Shape, square

Description automatically generated

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**

**ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA HỆ THỐNG THÔNG TIN**

——————

Logo, company name

Description automatically generated Logo

Description automatically generated

**Bài tập lớn 2**

**Cơ chế phân tán trong hệ quản trị RavenDB**

**Môn học:**

**CƠ SỞ DỮ LIỆU PHÂN TÁN**

Giảng viên hướng dẫn: **Nguyễn Minh Nhựt**

Lớp: IS211.O11.HTCL

**Sinh viên thực hiện:**

Đỗ Huỳnh Mỹ Tâm - 20520746

Lưu Vĩnh Phát - 20521733

Trần Văn Thế - 20520770

Nguyễn Minh Duy - 21522005

**Mục Lục**

[Chương 1. Giới thiệu 4](#_Toc155221994)

[1.1. Giới thiệu về hệ quản trị CSDL NoSQL 4](#_Toc155221995)

[1.2. Giới thiệu về RavenDB 5](#_Toc155221996)

[1.2.1.Tổng quan 5](#_Toc155221997)

[1.2.2.Đặc điểm 5](#_Toc155221998)

[1.2.3.Ưu điểm 6](#_Toc155221999)

[1.2.4.Nhược điểm 7](#_Toc155222000)

[Chương 2. Cài đặt ravendb 8](#_Toc155222001)

[2.1. Cài đặt ravendb trên hai máy 8](#_Toc155222002)

[2.2. Kết nối hai máy bằng Radmin 13](#_Toc155222003)

[Chương 3. Thiết kế CSDL 14](#_Toc155222004)

[3.1. Mô trả về cơ sở dữ liệu 14](#_Toc155222005)

[3.2. Dữ liệu các bảng 15](#_Toc155222006)

[3.2.1. Bảng CUAHANG 15](#_Toc155222007)

[3.2.2. Bảng HAISAN 16](#_Toc155222008)

[4. Chương 4:Thao tác dữ liệu qua lại giữa hai máy 20](#_Toc155222009)

[4.1. Tạo bảng 22](#_Toc155222010)

[4.1.1. Bảng hải sản 22](#_Toc155222011)

[4.1.2. Bảng cửa hàng 22](#_Toc155222012)

[4.1.3. Bảng hóa đơn 22](#_Toc155222013)

[4.1.4. Bảng Chi tiết hóa đơn 22](#_Toc155222014)

[4.1.5. Bảng khách hàng 23](#_Toc155222015)

[4.1.6. Bảng nhân viên bán hàng 23](#_Toc155222016)

[4.1.7. Bảng quản lý kho 24](#_Toc155222017)

[4.2. Thêm dữ liệu 24](#_Toc155222018)

[4.2.1. Bảng cửa hàng 24](#_Toc155222019)

[4.2.2. Bảng hải sản 25](#_Toc155222020)

[4.2.3. Bảng hóa đơn 26](#_Toc155222021)

[4.2.4. Bảng chi tiết hóa đơn 27](#_Toc155222022)

[4.2.5. Bảng khách hàng 28](#_Toc155222023)

[4.2.6. Bảng nhân viên 29](#_Toc155222024)

[4.2.7. Bảng nhân viên bán hàng 30](#_Toc155222025)

[4.2.8. Bảng quản lý kho 31](#_Toc155222026)

[4.3. Thêm, xóa, sửa trong RavenDB 31](#_Toc155222027)

[4.3.1. Thêm dữ liệu 31](#_Toc155222028)

[4.3.2. Xóa dữ liệu 32](#_Toc155222029)

[4.3.3. Sửa dữ liệu 33](#_Toc155222030)

[4.4. Truy vấn RavenDB 35](#_Toc155222031)

[Bảng phân công việc 37](#_Toc155222032)

[Tài liệu tham khảo 38](#_Toc155222033)

# Chương 1. Giới thiệu

## Giới thiệu về hệ quản trị CSDL NoSQL

Cơ sở dữ liệu NoSQL (Not Only SQL) là một loại cơ sở dữ liệu được thiết kế để xử lý và lưu trữ dữ liệu có cấu trúc, bán cấu trúc hoặc dữ liệu lớn và phức tạp mà không tuân theo mô hình quan hệ của cơ sở dữ liệu SQL truyền thống. Mục đích chính của việc sử dụng cơ sở dữ liệu NoSQL là danh cho các kho dữ liệu phân tán với nhu cầu lưu trữ dữ liệu lớn.

Khi nào sử dụng Cơ sở dữ liệu NoSQL:

* **Dữ liệu không có cấu trúc hoặc có cấu trúc linh hoạt:** Khi dữ liệu của bạn không phải là bảng quan hệ cố định và có thể thay đổi cấu trúc dễ dàng theo thời gian.
* **Khối lượng dữ liệu lớn và có tính mở rộng cao:** Cơ sở dữ liệu NoSQL thường đối phó tốt với việc lưu trữ và truy vấn dữ liệu lớn, cũng như việc mở rộng hệ thống theo nhu cầu.
* **Tốc độ truy vấn và hiệu suất:** Trong môi trường yêu cầu truy vấn nhanh và hiệu suất cao, NoSQL thường là lựa chọn phù hợp, đặc biệt là trong các ứng dụng web có lượng người dùng lớn.
* **Phân tích dữ liệu và kho lưu trữ dữ liệu đa dạng:** Khi bạn cần lưu trữ và xử lý dữ liệu đa dạng như văn bản, hình ảnh, video, dữ liệu người dùng, và dữ liệu đồng thời mà không cần phải chuẩn hóa chúng thành một cấu trúc chung.
* **Dự án phát triển nhanh và linh hoạt:** Trong những dự án có tính chất thay đổi nhanh về yêu cầu, NoSQL có thể giúp giảm thời gian và công sức cần thiết để điều chỉnh cấu trúc dữ liệu.
* **Hệ thống phân tán và tích hợp dữ liệu từ nhiều nguồn:** Khi bạn cần tích hợp dữ liệu từ nhiều nguồn khác nhau hoặc triển khai hệ thống phân tán, NoSQL có thể cung cấp sự linh hoạt và khả năng mở rộng.
* **Yêu cầu về khả năng chịu lỗi và sẵn sàng cao:** Một số hệ thống NoSQL được thiết kế để đảm bảo khả năng chịu lỗi và sẵn sàng cao, giúp đảm bảo tính ổn định của hệ thống trong môi trường sản xuất.

