|  |  |
| --- | --- |
|  | BỘ GIÁO DỤC VÀ ĐÀO TẠO  **TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP. HCM** |

**ĐỒ ÁN CƠ SỞ**

**thu thap va phan tich gia chung cu tren trang web batdongsan.com.vn**

Ngành: **Công Nghệ Thông Tin**

Chuyên ngành: **Khoa Học Dữ Liệu**

Giảng viên hướng dẫn: Lê Nhật Tùng

Sinh viên thực hiện:

Trần Văn Thiện MSSV:22864000 Lớp:22DKHA1  
Trần Minh Thông MSSV: 22864000 Lớp:22DKHA1

Trần Tuấn Kiệt MSSV:2286400013 Lớp: 22DKHA1

TP. Hồ Chí Minh, ngày 31 tháng 10 năm 2024

TRƯỜNG ĐẠI HỌC CÔNG NGHỆ TP.HCM

**KHOA CÔNG NGHỆ THÔNG TIN**

**HƯỚNG DẪN TRÌNH BÀY ĐỒ ÁN CƠ SỞ**

1. **BỐ CỤC**

Bố cục đồ án được minh họa qua mục lục:

|  |
| --- |
| **MỤC LỤC**  Trang phụ bìa  Lời cam đoan  Mục lục 1  Danh mục các ký hiệu, các chữ viết tắt 2  Danh mục các bảng 3  Danh mục các hình vẽ, đồ thị 4  Chương 1. TỔNG QUAN 6  1.1 … 6  1.1.1 ... 6  1.1.2… 7  1.2 …. 8  Chương 2. … 9  2.1 … 9  2.1.1… 9  2.1.2… 10  2.2 …  …  Chương … - KẾT LUẬN VÀ KIẾN NGHỊ xx  TÀI LIỆU THAM KHẢO xx  PHỤ LỤC (Nếu có) xx |

**LỜI CAM ĐOAN**

**Nhóm Lab42 xin cam đoan mọi thông tin và nghiên cứu này này là trung thực, được thu thập và phân tích đựa trên các nguồn chính thống và tinh cậy.**

**Chương 1. Tổng quát**

**1.1 Giới thiệu đề tài**

*1.1.1 Bối cảnh và lý do chọn đề tài*Trong những năm gần đây, tốc độ phát triển của các đô thị lớn ở Việt Nam, đặc biệt là TP.HCM, đã khiến giá nhà đất tăng mạnh. Điều này đặt ra thách thức cho các nhà đầu tư, nhà phát triển bất động sản và cả người mua nhà trong việc định giá, đầu tư và lựa chọn bất động sản phù hợp. Các yếu tố như vị trí, tiện ích, và tình hình kinh tế cũng ảnh hưởng lớn đến quyết định đầu tư.

*1.1.2 Mục đích của đề tài*  
Đề tài "Phân tích giá nhà" nhằm đưa ra một phương pháp khoa học để thu thập và phân tích dữ liệu về thị trường bất động sản, đặc biệt là phân khúc chung cư tại TP.HCM. Nghiên cứu này không chỉ hỗ trợ các nhà đầu tư xác định tiềm năng của từng khu vực mà còn giúp người mua có cái nhìn tổng quan về giá cả và điều kiện mua nhà.

*1.1.3 Lợi ích cho nhà đầu tư và người dùng cuối*  
Các nhà đầu tư sẽ có thể đưa ra các quyết định chính xác hơn dựa trên dữ liệu cụ thể và các phân tích về xu hướng giá cả. Người mua nhà cũng có thể sử dụng hệ thống để tra cứu các chung cư phù hợp với điều kiện tài chính và yêu cầu cá nhân của mình.

**1.2 Nhiệm vụ của đề tài**

Đề tài sẽ thực hiện một loạt các nhiệm vụ sau để hoàn thành mục tiêu đã đề ra:

*1.2.1 Thu thập dữ liệu*  
Áp dụng các kỹ thuật thu thập dữ liệu từ các nguồn đáng tin cậy, trong đó trang web Batdongsan.vn là một lựa chọn chủ yếu do chứa nhiều thông tin liên quan đến bất động sản ở Việt Nam. Quá trình này bao gồm việc xây dựng các công cụ thu thập (crawler) bằng các thư viện như Selenium để tự động hóa việc lấy dữ liệu từ trang web.

