**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA HỆ THỐNG THÔNG TIN**

**TRƯƠNG THỊ BÍCH CHI**

**LƯU NGUYỄN**

**ĐỌC VÀ TƯ VẤN SÁCH ĐIỆN TỬ**

**TRÊN ỨNG DỤNG DI ĐỘNG**

**READING AND RECOMMENDING E-BOOKS**

**ON MOBILE APPLICATION**

**KỸ SƯ NGÀNH HỆ THỐNG THÔNG TIN**

**TP. HỒ CHÍ MINH, 6/2019**

**ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN**

**KHOA HỆ THỐNG THÔNG TIN**

**TRƯƠNG THỊ BÍCH CHI – 15520062**

**LƯU NGUYỄN - 15520561**

**ĐỌC VÀ TƯ VẤN SÁCH ĐIỆN TỬ**

**TRÊN ỨNG DỤNG DI ĐỘNG**

**READING AND RECOMMENDING E-BOOKS**

**ON MOBILE APPLICATION**

**KỸ SƯ NGÀNH HỆ THỐNG THÔNG TIN**

**GIẢNG VIÊN HƯỚNG DẪN**

**THS. THÁI BẢO TRÂN**

**TP. HỒ CHÍ MINH, 6/2019**

DANH SÁCH HỘI ĐỒNG BẢO VỆ KHÓA LUẬN

Hội đồng chấm khóa luận tốt nghiệp, thành lập theo Quyết định số …………………… ngày ………………….. của Hiệu trưởng Trường Đại học Công nghệ Thông tin.

* 1. …………………………………………. – Chủ tịch.
  2. …………………………………………. – Thư ký.
  3. …………………………………………. – Ủy viên.
  4. …………………………………………. – Ủy viên.

|  |  |
| --- | --- |
| **ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |
|  | *TP. HCM, ngày…..tháng…..năm……..* |

**NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP**

**(CỦA CÁN BỘ HƯỚNG DẪN)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên khóa luận:** | | | |
| **ĐỌC VÀ TƯ VẤN SÁCH ĐIỆN TỬ TRÊN ỨNG DỤNG DI ĐỘNG** | | | |
| **Nhóm SV thực hiện:** | | | **Cán bộ hướng dẫn:** |
| Trương Thị Bích Chi  Lưu Nguyễn | 15520062  15520561 | | ThS. Thái Bảo Trân |
| **Đánh Giá Khóa Luận:**   1. **Về cuốn báo cáo:**   Số trang 110 Số chương 5  Số bảng số liệu 10 Số hình vẽ 44  Số tài liệu tham khảo 13 Sản phẩm 1  **Một số nhận xét về hình thức cuốn báo cáo:**   * Bố cục nội dung từng chương trình bày khá rõ ràng. * Tuy nhiên, khóa luận vẫn còn một số lỗi chính tả.  1. **Về nội dung nghiên cứu:**  * Nhóm tác giả không những đã vận dụng tốt những kiến thức đã học mà còn nghiên cứu thêm nhiều kiến thức, công nghệ mới chưa được học ở trường. * Nghiên cứu sử dụng **Etcd** vào việc xây dựng kho dữ liệu cho đề tài. * Tìm hiểu và nắm vững một số công nghệ lập trình di động và server để xây dựng hệ thống Server – Client. * Nghiên cứu thuật toán **Apriori** để giải quyết bài toán khuyến nghị sách cho người dùng dựa trên những dữ liệu lấy được. * Xây dựng ứng dụng đọc sách với các tính năng cơ bản, kết hợp với tính năng điều khiển bằng giọng nói đang dần phổ biến trên các thiết bị thông minh. * Tuy nhiên, ứng dụng vẫn tồn tại một số hạn chế như: dữ liệu chưa đủ nhiều và phong phú để áp dụng và kết hợp nhiều thuật toán, đưa ra kết quả tối ưu.  1. **Về chương trình ứng dụng:**  * Hoàn thành các chức năng cơ bản đã đề ra. Áp dụng được những gì đã được học và nghiên cứu lý thuyết vào sản phẩm cuối cùng. * Tuy nhiên, tốc độ gửi trả dữ liệu không ổn định, dựa nhiều vào đường truyền mạng và trạng thái server. Chức năng đọc sách chưa thực sự tốt, chỉ đọc được file pdf, không hỗ trợ thay đổi font chữ.  1. **Về thái độ làm việc của sinh viên:**  * Nhóm sinh viên có kiến thức nền tảng, có kỹ năng khá tốt * Độc lập nghiên cứu, biết nhiều những kiến thức và công nghệ mới.   **Đánh giá chung:** Khóa luận đạt yêu cầu của một khóa luận tốt nghiệp kỹ sư, xếp loại G**iỏi**  **Điểm từng sinh viên:**  **Trương Thị Bích Chi: 9.0/10**  **Lưu Nguyễn: 8.5/10** | | | |
|  | | **Người nhận xét**  **Thái Bảo Trân** | | |

|  |  |
| --- | --- |
| **ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |
|  | *TP. HCM, ngày…..tháng…..năm……..* |

**NHẬN XÉT KHÓA LUẬN TỐT NGHIỆP**

**(CỦA CÁN BỘ PHẢN BIỆN)**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên khóa luận:** | | | |
| **ĐỌC VÀ TƯ VẤN SÁCH ĐIỆN TỬ TRÊN ỨNG DỤNG DI ĐỘNG** | | | |
| **Nhóm SV thực hiện:** | | | **Cán bộ phản biện:** |
| Trương Thị Bích Chi  Lưu Nguyễn | 15520062  15520561 | | ThS. Nguyễn Hồ Duy Tri |
| **Đánh giá Khóa luận**   1. Về cuốn báo cáo:   Số trang 110 Số chương 5  Số bảng số liệu 10 Số hình vẽ 44  Số tài liệu tham khảo 13 Sản phẩm 1  **Một số nhận xét về hình thức cuốn báo cáo:**  Khóa luận được trình bày một cách rõ ràng, mạch lạc, theo đúng định dạng chung. Các chương mục được phân chia hợp lý, các hình ảnh minh họa, bảng biểu, các từ viết tắt được chú thích, trình bày cụ thể.  Tuy nhiên trong báo cáo vẫn còn nhiều lỗi chính tả, một vài nội dung còn nhầm lẫn, chưa đánh dấu rõ ràng các trích dẫn và phần tài liệu tham khảo vẫn chưa được định dạng đúng theo quy định.   1. **Về nội dung nghiên cứu:**   Bằng việc khảo sát một vài trang web và ứng dụng đọc sách hiện có, nhóm tác giả đã rút ra được những hạn chế, từ đó xây dựng thành công một ứng dụng trên nền tảng di động phổ biến đó là Android, với mục đích rèn luyện, khuyến khích thói quen đọc sách cho mọi người.  Ngoài những chức năng cơ bản như hiển thị thông tin sách, hỗ trợ tải và đọc sách, quản lý lịch sử đọc sách, quản lý thông tin cá nhân người dùng, chia sẻ sách với cộng đồng, hỗ trợ đánh giá, bình luận sách, nhóm tác giả còn xây dựng được những chức năng hữu ích khác như xác định thể loại sách do người dùng tải lên một cách tự động, đề xuất sách dựa trên lịch sử đọc.  Để lưu trữ dữ liệu, khóa luận tìm hiểu và cài đặt hệ quản trị cơ sở dữ liệu Etcd. Từ các tiêu đề sách, nhóm tác giả đã tiến hành các bước tiền xử lý nhằm tìm ra mô hình dự đoán thể loại sách. Đầu tiên chuỗi tiêu đề sẽ được tách thành các từ, sau đó, các từ này sẽ được chuyển thành vector dựa vào mô hình túi từ (bag of words), tiếp theo, nhóm sử dụng kĩ thuật TF-IDF để điều chỉnh lại vector có được ở bước trước, cuối cùng, áp dụng thuật toán Naive Bayes, trong đó sử dụng mô hình đa thức để xây dựng mô hình phân lớp thể loại sách. Bên cạnh đó, khóa luận cũng xử lý tương tự để tìm ra mô hình phân lớp cho 06 chức năng của phần mềm, dựa trên 54 mẫu câu mệnh lệnh sẵn có. Trong phần trình bày, nhóm tác giả đã thấy được các hạn chế khi xuất hiện một số từ phổ biến nhưng không mang lại ý nghĩa. Tuy nhiên, nhóm vẫn dựa vào TF-IDF để giảm đi mức độ quan trọng của các từ này trong văn bản, thay vì tìm cách loại bỏ như thường thấy, điều này cũng giúp làm giảm bớt đi kích thước của túi từ cần xét.  Khóa luận còn tìm hiểu và áp dụng thuật toán Apriori trên lịch sử đọc của tất cả người dùng để đưa ra những luật kết hợp thõa mãn ngưỡng độ hỗ trợ và độ tin cậy nhất định, từ đó xác định được sách cần tư vấn cho người dùng. Tuy nhiên, ngưỡng mà nhóm đưa ra, áp dụng trong sản phẩm chưa cao, điều đó dẫn đến hiệu quả của khuyến nghị vẫn còn hạn chế.   1. **Về chương trình ứng dụng:**   Khóa luận đã xây dựng được máy chủ trên nền tảng Google Cloud Platform dùng để lưu trữ thông tin, huyến luyện các mô hình phân lớp, mô hình khuyến nghị sách… bằng ngôn ngữ Python, Java. Nhóm tác giả đã tự lập trình thư viện hỗ trợ, cung cấp các hàm để kết nối và thao tác cơ bản với hệ quản trị Etcd.  Ứng dụng di động có giao diện gọn gàng, bắt mắt, dễ sử dụng, cung cấp đầy đủ các chức năng đã liệt kê. Bên cạnh đó, nhóm còn xây dựng thêm một chức năng rất thú vị là điều khiển bằng giọng nói tiếng Việt. Để lập trình ứng dụng này, nhóm đã sử dụng framework Annotation, thư viện Android PdfViewer để đọc tập tin sách, thư viện MPAndroidChart để vẽ các biểu đồ, thư viện Glide để hiển thị ảnh. Ngoài ra còn có các thư viện hỗ trợ khác như: Retrofit, GSON, Toasty, Fab, Lombok,…  Tuy nhiên, chức năng của ứng dụng còn chưa nhiều, lúc minh họa vẫn còn một vài lỗi, chưa sắp xếp được kết quả tìm kiếm sách bằng tiếng Việt.   1. **Về thái độ làm việc của sinh viên:**   Sinh viên có sự chủ động trong quá trình liên lạc và phản biện với cán bộ. Cả hai tác giả đều tuân thủ giờ giấc, trình bày báo cáo với tinh thần cầu thị, luôn khiêm tốn và biết lắng nghe. Sinh viên cũng thể hiện sự cố gắng trong quá trình chỉnh sửa lại khóa luận theo những góp ý của phản biện.  **Đánh giá chung:** Khóa luận đạt yêu cầu của một khóa luận tốt nghiệp kỹ sư, xếp loại Giỏi  **Điểm từng sinh viên:**  **Trương Thị Bích Chi: 8,5/10**  **Lưu Nguyễn: 8,5/10** | | | |
|  | | **Người nhận xét**  Nguyễn Hồ Duy Tri | | |
|  | |  | | |

**LỜI CẢM ƠN**

Chúng em xin gửi lời cảm ơn chân thành đến cô Thái Bảo Trân, người đã tận tình hướng dẫn và tạo mọi điều kiện thuận lợi có thể để hoàn thành khóa luận này. Cô luôn nhiệt tình trao đổi, và tận tình chỉ bảo giúp chúng em hoàn thành khóa luận với kết quả tốt nhất.

