặc sửa đổi các mục trong phiếu khảo sát.

### 3.2 Công cụ được sử dụng

- Hệ quản trị cơ sở dữ liệu: MongoDB.
- Công cụ hỗ trợ phát triển: MongoDB Compass để trực quan hóa dữ liệu.
- Mô hình lưu trữ: Dựa trên tài liệu (document-based) với MongoDB.

### 3.3 Thiết kế cơ sở dữ liệu

#### 3.3.1 Cấu trúc cơ sở dữ liệu

Cấu trúc của cơ sở dữ liệu: được thiết kế theo dạng document để lưu trữ các thông tin sinh viên và kết quả khảo sát.

Cấu trúc của phiếu khảo sát: Chứa các câu hỏi được chia thành hai phần chính:

- Phần 1: Các câu hỏi mức độ đồng ý với thang điểm từ 1 đến 5, chia thành các mục *Chương trình đào tạo*, *Các hoạt động hỗ trợ*, và *Cơ sở vật chất*.

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

- Phần 2: Câu hỏi về kiến nghị hoặc góp ý cho sinh viên.

Mô hình cơ sở dữ liệu bao gồm:

- Collection sinhVien: Lưu thông tin cá nhân sinh viên và kết quả khảo sát.

*Dưới đây là cấu trúc lược đồ của một document:*

Collection sinhVien

```
{  
  • hoten: họ tên sinh viên  
  • mssv: mã số sinh viên  
  • lop: lớp của sinh viên đang học  
  • khoa: khoa của ngành mà sinh viên đang học  
  • ngành: ngành mà sinh viên đang học  
  • sdt: số điện thoại của sinh viên  
  • mailSv: mail của sinh viên  
  {  
    • thoigiankhaosat: thời gian khảo sát  
    • phan1: mucDoDongY (câu trả lời dạng thang điểm từ 1 đến 5)  
      ○ I: Mục chươngTrinhDaoTao
```

Câu hỏi	Câu trả lời
Nội dung câu hỏi 1	Câu trả lời
Nội dung câu hỏi 2	Câu trả lời
Nội dung câu hỏi 3	Câu trả lời
Nội dung câu hỏi 4	Câu trả lời
Nội dung câu hỏi 5	Câu trả lời
Nội dung câu hỏi 6	Câu trả lời
Nội dung câu hỏi 7	Câu trả lời
Nội dung câu hỏi 8	Câu trả lời
Nội dung câu hỏi 9	Câu trả lời

○ II: Mục cacHoatDongHoTro

Câu hỏi	Câu trả lời
Nội dung câu hỏi 10	Câu trả lời
Nội dung câu hỏi 11	Câu trả lời

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

Câu hỏi	Câu trả lời
Nội dung câu hỏi 12	Câu trả lời
Nội dung câu hỏi 13	Câu trả lời
Nội dung câu hỏi 14	Câu trả lời
Nội dung câu hỏi 15	Câu trả lời

○ III: Mục coSoVatChat

Câu hỏi	Câu trả lời
Nội dung câu hỏi 16	Câu trả lời
Nội dung câu hỏi 17	Câu trả lời
Nội dung câu hỏi 18	Câu trả lời
Nội dung câu hỏi 19	Câu trả lời

},

{

- Phan2: kienNghì (câu trả lời dạng văn bản)

- Câu hỏi : Trình bày những kiến nghị về chương trình học, công tác tổ chức, chương trình đào tạo
- Trả lời: Có hoặc không có kiến nghị

}

}

### 3.3.2 Chèn dữ liệu mẫu

Dữ liệu mẫu bao gồm thông tin của 10 sinh viên, mỗi sinh viên có đầy đủ thông tin cá nhân và câu trả lời khảo sát.

Lệnh chèn dữ liệu mẫu:

Dữ liệu mẫu 1:

```
db.sinhVien.insertOne(
{
  "_id": "001",
  "sinhVien": {
    "thongtinsv": {
      "mssv": "001",
      "hoTen": "Nguyễn Thanh Tùng",
      "lop" : "DA22TTA",
```



Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
"khoa": "Kỹ thuật và Công nghệ",
"nganh": "Công nghệ thông tin",
"sdt": "0123456789",
"mailSv": "001@st.tvu.edu.vn",
},
"khaosat": {
  "ngayKhaoSat": "2024-11-20",
  "phan1": {
    "mucDoDongY": {
      "chuongTrinhDaoTao": {
        "1. Ngành học có mục tiêu rõ ràng và phù hợp với yêu cầu xã hội": 5,
        "2. Chương trình học cung cấp kiến thức thực tế": 5,
        "3. Cấu trúc chương trình đào tạo linh hoạt, tạo điều kiện thuận lợi cho sinh viên": 5,
        "4. Tỷ lệ phân bố giữa lý thuyết và thực hành hợp lý": 5,
        "5. Các kỳ thi, kiểm tra đảm bảo nghiêm túc, khách quan, công bằng": 5,
        "6. Tài liệu đáp ứng được nhu cầu học tập của SV": 5,
        "7. Giảng viên khuyến khích sinh viên chủ động tham gia vào các hoạt động học tập": 5,
        "8. Chương trình học giúp sinh viên phát triển những kỹ năng mềm": 5,
        "9. Chương trình cung cấp các kiến thức và kỹ năng cần thiết cho công việc": 5
      },
      "cacHoatDongHoTro": {
        "10. Các hoạt động định hướng nghề nghiệp truyền cảm hứng giúp bạn yêu nghề": 5,
        "11. Các hoạt động tư vấn, hỗ trợ việc làm của trường giúp sinh viên thuận lợi trong quá trình tìm việc": 5,
        "12. Các hoạt động ngoại khóa được phổ biến rộng rãi đến SV": 5,
        "13. Chương trình hoạt động ngoại khóa phù hợp và giúp SV rèn luyện kỹ năng": 5,
        "14. Giờ mở cửa thư viện phù hợp với nhu cầu học tập của SV": 5,
        "15. Thủ tục mượn trả sách nhanh chóng": 5
      }
    }
  }
}
```

```
    },  
    "coSoVatChat": {  
        "16. Phòng học thoáng mát, đủ ánh sáng": 5,  
        "17. Trang thiết bị phòng học hoạt động tốt": 5,  
        "18. Hệ thống internet, wifi đáp ứng yêu cầu": 5,  
        "19. Website của khoa, trường cung cấp thông tin cần  
thiết": 5  
    }  
}  
},  
"phan2": {  
    "kienNghị": "không có kiến nghị"  
}  
}  
}  
});
```

## CHƯƠNG 4 KẾT QUẢ NGHIÊN CỨU

### 4.1 Kết quả đạt được

Qua quá trình nghiên cứu và thực nghiệm, kết quả sau khi hoàn thành đề tài là một cơ sở dữ liệu cho việc khảo sát sinh viên cuối khóa. Mô hình này được triển khai bằng hệ quản trị cơ sở dữ liệu MongoDB với các đặc điểm chính:

- Cấu trúc lưu trữ dữ liệu phi quan hệ giúp dễ dàng mở rộng và tối ưu cho các dạng dữ liệu phức tạp.
- Tính linh hoạt trong việc bổ sung các câu hỏi khảo sát hoặc thay đổi cấu trúc phiếu khảo sát.
- Hỗ trợ truy vấn nhanh.

### 4.2 Các câu lệnh truy vấn

Một số câu lệnh truy vấn:

#### 1) Câu lệnh truy xuất ra thông tin một sinh viên