*1.2.2 Xây dựng cơ sở dữ liệu*  
Dữ liệu thu thập sẽ được tổ chức và lưu trữ trong MongoDB, một hệ quản trị cơ sở dữ liệu NoSQL mạnh mẽ, linh hoạt và phù hợp cho việc lưu trữ dữ liệu bán cấu trúc như thông tin bất động sản. Cơ sở dữ liệu này cần được tối ưu hóa để đảm bảo khả năng truy xuất nhanh và hiệu quả cho quá trình phân tích dữ liệu.

*1.2.3 Phân tích dữ liệu và trực quan hóa*  
Dữ liệu được xử lý và phân tích để đưa ra các biểu đồ trực quan giúp nhà đầu tư và người mua dễ dàng tiếp cận thông tin. Các thư viện như Matplotlib, Seaborn có thể được sử dụng để xây dựng các biểu đồ này, cung cấp thông tin về xu hướng giá cả theo khu vực, so sánh các tiện ích, và xác định các "điểm nóng" về giá trị bất động sản.

**1.3. Tính cấp thiết và ý nghĩa khoa học của đề tài**

*1.3.1 Tính cấp thiết của dự án*  
TP.HCM là thành phố lớn nhất Việt Nam với mật độ dân số cao và nhu cầu về nhà ở ngày càng tăng, đặc biệt là trong bối cảnh người dân đổ về đô thị để tìm kiếm cơ hội việc làm và cải thiện chất lượng cuộc sống. Với mức giá nhà đang tăng nhanh, nhiều người mua nhà gặp khó khăn trong việc tìm kiếm bất động sản phù hợp. Dự án này giúp người mua tiếp cận thông tin minh bạch, hỗ trợ họ trong việc lựa chọn đúng đắn và phù hợp với tài chính.

*1.3.2 Ý nghĩa khoa học*  
Đề tài không chỉ mang lại lợi ích thực tiễn mà còn có ý nghĩa khoa học khi ứng dụng các phương pháp xử lý dữ liệu hiện đại trong lĩnh vực bất động sản. Việc thu thập và phân tích dữ liệu một cách có hệ thống giúp tạo ra một bộ dữ liệu quan trọng cho các nghiên cứu tiếp theo và góp phần xây dựng nền tảng khoa học dữ liệu bất động sản ở Việt Nam.

**4. Mục tiêu**

*1.4.1 Mục tiêu chung*  
Xây dựng một cơ sở dữ liệu chi tiết và cập nhật về các đặc điểm và giá cả của các chung cư trên thị trường TP.HCM. Phân tích dữ liệu và phát triển hệ thống trực quan để cung cấp thông tin hữu ích cho người dùng.

*1.4.2 Mục tiêu cụ thể*

* Thu thập dữ liệu: Từ trang web Batdongsan.vn, thu thập thông tin chi tiết về giá cả, tiện ích, diện tích, vị trí của các chung cư.
* Thiết kế cơ sở dữ liệu: Lựa chọn MongoDB làm nền tảng để lưu trữ dữ liệu và tối ưu hóa cho khả năng truy xuất dữ liệu nhanh chóng.
* Phân tích dữ liệu: Sử dụng công cụ trực quan hóa dữ liệu để tạo ra các biểu đồ và đồ thị về xu hướng giá nhà, so sánh các khu vực, và phát hiện các khu vực tiềm năng.
* Xây dựng hệ thống truy vấn: Phát triển các chức năng tìm kiếm để người dùng có thể lọc và xem chi tiết các chung cư phù hợp với nhu cầu của họ.

**1.5 Đối tượng và phạm vi nghiên cứu**

*1.5.1 Đối tượng nghiên cứu*  
Đối tượng của nghiên cứu này là các dữ liệu liên quan đến chung cư tại TP.HCM, bao gồm các thông tin như giá bán, diện tích, tiện ích, và vị trí. Ngoài ra, đề tài cũng xem xét thị hiếu và nhu cầu của người mua nhà để từ đó tạo ra các báo cáo phân tích phù hợp.

*1.5.2 Phạm vi nghiên cứu*  
Phạm vi của đề tài là thị trường bất động sản chung cư tại TP.HCM. Đề tài sẽ tập trung vào việc thu thập dữ liệu từ các khu vực trung tâm, các quận có tốc độ phát triển cao, cũng như các khu vực đang nổi lên trong thời gian gần đây. Những dữ liệu này sẽ được giới hạn ở TP.HCM để đảm bảo tính chính xác và tập trung của nghiên cứu.