Ngoài ra có thể xem xét sử dụng NoSQL DB thay vì SQL DB:

* Tốc độ phát triển với cơ sở dữ liệu NoSQL có thể nhanh hơn nhiều so với cơ sở dữ liệu SQL.
* Cấu trúc của nhiều dạng dữ liệu khác nhau được xử lý và phát triển dễ dàng hơn với cơ sở dữ liệu SQL.
* Khi cơ sở dữ liệu SQL không đáp ứng được nhu cầu lưu trữ dữ liệu lớn.
* Khi hệ thống có lượng truy cập cao và không cho phép downtime.
* Các mô hình ứng dụng mới có thể được hỗ trợ dễ dàng hơn

Các kiểu cơ sở dữ liệu NoSQL:

* **Key-value pair:** lưu trữ dưới dạng các cặp key – value, nó được thiết kế theo cách để xử lý nhiều dữ liệu và tải nặng. vd: DynamoDB,Redis…
* **Column-based:** lưu trữ dưới dạng các cột, mỗi cột được xử lý riêng biệt, mang lại hiệu suất cao cho các truy vấn tổng hợp như SUM, COUNT, MIN,…vd: Hbase, Cassandra…
* **Graph-based:** lưu trữ các thực thể cũng như các mối quan hệ giữa các thực thể đó, thực thể được lưu trữ dưới dạng một node với mối quan hệ là các cạnh, một cạnh cho biết mối quan hệ giữa các node, mỗi node và cạnh có một mã định danh duy nhất. vd: CouchDB, Amazon SimpleDB…
* **Document-oriented:** lưu trữ và truy xuất dữ liệu dưới dạng một cặp giá trị khóa (key value pair) nhưng phần giá trị được lưu trữ dưới dạng tài liệu. vd: Neo4J, OrientDB…

Những hạn chế của CSDL NoSQL

* No schema
* Thiếu tính nhất quán (consistency): Một số hệ thống NoSQL ưu tiên việc lấy dữ liệu nhanh chóng hơn tính nhất quán của dữ liệu, dẫn đến mất mát dữ liệu hoặc dữ liệu không nhất quán trong một số trường hợp.
* Không thể thực hiện các truy vấn phức tạp như SQL: Các hệ thống NoSQL thường không hỗ trợ ngôn ngữ truy vấn phức tạp như SQL, điều này có thể khiến việc truy xuất dữ liệu phức tạp trở nên khó khăn.
* Hạn chế trong việc thay đổi cấu trúc dữ liệu: Một số NoSQL có hạn chế trong việc thay đổi cấu trúc dữ liệu một khi dữ liệu đã được lưu trữ. Điều này có thể gây khó khăn khi cần thay đổi cấu trúc dữ liệu.
* Khó khăn trong việc thực hiện các giao dịch phức tạp: Một số hệ thống NoSQL có khó khăn khi xử lý các giao dịch phức tạp hoặc cần đảm bảo tính nhất quán (ACID properties).
* ...

## 1.2. Giới thiệu về RavenDB

### 1.2.1.Tổng quan

RavenDB là một cơ sở dữ liệu hướng tài liệu hoàn toàn mã nguồn mở, tuân thủ hoàn toàn ACID, được viết bằng ngôn ngữ C#, phát triển bởi Hibernating Rhinos Ltd. Nó có khả năng chạy trên nhiều nền tảng như Windows, Linux và Mac OS. RavenDB lưu trữ dữ liệu dưới dạng các tài liệu JSON và có thể triển khai trong các cụm phân tán với đồng bộ hóa master-master.

Ban đầu có tên là "Rhino Divan DB", RavenDB bắt đầu vào năm 2008 như một dự án của Oren Eini (còn được gọi là Ayende Rahien) và được phát triển bởi Hibernating Rhinos Ltd. Công ty khẳng định đó là cơ sở dữ liệu tài liệu đầu tiên chạy một cách tự nhiên trên .NET Framework. Đó là một trong những cơ sở dữ liệu tài liệu sớm nhất cung cấp các đảm bảo ACID.

Năm 2019, Hibernating Rhinos bắt đầu cung cấp RavenDB dưới dạng dịch vụ đám mây được gọi là RavenDB Cloud.

RavenDB là một cơ sở dữ liệu mã nguồn mở có hỗ trợ transactional (giao dịch) được viết cho nền tảng .NET. RavenDB đưa ra mô hình dữ liệu linh hoạt (flexible data model) nhằm đáp ứng yêu cầu của các hệ thống thế giới thực (real-world systems).

RavenDB cho phép xây dựng những ứng dụng có hiệu suất cao(high-performance), độ trễ thấp(low-latency) một cách nhanh chóng và hiệu quả. RavenDB xứng đáng là một cơ sở dữ liệu đáng tin cậy.