```
db.sinhVien.find(  
  { _id: "001" }, // Điều kiện để tìm sinh viên cụ thể  
  { "sinhVien.thongtinsv": 1, _id: 0 } // Chỉ lấy thông tin sinh  
viên, ẩn `_id`  
);
```

Kết quả:

```
{  
  sinhVien: {  
    thongtinsv: {  
      mssv: '001',  
      hoTen: 'Nguyễn Thanh Tùng',  
      lop: 'DA22TTA',  
      khoa: 'Kỹ thuật và Công nghệ',  
      ngành: 'Công nghệ thông tin',  
      sdt: '0123456789',  
      mailSv: '001@st.tvu.edu.vn'  
    }  
  }  
}
```

## 2) Câu lệnh truy xuất ra thông tin một sinh viên và sinh viên trả lời khảo sát

```
db.sinhVien.find(  
  {  
    "sinhVien.thongtinsv.mssv": "010" // Tìm sinh viên có mssv là  
010  
  },  
  {  
    "sinhVien.thongtinsv": 1, // Chỉ lấy thông tin sinh viên  
    "sinhVien.khaosat": 1,    // Lấy câu trả lời khảo sát  
    "_id": 0                  // Loại bỏ `_id` trong kết quả  
  }  
);
```

Kết quả:

```
{  
  sinhVien: {  
    thongtinsv: {  
      mssv: '010',  
      hoTen: 'Trần Văn Quyết',  
      lop: 'DA22THB',  
      khoa: 'Su phạm',  
      ngành: 'Su phạm Tiểu học',  
      sdt: '0123456789',  
      mailSv: '010@st.tvu.edu.vn'  
    },  
    khaosat: {  
      ngayKhaoSat: '2024-11-20',  
      phan1: {  
        mucDoDongY: {  
          chươngTrìnhĐàoTạo: {  
            '1. Ngành học có mục tiêu rõ ràng và phù hợp với yêu  
cầu xã hội': 5,  
            '2. Chương trình học cung cấp kiến thức thực tế': 5,  
            '3. Cấu trúc chương trình đào tạo linh hoạt, tạo điều  
kiện thuận lợi cho sinh viên': 5,  
            '4. Tỷ lệ phân bố giữa lý thuyết và thực hành hợp lý':  
5,  

```

## Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
'5. Các kỳ thi, kiểm tra đảm bảo nghiêm túc, khách
quan, công bằng': 5,
'6. Tài liệu đáp ứng được nhu cầu học tập của SV': 5,
'7. Giảng viên khuyến khích sinh viên chủ động tham
gia vào các hoạt động học tập': 5,
'8. Chương trình học giúp sinh viên phát triển những
kỹ năng mềm': 5,
'9. Chương trình cung cấp các kiến thức và kỹ năng cần
thiết cho công việc ': 5
},
cacHoatDongHoTro: {
'10. Các hoạt động định hướng nghề nghiệp truyền cảm
hứng giúp bạn yêu nghề': 5,
'11. Các hoạt động tư vấn, hỗ trợ việc làm của trường
giúp sinh viên thuận lợi trong quá trình tìm việc': 5,
'12. Các hoạt động ngoại khóa được phổ biến rộng rãi
đến SV': 5,
'13. Chương trình hoạt động ngoại khóa phù hợp và giúp
SV rèn luyện kỹ năng': 5,
'14. Giờ mở cửa thư viện phù hợp với nhu cầu học tập
của SV': 5,
'15. Thủ tục mượn trả sách nhanh chóng': 5
},
coSoVatChat: {
'16. Phòng học thoáng mát, đủ ánh sáng': 4,
'17. Trang thiết bị phòng học hoạt động tốt': 4,
'18. Hệ thống internet, wifi đáp ứng yêu cầu': 4,
'19. Website của khoa, trường cung cấp thông tin cần
thiết': 4
}
}
},
phan2: {
kienNghị: 'không có kiến nghị'
}
}
}
```

### 3) Câu lệnh truy xuất ra thông tin một sinh viên và câu trả lời khảo sát mục coSoVatChat của sinh viên đó

```
db.sinhVien.find( {
  "sinhVien.thongtinsv.mssv": "004" // Thay "." bằng MSSV của sinh
  viên cần tìm
}, {
  "sinhVien.thongtinsv": 1,          // Chỉ lấy thông tin sinh viên
  "sinhVien.khaosat.phan1.mucDoDongY.coSoVatChat": 1, // Lấy mục
  coSoVatChat
  _id: 0 } );                        // Loại bỏ `_id` trong kết quả
```

Kết quả:

```
{
  sinhVien: {
    thongtinsv: {
      mssv: '004',
      hoTen: 'Bùi Ngọc Hằng',
      lop: 'DA22TTC',
      khoa: 'Kỹ thuật và Công nghệ',
      ngành: 'Công nghệ thông tin',
      sdt: '01234567893',
      mailSv: '004@st.tvu.edu.vn'
    },
    khaosat: {
      phan1: {
        mucDoDongY: {
          coSoVatChat: {
            '16. Phòng học thoáng mát, đủ ánh sáng': 5,
            '17. Trang thiết bị phòng học hoạt động tốt': 5,
            '18. Hệ thống internet, wifi đáp ứng yêu cầu': 4,
            '19. Website của khoa, trường cung cấp thông tin cần
            thiết': 5
          }
        }
      }
    }
  }
}
```

#### 4) Câu lệnh đếm số sinh viên của Khoa Kỹ thuật và Công nghệ khảo sát mục cơ sở vật chất của mục mức độ đồng ý trong phần 1

```
>db.sinhVien.find({"sinhVien.thongtinsv.khoa":"Kỹ thuật và Công  
nghệ"},  
{ "sinhVien.khaoSat.phan1.mucDoDongY.coSoVatChat":1}).count()
```

Kết quả

```
< 10
```

#### 5) Câu lệnh xuất ra danh sách các khoa tham gia khảo sát

```
>db.sinhVien.aggregate([  
  {  
    $group: {  
      _id: "$sinhVien.thongtinsv.khoa" // Nhóm theo tên khoa  
    }  
  },  
  {  
    $project: {  
      khoa: "$_id", // Đặt tên khoa vào trường `khoa`  
      _id: 0 // Ẩn `_id` mặc định  
    }  
  }  
]);
```