Chúng em xin chân thành biết ơn sự tận tình dạy dỗ và sự giúp đỡ của tất cả quý thầy cô tại trường Đại học Công Nghệ Thông Tin, đặc biệt là các thầy cô trong khoa Hệ Thống Thông Tin, các thầy cô đã tận tình truyền đạt những kiến thức quý giá cho chúng em trong suốt hơn 4 năm học qua, những kiến thức học được từ giảng đường sẽ là hành trang quý báu để chúng em có thể tiếp tục trên con đường học tập, nghiên cứu, làm việc cũng nhưng trong quá trình hoàn thiện bản thân của mình.

Chúng em xin chân thành cảm ơn quý thầy cô!

Xin trân trọng cảm ơn!

|  |  |
| --- | --- |
| **ĐẠI HỌC QUỐC GIA TP. HỒ CHÍ MINH**  **TRƯỜNG ĐẠI HỌC**  **CÔNG NGHỆ THÔNG TIN** | **CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**  **Độc Lập - Tự Do - Hạnh Phúc** |
|  |  |

**ĐỀ CƯƠNG CHI TIẾT**

|  |  |
| --- | --- |
| **TÊN ĐỀ TÀI:**  **Đọc và tư vấn sách điện tử trên ứng dụng di động**  **(Reading and Recommending e-Books on Mobile Application)** | |
| **Cán bộ hướng dẫn:** ThS. Thái Bảo Trân | |
| **Thời gian thực hiện:** Từ 2/2019 đến 6/2019 | |
| **Sinh viên thực hiện:** Trương Thị Bích Chi – 15520062  Lưu Nguyễn - 15520561 | |
| **NỘI DUNG ĐỀ TÀI**   1. **Lý do chọn đề tài**   Sách là nguồn tri thức vô hạn của nhân loại, mang cho người đọc những trải nghiệm, suy ngẫm phong phú trong cuộc sống, tăng vốn sống của bản thân. Đọc sách để nâng cao nhận thức, tiếp thu thông tin về mọi vấn đề cuộc sống là nhu cầu chính đáng của mỗi người. Đọc sách còn là nét đẹp văn hóa trong đời sống xã hội, làm tăng giá trị của bản thân trong mắt mọi người xung quanh. Một thói quen cần thiết nếu muốn phát triển toàn vẹn về nhân cách con người và nhận thức xã hội.  Theo [WorldAtlas](https://www.worldatlas.com/articles/the-countries-that-read-the-most.html), số liệu từ đầu năm 2017, Ấn Độ là quốc gia có số giờ đọc sách mỗi tuần tính trên đầu người cao nhất, 10 tiếng 42 phút. Trong top 5 còn có Thái Lan (9 tiếng 24 phút) và Philippines (7 tiếng 36 phút) lần lượt ở vị trí thứ 2 và 4. Còn ở Việt Nam, theo một [phóng sự của VTV](https://vtv.vn/doi-song/nguoi-viet-tre-it-doc-sach-2018060619510808.htm) vào năm 2018, con số được tổng hợp từ nhiều cơ quan có thẩm quyền là 1 cuốn/năm, nếu quy ra thời gian đọc sách mỗi tuần là vô cùng ít, đây là biểu hiện cho thấy văn hóa đọc tại nước ta chỉ đang dừng lại ở mức phong trào, biển hiệu chứ chưa thật sự đi sâu vào đời sống mỗi người.  **Đọc sách rất tốt nhưng làm sao biết được cuốn sách nào đáng để đọc?**  Chúng ta cần đọc nhiều sách hơn nên cần thay đổi thói quen đọc sách, đặc biệt là từ khi còn nhỏ. Vấn đề cần được giải quyết là làm thế nào để chọn được những cuốn sách đáng để đọc và phù hợp với lứa tuổi, phù hợp sở thích của mỗi người?  Dưới đây là thống kê một số kinh nghiệm từ người đọc giúp bạn có thể đánh giá sách và tìm được những cuốn sách đáng để đọc:   * **Giọng văn cuốn hút:** Nếu bạn cảm thấy bị giọng văn của người viết hấp dẫn ngay từ những trang đầu, cho dù đó là phần dẫn một cuốn tiểu thuyết hay lời mở đầu của một cuốn phi tiểu thuyết, nó có thể khiến bạn cười, thốt lên một lời cảm thán, phải suy nghĩ, thấy tò mò, … thì khả năng cao đó sẽ làm một cuốn đáng đọc với bạn. * **Có định hướng rõ ràng:** Một cuốn sách phi tiểu thuyết có thể nói về một mảng kiến thức, một ý tưởng xuyên suốt hay những hướng dẫn giúp vượt qua khó khăn, nhưng tất cả các chương phải xoay quanh một định hướng rõ ràng và bạn có thể chắt lọc được gì ra từ đó. * **Nhân vật chính hấp dẫn:** Cho dù đó có là một cuốn tiểu thuyết hay phi tiểu thuyết thì ít nhất nhân vật chính cũng nên là một người khiến bạn cảm thấy thú vị và muốn khám phá thêm. Những thứ làm nên sức hấp dẫn của nhân vật chính thường bao gồm suy nghĩ, cách thể hiện suy nghĩ, cách tương tác với người khác cũng như độ độc đáo của họ. * **Nhiều thông tin hữu ích:** Những thứ bạn có thể tiếp thu và áp dụng như kỹ năng sống, nghệ thuật giao tiếp, kỹ năng bán hàng… hay một cái nhìn cận cảnh về nhân vật hay nhóm người nào đó khiến bạn tò mò. Những chi tiết bạn có thể chắt lọc ra khi đọc: từ cuốn sách này, bạn có thể tìm đọc, nghiên cứu thêm về những ý tưởng, lĩnh vực được truyền đạt trong đó. * **Ham muốn đọc tiếp:** Những cuốn sách có thể khiến bạn không thể ngừng đọc và cảm nhận được sự thôi thúc lật tiếp trang sau để biết điều gì sẽ xảy ra. * **Khoảnh khắc khi đọc xong:** Cuốn sách có thể khiến người đọc thở dài, muốn viết, muốn nghĩ về gì đó hay nói lại với ai đó về nó, … ngay sau khi đọc xong chắc chắn không phải một cuốn sách tầm thường, thậm chí trong nhiều trường hợp có thể là cuốn sách thay đổi cuộc đời bạn.   **Vậy bạn nên chọn sách từ đâu?**   * Có một số nguồn gợi ý uy tín như [Goodreads](https://www.goodreads.com/) (bạn có thể xem bạn bè đang đọc gì, đặt mục tiêu đọc bao nhiều cuốn trong năm nay hay tham khảo gợi ý sách từ những cá nhân nổi bật), danh sách Best-seller của New York Times (cập nhật hàng tuần) hay các bảng xếp hạng sách hay nhất năm có thể tìm được qua một lượt tìm kiếm Google. * Bạn bè, người quen hay những người thành công, có sức ảnh hưởng lớn đến cộng đồng hoặc với bản thân đều có thể là nguồn sách giá trị. Bạn có thể tham khảo blog của họ, các blog sách nổi tiếng. * Tìm sách theo tác giả: Chắc hẳn những ai yêu sách sẽ dễ dàng liệt kê được những tác giả yêu thích của mình. Khi đã yêu thích lối viết của ai đó ở một hai cuốn sách, bạn sẽ thấy thích hầu hết các tác phẩm khác. Điển hình là Alexandre Dumas với những tác phẩm lấy bối cảnh nước Pháp kinh điển, có giả trị xuyên thời gian hay Azit Nexin, người đã tạo ra những câu chuyện cười khiến nhiều thế hệ phải suy ngẫm, gật gù đồng ý. * Chọn các cuốn theo lĩnh vực quan tâm: Nếu bạn làm kinh doanh nhưng vẫn thích tìm hiểu tâm lý học hay khoa học đời sống thì hãy tham khảo những danh sách nổi tiếng nhất về các lĩnh vực này trước tiên. * Chọn các tác phẩm kinh điển: Lý do những cuốn này được xếp vào hàng kinh điển cũng bởi chúng chứa đựng những tư tưởng lớn lao có thể áp dụng vào cuộc sống cho đến tận ngày nay. Bạn rất có thể sẽ cảm thấy sửng sốt khi nhận ra cho dù mọi thứ có thay đổi chóng mặt qua nhiều thập kỷ, thế kỷ thì những vấn đề con người gặp phải, cách họ ứng phó với nghịch cảnh hay cách họ đối xử với nhau vẫn không hề đổi thay quá nhiều.  1. **Mục tiêu đề tài**  * Như đã nêu, văn hoá đọc sách của người Việt chưa được phát triển rộng rãi, ảnh hưởng không nhỏ đến sự phát triển con người, nguyên liệu quan trọng để phát triển đất nước. Lí do chủ quan để giải thích cho tình trạng này, trẻ em không có thói quen đọc sách từ nhỏ, người lớn không có thời gian để kiếm tìm cuốn sách để đọc. * Mặc dù đã có rất nhiều nơi tóm tắt hay nhận xét nội dung cũng như cho đọc thử một phần, nhưng việc đó chưa đủ thông tin để tư vấn tốt cho người đọc có nên mua quyển sách đó để đọc hay không? Ứng dụng sẽ giúp người dùng tìm được cuốn sách hợp lý, phù hợp với sở thích và lịch sử của độc giả. * Để góp phần thay đổi văn hoá đọc của người Việt, nhóm hướng tới một ứng dụng di động đọc sách điện tử và tư vấn chọn sách phù hợp trên di động. Ứng dụng giúp người đọc có thể chọn được sách phù hợp để đọc, quản lý thời gian đọc, có thể đọc sách mọi lúc mọi nơi mà không tốn phí hoặc chi phí đầu tư không đáng kể. Khi chúng ta đọc sách vì thích và đam mê thì dần dần thói quen đọc sách sẽ được hình thành. Bên cạnh đó, ứng dụng sẽ tạo một môi trường trao đổi, quyên góp sách điện tử miễn phí cho cộng đồng người dùng, cũng như một số chức năng hữu ích cho việc chọn sách.  1. **Phạm vi đề tài**   Đề tài tập trung vào các nội dung chính sau:   * Xây dựng ứng dụng di động đọc sách điện tử. * Ứng dụng bao gồm chức năng: đọc sách điện tử; tìm kiếm sách; trao đổi sách; đánh giá sách; đề xuất người dùng, ... * Xây dựng các biểu mẫu báo cáo thống kê sách.  1. **Đối tượng**  * Các tựa sách có trong dữ liệu và quyên góp từ người dùng. * Người dùng  1. **Phương pháp thực hiện**  * Tìm hiểu các công nghệ áp dụng để xây dựng các chức năng đã nêu. * Xây dựng cơ sở dữ liệu, luồng dữ liệu. * Lên luồng sự kiện, các màn hình cần có. * Xây dựng liên kết giữa ứng dụng và máy chủ. * Hiện thực hoá ứng dụng theo thiết kế đã có. * Nghiên cứu một số thuật toán khai phá dữ liệu như lọc cộng tác, thuật toán xử lý văn bản, …tư vấn sách và đề xuất những quyển sách phù hợp nhất cho người đọc.  1. **Kết quả mong đợi của đề tài**   Xây dựng kho sách điện tử và tư vấn tìm kiếm sách phù hợp trên di động có khả năng đưa vào thực tiễn. | |
| **KẾ HOẠCH THỰC HIỆN** | |
| **Công việc cần thực hiện** | **Thời gian** |
| * Xác định đề tài thực hiện. * Viết mô tả cho đề tài. | 1 tuần |
| Tìm hiểu các công nghệ liên quan áp dụng cho đề tài. | 1 tuần |
| * Xây dựng cơ sở dữ liệu, luồng dữ liệu. * Lên luồng sự kiện, các màn hình cần có. * Xây dựng liên kết giữa ứng dụng và máy chủ. | 1 tuần |
| Trực quan hóa các biểu mẫu vào phần mềm dựa trên dữ liệu đã có. | 2 tuần |
| Xây dựng các chức năng cơ bản của ứng dụng đọc sách online. | 4 tuần |
| Xây dựng hệ thống đề xuất cho người dùng | 4 tuần |
| Viết báo cáo hoàn chỉnh cho đề tài. | 2 tuần |
| **Tổng cộng** | 15 tuần |
| **Xác nhận của CBHD**  **(Ký tên và ghi rõ họ tên)** | TPHCM, ngày 10 tháng 3 năm 2019  Sinh viên  (Ký tên và ghi rõ họ tên) |

