A logo with a red and blue letter and a drop of water

Description automatically generated

**BỘ CÔNG THƯƠNG**

**TRƯỜNG ĐẠI HỌC CÔNG NGHIỆP TP.HCM**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO MÔN HỌC**

**Mô Hình Hóa Dữ Liệu No SQL**

**MongoDB**

Đề tài: Hệ thống quản lý sàn thương mại điện tử

**Giảng viên: Trần Thế Trung**

Nhóm thực hiện: **Nhóm 4**

Lớp: DHKTPM17A

TP.Hồ Chí Minh – Ngày 17 Tháng 09 Năm 2023

**Thành Viên**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Mã sinh Viên** | **Họ Và Tên** | **Phân Công** | **Đánh Giá** |
| 20 | 21044141 | Lại Dương Minh Hiếu | **II.7** | Hoàn thành tốt |
| 21 | 21044471 | Trần Đăng Hiếu | **I.1, I.2** | Gương mẫu |
| 26 | 21050211 | Nguyễn Hồng Huy | **II.2, II.5** | Hoàn thành tốt |
| 46 | 20073871 | Nguyễn Thiên Phú | **II.6** | Hoàn thành tốt |
| 56 | 21044611 | Nguyễn Minh Tiến | **II.3, II.4** | Hoàn thành tốt |
| 63 | 21020241 | Nguyễn Đức Vinh | **II.1** | Hoàn thành tốt |

1. Chương 1

**1. Chọn hệ thống thông tin phù hợp với việc lưu trữ dữ liệu bằng NoSQL - MongoDB:**

* **Tên hệ thống:** Quản lý sàn thương mại điện tử
* **Phạm vi của hệ thống:**
* Quản lý thông tin về sản phẩm bao gồm: tên sản phẩm, mô tả, giá, số lượng tồn kho, danh mục, và hình ảnh sản phẩm.
* Quản lý đơn hàng gồm: tạo đơn hàng mới, cập nhật trạng thái đơn hàng, xác định các sản phẩm trong mỗi đơn hàng.
* **Lý do sử dụng NoSQL - MongoDB**:
* **Dữ liệu đa dạng**: sản phẩm, đơn hàng, danh mục sản phẩm, và danh sách khách hàng. MongoDB cho phép lưu trữ dữ liệu đa dạng mà không cần định nghĩa cấu trúc cụ thể trước đó.
* **Khả năng mở rộng**: MongoDB có khả năng mở rộng tốt, cho phép bạn mở rộng hệ thống khi có nhiều đơn hàng và sản phẩm tham gia vào nền tảng thương mại điện tử.
* **Hiệu suất truy vấn**: MongoDB cung cấp khả năng tối ưu hóa hiệu suất truy vấn, đặc biệt là khi bạn cần thực hiện các truy vấn phức tạp liên quan đến tìm kiếm sản phẩm, quản lý đơn hàng, và theo dõi lịch sử mua sắm của khách hàng.