Kết quả:

```
<{  
  khoa: 'Kỹ thuật và Công nghệ'  
}  
{  
  khoa: 'Su phạm'  
}
```

## 6) Câu lệnh xem danh sách các ngành trong mỗi khoa

```
>db.sinhVien.aggregate([
  {
    $group: {
      _id: {
        khoa: "$sinhVien.thongtinsv.khoa"
      },
      danhSachNganh: { $addToSet: "$sinhVien.thongtinsv.nganh" }
    }
  },
  {
    $project: {
      khoa: "$_id.khoa",
      danhSachNganh: 1,
      _id: 0
    }
  }
]);
```

Kết quả:

```
<{
  danhSachNganh: [
    'Su phạm mầm non',
    'Su phạm Tiểu học'
  ],
  khoa: 'Su phạm'
}
```

## 7) Câu lệnh đếm số sinh viên của Khoa Sư phạm tham gia khảo sát

```
>db.sinhVien.find({'sinhVien.thongtinsv.khoa':'Sư phạm'},
{'sinhVien.khaosat':1}).count()
```

Kết quả:

```
< 10
```



## 8) Câu lệnh xuất ra danh sách các lớp có mã lớp “DA22TTC” tham gia khảo sát

```
>db.sinhVien.find(
  {
    "sinhVien.thongtinsv.lop": "DA22TTC" // Điều kiện lọc lớp
    DA22TTC
  },
  {
    "sinhVien.thongtinsv": 1, // Lấy thông tin sinh viên
    "sinhVien.khaosat": 1,   // Lấy thông tin khảo sát
    _id: 0                    // Loại bỏ `_id` trong kết quả
  }
);
```

Kết quả: có 3 sinh viên thuộc lớp “DA22TTC” tham gia khảo sát

```
<{
  sinhVien: {
    thongtinsv: {
      mssv: '003',
      hoTen: 'Trần Thanh Tâm',
      lop: 'DA22TTC',
      khoa: 'Kỹ thuật và Công nghệ',
      ngành: 'Công nghệ thông tin',
      sdt: '01234567892',
      mailSv: '003@st.tvu.edu.vn'
    },
    khaosat: {
      ngayKhaoSat: '2024-11-20',
      phan1: {
        mucDoDongY: {
          chuongTrinhDaoTao: {
            '1. Ngành học có mục tiêu rõ ràng và phù hợp với yêu
            cầu xã hội': 4,
            '2. Chương trình học cung cấp kiến thức thực tế': 4,
            '3. Cấu trúc chương trình đào tạo linh hoạt, tạo điều
            kiện thuận lợi cho sinh viên': 4,
            '4. Tỷ lệ phân bố giữa lý thuyết và thực hành hợp lý':
            4,
```

## Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
'5. Các kỳ thi, kiểm tra đảm bảo nghiêm túc, khách
quan, công bằng': 4,
    '6. Tài liệu đáp ứng được nhu cầu học tập của SV': 4,
    '7. Giảng viên khuyến khích sinh viên chủ động tham
gia vào các hoạt động học tập': 5,
    '8. Chương trình học giúp sinh viên phát triển những
kỹ năng mềm': 5,
    '9. Chương trình cung cấp các kiến thức và kỹ năng cần
thiết cho công việc ': 5
    },
    cacHoatDongHoTro: {
        '10. Các hoạt động định hướng nghề nghiệp truyền cảm
hứng giúp bạn yêu nghề': 4,
        '11. Các hoạt động tư vấn, hỗ trợ việc làm của trường
giúp sinh viên thuận lợi trong quá trình tìm việc': 4,
        '12. Các hoạt động ngoại khóa được phổ biến rộng rãi
đến SV': 5,
        '13. Chương trình hoạt động ngoại khóa phù hợp và giúp
SV rèn luyện kỹ năng': 4,
        '14. Giờ mở cửa thư viện phù hợp với nhu cầu học tập
của SV': 5,
        '15. Thủ tục mượn trả sách nhanh chóng': 5
    },
    coSoVatChat: {
        '16. Phòng học thoáng mát, đủ ánh sáng': 4,
        '17. Trang thiết bị phòng học hoạt động tốt': 4,
        '18. Hệ thống internet, wifi đáp ứng yêu cầu': 4,
        '19. Website của khoa, trường cung cấp thông tin cần
thiết': 5
    }
    },
    phan2: {
        kienNghì: 'không có kiến nghị'
    }
    }
    }
    {
    {
```

## Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
sinhVien: {
  thongtinsv: {
    mssv: '004',
    hoTen: 'Bùi Ngọc Hằng',
    lop: 'DA22TTC',
    khoa: 'Kỹ thuật và Công nghệ',
    ngành: 'Công nghệ thông tin',
    sdt: '01234567893',
    mailSv: '004@st.tvu.edu.vn'
  },
  khaosat: {
    ngayKhaoSat: '2024-11-20',
    phan1: {
      mucDoDongY: {
        chuongTrinhDaoTao: {
          '1. Ngành học có mục tiêu rõ ràng và phù hợp với yêu cầu xã hội': 5,
          '2. Chương trình học cung cấp kiến thức thực tế': 5,
          '3. Cấu trúc chương trình đào tạo linh hoạt, tạo điều kiện thuận lợi cho sinh viên': 5,
          '4. Tỷ lệ phân bố giữa lý thuyết và thực hành hợp lý': 5,
          '5. Các kỳ thi, kiểm tra đảm bảo nghiêm túc, khách quan, công bằng': 5,
          '6. Tài liệu đáp ứng được nhu cầu học tập của SV': 5,
          '7. Giảng viên khuyến khích sinh viên chủ động tham gia vào các hoạt động học tập': 5,
          '8. Chương trình học giúp sinh viên phát triển những kỹ năng mềm': 5,
          '9. Chương trình cung cấp các kiến thức và kỹ năng cần thiết cho công việc ': 5
        },
        cacHoatDongHoTro: {
          '10. Các hoạt động định hướng nghề nghiệp truyền cảm hứng giúp bạn yêu nghề': 4,
          '11. Các hoạt động tư vấn, hỗ trợ việc làm của trường giúp sinh viên thuận lợi trong quá trình tìm việc': 4,
          '12. Các hoạt động ngoại khóa được phổ biến rộng rãi đến SV': 4,