**1.6 Phương pháp nghiên cứu**

*1.6.1 Phương pháp thu thập dữ liệu*  
Dữ liệu sẽ được thu thập từ các trang web như Batdongsan.vn bằng cách sử dụng kỹ thuật web scraping. Công cụ Selenium sẽ được sử dụng để tự động hóa quá trình thu thập dữ liệu và lưu trữ chúng trong cơ sở dữ liệu MongoDB. Để đảm bảo tính hợp pháp và tuân thủ quy định, việc thu thập dữ liệu sẽ thực hiện trong giới hạn tần suất cho phép.

*1.6.2 Phương pháp phân tích dữ liệu*  
Sau khi thu thập, dữ liệu sẽ được xử lý và phân tích bằng các công cụ như Pandas và Scikit-learn. Các kỹ thuật như phân tích hồi quy sẽ được áp dụng để hiểu mối quan hệ giữa giá cả và các đặc điểm của chung cư. Ngoài ra, các công cụ trực quan hóa như Matplotlib và Seaborn sẽ được sử dụng để tạo các biểu đồ trực quan giúp minh họa xu hướng giá cả và các "điểm nóng" về bất động sản.

*1.6.3 Phương pháp trực quan hóa dữ liệu*  
Để giúp người dùng dễ dàng tiếp cận và hiểu thông tin, đề tài sẽ sử dụng các biểu đồ và bản đồ để trình bày dữ liệu. Các biểu đồ giá nhà theo thời gian, bản đồ các khu vực tiềm năng và biểu đồ so sánh giữa các chung cư sẽ được sử dụng để người dùng có thể nắm bắt thông tin một cách trực quan.

**1.7 Kết luận và Kỳ vọng của đề tài**

Dự án này hy vọng sẽ cung cấp một cơ sở dữ liệu chi tiết, hữu ích cho các nhà đầu tư và người mua nhà, cũng như xây dựng một hệ thống truy vấn mạnh mẽ để người dùng có thể dễ dàng tìm kiếm thông tin về các chung cư phù hợp. Các kết quả phân tích không chỉ hỗ trợ các quyết định đầu tư mà còn giúp người mua tiếp cận thị trường bất động sản một cách dễ dàng hơn.

Ngoài ra, đề tài này kỳ vọng sẽ trở thành một nguồn dữ liệu tham khảo quan trọng cho các nghiên cứu tiếp theo về bất động sản ở Việt Nam. Việc áp dụng các phương pháp khoa học dữ liệu cũng tạo cơ hội để phát triển các công cụ hỗ trợ và phân tích thị trường trong tương lai.

**1.8 Tổng quan lý thuyết**

*1.8.1 Khái niệm và phân loại bất động sản*

Bất động sản là một loại tài sản bao gồm đất đai và những tài sản gắn liền với đất, ví dụ như nhà cửa, các công trình xây dựng. Bất động sản có thể chia thành nhiều loại như đất ở, đất thương mại, đất công nghiệp, và các loại hình bất động sản hỗn hợp khác. Trong phạm vi nghiên cứu này, chúng ta sẽ tập trung vào phân khúc chung cư - một dạng bất động sản nhà ở phổ biến trong các đô thị lớn như TP.HCM.

*1.8.2 Các yếu tố ảnh hưởng đến giá nhà*

Giá trị bất động sản nói chung và giá nhà nói riêng phụ thuộc vào nhiều yếu tố, bao gồm vị trí, cơ sở hạ tầng, tiện ích xung quanh, tình hình kinh tế, và chính sách của nhà nước. Những yếu tố này có tác động khác nhau ở mỗi khu vực. Trong TP.HCM, vị trí địa lý có thể coi là yếu tố quyết định lớn nhất đối với giá nhà. Ngoài ra, việc tiếp cận với các khu vực tiện ích như trường học, bệnh viện, trung tâm mua sắm, và giao thông công cộng cũng ảnh hưởng đáng kể đến giá trị của các căn hộ.

*1.8.3 Ứng dụng khoa học dữ liệu trong bất động sản*

Khoa học dữ liệu hiện nay được sử dụng rộng rãi trong lĩnh vực bất động sản để phân tích và dự đoán giá trị bất động sản. Các phương pháp khoa học dữ liệu, như phân tích hồi quy, phân tích cụm, và các mô hình học máy khác, có khả năng cung cấp các dự đoán chính xác hơn về xu hướng giá nhà, từ đó hỗ trợ các nhà đầu tư và người mua đưa ra quyết định thông minh hơn. Việc áp dụng các mô hình dữ liệu hiện đại vào nghiên cứu giá nhà sẽ là một bước tiến quan trọng trong ngành bất động sản ở Việt Nam.