MỤC LỤC

[DANH MỤC CÁC BẢNG i](#_Toc13329474)

[DANH MỤC CÁC HÌNH ii](#_Toc13329475)

[DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT iv](#_Toc13329476)

[TÓM TẮT v](#_Toc13329477)

[CHƯƠNG 1: TỔNG QUAN 1](#_Toc13329478)

[1.1. Đặt vấn đề 1](#_Toc13329479)

[1.2. Mục tiêu đề tài 1](#_Toc13329480)

[1.3. Khảo sát hiện trạng 2](#_Toc13329481)

[1.3.1. Khảo sát ứng dụng đọc sách điện tử 2](#_Toc13329482)

[1.3.2. Các nghiên cứu liên quan 5](#_Toc13329483)

[1.4. Đối tượng, phạm vi nghiên cứu 5](#_Toc13329484)

[1.4.1. Đối tượng 5](#_Toc13329485)

[1.4.2. Phạm vi nghiên cứu 5](#_Toc13329486)

[1.5. Nội dung thực hiện 6](#_Toc13329487)

[1.6. Bố cục báo cáo 7](#_Toc13329488)

[CHƯƠNG 2: CƠ SỞ LÝ THUYẾT 8](#_Toc13329489)

[2.1. Tổng quan về cơ sở dữ liệu Etcd 8](#_Toc13329490)

[2.1.1. Giới thiệu về Etcd 8](#_Toc13329491)

[2.1.2. Cấu trúc lưu trữ dữ liệu 8](#_Toc13329492)

[2.2. Phương thức hoạt động của hệ phân tán Etcd 9](#_Toc13329493)

[2.3. Các chức năng Etcd hỗ trợ 10](#_Toc13329494)

[2.4. Naive Bayes 12](#_Toc13329495)

[2.4.1. Túi từ (Bag of Words - Bow), TF-IDF 12](#_Toc13329496)

[2.4.2. Định lý Bayes 15](#_Toc13329497)

[2.4.3. Thuật toán phân cấp Naive Bayes 15](#_Toc13329498)

[2.4.4. Các mô hình thuật toán phân cấp Naive Bayes 16](#_Toc13329499)

[2.4.4.1. Tổng quan mô hình đa thức 16](#_Toc13329500)

[2.4.4.2. Tổng quan mô hình Bernoulli 17](#_Toc13329501)

[2.5. Tập phổ biến và luật kết hợp với thuật toán Apriori 17](#_Toc13329502)

[2.6. Các công cụ hỗ trợ hệ thống 19](#_Toc13329503)

[2.6.1. Python 19](#_Toc13329504)

[2.6.1.1. Giới thiệu 19](#_Toc13329505)

[2.6.1.2. Đặc điểm 19](#_Toc13329506)

[2.6.1.3. Ưu điểm 20](#_Toc13329507)

[2.6.1.4. Nhược điểm 20](#_Toc13329508)

[2.6.2. Java 20](#_Toc13329509)

[2.6.2.1. Giới thiệu 20](#_Toc13329510)

[2.6.2.2. Đặc điểm 21](#_Toc13329511)

[2.6.2.3. Ưu điểm 22](#_Toc13329512)

[2.6.2.4. Nhược điểm 22](#_Toc13329513)

[CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG 23](#_Toc13329514)

[3.1. Phân tích hiện trạng nhu cầu 23](#_Toc13329515)

[3.1.1. Yêu cầu chức năng 23](#_Toc13329516)

[3.1.2. Yêu cầu phi chức năng 23](#_Toc13329517)

[3.2. Phân tích và thiết kế thành phần dữ liệu 24](#_Toc13329518)

[3.2.1. Cấu trúc dữ liệu 24](#_Toc13329519)

[3.2.2. Mô tả các thuộc tính 25](#_Toc13329520)

[3.3. Phân tích thiết kế thành phần xử lý 30](#_Toc13329521)

[3.3.1. Sơ đồ Usecase 30](#_Toc13329522)

[3.3.2. Danh sách Usecase 30](#_Toc13329523)

[3.3.3. Sơ đồ tuần tự 32](#_Toc13329524)

[3.4. Thiết kế giao diện 45](#_Toc13329525)

[3.5. Xây dựng giao diện 46](#_Toc13329526)

[CHƯƠNG 4: CÀI ĐẶT VÀ TRIỂN KHAI HỆ THỐNG 50](#_Toc13329527)

[4.1. Môi trường triển khai 50](#_Toc13329528)

[4.2. Giải pháp công nghệ sử dụng ở phía Client (ứng dụng Android) 53](#_Toc13329529)

[4.2.1. Áp dụng Annotation để giảm thời gian lập trình 53](#_Toc13329530)

[4.2.2. Sử dụng thư viện Android PdfViewer để đọc và hiển thị file pdf từ bộ nhớ điện thoại 54](#_Toc13329531)

[4.2.3. Vẽ biểu đồ với MPAndroidChart 55](#_Toc13329532)

[4.2.4. Việc tải và sử dụng hình ảnh 55](#_Toc13329533)

[4.2.5. Giao tiếp với Server 58](#_Toc13329534)

[4.2.6. Xử lý kết quả trả về từ Server 62](#_Toc13329535)

[4.2.7. Xây dựng giao diện 62](#_Toc13329536)

[4.3. Giải pháp công nghệ sử dụng ở phía Server 62](#_Toc13329537)

[4.3.1. Ngôn ngữ Python 62](#_Toc13329538)

[4.3.1.1. Tổng quan 62](#_Toc13329539)

[4.3.1.2. Lý do sử dụng 63](#_Toc13329540)

[4.3.2. Nghiên cứu và triển khai thư viện hỗ trợ truy xuất dữ liệu trên cơ sở dữ liệu Etcd 63](#_Toc13329541)

[4.3.3. Ứng dụng thuật toán phân lớp Naive Bayes 82](#_Toc13329542)

[4.3.3.1. Chức năng phân lớp thể loại sách dựa vào tiêu đề 82](#_Toc13329543)

[4.3.3.2. Chức năng điều khiển ứng dụng bằng giọng nói 86](#_Toc13329544)

[4.3.4. Ứng dụng thuật toán Apriori trong việc tư vấn sách cho từng người dùng 90](#_Toc13329545)

[4.3.5. Dùng Google Cloud Platform để triển khai Server 94](#_Toc13329546)

[CHƯƠNG 5: KẾT LUẬN 95](#_Toc13329547)

[5.1. Kết quả đạt được 95](#_Toc13329548)

[5.2. Ưu điểm của hệ thống 95](#_Toc13329549)

[5.3. Hạn chế của hệ thống 95](#_Toc13329550)

[5.4. Hướng phát triển 96](#_Toc13329551)

[TÀI LIỆU THAM KHẢO 97](#_Toc13329552)

DANH MỤC CÁC BẢNG

[Bảng 3‑1. Bảng mô tả thuộc tính User 26](#_Toc13208352)

[Bảng 3‑2. Bảng mô tả thuộc tính category 26](#_Toc13208353)

[Bảng 3‑3. Bảng mô tả thuộc tính Book 27](#_Toc13208354)

[Bảng 3‑4. Bảng mô tả thuộc tính Reading history 28](#_Toc13208355)

[Bảng 3‑5. Bảng mô tả thuộc tính News 29](#_Toc13208356)

[Bảng 3‑6. Bảng mô tả thuộc tính Comment 29](#_Toc13208357)