```

## Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
'13. Chương trình hoạt động ngoại khóa phù hợp và giúp
SV rèn luyện kỹ năng': 4,
      '14. Giờ mở cửa thư viện phù hợp với nhu cầu học tập
của SV': 4,
      '15. Thủ tục mượn trả sách nhanh chóng': 4
    },
    coSoVatChat: {
      '16. Phòng học thoáng mát, đủ ánh sáng': 5,
      '17. Trang thiết bị phòng học hoạt động tốt': 5,
      '18. Hệ thống internet, wifi đáp ứng yêu cầu': 4,
      '19. Website của khoa, trường cung cấp thông tin cần
thiết': 5
    }
  },
  phan2: {
    kienNghĩ: 'không có'
  }
}
{
  sinhVien: {
    thongtinsv: {
      mssv: '005',
      hoTen: 'Lê Quỳnh Như',
      lop: 'DA22TTC',
      khoa: 'Kỹ thuật và Công nghệ',
      ngành: 'Công nghệ thông tin',
      sdt: '01234567894',
      mailSv: '005@st.tvu.edu.vn'
    },
    khaosat: {
      ngayKhaoSat: '2024-11-20',
      phan1: {
        mucDoDongY: {
          chuongTrinhDaoTao: {
            '1. Ngành học có mục tiêu rõ ràng và phù hợp với yêu
cầu xã hội': 4,
```

## Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
'2. Chương trình học cung cấp kiến thức thực tế': 4,
'3. Cấu trúc chương trình đào tạo linh hoạt, tạo điều
kiện thuận lợi cho sinh viên': 5,
'4. Tỷ lệ phân bố giữa lý thuyết và thực hành hợp lý':
5,
'5. Các kỳ thi, kiểm tra đảm bảo nghiêm túc, khách
quan, công bằng': 4,
'6. Tài liệu đáp ứng được nhu cầu học tập của SV': 5,
'7. Giảng viên khuyến khích sinh viên chủ động tham
gia vào các hoạt động học tập': 4,
'8. Chương trình học giúp sinh viên phát triển những
kỹ năng mềm': 4,
'9. Chương trình cung cấp các kiến thức và kỹ năng cần
thiết cho công việc ': 5
},
cacHoatDongHoTro: {
'10. Các hoạt động định hướng nghề nghiệp truyền cảm
hứng giúp bạn yêu nghề': 4,
'11. Các hoạt động tư vấn, hỗ trợ việc làm của trường
giúp sinh viên thuận lợi trong quá trình tìm việc': 4,
'12. Các hoạt động ngoại khóa được phổ biến rộng rãi
đến SV': 5,
'13. Chương trình hoạt động ngoại khóa phù hợp và giúp
SV rèn luyện kỹ năng': 4,
'14. Giờ mở cửa thư viện phù hợp với nhu cầu học tập
của SV': 5,
'15. Thủ tục mượn trả sách nhanh chóng': 5
},
coSoVatChat: {
'16. Phòng học thoáng mát, đủ ánh sáng': 5,
'17. Trang thiết bị phòng học hoạt động tốt': 5,
'18. Hệ thống internet, wifi đáp ứng yêu cầu': 4,
'19. Website của khoa, trường cung cấp thông tin cần
thiết': 5
}
}
},
phan2: {
kienNghị: 'không có kiến nghị'
```

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

```
}  
}  
}  
}
```

## CHƯƠNG 5 KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

### 5.1 Kết luận

#### 5.1.1 Kết luận chung

Đề tài "*Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL*" đã hoàn thành các mục tiêu đặt ra và đạt được những kết quả:

##### **Xây dựng mô hình cơ sở dữ liệu:**

Cơ sở dữ liệu NoSQL được cài đặt trong hệ quản trị cơ sở dữ liệu MongoDB được thiết kế với cấu trúc tối ưu, đáp ứng được yêu cầu lưu trữ thông tin đa dạng và linh hoạt của phiếu khảo sát.

Mô hình cho phép lưu trữ cả dữ liệu có cấu trúc (thông tin cá nhân sinh viên) và dữ liệu phi cấu trúc (câu trả lời dạng text, đánh giá mức độ hài lòng).

##### **Đáp ứng nhu cầu phân tích dữ liệu:**

Các truy vấn trên MongoDB được triển khai để phân tích dữ liệu khảo sát, từ đó trích xuất các thông tin hữu ích về mức độ hài lòng trung bình, các ý kiến đóng góp của sinh viên.

##### **Tính ứng dụng cao:**

Hỗ trợ việc khảo sát sinh viên cuối khóa, cung cấp thông tin quan trọng giúp nhà trường cải thiện chương trình đào tạo, các hoạt động hỗ trợ, và cơ sở vật chất.

##### **Hiệu quả của việc sử dụng NoSQL:**

Tính năng linh hoạt của MongoDB đảm bảo hệ thống có thể mở rộng và dễ dàng nâng cấp theo nhu cầu thực tế.

#### 5.1.2 Hạn chế

Do thời gian thực hiện đề tài có hạn nên cơ sở dữ liệu NoSQL chưa được triển khai thực tế tại các trường học để đánh giá hiệu quả và tính thực tiễn trong môi trường thực tế.

### 5.2 Hướng phát triển

Một số định hướng phát triển trong tương lai được đề xuất như sau:

### **Mở rộng quy mô dữ liệu:**

Thu thập dữ liệu khảo sát từ nhiều đối tượng sinh viên hơn, bao gồm các khoa khác nhau hoặc các khóa học trước đó.

Tích hợp các câu hỏi khảo sát động, cho phép tùy chỉnh nội dung phiếu khảo sát theo thời gian.

### **Tích hợp với hệ thống ứng dụng:**

Xây dựng giao diện người dùng trực quan (web hoặc ứng dụng di động) để sinh viên dễ dàng tham gia khảo sát và dễ dàng quản lý kết quả.

### **Tối ưu hiệu năng:**

Nghiên cứu và triển khai các giải pháp tối ưu hóa truy vấn và lưu trữ dữ liệu, đặc biệt khi hệ thống phải xử lý dữ liệu lớn.

Cân nhắc sử dụng kiến trúc phân tán hoặc cơ chế sharding của MongoDB để tăng khả năng chịu tải.