**1.9 Tình hình thị trường bất động sản tại TP.HCM**

*1.9.1 Tăng trưởng dân số và nhu cầu nhà ở*

TP.HCM là thành phố đông dân nhất Việt Nam và có tốc độ gia tăng dân số nhanh chóng. Điều này làm tăng nhu cầu về nhà ở, đặc biệt là ở các khu vực trung tâm thành phố và các quận đang phát triển nhanh. Theo thống kê, nhu cầu về nhà ở tại TP.HCM tăng đều đặn qua các năm, trong đó phần lớn là nhu cầu về căn hộ chung cư - loại hình phù hợp với người dân đô thị do tiện lợi và chi phí hợp lý.

*1.9.2 Các khu vực "nóng" của thị trường*

Các quận trung tâm như Quận 1, Quận 3, Bình Thạnh, và Quận 2 thường có giá nhà đất cao do có hệ thống cơ sở hạ tầng phát triển, tiện ích đa dạng và gần các trung tâm kinh tế. Bên cạnh đó, một số quận ven như Quận 9, Thủ Đức (nay là thành phố Thủ Đức), và Quận 7 cũng đang nổi lên với nhiều dự án bất động sản mới, tạo ra nhiều cơ hội cho các nhà đầu tư.

*1.9.3 Thách thức của thị trường bất động sản TP.HCM*

Thị trường bất động sản TP.HCM không chỉ mang lại nhiều cơ hội mà còn tiềm ẩn nhiều thách thức. Việc giá nhà đất tăng cao, hạ tầng giao thông chưa đáp ứng kịp nhu cầu, và tình trạng "bong bóng" bất động sản là những yếu tố đáng lo ngại. Đối với người mua nhà, việc tìm kiếm một căn hộ có giá hợp lý và đáp ứng nhu cầu trở thành một vấn đề nan giải. Đề tài này sẽ giúp cung cấp thông tin và dự đoán xu hướng thị trường để người dùng có thể đưa ra quyết định tốt hơn.

**1.10 Kết quả dự kiến và ứng dụng**

*1.10.1 Kết quả dự kiến*

Dự án sẽ cung cấp một cơ sở dữ liệu chi tiết về thị trường chung cư tại TP.HCM, cùng với hệ thống trực quan hóa dữ liệu để người dùng có thể dễ dàng xem các thông tin về giá cả, diện tích, tiện ích và các yếu tố khác. Ngoài ra, các mô hình dự đoán sẽ giúp nhà đầu tư và người mua nắm bắt được xu hướng giá trong tương lai và có quyết định phù hợp.

*1.10.2 Ứng dụng thực tiễn*

Các nhà đầu tư có thể sử dụng các kết quả phân tích và dự đoán để xác định các khu vực tiềm năng, tối ưu hóa chiến lược đầu tư. Người mua nhà có thể dễ dàng tìm kiếm và so sánh các chung cư để đưa ra quyết định phù hợp với nhu cầu cá nhân. Dự án này cũng sẽ là một nền tảng tham khảo cho các nhà nghiên cứu và phát triển công nghệ trong lĩnh vực bất động sản ở Việt Nam.

**1.11 Đóng góp của đề tài**

*1.11.1 Đóng góp thực tiễn*

Đề tài này mang lại lợi ích thực tiễn cho cả nhà đầu tư và người mua nhà, giúp họ ra quyết định chính xác và tiết kiệm thời gian trong quá trình tìm kiếm bất động sản. Đây là một công cụ hữu ích, minh bạch giúp tối ưu hóa quá trình mua bán và đầu tư bất động sản.

*1.12.2 Đóng góp khoa học*

Đề tài cũng có ý nghĩa khoa học khi áp dụng các phương pháp xử lý dữ liệu và học máy hiện đại vào phân tích thị trường bất động sản. Nó góp phần xây dựng cơ sở dữ liệu lớn cho các nghiên cứu tiếp theo và tạo điều kiện để phát triển các công nghệ tiên tiến trong lĩnh vực này.

**Chương 2. Cở sở lí thuyết**

**2.1 Selenium Webdriver**

*2.1.1 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.

*2.1.2 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.

*2.1.3 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.

**2.2 Cở sở dữ liệu phân tán**

*2.2.1 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.

*2.2.2 Các loại cơ sở dữ liệu phân tán*

- 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.

*2.2.3 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.