[Bảng 3‑7. Danh sách usecase 33](#_Toc13208358)

[Bảng 3‑8. Danh sách màn hình 46](#_Toc13208359)

[Bảng 4‑1. So sánh Glide và Picasso 59](#_Toc13208360)

[Bảng 4‑2. So sánh Retrofit và Volley 62](#_Toc13208361)

DANH MỤC CÁC HÌNH

[Hình 1‑1. App waka 3](#_Toc13244847)

[Hình 1‑2. App waka 3](#_Toc13244848)

[Hình 1‑3. Web sachtot 4](#_Toc13244849)

[Hình 1‑4. Web gacsach 4](#_Toc13244850)

[Hình 2‑1. Phương thức hoạt động của hệ phân tán Etcd 9](#_Toc13244851)

[Hình 3‑1. Sơ đồ usecase 30](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244852)

[Hình 3‑2. Sơ đồ tuần tự usecase đăng kí 32](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244853)

[Hình 3‑3. Sơ đồ tuần tự usecase đăng nhập 33](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244854)

[Hình 3‑4. Sơ đồ tuần tự usecase đăng xuất 33](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244855)

[Hình 3‑5. Sơ đồ tuần tự usecase tra cứu người dùng 34](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244856)

[Hình 3‑6. Sơ đồ tuần tự usecase khởi tạo màn hình trang chủ 34](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244857)

[Hình 3‑7. Sơ đồ tuần tự usecase tra cứu sách 35](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244858)

[Hình 3‑8. Sơ đồ tuần tự usecase xem bảng tin chung 36](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244859)

[Hình 3‑9. Sơ đồ tuần tự usecase thống kê hệ thống 36](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244860)

[Hình 3‑10. Sơ đồ tuần tự usecase quản lý thông tin người dùng 37](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244861)

[Hình 3‑11. Sơ đồ tuần tự usecase quản lý lịch sử đọc sách 38](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244862)

[Hình 3‑12. Sơ đồ tuần tự usecase thêm sách – bài đăng 39](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244863)

[Hình 3‑13. Sơ đồ tuần tự usecase xem sách – bài đăng 40](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244864)

[Hình 3‑14. Sơ đồ tuần tự usecase chỉnh sửa sách – bài đăng 40](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244865)

[Hình 3‑15. Sơ đồ tuần tự usecase xóa sách – bài đăng 41](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244866)

[Hình 3‑16. Sơ đồ tuần tự usecase xem chi tiết sách 41](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244867)

[Hình 3‑17. Sơ đồ tuần tự usecase bình luận sách 42](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244868)

[Hình 3‑18. Sơ đồ tuần tự usecase đánh giá sách 42](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244869)

[Hình 3‑19. Sơ đồ tuần tự usecase đọc sách 43](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244870)

[Hình 3‑20. Sơ đồ usecase điều khiển bằng giọng nói 43](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244871)

[Hình 3‑21. Sơ đồ usecase lập lịch nhắc nhở 44](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244872)

[Hình 3‑22. Màn hình đăng nhập 46](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244873)

[Hình 3‑23. Màn hình đăng kí 46](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244874)

[Hình 3‑24. Màn hình trang chủ 46](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244875)

[Hình 3‑25. Màn hình quản lý thông tin 46](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244876)

[Hình 3‑26. Màn hình lịch sử đọc sách 47](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244877)

[Hình 3‑27. Màn hình chi tiết sách 47](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244878)

[Hình 3‑28. Màn hình bảng tin chia sẻ 47](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244879)

[Hình 3‑29. Màn hình tra cứu sách 47](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244880)

[Hình 3‑30.Màn hình thêm sách 48](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244881)

[Hình 3‑31. Màn hình tra cứu người dùng 48](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244882)

[Hình 3‑32. Màn hình trang cá nhân 48](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244883)

[Hình 3‑33. Màn hình thống kê 48](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244884)

[Hình 3‑34. Màn hình danh sách nhắc nhở 49](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244885)

[Hình 3‑35. Màn hình đặt lịch nhắc nhở 49](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244886)

[Hình 4‑1. Sơ đồ thị trường hệ điều hành điện thoại toàn thế giới từ tháng 5 2018 đến tháng 6 2019 50](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244887)

[Hình 4‑2. Sơ đồ thư viện hỗ trợ truy xuất dữ liệu trên cơ sở dữ liệu Etcd 63](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244888)

[Hình 4‑3. Màn hình lịch sử đọc sách 93](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244889)

[Hình 4‑4. Màn hình chính với danh sách sách gợi ý 94](file:///C:\Users\luung\Downloads\Microsoft.SkypeApp_kzf8qxf38zg5c!App\All\BÁO-CÁO-KLTN-ĐỌC-VÀ-TƯ-VẤN-SÁCH-ĐIỆN-TỬ-15520062-15520561Version-phản-biện%20(2).docx#_Toc13244890)

# DANH MỤC THUẬT NGỮ VÀ TỪ VIẾT TẮT

|  |  |  |
| --- | --- | --- |
| **STT** | **Từ viết tắt** | **Nội dung** |
| 1 | Etcd | Etc distributed |
| 2 | BOW | Bag Of Words |
| 3 | TF | Term Frequency |

TÓM TẮT

Đề tài khóa luận “Đọc và tư vấn sách điện tử trên ứng dụng di động” mang tính thực tế hướng đến đối tượng người dùng mong muốn có một nơi để chia sẻ, nhận được những lời khuyên hữu ích về sách cũng như đọc sách ngay trên thiết bị di động của mình, vật bất ly thân với đa số mọi người hiện nay.

Sau khi tìm hiểu, khảo sát các ứng dụng đọc sách hiện nay, nhóm tác giả đưa ra hướng giải quyết sau:

* Tìm hiểu và xây dựng phương pháp khuyến nghị dựa vào lịch sử đọc, nhận xét của người dùng cho từng cuốn sách cũng như thể loại.
* Xây dựng ứng dụng với chức năng đọc sách với giao diện thân thiện, dễ sử dụng. Ngoài ra, ứng dụng còn cung cấp các chức năng khác như: đăng nhập, đăng xuất, tìm kiếm, đánh giá sách, chia sẻ sách.
* Công nghệ sử dụng xây dựng hệ thống: Java Android, Python, Etcd, Google Cloud Platform.

# CHƯƠNG 1: TỔNG QUAN

## Đặt vấn đề

Sách là nguồn tri thức vô hạn của nhân loại, mang cho người đọc những trải nghiệm, suy ngẫm phong phú trong cuộc sống, tăng vốn sống của bản thân. Đọc sách để nâng cao nhận thức, tiếp thu thông tin về mọi vấn đề cuộc sống là nhu cầu chính đáng của mỗi người. Đọc sách còn là nét đẹp văn hóa trong đời sống xã hội, làm tăng giá trị của bản thân trong mắt mọi người xung quanh. Một thói quen cần thiết nếu muốn phát triển toàn vẹn về nhân cách con người và nhận thức xã hội.

Theo [WorldAtlas](https://www.worldatlas.com/articles/the-countries-that-read-the-most.html), số liệu từ đầu năm 2017, Ấn Độ là quốc gia có số giờ đọc sách mỗi tuần tính trên đầu người cao nhất, 10 tiếng 42 phút. Trong top 5 còn có Thái Lan (9 tiếng 24 phút) và Philippines (7 tiếng 36 phút) lần lượt ở vị trí thứ 2 và 4. Còn ở Việt Nam, theo một [phóng sự của VTV](https://vtv.vn/doi-song/nguoi-viet-tre-it-doc-sach-2018060619510808.htm) vào năm 2018, con số được tổng hợp từ nhiều cơ quan có thẩm quyền là 1 cuốn/năm, nếu quy ra thời gian đọc sách mỗi tuần là vô cùng ít, đây là biểu hiện cho thấy văn hóa đọc tại nước ta chỉ đang dừng lại ở mức phong trào, biển hiệu chứ chưa thật sự đi sau vào đời sống mỗi người.

Với sự phát triển của công nghệ thông tin, việc đọc sách sẽ dễ dàng hơn cho mọi người vì sách sẽ dễ dàng tìm kiếm hơn nhưng lại tạo ra một rào cản rất lớn cho những ai chưa có thói quen đọc sách.

Nhằm hỗ trợ tạo ra một thói quen đọc sách, đặc biệt dành cho người Việt, nhóm tác giả cố gắng tạo ra một phương pháp giúp người dùng nhận được những cuốn sách phù hợp, tạo ra sự thích thú cũng như hoàn thiện thói quen đọc sách. Bên cạnh đó tạo ra một cộng đồng chia sẻ, đánh giá sách đáng tin cậy.

## Mục tiêu đề tài

Mặc dù đã có rất nhiều nơi tóm tắt hay nhận xét nội dung cũng như cho đọc thử một phần, nhưng việc đó chưa đủ thông tin để tư vấn tốt cho người đọc có nên mua quyển sách đó để đọc hay không? Ứng dụng sẽ giúp người dùng tìm được cuốn sách hợp lý, phù hợp với sở thích và lịch sử của độc giả.

Để góp phần thay đổi văn hoá đọc của người Việt, nhóm hướng tới một ứng dụng di động đọc sách điện tử và tư vấn chọn sách phù hợp trên di động. Ứng dụng giúp người đọc có thể chọn được sách phù hợp để đọc, quản lý thời gian đọc, có thể đọc sách mọi lúc mọi nơi mà không tốn phí hoặc chi phí đầu tư không đáng kể. Khi chúng ta đọc sách vì thích và đam mê thì dần dần thói quen đọc sách sẽ được hình thành. Bên cạnh đó, ứng dụng sẽ tạo một môi trường trao đổi, quyên góp sách điện tử miễn phí cho cộng đồng người dùng, cũng như một số chức năng hữu ích cho việc chọn sách.