## DANH MỤC TÀI LIỆU THAM KHẢO

- [1] “TÌM HIỂU VỀ NOSQL”. Truy cập: 27 Tháng Mười-Một 2024. [Online].  
Available at: <https://viblo.asia/p/tim-hieu-ve-nosql-Zzb7vDNYMjKd>
- [2] “NoSQL là gì? Tổng quan về NoSQL database”. Truy cập: 15 Tháng Chạp 2024. [Online]. Available at: <https://bizflycloud.vn/tin-tuc/nosql-la-gi-20181013113252686.htm>
- [3] “NoSQL”, *Wikipedia tiếng Việt*. 23 Tháng Ba 2024. Truy cập: 23 Tháng Mười-Một 2024. [Online]. Available at:  
<https://vi.wikipedia.org/w/index.php?title=NoSQL&oldid=71237922>
- [4] J. Carder, “RDBMS vs. NoSQL: Key Differences & Migration Tips | OpenLogic”. Truy cập: 15 Tháng Chạp 2024. [Online]. Available at:  
<https://www.openlogic.com/blog/rdbms-vs-nosql>
- [5] “NoSQL Là Gì? So sánh RDBMS và NoSQL - VinaHost”. Truy cập: 15 Tháng Chạp 2024. [Online]. Available at: <https://blog.vinahost.vn/nosql-la-gi/>
- [6] A. Singh, “The Power of MongoDB: Understanding Its Features and Capabilities”, Medium. Truy cập: 15 Tháng Chạp 2024. [Online]. Available at:  
[https://medium.com/@Ambarish\\_224/the-power-of-mongodb-understanding-its-features-and-capabilities-6f70dac2d306](https://medium.com/@Ambarish_224/the-power-of-mongodb-understanding-its-features-and-capabilities-6f70dac2d306)
- [7] “Mongodb là gì? tính năng nổi bật của mongodb mà bạn cần biết”, 200Lab Blog. Truy cập: 31 Tháng Chạp 2024. [Online]. Available at:  
<https://200lab.io/blog/mongodb-la-gi/>
- [8] Andreas Meier, Michael Kaufmann, “SQL & NoSQL Databases\_ Models, Languages, Consistency Options and Architectures for Big Data Management- Springer Vieweg (2019)”.
- [9] Kyle Banker, Peter Bakkum, Shaun Verch, Douglas Garrett, Tim Hawkins, “MongoDB-in-Action-2nd-Edition”.
- [10] Shannon Bradshaw, Eoin Brazil, Kristina Chodorow, “MongoDB\_ The Definitive Guide\_ Powerful and Scalable Data Storage-O’Reilly Media (2019)”.
- [11] K. Nguyễn, “Hướng dẫn cài đặt MongoDB trên hệ điều hành Windows”, Blog Chia Sẻ Kiến Thức. Truy cập: 28 Tháng Mười-Một 2024. [Online].  
Available at: <https://blogchiasekienthuc.com/thu-thuat-hay/cai-dat-mongodb-tren-windows.html>

## PHỤ LỤC

### 1. Thông tin mẫu phiếu khảo sát

**PHIẾU KHẢO SÁT SINH VIÊN CUỐI KHÓA**

Nhằm đáp ứng yêu cầu đánh giá chất lượng đào tạo và thực hiện cải tiến trong tương lai, mong các bạn SV vui lòng dành ít thời gian giúp chúng tôi hoàn thành bảng câu hỏi này.

**Đánh dấu (✓) vào ô mà bạn chọn**

**Mã lớp:**.....**Ngày khảo sát:**...../...../.....

**1. Mức độ đồng ý của bạn đối với những nội dung sau đây như thế nào?**

Stt	Nội dung	1. Rất không đồng ý	2. Không đồng ý	3. Trung bình	4. Đồng ý	5. Rất đồng ý
I	<b>Chương trình đào tạo</b>					
1.1.	Ngành học có mục tiêu rõ ràng và phù hợp với yêu cầu xã hội					
1.2.	Chương trình học cung cấp kiến thức thực tế					
1.3.	Cấu trúc chương trình đào tạo linh hoạt, tạo điều kiện thuận lợi cho sinh viên					
1.4.	Tỷ lệ phân bổ giữa lý thuyết và thực hành hợp lý					
1.5.	Các kỹ thuật, kiểm tra đảm bảo nghiêm túc, khách quan, công bằng					
1.6.	Tài liệu đáp ứng được nhu cầu học tập của SV					
1.7.	Giảng viên khuyến khích sinh viên chủ động tham gia vào các hoạt động học tập					
1.8.	Chương trình học giúp sinh viên phát triển những kỹ năng mềm					
1.9.	Chương trình cung cấp các kiến thức và kỹ năng cần thiết cho công việc					
II	<b>Các hoạt động hỗ trợ</b>					
1.10.	Các hoạt động định hướng nghề nghiệp truyền cảm hứng giúp bạn yêu nghề					
1.11.	Các hoạt động tư vấn, hỗ trợ việc làm của trường giúp sinh viên thuận lợi trong quá trình tìm việc					
1.12.	Các hoạt động ngoại khóa được phổ biến rộng rãi SV					
1.13.	Chương trình hoạt động ngoại khóa phù hợp và giúp SV rèn luyện kỹ năng					
1.14.	Giờ mở cửa thư viện phù hợp với nhu cầu học tập của SV					
1.15.	Thủ tục mượn trả sách nhanh chóng					
III	<b>Cơ sở vật chất</b>					
1.16.	Phòng học thoáng mát, đủ ánh sáng					
1.17.	Trang thiết bị phòng học hoạt động tốt					
1.18.	Hệ thống internet, wifi đáp ứng yêu cầu					
1.19.	Website của khoa, trường cung cấp thông tin cần thiết					

**3. Những kiến nghị của bạn về chương trình học, về công tác tổ chức, đào tạo của nhà trường:**

.....

.....

.....

.....

**Thông tin đáp viên:** Họ và tên:.....số điện thoại .....email .....

**Cảm ơn bạn, chúc bạn thành công!**

End of document

Hình 1 Mẫu phiếu khảo sát

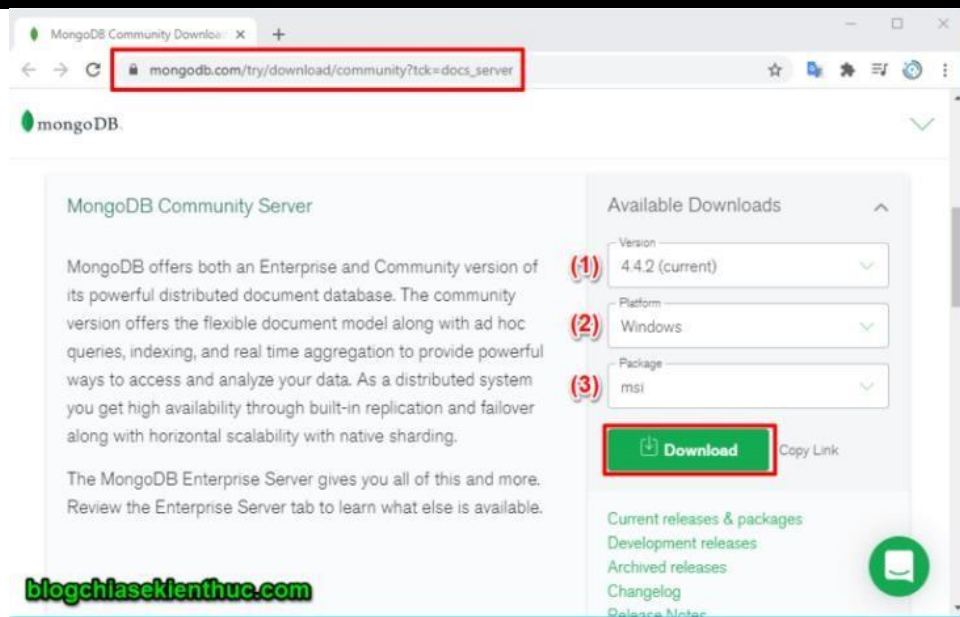
[Nguồn: <https://dbcl.tvu.edu.vn/post/%C4%90BCL-KSCK-BMKS>]

### 2. Cách cài đặt MongoDB

**Bước 1:** Truy cập vào đường dẫn sau để download file cài đặt của MongoDB tại đây[11]:

[https://www.mongodb.com/try/download/community?tick=docs\\_server](https://www.mongodb.com/try/download/community?tick=docs_server)

1. Là phiên bản hiện tại của Mongodb, phiên bản cộng đồng (Free).
2. Hệ điều hành (ở đây chúng ta chọn Windows).
3. Loại file cài đặt (ở đây chúng ta chọn file \*.msi).