### 1.2.2.Đặc điểm

* **ACID Database Transactions:** RavenDB là một trong những cơ sở dữ liệu nonrelational đầu tiên cung cấp tính ACIDity không chỉ cho một giá trị duy nhất mà còn cho nhiều giá trị trong suốt cơ sở dữ liệu. Là một cơ sở dữ liệu phân tán, nó cũng cung cấp các bảo đảm ACID trong toàn bộ cơ sở dữ liệu. Các nhà phát triển được miễn xử lý nhiều kịch bản chuyển dữ liệu một phần và sự phức tạp của việc lưu trữ dữ liệu và được tự do tập trung vào việc xây dựng các ứng dụng mạnh mẽ và cung cấp giá trị cho doanh nghiệp.
* **High Performance:** RavenDB có thể thực hiện hơn 150.000 lần ghi mỗi giây và 1 triệu lần đọc trên phần cứng hàng hóa đơn giản.
* **Advanced Query Engine:** **Công cụ truy vấn nâng cao** Ánh xạ tài liệu thành các chỉ mục, phân tích văn bản và dữ liệu không gian, chiếu dữ liệu thành hình dạng mới.
* **Database Management Studio:** Giao diện người dùng đồ họa (GUI) được tải đầy đủ, giám sát cả số liệu vận hành và hiệu suất của cơ sở dữ liệu của bạn.
* **Multi-Model Architecture:** **• Kiến trúc đa mô hình** Cơ sở dữ liệu đa mô hình cho tất cả các dịch vụ như lưu trữ tài liệu, quầy phân phối và các dịch vụ khác.
* **Multi-Platfrom:** RavenDB cho phép di chuyển dữ liệu giữa các hệ điều hành khác nhau một cách dễ dàng. Ta có thể triển khai RavenDB trên các phần cứng khác nhau chạy trên các hệ điều hành khác nhau, từ các máy chủ mạnh đến các đơn vị nhỏ và có cơ sở dữ liệu không chỉ hoạt động hoàn hảo trên mỗi máy chủ mà còn thực sự sử dụng các tài nguyên độc đáo của nó và tận dụng tối đa nó
* **High Availability and Scalability** **• Tính sẵn sàng và khả năng mở rộng cao** RavenDB có tính khả dụng cao trong kiến trúc  
  cơ sở dữ liệu phân tán mà không làm tăng độ phức tạp
* **Ease of Use:**
* **Free License:** Giấy phép cộng đồng miễn phí dành cho Cơ sở dữ liệu RavenDB OnPrem NoSQL hoặc dịch vụ DbaaS trên nền tảng đám mây được quản lý đi kèm với các công cụ bổ sung để nâng cao các bạn làm việc với dữ liệu của mình
* **Integration with Relational Databases:** **• Tích hợp với cơ sở dữ liệu quan hệ** RavenDB cung cấp khả năng gửi và chuyển đổi dữ liệu từ nơi này sang nơi khác. Ngoài ra, còn hỗ trợ tích hợp cho các quy trình ETLcho các cơ sở dữ liệu quan hệ như Microsoft SQL, Oracle, PostgreSQL, MySQL,...

### 1.2.3.Ưu điểm

RavenDB rất dễ cài đặt và bảo mật, dễ sử dụng và đưa vào sản xuất nhanh chóng. Ngôn ngữ truy vấn dễ dàng vì RQL sử dụng cú pháp SQL đến 80%.

RavenDB là một CSDL đa mô hình. Document, key/value, Counter, Graph API và Time Series cho phép sử dụng nhiều phiên bản cho các chức năng khác nhau mà không phải mua cơ sở dữ liệu khác nhau, hữu dụng cho microservice.

RavenDB dễ dàng mở rộng quy mô. Cơ sở dữ liệu phân tán cho phép thiết lập các nút (nodes) ngay lập tức. Ngoài ra, nó sử dụng phần cứng tối đa, RavenDB hoạt động trên Raspberry Pi và chip ARM vì nó tận dụng tối đa phần cứng mà nó chạy

### 1.2.4.Nhược điểm

RavenDB chưa có cộng đồng đủ mạnh để hỗ trợ việc giải đáp các thắc mắc hoặc các vấn đề hay gặp phải khi sử dụng nó.

Tài liệu vẫn chưa có nhiều về các ví dụ thực hành, hoặc các trường hợp phổ biến khi sử dụng RavenDB. Ngoài ra, vẫn còn thiếu các video hướng dẫn giải quyết các vấn đề phổ biến.

Không thể kiểm tra sharding, replication, hoặc truy cập được xác thực nếu không mua giấy phép.

Giấy phép là điều bắt buộc đối vớis bất cứ điều gì khác ngoài sự phát triển (ngay cả UAT cũng yêu cầu giấy phép)

# Chương 2. Cài đặt ravendb

## Cài đặt ravendb trên hai máy

Bước 1: Tải RavenDB về máy theo đường link: [Download RavenDB NoSQL  
Document Database](https://ravendb.net/download)

A screenshot of a computer

Description automatically generated

Bước 2: Giải nén file RavenDB

A screenshot of a computer

Description automatically generated

Bước 3: Truy cập theo đường dẫn RavenDB-5.4.5-windows-x64\Server.

Bước 4: Bật cmd tại file Server.

Bước 5: Gõ lệnh .\Raven.Server rồi đợi khoảng vài giây để nó khởi động.