## Khảo sát hiện trạng

### Khảo sát ứng dụng đọc sách điện tử

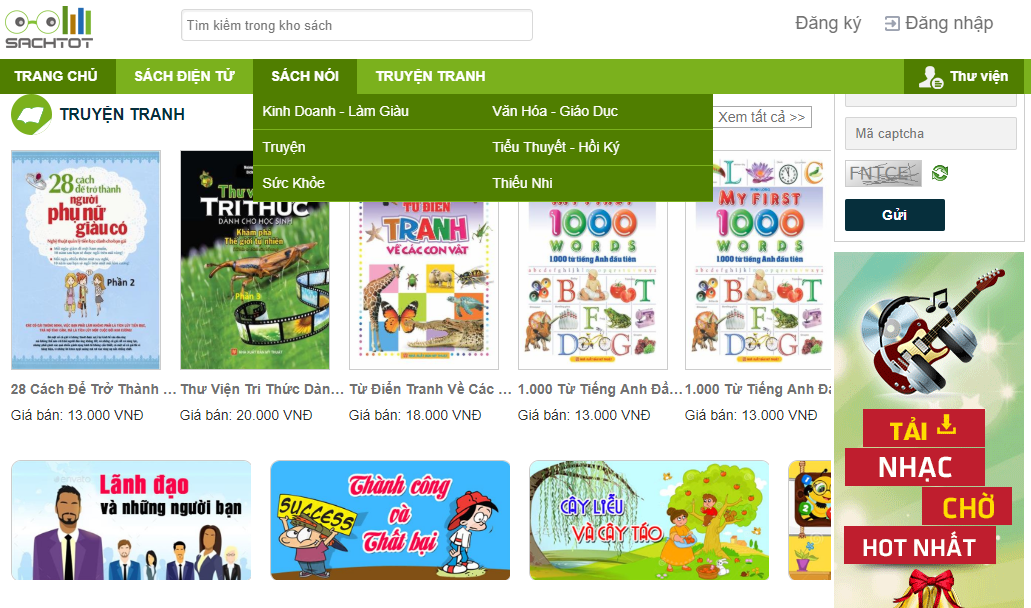
* Ứng dụng Waka 4.0 – Ebook & Audiobook

Waka[[1]](#footnote-2) là Nền tảng xuất bản điện tử số một Việt Nam, trực thuộc Công ty Cổ phần Bạch Minh (Vega Corporation). Ra đời từ năm 2014, đến nay, Waka có hơn 3,2 triệu người dùng, hơn 13.000 nội dung điện tử đa dạng về thể loại và hỗ trợ trên nhiều nền tảng khác nhau (máy tính, máy tính bảng, di động). Với một giao diện cực kì đơn giản cho mọi người và khả năng khai thác tốt nguồn sách đang sở hữu, Waka đã cung cấp một nơi tuyệt vời đối với những người yêu thích đọc sách tại Việt Nam.

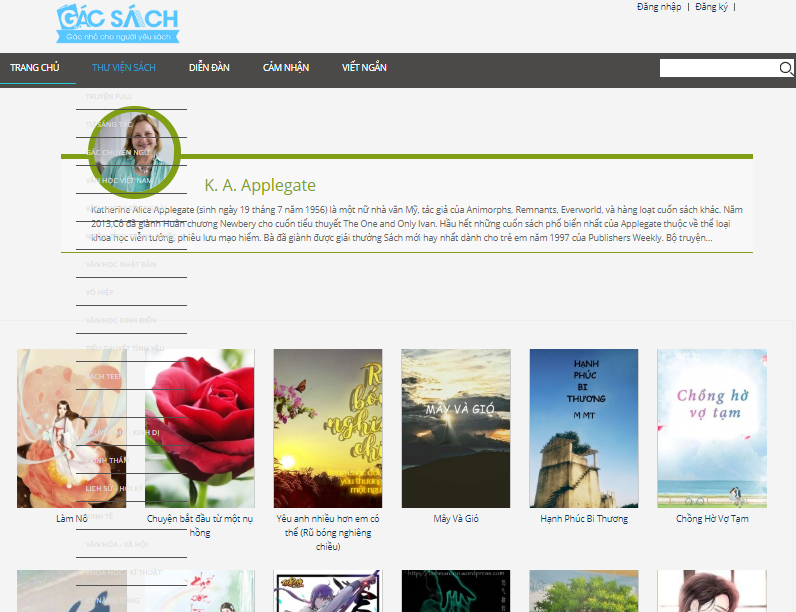
|  |  |
| --- | --- |
| Hình 1‑1. App waka | Hình 1‑2. App waka |

* Trang web Sachtot.vn

Sách tốt[[2]](#footnote-3) là một trong những hệ thống gồm: sách điện tử, sách nói, truyện tranh có bản quyền lớn nhất dành cho mọi loại khách hàng. Với số lượng lên đến vài chục ngàn tựa sách được cung cấp từ nhà xuất bản, các công ty uy tín tại Việt Nam. Ngoài ra, Sachtot.vn còn có ứng dụng trên Apple Store hoặc Google Play, đọc giả có thể đọc sách bằng điện thoại một cách cực kì thuận tiện.



Hình 1‑3. Web sachtot

* Trang web gacsach.vn

Hình 1‑4. Web gacsach

Gác sách[[3]](#footnote-4) có lượt truy cập từ các độc giả khá cao, khoảng 150.000 lượt mỗi ngày. Hệ thống không chỉ có các loại sách đủ thể loại do những tác giả nổi tiếng viết mà còn có nhiều loại truyện tự sáng tác phù hợp với nhiều lứa tuổi.

* Nhận xét: Nhìn chung ta thấy các ứng dụng đọc sách online hiện nay đều có chức năng chính là giúp người dùng đọc sách và tư vấn sách. Tuy nhiên việc chọn lựa sách đọc vẫn tùy thuộc vào người dùng khi tham khảo hệ thống phân loại cơ bản như sách mới nhất, sách đọc nhiều nhất … của ứng dụng, với dữ liệu được thu thập nhiều năm thì chúng vẫn đang đóng vai trò chính trong việc giới thiệu sách cho người dùng bên cạnh hệ thống đánh giá và bình luận. Đối với người mới hoặc những người dùng khó tính việc dựa trên hệ thống gợi ý này có thể gây phản tác dụng khi họ không cảm thấy thích thú với những gì đa số mọi người đọc.

### Các nghiên cứu liên quan

* Thuật toán phân lớp.
* Thuật toán khuyến nghị.

## Đối tượng, phạm vi nghiên cứu

### Đối tượng

Mọi đối tượng có dùng điện thoại thông minh chạy hệ điều hành Android – đặc biệt ứng dụng quan tâm đến những người chưa có thói quen đọc sách.

### Phạm vi nghiên cứu

Về công nghệ:

* Ngôn ngữ lập trình: Java, Python.
* Thư viện: Weka, Retrofit, RxJava, scikit learn
* Framework: Flask Framework.
* Hệ quản trị cơ sở dữ liệu: Etcd.
* Quản lý source code với Git.

Về nghiên cứu:

* Viết thư viện hỗ trợ truy xuất dữ liệu trên Etcd database cluster
* Thuật toán phân lớp
* Thuật toán khuyến nghị

## Nội dung thực hiện

**Bước 1:**

* Khảo sát các ứng dụng đọc sách.
* Xác định những chức năng cần có và thứ tự ưu tiên.
* Tìm hiểu các công nghệ, thư viện hỗ trợ việc lập trình ứng dụng.
* Tìm kiếm và nghiên cứu nguồn dữ liệu.

**Bước 2:**

* Thiết kế các thành phần hệ thống.
* Thiết kế cơ sở dữ liệu.
* Thiết kế server.
* Thiết kế giao diện.
* Thực hiện lấy dữ liệu.

**Bước 3:**

* Xây dựng server.
* Tạo API tương tác với server.
* Tạo giao diện.
* Tạo các chức năng đã chọn.

**Bước 4:**

* Xây dựng hệ thống khuyến nghị từ thuật toán tìm hiểu.

**Bước 5:**

* Hoàn chỉnh và cải thiện hiệu năng hệ thống.
* Nhận xét và đánh giá

## Bố cục báo cáo

#### *Chương 1*: Tổng quan về đề tài

Giới thiệu tổng quan nhất về nội dung đề tài. Nội dung chương 1 bao gồm: Đặt vấn đề và lý do chọn đề tài, khảo sát hiện trạng, mục tiêu và phạm vi nghiên cứu, công cụ xây dựng hệ thống và bố cục của báo cáo.

#### *Chương 2*: Cơ sở lý thuyết

Giới thiệu lý thuyết, các thư viện, công cụ sử dụng trong đề tài. Trình bày ưu nhược điểm của từng mục và lí do lựa chọn. Trình bày chi tiết lý thuyết áp dụng trong đề tài.

#### *Chương 3*: Phân tích và thiết kế hệ thống

Trình bày sơ đồ tổ chức của ứng dụng, phân tích thiết kế hệ thống, thiết kế cơ sở dữ liệu, thiết kế giao diện và luồng chạy của ứng dụng.

#### *Chương 4*: Cài đặt và triển khai hệ thống

Trình bày những công cụ đã sử dụng để hiện thực sản phẩm của đề tài kèm ví dụ. Cách áp dụng các thuật toán phân lớp, khai thác dữ liệu vào bài toán đề xuất sách cho người dùng.

#### *Chương 5*: Kết luận

Tóm tắt đề tài, thành công và hạn chế, hướng phát triển tương lai về mặt chức năng và khả năng đề xuất cho người dùng.

# CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

## Tổng quan về cơ sở dữ liệu Etcd[[4]](#footnote-5)

## Giới thiệu về Etcd

Được phát triển bởi nhóm CoreOS, là một hệ thống phân tán, mã nguồn mở, dùng để lưu trữ các giá trị key-value. Các ứng dụng có độ phức tạp cao, từ một ứng dụng web đơn giản đến Kubernetes, có thể đọc dữ liệu từ và ghi dữ liệu vào etcd.

### Cấu trúc lưu trữ dữ liệu

Cùng là NoSql giống MongoDB nhưng cấu trúc lưu trữ dữ liệu của Etcd hoàn toàn khác biệt.

Cụ thể, MongoDB được thiết kế theo kiểu hướng đối tượng, các bảng trong MongoDB được cấu trúc rất linh hoạt, lưu trữ dữ liệu theo hướng tài liệu (document), các dữ liệu được lưu trữ trong document kiểu json.

Etcd chỉ lưu trữ dữ liệu theo cặp giá trị key-value, các thuộc tính của một đối tượng sẽ được làm phẳng mà không phân cấp bậc.