**2.3 MongoDb**

*2.3.1 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.

*2.3.2 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.

**2.4 Python**

*2.4.1 Giới thiệu về python*

Python là một ngôn ngữ lập trình bậc cao, được phát triển bởi Guido van Rossum và phát hành lần đầu vào năm 1991. Được thiết kế với mục tiêu đơn giản và dễ đọc, Python trở thành lựa chọn hàng đầu cho nhiều lập trình viên và nhà phát triển trên khắp thế giới. Với sự linh hoạt, cú pháp đơn giản, khả năng mở rộng và hỗ trợ đa nền tảng, Python đã trở thành lựa chọn hàng đầu cho nhiều công ty và tổ chức lớn, từ các công ty công nghệ như Google, Facebook, và Netflix, các viện nghiên cứu và các trường đại học. Python nổi bật với tính linh hoạt và đa năng, được sử dụng trong rất nhiều lĩnh vực khác nhau như phát triển web, khoa học dữ liệu, trí tuệ nhân tạo, tự động hóa, và lập trình nhúng.

*2.4.2 Đặc điểm chính*

- Cú pháp đơn giản là dễ hiểu: Python có cú pháp rõ ràng, dễ đọc và gần với ngôn ngữ tự nhiên, giúp người mới học lập trình dễ dàng tiếp cận.

- Ngôn ngữ lập trình bậc cao: Python là ngôn ngữ lập trình bậc cao, cho phép lập trình viên tập trung vào giải quyết vấn đề vì quan tâm đến các kỹ thuật như quản lý bộ nhớ.

- Thư viện phong phú: Python có một kho thư viện phong phú và mạnh mẽ, hỗ trợ hầu hết mọi nhu cầu lập trình.

- Mã nguồn mở: Python là mã nguồn mở và hoàn toàn miễn phí, giúp người dùng có thể tiếp cận dễ dàng tiếp cận.

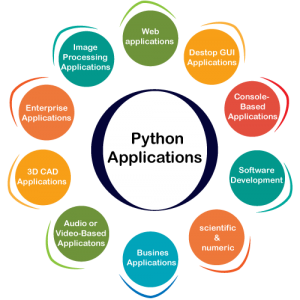
*2.4.3 Một số thứ viện trong python*

Python có một kho thư viện phong phú và mạnh mẽ, hỗ trợ hầu hết mọi nhu cầu lập trình. Một số thư viện nổi tiếng bao gồm:

* Khoa học dữ liệu và học máy: Pandas, NumPy, Scikit-learn, TensorFlow, PyTorch.
* Phát triển web: Django, Flask.
* Thu thập dữ liệu: Selenium, BeautifulSoup, Scrapy.
* Xử lý hình ảnh: OpenCV, PIL.
* Xử lý ngôn ngữ tự nhiên: NLTK, Transformers.

Các thư viện này giúp Python trở thành một ngôn ngữ mạnh mẽ cho xử lý, hỗ trợ đa dạng các tác vụ từ cơ bản đến nâng cao.

*2.4.4 Ứng dụng python trong một số lĩnh vực*



Hình 1.1 Một số tính năng và ứng dụng python

* Ứng dụng khoa học dữ liệu và học máy: python là ngôn ngữ chủ đạo trong các lĩnh vực khoa học dữ liệu và học máy, các ứng dụng bao gồm phân tích dữ liệu, xử lý dữ liệu lớn, và phát triển mô hình học máy, dự đoán, phân loại hình ảnh, cho đến xử lý ngôn ngữ tự nhiên.
* Ứng dụng trong xử lý ngôn ngữ tự nhiên: Các ứng dụng Python trong xử lý ngôn ngữ tự nhiên bao gồm chatbot, phân tích cảm xúc, dịch máy, và tóm tắt văn bản. Python được sử dụng để phân tích, hiểu và xử lý văn bản trong các ngôn ngữ tự nhiên.
* Ứng dụng trong xử lý ảnh: Python là lựa chọn hàng đầu để xây dựng các ứng dụng AI như nhận diện hình ảnh, phân tích video, và phát triển các hệ thống tự học, giúp xử lý ảnh, nhận diện khuôn mặt, nhận dạng đối tượng, và nhiều tác vụ thị giác máy tính khác.
* Ứng dụng tài chính: Python được sử dụng rộng rãi trong lĩnh vực tài chính để phân tích thị trường, dự đoán xu hướng, và xây dựng các mô hình tài chính. Matplotlib giúp phân tích và trực quan hóa dữ liệu tài chính, hỗ trợ quyết định đầu tư và quản lý rủi ro.