A screenshot of a computer

Description automatically generated

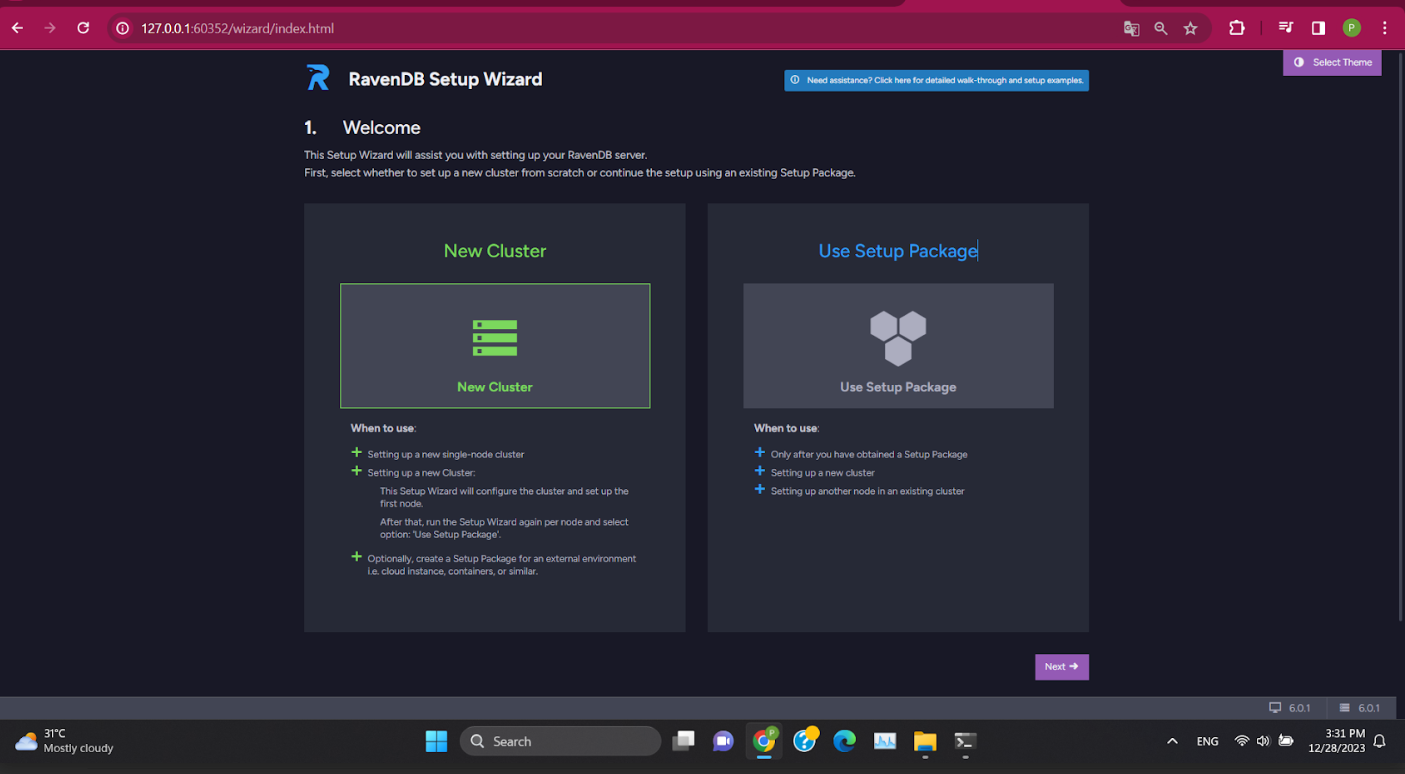
Bước 6: Truy cập vào server bằng trình duyệt web dựa vào IP và Port đã được cho sẵn.



Bước 7: Kéo chính sách của RavenDB tới cuối cùng, sau đó chọn Accept.

Ở máy A:

Bước 8.1: Máy A sẽ tạo Setup Package. Chọn New Cluster, sau đó chọn Next.



Bước 9.1: Chọn Unsecure và nhấn Next.

**A screenshot of a computer

Description automatically generated**

Bước 10.1: Thêm IPv4 của Máy A vào Node A.

A screenshot of a computer

Description automatically generated

Bước 11.1: Chọn Add Node B và thêm IPv4 của Máy B vào.

A screenshot of a computer

Description automatically generated

**Bước 12.1:** Chọn Finish, Set up thành công Node A và N duyệt tải về file zip để Node B có thể chọn Set up Package.

**Bước 13.1:** Restart server. Hoàn thành set up.

**Bước 14.1:** Sau khi set up Máy 2 thành công sẽ thấy Node A và Node B ở Manage Server -> Cluster.

Ở máy B:

Bước 8.2: Máy B sẽ sử dụng Setup Package mà máy A đã tạo. Chọn Use SetupPackage, sau đó chọn Next.

Bước 9.2: Chọn Browse -> Chọn file zip Unsecure.Cluster.Settings.

Bước 10.2: Chọn Node tag có IP (Radmin) là của máy B, sau đó chọn Next.

Bước 11.2: Lúc này ta đã config xong cho server, chọn Restart Server.

Sau khi restart:

## Kết nối hai máy bằng Radmin

* Ở máy 1: Tạo network mới với Network name và Password.
* Ở máy 2: Tham gia vào network của máy 1.
* Bước 1: Chọn “Join network”.
* Bước 2: Tại tab Private Network, ta nhập Network name và Password mà
* máy 1 đã tạo trước đó.
* Bước 3: Chọn nút “Join” để tham gia vào network của máy 1.
* Ở cả 2 máy: Tắt tường lửa (firewall) để 2 máy có thể kết nối được với nha

# Chương 3. Thiết kế CSDL

## Mô trả về cơ sở dữ liệu

CSDL quản lý các chi nhánh thuộc nhà phân phối và bán lẻ Gido Foods

Hiện Gido Foods có 2 chi nhánh, hay 2 cửa hàng. Cửa hàng chính là cửa hàng 1, nằm ở 19A phường Linh Trung, Thủ Đức