Hình 2. Hướng dẫn cài đặt MongoDB

**Bước 2:** Các bạn có thể bấm vào biểu tượng mũi tên bên cạnh để chọn các option phù hợp, rồi bấm vào nút Download để tải file cài đặt về.

Sau khi tải file cài đặt về, các bạn nhấp đúp chuột vào file để bắt đầu quá trình cài đặt và bấm Next để chuyển sang bước tiếp theo.



Hình 3. Hướng dẫn cài đặt MongoDB

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

**Bước 3:** Ở bước này các bạn tích vào I accept the terms in the License Agreement, sau đó bấm vào Next để sang bước tiếp theo[11].



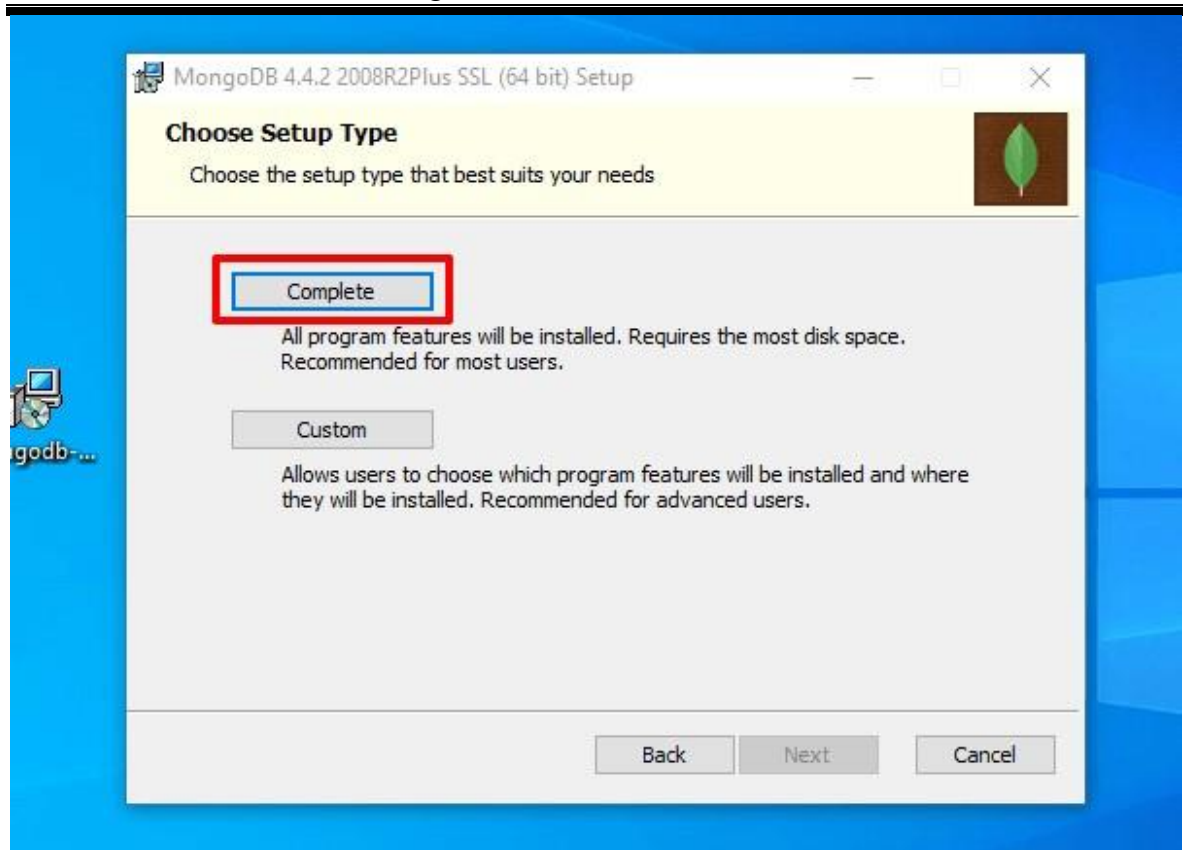
Hình 4. Hướng dẫn cài đặt MongoDB

Bước 4: Ở đây sẽ có hai option đó là hoặc bạn bấm vào Complete hoặc Custom (Tùy chọn).

Nếu bạn chọn Complete thì tất cả các chức năng sẽ được cài đặt và tất nhiên là sẽ chiếm nhiều dung lượng ổ cứng hơn.

Còn nếu bạn chọn Custom thì bạn sẽ chỉ chọn những chức năng bạn muốn cài đặt và tùy thuộc vào chức năng bạn chọn dung lượng ổ cứng cài đặt sẽ giảm đi.

Chúng ta thấy MongoDB gợi ý chúng ta (người dùng mới) thì nên cài đặt theo option đầu tiên (Complete). Vì vậy các bạn cứ bấm vào Complete để tiếp tục nha[11].



