**Chương 1. Tổng quát**

* 1. **Giới thiệu đề tài**

Trong bối cảnh giá nhà đang tăng cao, việc đầu tư, mua bán và phát triển bất động sản trong thị trường hiện nay rất quan trọng, Các nhà đầu tư cần biết rõ thị hiếu của khách hàng, bên cạch đó xác định các khu vực tiềm năng để tối ưu hóa lợi nhuận đạt được. Đề tài ‘Phân tích giá nhà ‘ nhằm mục đích cung cấp một cách tiếp cận khoa học cho các nhà đầu tư, vừa tạo điều kiện dễ dàng cho người dùng tiếp cận, tra cứu nhưng thông tin chung cư phù hợp với điều kiện của mình.

* 1. **Nhiệm vụ đề tài**

Áp dụng những kĩ thuật thu thập dữ liệu để thu thập thông tin về các đặc điểm và giá hiện tại của các chung cư. Từ đó xây dụng cở sơ dữ liệu phù hợp để tối ưu quá trình xử lí, sử dụng và quản lí dữ liệu một cách mạnh mẽ. Từ đó phân tích dữ liệu qua các biểu đồ trực quan để giúp người dùng tiếp cận một cách hiệu quả. Điều này giúp các nhà đầu tư và người dùng có cái nhìn sâu sắc và tổng quát về thị trường bất động sản, qua đó định hướng được chiến lược phát triển và đầu tư và mua bán.

* + 1. *Tính cấp thiết đề tài*
    2. *Ý nghĩa khóa học và thực tiễn của đề tài*
  1. **Mục tiêu**

Đề tài cung cấp một bộ cở sở dữ liệu về nhưng đặc điểm và giá cả của nhưng chung cư trong thị trường bất động sản HCM, xây đựng một hệ thống truy vấn mạnh mẽ và phân tích các điểm nóng bất động sản tại HCM.

* + 1. *Mục tiêu tổng quát*
    2. *Mục tiêu cụ thể*

Mục tiêu của chúng tôi là xây dựng một bộ dữ liệu tổng quát về thông tin các đặc điểm, tiện ích và giá cả tại các chung cư trên địa bàn HCM. Quy trình thực hiện bao gồm: Lựa chọn trang web, thiết kế cơ sở dữ liệu, thu thập dữ liệu, lưu dữ liệu và cuối cùng là phân tích. Với từng bước một, chúng tôi lựa chọn web batdongsan để thu thap dữ iệu, sử dụng Môngdb để tổ chức và lưu trữ dữ liệu cuối cùng Selenium và sklieam để thu thập và phân tích dữ liệu

* 1. **Đối tượng và phạm vi**
     1. **Đối tượng**
     2. **Phạm vi**
  2. **Phương pháp nghiên cứu**
  3. **Đóng góp của đề tài**

**Chương 2. Cở sở lí thuyết**

* 1. **Selenium Webdriver** 
     1. **Giới thiệu về selenium**

Selenium là một bộ công cụ kiểm thử tự động mã nguồn mở chuyên dụng cho các ứng dụng web, hỗ trợ hoạt động trên nhiều trình duyệt và hệ điều hành khác nhau như Mac, Linux, Windows. Với Selenium, các nhà phát triển và tester hoàn toàn có thể viết các script kiểm thử bằng nhiều ngôn ngữ lập trình như Java, PHP, C#, Ruby hoặc Python. Selenium được sử dụng để tự động hóa các thao tác trên trình duyệt, hay nói cách khác, nó hỗ trợ giả lập các tương tác trên trình duyệt tương tự như một người dùng thực sự. Nhờ đó, các nhà phát triển và tester có thể lập trình để tự động mở trình duyệt, truy cập một liên kết, nhập liệu, tải lên hoặc tải xuống dữ liệu từ trang web, và thậm chí lấy thông tin từ trang.