Ví dụ: MongoDB có đối tượng User có cấu trúc như sau:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| { |  |  |  |  |  |
|  | id: | "12345" |  |  |  |
|  | usernam: | "Nguyễn Văn A" | |  |  |
|  | contact: { |  |  |  |  |
|  |  | phone: | "0935734123", | |  |
|  |  | email: | "nguyenvana@gmail.com", | | |
|  | }, |  |  |  |  |
|  | access: { |  |  |  |  |
|  |  | level: | 5, |  |  |
|  |  | group: | dev, |  |  |
|  | } |  |  |  |  |
| } |  |  |  |  |  |

Khi muốn lưu trữ User ở Etcd thì các key sẽ được làm phẳng theo dạng:

/user/id: "12345",

/user/username: "Nguyen Van A",

/user/contact/phone: "0935734123",

/user/contact/email: "nguyenvana@gmail.com",

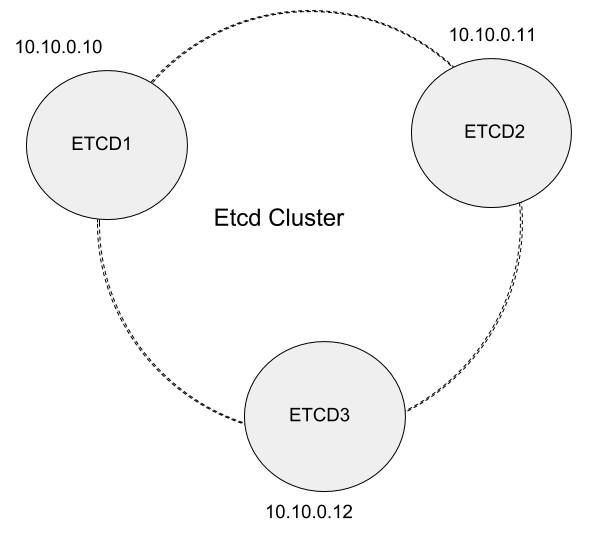
/user/access/level: 5,

/user/access/group: dev

Tuy nhiên, trong database thì tất cả các cặp key-value của đối tượng User sẽ không được lưu trữ chung với nhau, nếu muốn truy xuất giá trị của thuộc tính thì phải dựa vào key đã được làm phẳng để lấy value.

Do đó, nhóm đã nghiên cứu và viết thư viện để hỗ trợ lưu trữ và truy xuất thông tin cho những đối tượng cần sử dụng trong ứng dụng. Chi tiết của thư viện này sẽ được trình bày trong chương 4.

### Phương thức hoạt động của hệ phân tán Etcd



Hình 2‑1. Phương thức hoạt động của hệ phân tán Etcd[[5]](#footnote-6)

Hệ thống phân tán Etcd được hoạt động dựa trên cụm Etcd (Etcd cluster), mỗi cụm sẽ gồm nhiều node.

Etcd cluster sẽ bao gồm ít nhất 3 node (đặt tại 3 máy khác nhau). Trong đó có 1 node sẽ đóng vai trò “Leader”, các node còn lại sẽ là “Follower”, trong cùng 1 thời điểm thì chỉ có duy nhất 1 leader.

Etcd phân tán dữ liệu theo kỹ thuật replicate (nhân bản), do đó tất cả dữ liệu sẽ lưu trữ như nhau tại tất cả các node (hiện nay trong nhiều bài toán phân tán dữ liệu thì kỹ thuật sharding (phân mảnh) được sử dụng nhiều hơn, tại mỗi node sẽ lưu trữ một phần dữ liệu, gộp hết tất cả các mảnh tại các node lại thì được bộ dữ liệu hoàn chỉnh).

Quy trình đọc, ghi dữ liệu như sau: Khi người dùng gửi request (yêu cầu) đọc hoặc ghi dữ liệu đến etcd cluster (có thể request tại bất kì node nào) thì request sẽ được chuyển đến cho leader, sau đó leader sẽ gửi thông báo đến tất cả các node đang hoạt động trong cụm. Khi tất cả các cụm respone (phản hồi) đồng ý tiếp nhận dữ liệu thì node leader sẽ thực thi ghi/đọc dữ liệu. Trường hợp là ghi dữ liệu thì sau khi leader ghi dữ liệu thành công sẽ gửi thông báo đến các node còn lại và các node này sẽ ghi dữ liệu đó xuống.

Nếu trong quá trình ghi dữ liệu nếu có bất kì node nào bị ngưng kết nối thì dữ liệu mới sẽ không request được tới node đó. Sau khi node đó kết nối lại thì tất cả các dữ liệu mới nhất sẽ được request để ghi vào node đó. Do đó trong cùng thời điểm (có thể cách biệt nhau 1 khoảng thời gian nhỏ) thì toàn bộ dữ liệu tại các node sẽ là như nhau.

Khi node leader bị ngắt kết nối thì tất cả các node còn lại sẽ bầu chọn leader mới.

### Các chức năng Etcd hỗ trợ[[6]](#footnote-7)

* ***Get:*** lấy value dựa vào key

VD: key = “foo”, value = “bar”

**$ etcdctl get foo**

**foo**

**bar**

* ***Get\_prefix:*** lấy value có key bắt đầu bằng những kí tự nào đó:

**$ etcdctl get --prefix foo**

**foo**

**bar**

**foo1**

**bar1**

**foo2**

**bar2**

**foo3**

**bar3**

* ***Put:*** Ghi key-value xuống database

VD: key = “foo”, value = “bar”

**$ etcdctl put foo bar**

**OK**

* ***Set:*** Cập nhật value của key

**$ etcdctl set foo bar11**

**OK**

* ***Delete:*** Xóa cặp key-value khi biết key

Giả sử trong etcd đã lưu các key-value như sau:

foo = bar

foo1 = bar1

foo3 = bar3

zoo = val

zoo1 = val1

zoo2 = val2

a = 123

b =456

c = 789

Xóa key “foo” (kết quả 1 dòng được xóa)

**$ etcdctl del foo**

**1**

* ***Delete\_prefix:*** Xóa những key bắt đầu bằng những kí tự nào đó

**$ etcdctl del foo foo9**

**2 # two keys are deleted**

* ***Watch:*** Xem lịch sử thay đổi giá trị value của key

**$ etcdctl watch foo**

**# in another terminal: etcdctl put foo bar**

**PUT**

**foo**

**bar**

* ***Lease:*** Gán time to live (TTL) cho key, sau khoảng thời gian được gán thì key đó sẽ bị xóa

**$ etcdctl lease grant 10**

**lease 32695410dcc0ca06 granted with TTL(10s)**

* ***Revoke:*** Thu hồi lease

**$ etcdctl lease revoke 32695410dcc0ca06**

**lease 32695410dcc0ca06 revoked**

* ***Snapshot***: Lưu lại trạng thái dữ liệu ở thời điểm hiện tại

### Naive Bayes

### Túi từ (Bag of Words - Bow), TF-IDF

Túi từ (Bag of Words[[7]](#footnote-8)) là một thuật toán hỗ trợ xử lý ngôn ngữ tự nhiên và mục đích là phân loại ký tự hay văn bản. Ý tưởng của túi từ là phân tích và phân nhóm dựa theo "Túi từ" (Kho ngữ liệu – Corpus[[8]](#footnote-9)). Với bộ dữ liệu mới, tiến hành tìm ra số lần từng từ của nó xuất hiện trong "túi". Tuy nhiên túi từ vẫn tồn tại khuyết điểm, nên TF-IDF là phương pháp khắc phục. Có thể ứng dụng túi từ kết hợp TF-IDF vào việc tìm kiếm, phân loại tài liệu, lọc thư làm phiền, xác định ý định của người dùng...

Ví dụ các bước thực hiện:

* Hình thành vector

“The quick brown fox jumps over the lazy dog and”

“Never jump over the lazy dog quickly”

Từ 2 câu trên, tiến hành tạo từ điển chứa các từ xuất hiện trong từng câu.

|  |  |  |
| --- | --- | --- |
| { |  |  |
|  | "brown": | 0 |
|  | "dog": | 1 |
|  | "fox": | 2 |
|  | "jump": | 3 |
|  | "jumps": | 4 |
|  | "lazy": | 5 |
|  | "never": | 6 |
|  | "over": | 7 |
|  | "quick": | 8 |
|  | "quickly": | 9 |
|  | "the": | 10 |
| } |  |  |

Dựa vào từ điển vừa tạo, tiến hành tạo vector lưu trữ số lần xuất hiện của từ trong từ điển ứng với mỗi câu. Và do từ điển đang có 11 từ nên mỗi vector sẽ có 11 phần tử như sau:

[1,1,1,0,1,1,0,1,1,0,2]

[0,1,0,1,0,1,1,1,0,1,1]

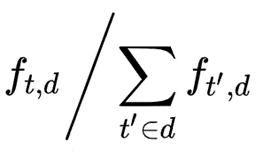
(Ở câu số 1, "the" xuất hiện 2 lần nên phần tử thứ 11 sẽ mang giá trị 2. Tương tự "brown", "dog", "fox", "jumps", "lazy", "over", "quick" xuất hiện 1 lần. Và vector chỉ mang giá trị 0 khi từ đó không xuất hiện trong câu như "jump", "over", "quickly")

* Đánh lại trọng số với TF-IDF

Trong hầu hết các ngôn ngữ, có một số từ có xu hướng xuất hiện thường xuyên như trong tiếng anh có "is", "the"... tương tự tiếng việt có các từ như "là", "của", "cứ"... Chính vì vậy nếu chỉ xét theo tần số xuất hiện của từng từ thì việc phân loại văn bản rất có thể cho kết quả sai dẫn tỷ lệ chính xác sẽ thấp. Vậy giải pháp là gì ?

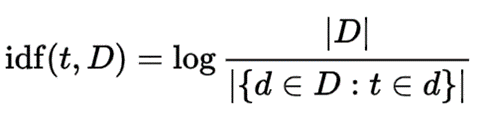
Phương pháp phổ biến là sử dụng một phương pháp thống kê có tên là TF-IDF, giá trị TF-IDF của một từ là một con số thu được qua thống kê thể hiện mức độ quan trọng của từ này trong một văn bản, mà bản thân văn bản đang xét nằm trong một tập hợp các văn bản.

Tại sao phương pháp lại có tên là TF-IDF?

Đầu tiên, TF(Term Frequency) là tần số xuất hiện của 1 từ trong 1 văn bản có cách tính như sau:

* f(t,d) - số lần xuất hiện từ t trong văn bản d.
* mẫu số là tổng số từ trong văn bản d

Tiếp theo là IDF (Inverse Document Frequency): Tần số nghịch của 1 từ trong tập văn bản (corpus).

Mục đích của việc tính IDF là giảm giá trị của các từ thường xuyên xuất hiện như "is", "the"... Do các từ này không mang nhiều ý nghĩa trong việc phân loại văn bản.

|D|: tổng số văn bản trong tập D, mẫu số là số văn bản có chứa từ t . Nếu từ đó không xuất hiện ở bất cứ 1 văn bản nào trong tập thì mẫu số sẽ bằng 0 => phép chia cho không không hợp lệ, vì thế với trường hợp này thường cộng thêm 1 vào mẫu số.