Lược đồ cơ sở dữ liệu của mỗi cửa hàng như sau:

|  |  |  |
| --- | --- | --- |
| Quan hệ | Thuộc tính | Ý nghĩa |
| HAISAN  (Sản phẩm) | **MAHS (Khóa chính)** | Mã sản phẩm hoặc mã định danh |
| TenHS | Tên |
| NgayNhap | Ngày nhập |
| Loai | Loại hoặc danh mục |
| XuatXu | Nguồn gốc |
| GiaBan | Giá bán |
| DVT | Đơn vị đo lường |
| CUAHANG  (Cửa hàng) | **MACH (Khóa chính)** | Mã cửa hàng hoặc mã định danh |
| TenCH | Tên |
| DiaChi | Địa chỉ |
| SoDT | Số điện thoại |
| NHANVIEN  (Nhân viên) | **MaNV (Khóa chính)** | Mã nhân viên hoặc mã định danh |
| TenNV | Tên |
| DiaChi | Địa chỉ |
| SoDT | Số điện thoại |
| Luong | Lương |
| MACH (Khóa ngoại) | Tham chiếu đến mã cửa hàng mà nhân viên làm việc |
| KHOHS\_QLKHO (Quản lý Kho hàng) | **MACH (Khóa ngoại)** | Tham chiếu đến mã cửa hàng |
| **MAHS (Khóa ngoại)** | Tham chiếu đến mã sản phẩm |
| SoLuong | Số lượng của sản phẩm trong kho |
| NgayCapNhat | Ngày cập nhật thông tin kho lần cuối |
| KHOHS\_NVBH (Quản lý Kho bởi Nhân viên) | **MACH (Khóa ngoại)** | Tham chiếu đến mã cửa hàng |
| **MAHS (Khóa ngoại)** | Tham chiếu đến mã sản phẩm |
| TinhTrang | Trạng thái của sản phẩm trong kho do nhân viên quản lý |
| KHACHHANG (Khách hàng) | **MAKH (Khóa chính)** | Mã khách hàng hoặc mã định danh |
| TenKH | Tên |
| NgaySinh | Ngày sinh |
| GioiTinh | Giới tính |
| DiaChi | Địa chỉ |
| Phone | Số điện thoại |
| NgayDK | Ngày đăng ký |
| HOADON  (Hoá đơn) | **MaHD (Khóa chính)** | Mã hóa đơn hoặc mã định danh |
| MANV (Khóa ngoại) | Tham chiếu đến mã nhân viên |
| MAKH (Khóa ngoại) | Tham chiếu đến mã khách hàng |
| NgayLap | Ngày tạo hóa đơn |
| TongTien | Tổng số tiền trên hóa đơn |
| MACH (Khóa ngoại) | Mã cửa hàng |
| CTHD  (Chi tiết Hóa đơn) | **MaHD (Khóa ngoại)** | Tham chiếu đến mã hóa đơn |
| **MAHS (Khóa ngoại)** | Tham chiếu đến mã sản phẩm trong chi tiết hóa đơn |
| SoLuong | Số lượng của sản phẩm trong chi tiết hóa đơn |
| MACH | Mã cửa hàng |

## Dữ liệu các bảng

### Bảng CUAHANG

createCuaHang("CH01", "Hai San Good 1", "Quan 7, HCM", "076584848");

createCuaHang("CH02", "Hai San Good 2", "Thu Duc, HCM", "076584849");

### Bảng HAISAN

createHaiSan("HS01", "Tom Hum", "Tom", "Han Quoc" ,900, "kg");

createHaiSan("HS02", "Tom Cang Xanh", "Tom", "Viet Nam", 450, "kg");

createHaiSan("HS03", "Cua Bien", "Cua", "Viet Nam", 600, "kg");

createHaiSan("HS04", "So Diep", "So", "Nhat Ban", 400, "kg");

createHaiSan("HS05", "Hau Bien", "Hau", "Viet Nam", 150, "kg");

createHaiSan("HS06", "Ca Hoi", "Ca", "NA Uy", 500, "kg");

createHaiSan("HS07", "Muc Khong Lo", "Muc", "Han Quoc", 900, "kg");

createHaiSan("HS08", "Muc Trung", "Muc", "Viet Nam", 700, "kg");

3.3.3.Bảng NHANVIEN CH01

createNhanVien("NV01", "Do Huynh My Tam", "Quan 3 HCM", "07699889", 15000000, "CH01");

createNhanVien("NV03","Luu Vinh Phat", "Quan 2, HCM", "0764484234", 15000000, "CH01");

createNhanVien("NV05", "Nguyen Minh Duy", "Quan 4, HCM", "0934484233", 1400000, "CH01");

createNhanVien("NV07", "Tran Van The", "Quan 5, HCM", "0934484235", 1200000, "CH01");

createNhanVien("NV09", "Nguyen Ngoc Tho", "Quan 6, HCM", "0934484236", 1200000, "CH01");

3.3.4.Bảng NHANVIEN CH02

createNhanVien("NV02", "Nguyen Ngoc Thu", "Quan 7 HCM", "07699889", 15000000, "CH02");

createNhanVien("NV04", "Tran Thi Thanh", "Quan 8, HCM", "0764484234", 11000000, "CH02");

createNhanVien("NV06", "Do Ngoc", "Quan 1, HCM", "0934484233", 1200000, "CH02");

createNhanVien("NV08", "Vo Thi Ha Trang", "Thu Duc", "0934484235", 1700000, "CH02");

createNhanVien("NV10", "Nguyen Ngoc Anh", "Quan 6, HCM", "0934484236", 9000000, "CH02");

3.3.5.Bảng KHACHHANG

createKhachHang("KH01", "Trum Bom Hang", "Quan 3, HCM", "076888899", "12/12/2000", "Nam", "29/12/2023");

createKhachHang("KH02", "Trum Quan 6", "Quan 6, HCM", "076888897", "12/09/2000", "Nam", "29/12/2023");

createKhachHang("KH03", "Nguyen Thanh Tu", "Quan 3, HCM", "076888896", "27/08/1999", "Nu", "28/12/2023");

createKhachHang("KH04", "Nguyen Van B", "Quan 8, HCM", "076888895", "12/12/1999", "Nam", "27/12/2023");

createKhachHang("KH05", "Ong Ke", "Quan 7, HCM", "076888894", "03/12/2000", "Nam", "29/11/2023");

createKhachHang("KH06", "Anh Dep Trai", "Quan 1, HCM", "076888893", "15/12/1998", "Nu", "29/11/2023");

createKhachHang("KH07", "Cau Be", "Quan 2, HCM", "076888892", "02/12/2002", "Nam", "29/11/2023");

createKhachHang("KH08", "Mo Hai", "Quan 2, HCM", "076888891", "12/11/2001", "Nu", "29/10/2023");

createKhachHang("KH09", "Be Ba", "Quan 1, HCM", "076888890", "12/11/2000", "Nam", "29/10/2023");

createKhachHang("KH10", "Doc La", "Binh Duong", "076888877", "12/11/2000", "Nam", "01/01/2024");