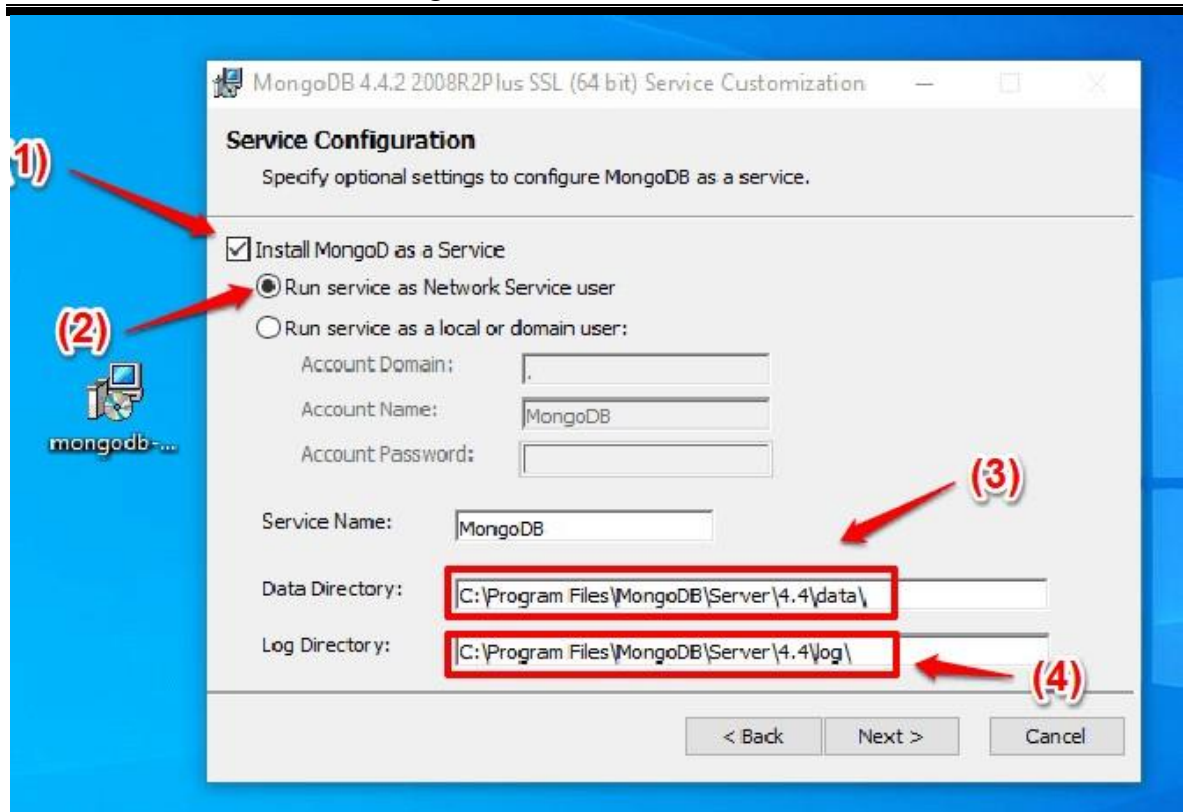
Hình 5. Hướng dẫn cài đặt MongoDB

Bước 5: Tiếp đến bước cấu hình service[11].

- (1) Cài đặt MongoDB như là một service. Nó giống như những service bình thường của hệ điều hành. Bạn có thể bật/tắt được.
- (2) Chạy service với vai trò là người sử dụng mạng dịch vụ.
- (3) Nơi lưu trữ Database.
- (4) Nơi chứa file log (các server chạy đều có file log, các bạn có thể xem log file xem server đã chạy các service nào, gặp lỗi ở đâu...) Sau đó các bạn bấm Next để tiếp tục.

Sau đó các bạn bấm Next để tiếp tục[11].

Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL



Hình 6. Hướng dẫn cài đặt MongoDB

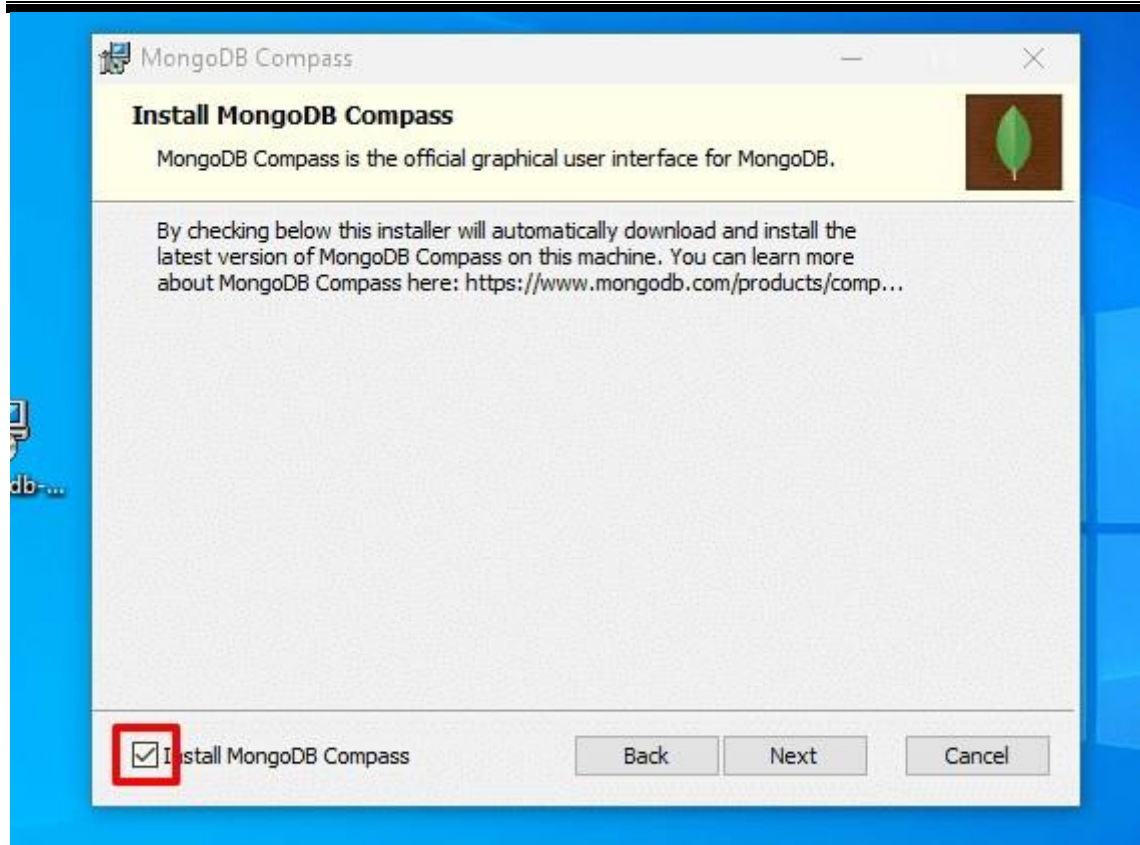
Bước 6: Bước tiếp theo này thì các bạn sẽ lựa chọn là có cài MongoDB Compass hay không.

Cho bạn nào chưa biết thì MongoDB Compass là một công cụ, chính xác hơn là một giao diện người dùng giúp cho việc thao tác với dữ liệu trong MongoDB trực quan hơn.

Bạn tích vào ô Install MongoDB Compass để cài đặt và bấm Next để đến bước cài đặt.