Và cuối cùng TF-IDF bằng:

Những từ có giá trị TF-IDF cao là những từ xuất hiện nhiều trong văn bản này, và xuất hiện ít trong các văn bản khác. Việc này giúp lọc ra những từ phổ biến và giữ lại những từ có giá trị cao (từ khoá của văn bản đó).

### Định lý Bayes

Một định lý toán học để tính xác suất xảy ra của một sự kiện ngẫu nhiên y khi biết sự kiện liên quan x đã xảy ra.

Ta có 2 sự kiện ngẫu nhiên x và y, nếu x và y là 2 sự kiện độc lập, ta có xác suất để xảy ra x và y đồng thời là:

*(1)*

* P(x): xác suất xảy ra x riêng biệt.
* P(y): là xác suất xảy ra y riêng biệt.

Nếu x và y là 2 sự kiện liên quan đến nhau, và xác suất xảy ra sự kiện y lớn hơn 0, ta có thể định nghĩa xác suất xảy ra y khi biết x xảy ra như sau:

Ta có thể viết lại thành:

*Khi x và y là 2 sự kiện độc lập ta có P(y|x) = P(x), ta thu được công thức như (1)*

### Thuật toán phân cấp Naive Bayes

Một thuật toán phân loại dựa trên tính toán xác suất áp dụng định lý Bayes. Thuật toán này thuộc nhóm “Học có giám sát” (Supervised Learning). Với công thức tính xác suất:

Do đó ta có:

Trên thực tế thì ít khi tìm được dữ liệu mà các thành phần là hoàn toàn độc lập với nhau. Tuy nhiên giả thiết này giúp cách tính toán trở nên đơn giản, huấn luyện dữ liệu nhanh, đem lại hiệu quả bất ngờ với các lớp bài toán nhất định.

Cách xác định các thành phần (class) của dữ liệu dựa trên giả thiết này có tên là Thuật toán phân cấp Naïve Bayes.

### Các mô hình thuật toán phân cấp Naive Bayes

Một trong các bài toán nổi tiếng hiệu quả khi sử dụng Thuật toán phân cấp Naïve Bayes là bài toán phân loại ký tự.

Trong bài toán này, mỗi văn bản được thể hiện thành dạng bag of words, hiểu nôm na là thể hiện xem có bao nhiêu từ xuất hiện và tần suất xuất hiện trong văn bản, nhưng bỏ qua thứ tự các từ

Có 2 mô hình thuật toán Naive Bayes thường sử dụng là mô hình Bernoulli và mô hình đa thức (Multinomial). Trong bài toán này ta chỉ tìm hiểu mô hình đa thức.

### Tổng quan mô hình đa thức

Mô hình này chủ yếu được sử dụng trong phân loại văn bản mà feature vectors được tính bằng Bags of Words. Lúc này, mỗi văn bản được biểu diễn bởi một vector có độ dài d chính là số từ trong từ điển. Giá trị của thành phần thứ i trong mỗi vector chính là số lần từ thứ i xuất hiện trong văn bản đó.

Ta tính xác suất từ xuất hiện trong văn bản P(xi∣y) bằng công thức:

* Ni là tổng số lần từ xi xuất hiện trong văn bản.
* Nc là tổng số lần từ của tất cả các từ x1,…xn xuất hiện trong văn bản.

Công thức trên có hạn chế là khi từ xi không xuất hiện lần nào trong văn bản, ta sẽ có Ni=0. Điều này làm cho P(xi∣y) = 0.

Để khắc phục vấn đề này, người ta sử dụng kỹ thuật gọi là Laplace Smoothing bằng cách cộng thêm vào cả tử và mẫu để giá trị luôn khác 0.

* α thường là số dương, bằng 1.
* dα được cộng vào mẫu để đảm bảo ∑i=1dP(xi∣y)=1

### Tổng quan mô hình Bernoulli

Ở mô hình này, các feature vector là các giá trị nhị phân 0, 1. Trong đó 1 thể hiện từ có xuất hiện trong văn bản, 0 thể hiện từ đó không xuất hiện trong văn bản.

Xác suất P(xi∣y) được tính bằng công thức bên dưới, với P(i∣y) là tỉ lệ số lần từ x\_ixi xuất hiện trong toàn bộ tập dữ liệu huấn luyện có nhãn y

Nhiều tài liệu biểu diễn công thức dưới dạng khác

### Tập phổ biến và luật kết hợp với thuật toán Apriori

Apriori là thuật toán khả sinh được đề xuất bởi R. Agrawal và R. Srikant vào năm 1993 để khai thác các tập item đối với các luật kết hợp kiểu bool. Tên của thuật toán dựa trên việc thuật toán sử dụng tri thức trước (prior knowledge) của các thuộc tính tập item phổ biến, chúng ta sẽ thấy sau đây. Apriori dùng cách tiếp cận lặp được biết đến như tìm kiếm level-wise, với các tập k item được dùng để thăm dò các tập (k+1) item. Đầu tiên, tập các tập 1 item phổ biến được tìm thấy bằng cách quét cơ sở dữ liệu để đếm số lượng từng item, và thu thập những item thỏa mãn độ hỗ trợ tối thiểu. Tập kết quả đặt là L1. Tiếp theo, L1được dùng để tìm L2, tập các tập 2 item phổ biến, nó được dùng để tìm L3, và cứ thế tiếp tục, cho tới khi tập k item phổ biến không thể tìm thấy. Việc tìm kiếm cho mỗi Lk đòi hỏi một lần quét toàn bộ cơ sở dữ liệu.

Trước khi đi vào chi tiết của thuật toán Apriori đầu tiên chúng ta sẽ tìm hiểu xác định một vài thuật ngữ phổ biến được sử dụng trong thuật toán.

* Itemset: Là tập hợp của những item trong cơ sở dữ liệu mà nó được xác định bởi I={i1, i2,…. in}, trong đó n là số lượng item.
* Transaction: Là một thành phần cơ sở dữ liệu mà nó bao gồm tập hợp các item. Transaction được ký hiệu là T và T http://ait.edu.vn/Hoc_thuat/Hinh/icon_con.jpg I. Một Transaction chứa tâp hợp các item T={i1, i2,…. in}.
* Minimum support: Là điều kiện cần được đáp ứng bởi các item đề ra để có thể tiến hành xử lý item kế tiếp có thể. Minimum support có thể được xem như là một điều kiện giúp loại bỏ các tâp không phổ biến trong bất kỳ cơ sở dữ liệu. Thường sử dụng Minimum support cho mô hình tỷ lệ phần trăm.
* Frequent itemset (tập phổ biến): Các itemset đáp ứng các tiêu chí điều kiện minimum support thì được gọi là tập phổ biến. Nó được ký hiệu là Li trong đó i chỉ i-itemset.
* Candidate itemset (ứng viên tập phổ biến): Ứng viên tập phổ biến là các item chỉ được xem xét xử lý. Ứng viên tập phổ biến là tất cả các kết hợp có thể có của tập phổ biến. Nó thường được ký hiệu Ci trong đó i chỉ i-itemset.
* Support: Độ hữu dụng của một luật có thể được đo với sự giúp đỡ của ngưỡng hỗ trợ. Support giúp chúng ta đo như thế nào các giao tác có  tập phổ biến mà nó phù hợp với ý nghĩa cả hai khía cạnh trong luật kết hợp.

Xem xét hai item A và b. Để tính toán support của A->B theo công thức như sau:

Confidence – Confidence chỉ sự chắc chắn của các luật. Thông số này cho phép chúng ta đếm mức độ thường xuyên một giao tác của tập phổ biến phù hợp với ý nghĩa cả phía cạnh bên trái với phía cạnh bên phải. các tập phổ biến không đáp ứng các điều kiện trên có thể được loại bỏ.

Xem xét hai item A và B. Để tính toán confidence của A->B theo công thức sau:

Chú ý: Conf(A B) có thể không bằng conf(BA)

### Các công cụ hỗ trợ hệ thống

### Python

### Giới thiệu

Là một ngôn ngữ lập trình bậc cao, thông dịch, hướng đối tượng, đa mục đích và cũng là một ngôn ngữ lập trình động. Cú pháp của Python thường khá dễ học và ngôn ngữ này cũng mạnh mẽ và linh hoạt không kém các ngôn ngữ khác trong việc phát triển các ứng dụng.

Python được phát triển bởi Guido van Rossum vào cuối những năm 80 và đầu những năm 90 tại Viện toán-tin ở Hà Lan. Python kế thừa từ nhiều ngôn ngữ như ABC, Module-3, C, C++, Unix Shell, …

### Đặc điểm

* Dễ dàng để sử dụng: Python là một ngôn ngữ bậc cao, số lượng từ khóa ít, cấu trúc đơn giản hơn và cú pháp được định nghĩa khá rõ ràng.
* Có một thư viện chuẩn khá lớn, dễ dàng tương thích và tích hợp với UNIX, Windows, và Macintosh.
* Một ngôn ngữ thông dịch, hướng đối tượng và hỗ trợ các phương thức lập trình theo hàm và cấu trúc.

### Ưu điểm

* Cú pháp đơn giản, ngôn ngữ dễ học, mã nguồn dễ đọc và dễ tìm hiểu.
* Tốc độ xử lý của Python nhanh hơn so với PHP.
* Thư viện có tiêu chuẩn cao, có khối cơ sở dữ liệu khá lớn nhằm cung cấp giao diện cho tất cả các cơ sở dữ liệu thương mại.
* Python được biên dịch và chạy trên tất cả nên tảng lớn hiện nay.
* Cộng đồng sử dụng và hỗ trợ lớn.

### Nhược điểm

* Python không có các thuộc tính như protected, private hay public, không có vòng lặp do...white và switch...case.
* Tốc độ xử lý của Python nhanh hơn PHP nhưng chậm hơn so với C++, Java.

### Java

### Giới thiệu

Là một ngôn ngữ bậc cao, thông dịch, hướng đối tượng, đa mục đích và cũng là một ngôn ngữ lập trình động. Khác với phần lớn ngôn ngữ lập trình thông thường, thay vì biên dịch mã nguồn thành mã máy hoặc thông dịch mã nguồn khi chạy, Java được thiết kế để biên dịch mã nguồn thành bytecode, bytecode sau đó sẽ được môi trường thực thi (runtime environment) chạy. Cú pháp Java được vay mượn nhiều từ C & C++ nhưng có cú pháp hướng đối tượng đơn giản hơn và ít tính năng xử lý cấp thấp hơn. Do đó việc viết một chương trình bằng Java dễ hơn, đơn giản hơn, đỡ tốn công sửa lỗi hơn.