**2. Đặc tả sơ lược về hệ thống hoặc yêu cầu người dùng (user requirements) của hệ thống:**

* **Hệ thống phần cứng**: Hệ thống sử dụng một số máy chủ để chạy MongoDB và máy chủ web để cung cấp giao diện cho người dùng và quản trị viên. Hệ thống cần đảm bảo có đủ dung lượng lưu trữ và tài nguyên để xử lý lượng dữ liệu và truy vấn lớn.
* **Mô tả nhóm người**: **Người dùng cuối** (khách hàng) có thể xem sản phẩm, thêm sản phẩm vào giỏ hàng, và tạo đơn hàng.
* **Người bán hàng** có thể thêm, sửa, xóa sản phẩm và quản lý đơn hàng.
* **Mục đích chính:** Được sử dụng để quản lý sản phẩm, đơn hàng và thực hiện các giao dịch thương mại điện tử.
* **Cách thức thông tin dữ liệu được phát sinh (lấy về) từ người dùng hoặc hệ thống**: Thông tin sản phẩm và đơn hàng được thêm và cập nhật bởi quản trị viên và khách hàng thông qua giao diện web
* **Tần suất phát sinh dữ liệu:** Tần suất thêm sản phẩm và đơn hàng thường xuyên, có thể hàng giờ hoặc hàng ngày. Thông tin về đơn hàng và sản phẩm được cập nhật khi có giao dịch mới hoặc sự thay đổi trạng thái của đơn hàng.
* **Những yếu tố có thể ảnh hưởng (không tốt) đến thông tin dữ liệu của hệ thống:**
* **Sự cố phần cứng**: Mất dữ liệu có thể xảy ra nếu máy chủ gặp sự cố hoặc thiết bị lưu trữ bị hỏng. Để đảm bảo tính toàn vẹn dữ liệu, hệ thống cần có các biện pháp sao lưu và khôi phục dữ liệu định kỳ.
* **Sự cố mạng**: Sự cố mạng có thể ảnh hưởng đến khả năng truy cập hệ thống và dữ liệu từ xa. Điều này có thể dẫn đến mất kết nối và ngừng hoạt động tạm thời.
* **Lỗi phần mềm**: Lỗi phần mềm có thể xảy ra trong quá trình phát triển hoặc sử dụng hệ thống, dẫn đến sự cố và mất dữ liệu. Để đối phó với điều này, cần thực hiện kiểm thử phần mềm thường xuyên và cập nhật hệ thống.
* **Lưu trữ dữ liệu lâu dài:** Với dữ liệu phải được lưu trữ trong một khoảng thời gian dài, cần có kế hoạch để quản lý lưu trữ và đảm bảo tính toàn vẹn và sẵn sàng cho việc truy vấn trong tương lai.
* **Thời gian xử lý truy vấn:** Hệ thống cần xử lý các truy vấn truy xuất sản phẩm và đơn hàng nhanh chóng để đảm bảo trải nghiệm người dùng mượt mà và không gây chờ đợi lâu.
* **Bảo mật và quyền truy cập:** Đảm bảo tính riêng tư và bảo mật của thông tin khách hàng là quan trọng. Người dùng chỉ nên có quyền truy cập vào dữ liệu mà họ được phép thấy.
* **Hiệu suất hệ thống:** Hệ thống cần đảm bảo hiệu suất đủ tốt để xử lý đồng thời các giao dịch và truy vấn từ nhiều người dùng.
* **Khả năng mở rộng**: Với sự phát triển của thương mại điện tử, hệ thống cần phải mở rộng để xử lý lượng dữ liệu và giao dịch lớn hơn theo thời gian.
* **Sao lưu và phục hồi dữ liệu**: Hệ thống cần có quy trình sao lưu định kỳ và khả năng phục hồi dữ liệu nhanh chóng trong trường hợp sự cố xảy ra.
* **trữ:** Người dùng muốn lưu trữ thông tin dữ liệu sản phẩm và đơn hàng trong một khoảng thời gian là 5 năm.
* **Thông tin dữ liệu thường xuyên được xử lý và truy vấn:**
* **Thông tin sản phẩm:** Tên, mô tả, giá, số lượng hàng tồn kho, danh mục sản phẩm, hình ảnh sản phẩm. (**Người truy vấn chủ yếu:** Khách hàng)
* **Thông tin đơn hàng**: Mã đơn hàng, trạng thái đơn hàng, sản phẩm trong đơn hàng, thông tin người mua, tổng giá trị đơn hàng. (**Người truy vấn chủ yếu:** Khách hàng)
* **Thông tin đánh giá và đánh giá sản phẩm**: các đánh giá và bài viết đánh giá sản phẩm.( **Người truy vấn chủ yếu:** khách hàng**,** người bán hàng.)
* **Thời gian lưu trữ:** 5 năm
* **Tận dụng lại hệ thống phần cứng hiện có:**

1. Chương II
2. Danh sách các actor và use case tương ứng:

* Khách hàng mua hàng:
* Tìm sản phẩm
* Xem thông tin chi tiết của sản phầm
* Đặt sản phẩm
* Hủy đặt tour
* Hủy đặt sản phẩm
* Thanh toán
* Xem đơn hàng đã mua
* Thêm sản phẩm giỏ hàng
* Xóa sản phẩm giỏ hàng
* Chỉnh sửa số lượng sản phẩm giỏ hàng
* Chỉnh sửa thông tin cá nhân
* Đánh giá sản phẩm đã mua
* Sử dụng mã giảm giá
* Người bán hàng:
* Đăng sản phẩm
* Hủy sản phẩm
* Thanh toán
* Tìm sản phẩm
* Cập nhật thông tin sản phẩm
* Xem sản phẩm
* Xem các báo cáo thống kê
* Chỉnh sửa thông tin cửa hàng
* Tạo mã giảm giá sản phẩm của họ
* Quản trị viên hệ thống:
* Xem khách hàng
* Xem người bán hàng
* Xóa sản phẩm của các người bán hàng
* Xóa người bán hàng
* Thống kê doanh thu hệ thống
* Tạo mã giám giá chung