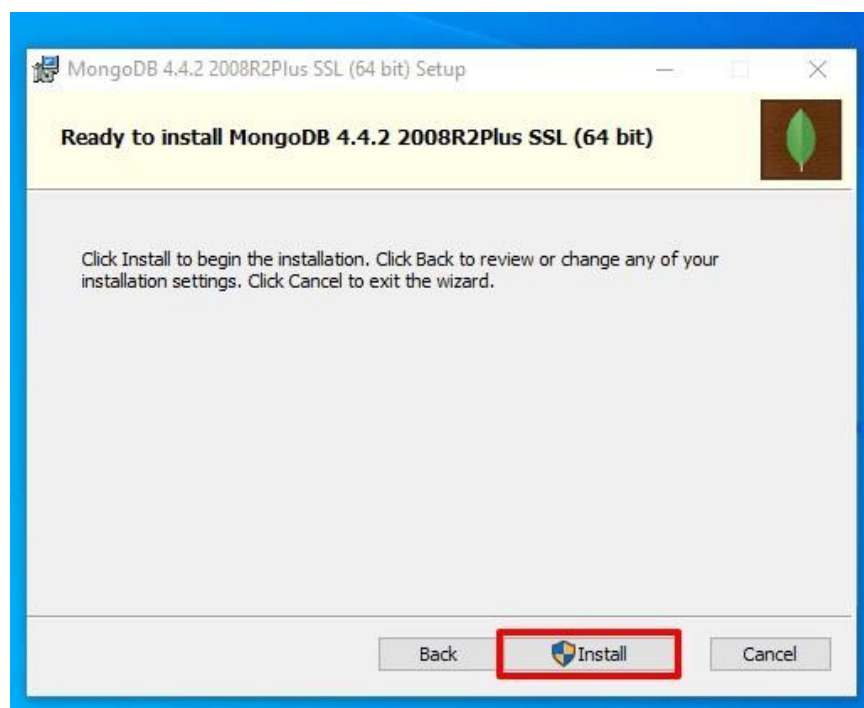


Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL



Hình 7. Hướng dẫn cài đặt MongoDB

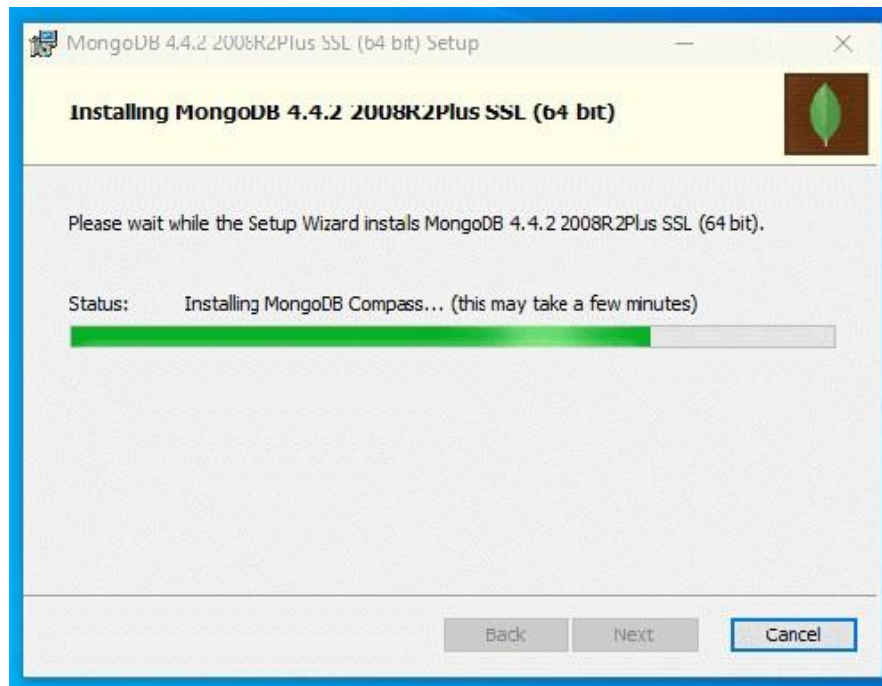
Bước 7: Đến bước này thì quá trình cấu hình đã xong và bắt đầu cài đặt.  
Các bạn bấm vào Install để bắt đầu quá trình cài đặt[11].



Hình 8. Hướng dẫn cài đặt MongoDB

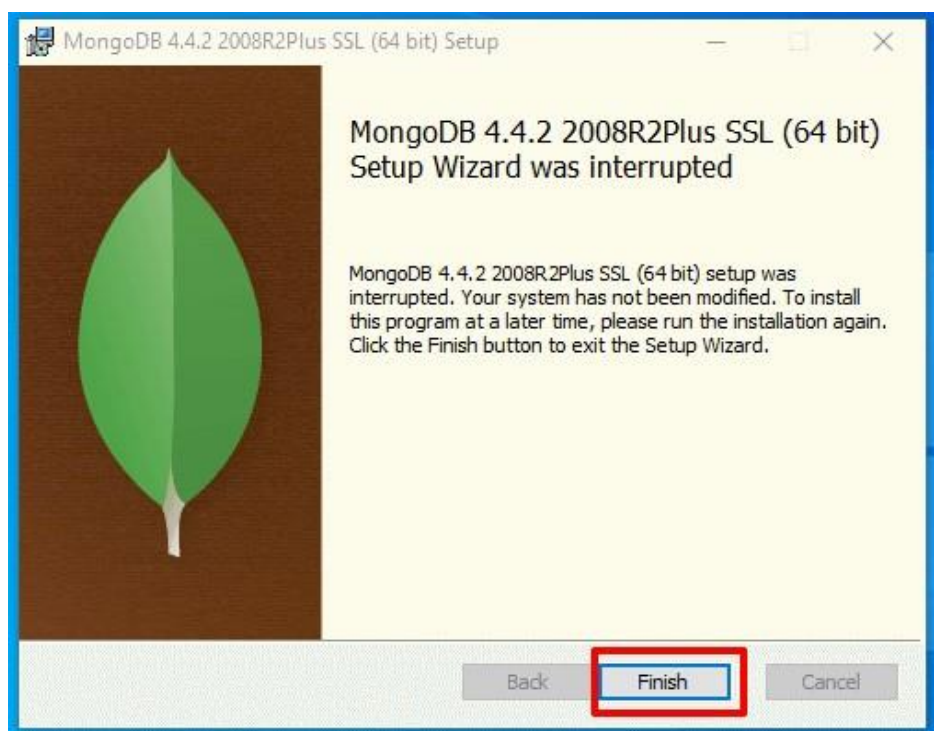
Thiết kế và cài đặt cơ sở dữ liệu cho việc khảo sát mức độ hài lòng của sinh viên trong việc khảo sát sinh viên cuối khóa bằng NoSQL

Quá trình cài đặt đang diễn ra, nhanh thì có thể 2-3 phút là xong, chậm có thể mất 5-7 phút tùy thuộc vào kết nối mạng và cấu hình máy nữa.



Hình 9. Hướng dẫn cài đặt MongoDB

Bước 8: Sau khi cài đặt xong, bấm Finish để hoàn tất quá trình cài đặt.



Hình 10. Hướng dẫn cài đặt MongoDB

Sau khi cài đặt thì các bạn có thể khởi động MongoDB Compass