**2.5 Maplotlib**

*2.5.1 Giới thiệu Maplotlib*

Matplotlib là một thư viện trong Python, được sử dụng rộng rãi để tạo các biểu đồ từ đơn giản đến phức tạp. Matplotlib ra đời với mục tiêu giúp các nhà phân tích và lập trình viên dễ dàng trực quan hóa dữ liệu. Thư viện này hỗ trợ nhiều loại biểu đồ như biểu đồ đường, biểu đồ cột, biểu đồ tròn, biểu đồ phân tán và nhiều loại biểu đồ khác. Một trong những ưu điểm lớn của Matplotlib là tính tùy chỉnh cao, cho phép điều chỉnh chi tiết biểu đồ như màu sắc, nhãn... Matplotlib dễ dàng tích hợp với các thư viện khác như Pandas, NumPy và SciPy, giúp cho việc trực quan hóa dữ liệu trong các dự án khoa học dữ liệu, học máy và phân tích tài chính trở nên đơn giản và hiệu quả.

*2.5.2 Đặc điểm của Maplotlib*

* Hỗ trợ đa dạng loại biểu đồ: Matplotlib cho phép tạo ra nhiều loại biểu đồ khác nhau như biểu đồ đường, biểu đồ cột, biểu đồ tròn, biểu đồ phân tán, biểu đồ hộp...
* Tính tùy chỉnh cao: Matplotlib cho phép người dùng tùy chỉnh gần như mọi chi tiết trên biểu đồ, từ màu sắc, kiểu đường, cho đến cỡ chữ, nhãn trục, và nền biểu đồ.
* Tương thích với các thư viện khác: Matplotlib dễ dàng tích hợp với các thư viện phân tích dữ liệu như Pandas, NumPy, SciPy, và các thư viện học máy như Scikit-Learn.

2.5.3 Một số hạn chế Maplotlib

* Hạn chế khi sử dụng trên các ứng dụng web: Matplotlib không dễ dàng tích hợp vào các ứng dụng web để tạo biểu đồ trực tiếp. Điều này làm giảm tính linh hoạt khi xây dựng các ứng dụng trực quan hóa dữ liệu tương tác trên web.
* Hiệu suất thấp khi xử lý dữ liệu lớn: Matplotlib không được tối ưu hóa cho việc xử lý và trực quan hóa các tập dữ liệu lớn. Khi làm việc với dữ liệu có hàng trăm nghìn điểm trở lên, hiệu suất của Matplotlib giảm đáng kể, khiến biểu đồ hiển thị chậm hoặc không phản hồi.

Mặc dù có những hạn chế, Matplotlib vẫn là một công cụ quan trọng và mạnh mẽ trong Python để trực quan hóa dữ liệu. Tuy nhiên, khi yêu cầu các tính năng cao cấp hơn về hiệu suất, hoặc đồ họa, người dùng có thể kết hợp Matplotlib với các thư viện khác hoặc chuyển sang các công cụ trực quan hóa khác phù hợp hơn.

*2.5.4 Ứng dụng Maplotlib*

Matplotlib được sử dụng rộng rãi trong các lĩnh vực như khoa học dữ liệu, tài chính, thống kê, học máy và bất kỳ lĩnh vực nào cần trực quan hóa dữ liệu. Nó đặc biệt hữu ích trong việc khám phá và phân tích dữ liệu, tạo báo cáo và trình bày kết quả.

**Chương 3. Phương pháp thực nghiệm**

Nhóm chúng tôi đã tiến hành thu thập dữ liệu từ trang web batdongsan.com.vn thông qua công cụ tự động hóa Selenium WebDriver. Quá trình này giúp chúng tôi thu thập thông tin từ nhiều mục khác nhau trên trang web, bao gồm thông tin về bất động sản, giá cả, diện tích, vị trí, và các đặc điểm chi tiết khác. Sau khi thu thập, chúng tôi chuyển đổi và trích xuất dữ liệu thu được ra thành file JSON để dễ dàng xử lý và phân tích trong các bước tiếp theo.