* + 1. *Giới thiệu về selenium webdriver*

**Selenium WebDriver** là một thành phần quan trọng trong bộ công cụ Selenium, cho phép bạn tương tác trực tiếp với các trình duyệt web một cách linh hoạt và hiệu quả. WebDriver cung cấp một API đơn giản và nhất quán để điều khiển các trình duyệt như Chrome, Firefox, Safari và Edge. WebDriver hoạt động bằng cách gửi lệnh trực tiếp đến trình duyệt thông qua giao thức gốc của nó, giúp giảm thiểu độ trễ và tăng tính ổn định. Điều này cho phép bạn mô phỏng chính xác các hành động của người dùng, bao gồm nhấp chuột, nhập văn bản, kéo thả, và tương tác với các phần tử động trên trang web.

* + 1. *Ưu điểm và hạn chế*

Ưu điểm:

* Miễn phí và mã nguồn mở: Selenium là một công cụ hoàn toàn miễn phí và mã nguồn mở, cho phép người dùng tùy chỉnh và mở rộng theo nhu cầu.
* Hỗ trợ đa nền tảng và đa trình duyệt: Có khả năng chạy trên nhiều hệ điều hành như Windows, macOS, Linux và hỗ trợ các trình duyệt phổ biến như Chrome, Firefox, Safari, Edge.
* Đa dạng ngôn ngữ lập trình: Hỗ trợ nhiều ngôn ngữ lập trình như Java, Python, C#, Ruby, JavaScript ...

Hạn chế:

* Chỉ hỗ trợ ứng dụng web: Selenium chỉ có thể tự động hóa kiểm thử cho các ứng dụng web, không hỗ trợ kiểm thử ứng dụng desktop hoặc mobile.
* Không hỗ trợ kiểm thử hình ảnh và CAPTCHA: Selenium không thể xử lý kiểm thử dựa trên hình ảnh hoặc tương tác với CAPTCHA, giới hạn trong việc kiểm tra giao diện người dùng chi tiết.
* Khó khăn với các phần tử web động: Việc xử lý các phần tử động hoặc các ứng dụng sử dụng nhiều JavaScript có thể phức tạp và dễ gây lỗi.
  1. **Cở sở dữ liệu phân tán**
     1. *Giới thiệu về NoSQL*

Not only SQL **(**NoSQL) là một loại cở sở dữ liệu không sử dụng các mô hình quan hệ truyền thống để lưu trữ và quản lí dữ liệu. NoSQL cho phép thay đổi linh hoạt trong cách tổ chức dữ liệu, phù hợp với dữ liệu phi cấu trúc và bán cầu trúc.

Cở sở dữ liệu NoSQL sử dụng các mô hình lưu trữ khác nhau như lưu trữ theo tài liệu, cặp khóa – giá trị, cột hay đồ thị, điều này cho phép NoSQL lữu trữ dữ liệu linh hoạt hơn so với các cơ sở dữ liệu quan hệ, nơi mà dữ liệu được lưu trữ dưới dạng bảng có cấu trúc cố định.

* + 1. *Các loại cơ sở dữ liệu phân tác*
* Cơ sở dữ liệu tài liệu: dữ liệu được lưu trữ dưới dạng tài liệu thường là BSON hay JSON.
* Cơ sở dữ liệu khóa – giá trị: Lưu trữ dưới dạng khóa – giá trị. Mỗi khóa trỏ đến một giá trị duy nhất.
* Cơ sở dữ liệu cột: Lưu trữ dữ liệu dưới dạng cột thay vì hàng như SQL.
* Cơ sở dữ liệu đồ thị: Tối ưu cho việc lưu trữ và truy vấn dữ liệu quan hệ phức tạp thông qua đồ thị.
  + 1. *Lợi ích của NoSQL*

NoSQL có nhiều lợi ích cho việc lưu trữ và quản lí dữ liệu. Đầu tiên, với khả năng phân tán dữ liệu ở nhiều máy chủ khác nhau, điều này giúp dễ dàng lưu trữ dữ liệu lớn mà không cần năng cấp phần cứng máy chủ hiện .