3.3.6.Bảng KHOHS\_QLKHO CH01

createKhoHS("CH01", "HS01", 10, "01/01/2024");

createKhoHS("CH01", "HS02", 9, "01/01/2024");

createKhoHS("CH01", "HS03", 8, "01/01/2024");

createKhoHS("CH01", "HS04", 0, "01/01/2024");

createKhoHS("CH01", "HS05", 5, "01/01/2024");

createKhoHS("CH01", "HS06", 4, "01/01/2024");

createKhoHS("CH01", "HS07", 7, "01/01/2024");

createKhoHS("CH01", "HS08", 0, "01/01/2024");

3.3.7.Bảng KHOHS\_QLKHO CH02

createKhoHS("CH02", "HS01", 10, "02/01/2024");

createKhoHS("CH02", "HS02", 9, "02/01/2024");

createKhoHS("CH02", "HS03", 8, "02/01/2024");

createKhoHS("CH02", "HS04", 0, "02/01/2024");

createKhoHS("CH02", "HS05", 5, "03/01/2024");

createKhoHS("CH02", "HS06", 4, "03/01/2024");

createKhoHS("CH02", "HS07", 7, "03/01/2024");

createKhoHS("CH02", "HS08", 0, "03/01/2024");

3.3.8.Bảng HOADON

createHoaDon("HD01", "NV01", "KH01", "02/01/2024", 1350, "CH01");

createHoaDon("HD03", "NV03", "KH03", "01/01/2024", 1800, "CH01");

createHoaDon("HD05", "NV05", "KH05", "03/01/2024", 1200, "CH01");

createHoaDon("HD07", "NV07", "KH07", "03/01/2024", 700, "CH01");

createHoaDon("HD02", "NV02", "KH02", "03/01/2024", 1400, "CH02");

createHoaDon("HD04", "NV06", "KH06", "03/01/2024", 900, "CH02");

createHoaDon("HD06", "NV04", "KH08", "02/01/2024", 1000, "CH02");

createHoaDon("HD08", "NV08", "KH04", "01/01/2024", 300, "CH02");

3.3.9.Bảng CTHD

createCthd("HD01", "HS01", 1, "CH01");

createCthd("HD01", "HS02", 1, "CH01");

createCthd("HD03", "HS07", 2, "CH01");

createCthd("HD05", "HS03", 2, "CH01");

createCthd("HD07", "HS08", 1, "CH01");

createCthd("HD02", "HS08", 2, "CH02");

createCthd("HD04", "HS02", 2, "CH02");

createCthd("HD06", "HS03", 1, "CH02");

createCthd("HD06", "HS04", 1, "CH02");

createCthd("HD08", "HS05", 2, "CH02");