Nhằm tối ưu hóa việc lưu trữ, truy xuất và quản lý dữ liệu, chúng tôi quyết định lưu trữ tất cả các thông tin này trong cơ sở dữ liệu MongoDB. MongoDB là một hệ quản trị cơ sở dữ liệu NoSQL mạnh mẽ, cho phép lưu trữ dữ liệu dưới dạng tài liệu linh hoạt, giúp việc xử lý dữ liệu không cần tuân theo cấu trúc bảng chặt chẽ. Bằng cách sử dụng MongoDB, chúng tôi có thể thực hiện các truy vấn một cách nhanh chóng, hỗ trợ tìm kiếm, trích xuất và phân tích thông tin hữu ích từ dữ liệu bất động sản mà chúng tôi đã thu thập được.

***3.1 Thu thập dữ liệu***

Quá trình thu thập dữ liệu từ trang web batdongsan.com.vn của nhóm chúng tôi được thực hiện thông qua các bước chi tiết và khoa học nhằm đảm bảo tính đầy đủ, chính xác và hiệu quả. Dưới đây là từng bước trong quá trình thu thập và xử lý dữ liệu:

1. Thu thập đường dẫn các trang chung cư
   1. Đầu tiên, chúng tôi truy cập vào trang web batdongsan.com.vn và thực hiện lấy mẫu từ một số trang bất động sản để phân tích và xác định cấu trúc URL của các trang về chung cư. Việc lấy mẫu này giúp chúng tôi hiểu rõ cách thức mà trang web tổ chức dữ liệu, cũng như xác định các quy luật và mẫu chung trong cấu trúc đường dẫn của trang.
   2. Sau khi có được cấu trúc URL, chúng tôi sử dụng Selenium Webdriver để tiến hành thu thập các đường dẫn liên quan đến các trang chung cư một cách tự động.. Quy trình này đảm bảo rằng chúng tôi có thể thu thập đường dẫn của tất cả các trang liên quan đến chung cư mà không bỏ sót bất kỳ trang nào, giúp xây dựng một danh sách URL đầy đủ.
   3. Sau khi hoàn tất việc thu thập đường dẫn, chúng tôi lưu trữ tất cả các URL này để tiến hành thu thập những thuộc tính quan trọng.
2. Trích xuất các thuộc tính của chung cư
   1. Sau khi thu thập đầy đủ các đường link, chúng tôi tiến hành truy cập vào từng trang cụ thể để thu thập thông tin chi tiết về từng bất động sản chung cư. Đầu tiên, chúng tôi lấy mẫu một vài trang chung cư và phân tích để xác định những thuộc tính quan trọng cần thu thập. Những thuộc tính này thường bao gồm thông tin về giá bán, diện tích, số phòng, địa chỉ, tiện ích, và các thông tin khác có liên quan đến đặc điểm và điều kiện của bất động sản
   2. Tiếp theo, sau khi xác định các thuộc tính quan trọng, chúng tôi tiến hành phân tích và xác định các thẻ HTML cụ thể chứa dữ liệu này trong cấu trúc trang web. Điều này bao gồm việc tìm kiếm các thẻ chứa giá cả, diện tích, địa chỉ, và các đặc điểm chi tiết khác của bất động sản...
   3. Bằng cách xác định các thẻ HTML cụ thể, chúng tôi có thể cấu hình Selenium để thu thập đúng những phần thông tin cần thiết, đảm bảo rằng không bỏ sót dữ liệu quan trọng và giảm thiểu nguy cơ thu thập dữ liệu không liên quan hoặc nhiễu.
   4. Chúng tôi sử dụng vòng lặp để lần lượt truy cập vào từng URL trong danh sách đã lưu và thực hiện trích xuất các thuộc tính theo cách tự động. Nhờ vậy, toàn bộ quy trình thu thập dữ liệu từ các trang chi tiết của từng chung cư trở nên tự động.
   5. Sau khi hoàn tất việc thu thập, toàn bộ dữ liệu được lưu trữ trong file JSON để thuận tiện cho việc quản lý và xử lý sau này.

3.1 Mô tả dữ liệu

Dữ liệu chúng tôi thu thập bao gồm 13.000 mẫu, mỗi mẫu chứa thông tin chi tiết về một chung cư. Các đặc trưng được thu thập gồm 11 thuộc tính quan trọng, thể hiện các khía cạnh chính của từng bất động sản. Dưới đây là bảng mô tả các đặc trưng của dữ liệu:

|  |  |
| --- | --- |
| Tên biến | Mô tả |
| Title | Thông tin chính xác địa chỉ chung cư |
| Name Project | Tên dự án |
| Arena | Diện tích chung cư |
| Price | Giá chung cư |
| Bedrooms | Số lượng phòng ngủ |
| Bathrooms | Số lượng phòng vệ sinh |
| LegalInformation | Pháp lý mua bán |
| District | Quận |
| ProjectProgress | Tiến độ dự án |