Sàn thương mại điện tử là 1 hệ thống cho phép người bán hàng có thể tạo ra các gian hàng của mình một các trực tuyến. Và người mua hàng có thể mua hàng của người bán hàng. Số tiền giao dịch giữa người bán và người mua là 30 tỉ VND/ 1 năm. Số lượt truy cập vào hệ thống hằng tháng là 500 ngàn và trong số đấy có 100 ngàn lượt truy cập vào xem các sản phẩm. Tỉ lệ mua hàng là 2% của số lượt xem. Giỏ hảng của mỗi người chứa tổng giá trị sản phẩm là 1 triệu VND. Lượng sản phẩm bán hàng tháng là 2000 sản phẩm. Tỉ lệ đánh giá sản phẩm của người mua hàng mỗi tháng là 3000 đánh giá. Tỉ lệ giao dịch là 5 phút/1 giao dịch.Người bán hàng đăng 2 bài / ngày, chỉnh sửa 2 bài/ giờ. Tài khoản mới là mỗi tháng là 2500 tài khoản. Tỉ lệ sử dụng phiếu giảm giá là 15% của tổng lượt mua. Tỉ lệ hủy là 10% lượt mua.Lợi nhuận mỗi tháng của sàn là 500 triệu VND. Mỗi người bán hành sẽ xem báo cáo 2 lần mỗi tuần. Tất cả thông tin khách hàng và nhà bán hàng lưu 10 năm, thông tin sản phẩm lưu 5 năm

1. Danh sách các CRUD của hệ thống ứng với Actor, operation, Information, Frequency, critically...

**Main Data**

* Người quản lý 5
* Người bán hàng 3.000
* Người mua hàng 200.000

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Actor** | **CRUD Operations** | **Data in Operation** | **Op Type** | **Rate** | **Data Durability** | **Data Read/Written** | **Data life** | **Response Latency** | **Query Time** | **Data Read** | **Data Freshess** |
| Người bán hàng | Đăng 2 bài bán sản phẩm 1 ngày | Id\_ SanPham ,thongTinSanPham | Write | 250 sản phẩm / giờ | Majority | 400 bytes | 5 năm |  |  |  |  |
| Người bán hàng | Cập nhật thông tin bài đăng sản phẩm của 2 sản phẩm mỗi ngày | id\_SanPham, thongTinCapNhat | Write | 250 sản phẩm / giờ | Majority | 400 bytes | 5 năm |  |  |  |  |
| Người bán hàng | Xóa 2 bài đăng sản phẩm mỗi ngày | id\_ SanPham | Write | 250 sản phẩm /giờ |  |  |  |  |  |  |  |
| Khách hàng | Xem sản phẩm | id\_ SanPham,thongTinSanPham | Read | 27777777 lượt xem/ giờ |  |  |  | 200ms | 300ms |  | < 1000ms |
| Khách hàng | Hủy đặt hàng | id\_khachHang,id\_donHang | Write | 55  hủy đặt hàng /giờ | Majority |  |  |  |  |  |  |
| Người bán hàng | Xem thống kê | id\_nguoiBan,thongTinThongKe | Read | 0.0119  lượt xem / giờ |  |  |  | 200ms | 500ms | collection scan | < 1 giờ |
| Người mua hàng | Thanh toán | id\_SanPham, id\_khachHang | Write | 41666.667  thanh toán tour / giờ | Majority | 500bytes | 5 năm | 100ms |  |  | < 1 giờ |

1. Phân tích chi tiết 2 thao tác (Operation) quan trọng nhất:

**Use Case: Đăng sản phẩm.**

|  |  |
| --- | --- |
| Actor | Người bán hàng |
| Description | Người bán hàng tiến hành nhập các thông tin mới của sản phẩm như: mã sản phẩm, thông tin sản phẩm và dữ liệu được nhập sẽ gửi về server |
| Operation type | Write |
| Data in Operation | id\_SanPham, thongTinSanPham |
| Frequency | 250 sản phẩm / giờ |
| Data size | 350 bytes |
| Data life | 5 năm |
| Data durability | Majority |