# Chương 4:Thao tác dữ liệu qua lại giữa hai máy

Generic create các bảng

A screen shot of a computer program

Description automatically generated

Tạo kết nối CH1

A computer screen shot of a computer code

Description automatically generated

CH1 kết nối CH2

A computer screen shot of a computer code

Description automatically generated

Tạo kết nối CH2

A computer screen with text

Description automatically generated

CH2 kết nối CH1

A computer screen with text

Description automatically generated

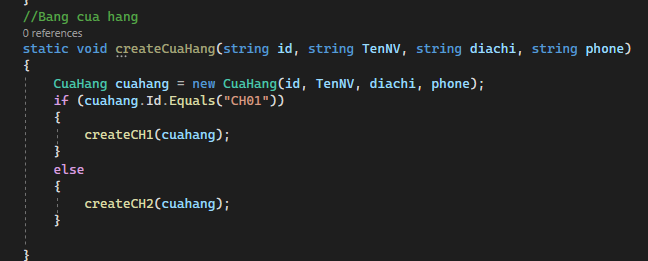
## Tạo bảng

### Bảng hải sản

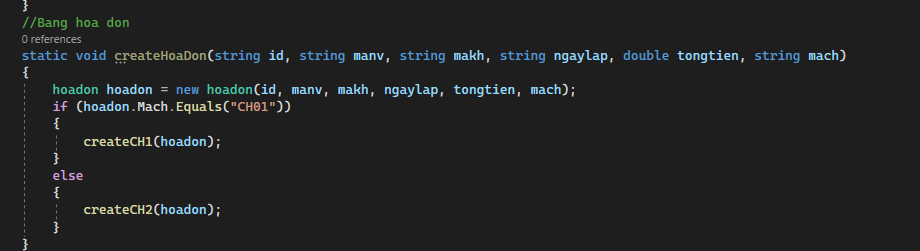
A black background with blue text

Description automatically generated

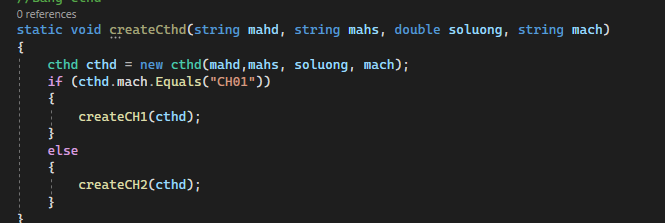
### Bảng cửa hàng



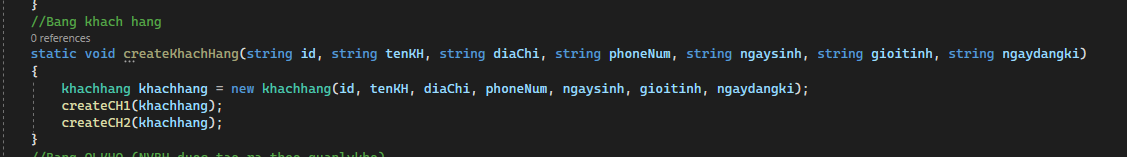
### Bảng hóa đơn



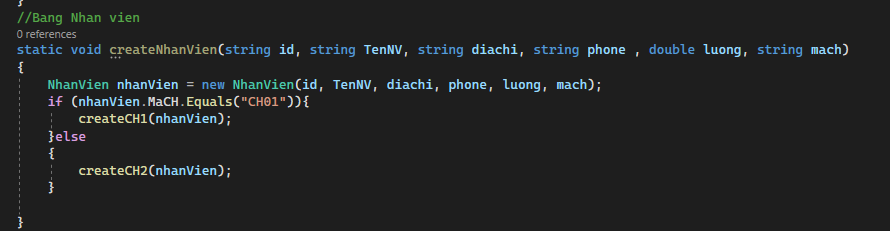
### Bảng Chi tiết hóa đơn



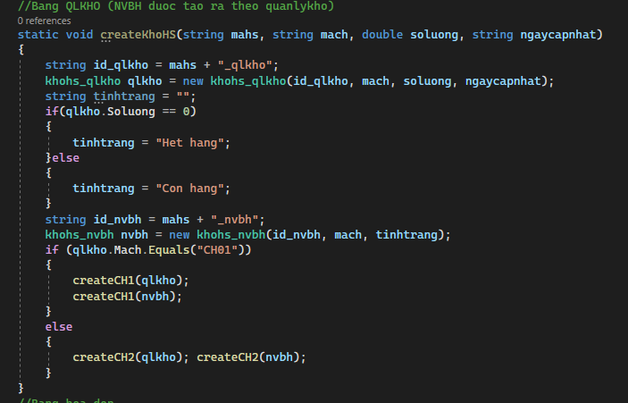
### Bảng khách hàng



### Bảng nhân viên bán hàng



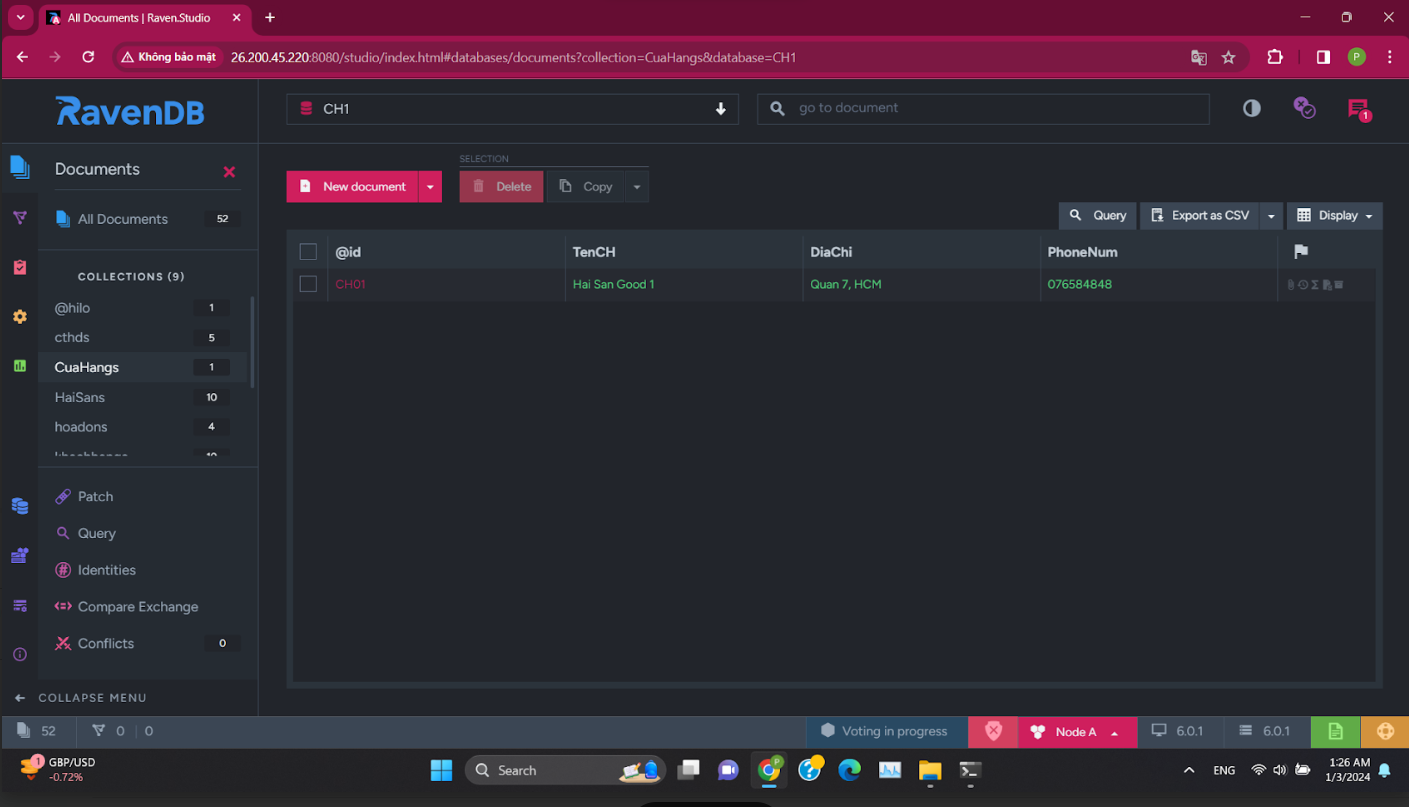
### Bảng quản lý kho



## Thêm dữ liệu

### Bảng cửa hàng

CN1:



CN2:

A screenshot of a computer

Description automatically generated

### Bảng hải sản

CN1:

A screenshot of a computer

Description automatically generated

CN2:

A screenshot of a computer

Description automatically generated

### Bảng hóa đơn

CN1:

A screenshot of a computer

Description automatically generated

CN2:

A screenshot of a computer

Description automatically generated

### Bảng chi tiết hóa đơn

CN1:

A screenshot of a computer

Description automatically generated

CN2:

A screenshot of a computer

Description automatically generated

### Bảng khách hàng

CN1:

A screenshot of a computer

Description automatically generated

CN2:

A screenshot of a computer

Description automatically generated

### Bảng nhân viên

CN1:

A screenshot of a computer

Description automatically generated

### Bảng nhân viên bán hàng

CN1:

A screenshot of a computer

Description automatically generated

### Bảng quản lý kho

CN1:

A screenshot of a computer

Description automatically generated

## Thêm, xóa, sửa trong RavenDB

### Thêm dữ liệu

CH2: thêm dữ liệu vào bảng NHANVIEN cho CH1

A black screen with white text

Description automatically generated

Kết quả thêm thành công NHANVIEN NV11

A screenshot of a computer

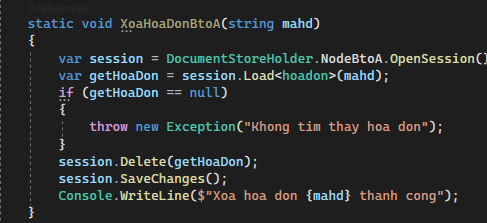
Description automatically generated

Bảng NHANVIEN sau khi thêm NHANVIEN NV11

A screenshot of a computer screen

Description automatically generated

### Xóa dữ liệu



CH1: bảng HOADON trước khi xoá

A screenshot of a computer

Description automatically generated

Tiến hành xoá HOADON HD12 bên CH2

A computer screen with text

Description automatically generated

Kết quả xoá HOADON HD12 thành công

A screenshot of a computer

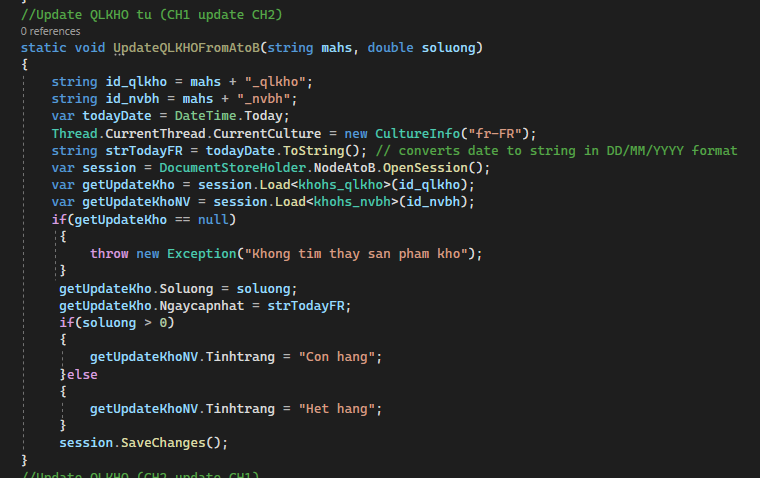
Description automatically generated

Bảng HOADON sau khi xoá

A screenshot of a computer

Description automatically generated

### Sửa dữ liệu



Ở bảng HAISAN thì HS04 có số lượng bằng 0

A screenshot of a computer

Description automatically generated

Và HS04 có tình trạng là hết hàng

A screenshot of a computer

Description automatically generated

Tiến hành cập nhật số lượng cho HS04 là 2

A black background with white text

Description automatically generated

Cập nhật số lượng thành công

A screenshot of a computer

Description automatically generated

NVBH sau khi cập nhật HS04

A screenshot of a computer

Description automatically generated

## Truy vấn RavenDB

Query 1: Tìm nhân viên có lương trên 10 triệu

A screen shot of a computer program

Description automatically generated

A screen shot of a computer

Description automatically generated

Query 2: Tìm nhân viên tham gia hoá đơn có trị giá cao nhất cả 2 cửa hàng

A screen shot of a computer program

Description automatically generated

A screenshot of a computer

Description automatically generated

# Bảng phân công việc

|  |  |  |  |
| --- | --- | --- | --- |
| **Họ tên** | **Mã số sinh viên** | **Công việc** | **Đánh giá hoàn thành (100%)** |
| Đỗ Huỳnh Mỹ Tâm | 20520746 | Làm slide, thiết kế CSDL. | 100% |
| Lưu Vĩnh Phát | 20521733 | Thực hiện các thao tác giữa 2 máy, code. | 100% |
| Trần Văn Thế | 20520770 | Viết docs, thực hiện các thao tác giữa 2 máy. | 100% |
| Nguyễn Minh Duy | 21522005 | Viết docs, thực hiện các thao tác giữa 2 máy. | 100% |

# Tài liệu tham khảo

1. Thu Ha, Cơ sở dữ liệu NoSQL là gì và tất cả những gì cần biết về NoSQL  
Database: [https://itguru.vn/blog/co-so-du-lieu-nosql-la-gi-va-tat-ca-nhung-gi-canbiet-ve-nosql-database/](https://itguru.vn/blog/co-so-du-lieu-nosql-la-gi-va-tat-ca-nhung-gi-can-biet-ve-nosql-database/)  
2. Officical Document, RavenDB: <https://ravendb.net/why-ravendb>

3.RavenDB Wikipedia: <https://en.wikipedia.org/wiki/RavenDB>

4. Các Báo cáo tham khảo được Giáo viên thực hành cung cấp thông qua website môn học