Bảng 1.2 Bảng mô tả các biến

**3.2. Cài đặt MongoDB và Kết nối**cần cài đặt thư viện pymongo để kết nối Python với MongoDB bằng lệnh sau: pip install pymongo  
Sau đó, khởi tạo kết nối MongoDB và tạo một cơ sở dữ liệu cùng với một bộ sưu tập để lưu trữ dữ liệu chung cư.  
*3.2.1 Mã Python để Lưu Dữ Liệu vào MongoDB*giả sử đã có file JSON chứa dữ liệu chung cư. Chúng ta sẽ đọc file JSON này và lưu từng mẫu vào MongoDB.  
  
import json

from pymongo import MongoClient

# Kết nối MongoDB

client = MongoClient("mongodb://localhost:27017/") # Thay đổi URI nếu cần thiết

db = client["batdongsan\_db"] # Tên cơ sở dữ liệu

collection = db["chung\_cu\_data"] # Tên bộ sưu tập

# Đọc dữ liệu từ file JSON

with open("property\_data.json", "r", encoding="utf-8") as file:

data = json.load(file)

# Lưu từng mẫu dữ liệu vào MongoDB

if isinstance(data, list):

collection.insert\_many(data) # Nếu dữ liệu là danh sách các bản ghi

else:

collection.insert\_one(data) # Nếu chỉ có một bản ghi

print("Đã lưu dữ liệu vào MongoDB.")

**3.3. Các Truy vấn Mẫu trong MongoDB**  
đây là một số truy vấn mẫu để có thể phân tích và tìm kiếm thông tin từ bộ sưu tập chung\_cu\_data trong MongoDB.  
  
*3.3.1. tìm chung cư theo tên dự án*  
  
project\_name = "Dự án A" # Thay bằng tên dự án cụ thể

result = collection.find({"Name Project": project\_name})

for chung\_cu in result:

print(chung\_cu)   
  
*3.3.2. Tìm chung cư có diện tích lớn hơn một giá trị cụ thể*  
  
arena\_threshold = 50 # Diện tích lớn hơn 50m²

result = collection.find({"Arena": {"$gt": arena\_threshold}})

for chung\_cu in result:

print(chung\_cu)  
  
*3.3.3. Tìm chung cư ở một quận cụ thể và có giá trong một khoảng nhất định*  
  
district = "Quận 1"

price\_min = 2 # Giá tối thiểu (triệu đồng)

price\_max = 5 # Giá tối đa (triệu đồng)

result = collection.find({

"District": district,

"Price": {"$gte": price\_min, "$lte": price\_max}

})

for chung\_cu in result:

print(chung\_cu)  
3.4. district = "Quận 1"

price\_min = 2 # Giá tối thiểu (triệu đồng)

price\_max = 5 # Giá tối đa (triệu đồng)

result = collection.find({

"District": district,

"Price": {"$gte": price\_min, "$lte": price\_max}

})

for chung\_cu in result:

print(chung\_cu)  
  
*3.3.4 Tìm chung cư có số lượng phòng ngủ cụ thể*  
  
bedrooms\_count = 2 # Số phòng ngủ

result = collection.find({"Bedrooms": bedrooms\_count})

for chung\_cu in result:

print(chung\_cu)  
 *3.3.5 Tìm chung cư có tiến độ dự án là "Đang xây dựng"*  
result = collection.find({"ProjectProgress": "Đang xây dựng"})

for chung\_cu in result:

print(chung\_cu)  
 *3.3.6 Đếm số lượng chung cư ở mỗi quận*  
pipeline = [

{"$group": {"\_id": "$District", "count": {"$sum": 1}}}

]

result = collection.aggregate(pipeline)

for doc in result:

print(f"Quận: {doc['\_id']}, Số lượng: {doc['count']}")  
  
*3.3.7 Tính giá chung bình của chung cư trong một quận cụ thể*  
  
district = "Quận 1"

pipeline = [

{"$match": {"District": district}},

{"$group": {"\_id": "$District", "average\_price": {"$avg": "$Price"}}}

]

result = collection.aggregate(pipeline)

for doc in result:

print(f"Quận: {doc['\_id']}, Giá trung bình: {doc['average\_price']}")  
  
**3.4 Đóng Kết Nối MongoDB**Sau khi hoàn thành các thao tác, đóng kết nối MongoDB để giải phóng tài nguyên.  
client.close()