**Use Case: Đặt hàng.**

|  |  |
| --- | --- |
| Actor | Khách hàng |
| Description | Khách hàng sau khi lựa chọn được sản phẩm ưng ý, tiến hành nhập các thông tin của khách hàng nếu như khách hàng chưa đăng ký tài khoản sau đó có thể chọn dịch vụ đi kèm và chọn xác nhận đặt hàng |
| Operation type | Write |
| Data in Operation | id\_SanPham, id\_KhachHang, idMaGiamGia |
| Frequency | 555555 lượt đặt / giờ |
| Data size | 30 bytes |
| Data life | 5 years |
| Data durability | Majority |

1. Đưa ra mô hình dữ liệu ở mức logic: gồm các thực thể lưu trữ dữ liệu (tên thực thể, cấu trúc lưu trữ) mỗi quan hệ giữa chúng

A diagram of a computer

Description automatically generated with medium confidence

1. Đưa ra các data size của document lưu trữ ứng với thao tác ghi (Write) quan trọng:

* **Sản Phẩm**

id\_sanPham: String (12) 12 bytes

tenSanPham: String (30) 30 bytes

moTa: String (200) 200 bytes

gia: double 8 bytes

soLuong: int 4byte

danhGia:[

{

maKhachHang: String(12), 12bytes

noiDung: String(100), 100bytes

luongLike: int 4bytes

ngayGio: ISO Date() 8bytes

} (124bytes)

]

* **Data size**: 12 + 30 + 200 + 8 + 4 +124 ≈ 400 bytes
* **Khách Hàng:**

id\_khachHang: String (12) 12 bytes

tenKhachHang: String (30) 30 bytes

email: String (32) 32 bytes

sdt: String(10) 10 bytes

diachi: String(100) 100bytes

taiKhoan[

{

Username: String(20) 20bytes

Password: String(20) 20bytes

}

] (40bytes\*2= 80 bytes)

* **Data size**: 12 + 30 + 32 + 10 +100 +80 ≈ 300 bytes.
* **Người bán hàng**

id\_nguoiBan: String (12) 12bytes

tenNguoiBan: String (30) 30bytes

email: String (32) 32 bytes

sdt: String(10) 10 bytes

diaChi: String(100) 100bytes

taiKhoan[

{

Username: String(20) 20bytes

Password: String(20) 20bytes

}

] (40bytes)

* **Data size**: 12 + 30 + 32 + 10 +100 +80 ≈ 300 bytes.
* **Hóa đơn**

id\_HoaDon: String (10) 10 bytes

id\_KhachHang: String (12) 12 bytes

thongTinKhachHang: {

id\_khachHang: String (10) 10 bytes

tenKhachHang: String (30) 30 bytes

} (40bytes)

thongTinNguoiBan: {

id\_khachBan: String (10) 10 bytes

tenKhachBan: String (30) 30 bytes

} (40bytes)

thongTinSanPham: [

{

id\_sanPham: String (12) 12 bytes

tenSanPham: String (30) 30 bytes

soLuong: int 4byte

} (46 bytes\*5 = 230 bytes)

]

phieuGiamGia:[

{

maGiamGia: String (12) 12bytes

giaTriGiamGia: double 8bytes

} (20\*5 = 100bytes)

]

tongThanhTien: double 8 bytes

* **Data size**: 10+12+40+40+12+30+4+230+100 ≈ 500 bytes

1. Đưa ra các câu truy vấn có thể có, đưa ra chỉ mục cần tạo:

1. Tìm kiếm danh sách sản phẩm theo tên (Tìm kiếm tương đối, không phân biệt hoa thường):

- Lệnh truy vấn (tìm tất cả sản phẩm có chứa từ “cà phê” trong tên sản phẩm):

db.sanPham.find({tenSanPham : {$regex : '^.\*cà phê.\*$', $options: 'i'}})

- Chỉ mục: {tenSanPham: 1} (Chỉ mục theo trường "tenSanPham" với thứ tự tăng dần):

db.sanPham.createIndex({tenSanPham: 1})

2. Tìm kiếm danh sản phẩm theo khoảng giá:

- Lệnh truy vấn:

db.sanPham.find({gia: {$gte: 30000,$lte: 150000}})

- Chỉ mục: {gia: 1} (Chỉ mục theo trường "gia" với số tăng dần)

db.sanPham.createIndex({gia: 1})