Thứ hai, khả năng lưu trữ dữ liệu phi cấu trúc và bán cấu trúc với nhiều loại cơ sở dữ liệu khác nhau giúp dễ dàng phát triển các ứng dụng một cách hiệu quả.

Thứ ba, tính linh hoạt của NoSQL còn thể hiện rõ hơn khi thay đổi cấu trúc của dữ liệu một cách dễ dàng mà không cần thay đổi hay phát triển lại như trong cơ sở dữ liệu quan hệ, giảm thiếu thời gian và công sức của nhà phát triển.

Cuối cùng, NoSQL thường cung cấp hiệu suất tốt hơn cho các ứng dụng yêu cầu xử lý dữ liệu lớn hoặc có yêu cầu cao về tốc độ truy cập cao. NoSQL có thể xử lý hàng triệu yêu cầu mỗi giây mà không làm giảm tốc độ hoặc hiệu suất.

* 1. **MongoDb**
     1. *Giới thiệu về MongoDB*

MongoDB là một hệ cơ sở quản trị dữ liệu NoSQL nổi bất, được thiết kế để lưu trữ và quản trị dữ liệu phi cấu trúc hoặc bán cấu trúc. MongoDB được thiết kế nhầm đáp ứng sử phát triển của dữ liệu trong bối cảnh yêu cầu về tốc độ xử lí các dữ liệu lớn.

MongoDB lưu trữ dữ liệu dưới dạng document với dịnh dạng BSON, điều này giúp người dùng lưu trữ dữ liệu một cách linh hoạt, khi thay đổi cấu trúc của một dữ liệu, chúng ta không cần sửa đổi toàn bộ dữ liệu trong cở sở dữ liệu.

MongoDB sử dụng chỉ mục để cải thiện tốc độ truy vấn và tìm kiếm, bên cạnh đó MongoDB hỗ trợ nhiều ngôn ngữ lập trình khác nhau, điều này làm chúng đễ đang tích hợp vào trong các ứng dụng một cách dễ dàng và hiệu quả.

* + 1. *Ưu điểm và hạn chế*

Ưu điểm:

* Hỗ trợ phân tán dữ liệu trên nhiều máy chủ: MongoDB cung cấp khả năng phân tán dữ liệu thông qua tính năng sharding, cho phép chia nhỏ cơ sở dữ liệu và phân phối chúng trên nhiều máy chủ khác nhau, giúp hệ thống dễ dàng mở rộng quy mô khi lượng dữ liệu và số lượng truy vấn tăng lên.
* Sử dụng chỉ mục để tăng tốc độ truy vấn: MongoDB hỗ trợ tạo chỉ mục trên các trường dữ liệu trong bộ sưu tập (collection), tương tự như trong các hệ quản trị cơ sở dữ liệu quan hệ, giúp tối ưu hóa hiệu suất truy vấn bằng cách giảm thiểu số lượng tài liệu cần duyệt qua khi tìm kiếm.
* Cho phép thực hiện các phép tính mà không cần chuyển đổi: Với Aggregation Framework, MongoDB cho phép thực hiện các phép tính phức tạp, như tổng hợp, phân nhóm, và biến đổi dữ liệu, trực tiếp trên cơ sở dữ liệu mà không cần chuyển dữ liệu về ứng dụng để xử lý.

Hạn chế:

* Quản lý dữ liệu phức tạp: MongoDB không hỗ trợ liên kết (join) phức tạp giữa các bộ sưu tập như trong cơ sở dữ liệu quan hệ.
* **Lưu trữ dưới dạng BSON tiêu tốn nhiều bộ nhớ hơn:** Mặc dù BSON cho phép lưu trữ các kiểu dữ liệu phong phú hơn và hỗ trợ truy vấn hiệu quả, nhưng nó thường chiếm nhiều dung lượng hơn so với định dạng văn bản thuần túy.
* **Chi phí cao do tiêu tốn nhiều bộ nhớ:** Do BSON tiêu tốn nhiều dung lượng và MongoDB thường yêu cầu nhiều bộ nhớ để duy trì hiệu suất cao, chi phí vận hành có thể tăng lên đáng kể.