3. Tìm kiếm danh sách sản phẩm theo từ khóa trong mô tả:

- Lệnh truy vấn:

db.sanPham.find({$text:{$search: "cà phê kem Việt Nam Mỹ"}})

- Chỉ mục: {moTa: 1} (Chỉ mục theo trường "moTa" với thứ tự tăng dần)

db.sanPham.createIndex({moTa: 1})

4. Tìm kiếm danh sách người bán hàng theo tên (Tìm kiếm tương đối, không phân biệt hoa thường):

-Lệnh truy vấn:

db.nguoiBan.find({tenNguoiBan : {$regex : '^.\*Coffee Shop.\*$', $options: 'i'}})

- Chỉ mục: {tenNguoiBan: 1} (Chỉ mục theo trường "tenNguoiBan" với thứ tự tăng dần)

db.nguoiBan.createIndex({moTa: 1})

5. Cập nhật thông tin của một của sản phẩm theo mã sản phẩm:

- Lệnh truy vấn:

db.sanPham.updateOne({

maSanPham : 'SP35501296'

},{

$set: {

tenSanPham: 'Cà Phê Cao Cấp Trung Nguyên', gia : 100000

}

}

)

6. Cập nhật danh sách đánh giá sản phẩm theo mã sản phẩm:

- Lệnh truy vấn:

db.sanPham.updateOne({

maSanPham : 'SP35501296'

},{

$addToSet: {

danhGia: {

maKhachHang: 'KH103279',

noiDung: 'Cà phê ngon',

luongLike: 0,

ngayGio : new ISODate("2023-09-15T12:30:15Z")

}

}

}

)

7. Xóa một sản phẩm theo mã sản phẩm:

- Lệnh truy vấn:

db.sanPham.deleteOne({

maSanPham : 'SP35501296'

},{

writeConcern: "majority",

}

)

1. Phác thảo sơ lược về hệ thống server liên quan đến sharding, replicate set:

* **Sharding:**
* Sahrding là một phương pháp để chia dữ liệu thành các phần nhỏ hơn gọi là shard và lưu trữ chúng trên nhiều máy chủ (node) khác nhau. Mục tiêu chính của sharding là tăng khả năng chịu tải và khả năng mở rộng của hệ thống.
* Dung lượng dữ liệu: Hệ thống lưu trữ nhiều thông tin đơn hàng và doanh thu trong 5 năm, và dữ liệu khách hàng quan tâm lưu trữ trong 5 năm. Điều này có thể dẫn đến một lượng lớn dữ liệu. Nếu dung lượng dữ liệu này tiếp tục tăng lên, sharding có thể hữu ích để phân chia dữ liệu vào các shard riêng biệt để quản lý tải.
* Tải công việc: Hệ thống phải xử lý hàng triệu lượt xem từ khách hàng mỗi ngày, và có nhiều tác vụ đặt tour, cập nhật và xóa tour từ người quản lý và nhân viên. Nếu tải công việc này trở nên quá lớn cho một node duy nhất, sharding có thể giúp phân tải công việc và đảm bảo hiệu suất.
* A diagram of a diagram of a router

  Description automatically generatedCơ chế Sharding: Hệ thống có thể sử dụng cơ chế sharding để chia dữ liệu thành các phần nhỏ dựa trên một thuộc tính cụ thể, chẳng hạn như địa điểm đặt tour hoặc thời gian đặt tour. Mỗi phần dữ liệu được lưu trữ trên một shard riêng biệt.
* **Replica Set:**
* Một replicaset là một nhóm các máy chủ (nodes) MongoDB hoạt động cùng nhau để đảm bảo sự sao lưu và sẵn sàng dự phòng.
* Sao lưu và sẵn sàng dự phòng: Hệ thống quản lý tour quan trọng về việc duy trì dữ liệu và hoạt động ổn định, một replicaset với ít nhất 3 node. Điều này cho phép có một node primary (hoạt động), một node secondary (sẵn sàng dự phòng), và một node arbiter (để đảm bảo bầu cử primary nếu cần).
* Khi hệ thống đang phát triển và dự kiến có sự gia tăng đáng kể về dung lượng dữ liệu và tải công việc, thì sharding và sử dụng replicaset với ít nhất 3 node có thể là một lựa chọn tốt để đảm bảo hiệu suất và độ tin cậy của hệ thống.

A diagram of a diagram

Description automatically generated