Java được khởi đầu bởi James Gosling và bạn đồng nghiệp ở Sun Microsystems năm 1991. Ban đầu ngôn ngữ này được gọi là Oak (có nghĩa là cây sồi; do bên ngoài cơ quan của ông Gosling có trồng nhiều loại cây này), họ dự định ngôn ngữ đó thay cho C++, nhưng các tính năng giống Objective C. Không nên lẫn lộn Java với JavaScript, hai ngôn ngữ đó chỉ giống tên và loại cú pháp như C. Công ty Sun Microsystems đang giữ bản quyền và phát triển Java thường xuyên. Tháng 03/2014, công ty Sun Microsystems tiếp tục cho ra bản JDK 1.8.[[9]](#footnote-10)

Java được tạo ra với tiêu chí "Viết (code) một lần, thực thi khắp nơi" ("Write Once, Run Anywhere" (WORA)). Chương trình phần mềm viết bằng Java có thể chạy trên mọi nền tảng (platform) khác nhau thông qua một môi trường thực thi với điều kiện có môi trường thực thi thích hợp hỗ trợ nền tảng đó.

### Đặc điểm

* Máy ảo java(JVM – Java Virtual Machine): Một chương trình viết bằng ngôn ngữ lập trình Java sẽ được biên dịch ra mã của máy ảo java (mã java bytecode). Sau đó máy ảo Java chịu trách nhiệm chuyển mã java bytecode thành mã máy tương ứng.).
* Thông dịch: Java là một ngôn ngữ lập trình vừa biên dịch vừa thông dịch. Chương trình nguồn viết bằng ngôn ngữ lập trình Java có đuôi \*.java đầu tiên được biên dịch thành tập tin có đuôi \*.class và sau đó sẽ được trình thông dịch thông dịch thành mã máy.
* Độc lập nền: Một chương trình viết bằng ngôn ngữ Java có thể chạy trên nhiều máy tính có hệ điều hành khác nhau (Windows, Unix, Linux, …) với điều kiện ở đó có cài đặt máy ảo java (Java Virtual Machine).
* Hướng đối tượng: Hướng đối tượng trong Java tương tự như C++ nhưng Java là một ngôn ngữ lập trình hướng đối tượng hoàn toàn. Tất cả mọi thứ đề cập đến trong Java đều liên quan đến các đối tượng được định nghĩa trước, thậm chí hàm chính của một chương trình viết bằng Java (đó là hàm main) cũng phải đặt bên trong một lớp. Hướng đối tượng trong Java không có tính đa kế thừa (multi inheritance) như trong C++ mà thay vào đó Java đưa ra khái niệm interface để hỗ trợ tính đa kế thừa.
* Đa nhiệm - đa luồng (MultiTasking - Multithreading): Java hỗ trợ lập trình đa nhiệm, đa luồng cho phép nhiều tiến trình, tiểu trình có thể chạy song song cùng một thời điểm và tương tác với nhau.
* Khả chuyển (portable): Chương trình ứng dụng viết bằng ngôn ngữ Java chỉ cần chạy được trên máy ảo Java là có thể chạy được trên bất kỳ máy tính, hệ điều hành nào có máy ảo Java. “Viết một lần, chạy mọi nơi” (Write Once, Run Anywhere).
* Hỗ trợ mạnh cho việc phát triển ứng dụng: Công nghệ Java phát triển mạnh mẽ cung cấp nhiều công cụ, thư viện lập trình phong phú hỗ trợ cho việc phát triển nhiều loại hình ứng dụng khác nhau cụ thể như:
  + J2SE (Java 2 Standard Edition) hỗ trợ phát triển những ứng dụng đơn, ứng dụng client-server.
  + J2EE (Java 2 Enterprise Edition) hỗ trợ phát triển các ứng dụng thương mại.
  + J2ME (Java 2 Micro Edition) hỗ trợ phát triển các ứng dụng trên các thiết bị di động, không dây,...

### Ưu điểm

* Ưu điểm nổi bật của Java là độc lập với hệ thống xử lý và hệ điều hành nên nó có thể hoạt động trên bất cứ một môi trường nào.
* Là một ngôn ngữ hướng đối tượng.
* Công nghệ mở, cộng đồng hỗ trợ lớn nên sẽ có nhiều hỗ trợ khi cần thiết.

### Nhược điểm

Quản lý bộ nhớ với Java tốn kém ảnh hưởng đến tốc độ xứ lý của ứng dụng

# CHƯƠNG 3: PHÂN TÍCH VÀ THIẾT KẾ HỆ THỐNG

## 3.1. Phân tích hiện trạng nhu cầu

### 3.1.1. Yêu cầu chức năng

* Người dùng:
  + Xem thông tin sách.
  + Xem lịch sử đọc sách và quản lý thông tin cá nhân.
  + Chia sẻ và đọc sách được chia sẻ.
  + Đánh giá và bình luận sách.
* Hệ thống:
  + Đề xuất sách dựa trên lịch sử đọc sách.
  + Tự động phân loại thể loại sách người dùng chia sẻ.
* Người quản lí:
  + Quản lý người dùng.
  + Quản lý sách.
  + Tạo biểu mẫu thống kê.

### 3.1.2. Yêu cầu phi chức năng

* Giao diện thân thiện
* Ứng dụng tiện ích
* Tốc đổ xử lý nhanh

## Phân tích và thiết kế thành phần dữ liệu

### Cấu trúc dữ liệu

|  |  |
| --- | --- |
| User |  |
| Category |  |
| Book |  |
| Read |  |
| News | A screenshot of a cell phone  Description automatically generated |
| Comment | A screenshot of a cell phone  Description automatically generated |

### Mô tả các thuộc tính

#### ***User***

Lưu trữ thông tin người dùng

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | **Tên key** | **Kiểu dữ liệu** | **Ý nghĩa** | **Ghi chú** |
| 1 | /user/user\_username | Text | Tên người dùng | Không được trùng lặp |
| 2 | /user/user\_fullname | Text | Tên đầy đủ |  |
| 3 | /user/ user\_ password | Text | Mật khẩu |  |
| 4 | /user/ user\_ date\_of\_birth | Date | Ngày sinh |  |
| 5 | /user/ user\_ address | Text | Địa chỉ |  |
| 6 | /user/ user\_ avatar | Text | Ảnh đại diện | Chuỗi url |
| 7 | /user/user\_created\_date | Date | Ngày tạo tài khoản |  |
| 8 | /user/user\_is\_admin | Text | Quản trị hệ thống | - 0: người dùng bình thường  - 1: Quản trị hệ thống |

Bảng 3‑1. Bảng mô tả thuộc tính User

#### ***Category***

Lưu trữ danh sách thể loại

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên key | Kiểu dữ liệu | Ý nghĩa | Ghi chú |
| 1 | /category | Text | Danh sách thể loại sách | Ngăn cách nhau bởi dấu “,” |

Bảng 3‑2. Bảng mô tả thuộc tính category

#### ***Book***

Lưu trữ thông tin sách

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên key | Kiểu dữ liệu | Ý nghĩa | Ghi chú |
| 1 | /book/book\_id | Text | Id sách | Không được trùng lặp |
| 2 | /book/book\_category | Text | Thể loại sách |  |
| 3 | /book/book\_type | Text | Loại sách | “ebook/paper” |
| 4 | /book/book\_title | Text | Tiêu đề |  |
| 5 | /book/book\_author | Text | Tác giả | Chuỗi url |
| 6 | /book/book\_public\_date | Date | Ngày xuất bản | Ngăn cách nhau bởi dấu “,” |
| 7 | /book/book\_rating | Text | Giá trị rating |  |
| 8 | /book/book\_rated\_time | Text | Số lượt rating |  |
| 9 | /book/book\_download | Text | Số lượt download |  |
| 10 | /book/book\_read\_time | Text | Số lượt đọc sách |  |
| 11 | /book/book\_page | Text | Số trang |  |
| 12 | /book/book\_description | Text | Tóm tắt |  |
| 13 | /book/book\_image | Text | Hình ảnh |  |
| 14 | /book/book\_file | Text | File sách |  |
| 15 | /book/book\_created\_time | Datetime | Thời gian tạo |  |
| 16 | /book/is\_deleted | Text | Tình trạng (đã xóa) | - 0: Chưa xóa  - 1: Đã xóa |

Bảng 3‑3. Bảng mô tả thuộc tính Book

#### ***Read***

Lưu trữ các cuốn sách người dùng đã đọc

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên key | Kiểu dữ liệu | Ý nghĩa | Ghi chú |
| 1 | /read/{username}/{read\_book\_id} | Text | Id quyển sách |  |
| 2 | /read/{username}/{read\_book\_id}/read\_started\_time | Datetime | Thời gian bắt đầu đọc |  |
| 3 | /read/{username}/{read\_book\_id}/read\_ended\_time | Datetime | Thời gian hoàn thành đọc sách |  |
| 4 | /read/{username}/{read\_book\_id}/read\_is\_deleted | Text | Tình trạng | - 0: Chưa xóa  -1 : Đã xóa lịch sử đọc |

Bảng 3‑4. Bảng mô tả thuộc tính Reading history

#### ***News***

Lưu trữ thông tin bài đăng người dùng chia sẻ về sách

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên key | Kiểu dữ liệu | Ý nghĩa | Ghi chú |
| 1 | /news/{news\_id} | Text | Id bài đăng | Không được trùng lặp |
| 2 | /news /{news\_id}/ news\_username | Text | Tên người dùng |  |
| 3 | /news /{news\_id}/ news\_ book\_id | Text | Id sách |  |
| 4 | /news /{news\_id}/ news\_ content | Text | Nội dung |  |
| 5 | /news /{news\_id}/ news\_ created\_time | Datetime | Thời gian tạo |  |
| 6 | /news /{news\_id} / news\_ is\_deleted | Text | Tình trạng (đã xóa) | - 0: Chưa xóa  -1: Đã xóa |

Bảng 3‑5. Bảng mô tả thuộc tính News

#### ***Comment***

Lưu trữ thông tin bình luận

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| STT | Tên key | Kiểu dữ liệu | Ý nghĩa | Ghi chú |
| 1 | /comment/{book\_id}/comment\_id | Text | Id bình luận | Không được trùng lặp |
| 2 | /comment/{book\_id}/ comment\_ username | Text | Tên người dùng |  |
| 3 | /comment/{book\_id}/comment\_ content | Text | Nội dung bình luận |  |
| 4 | /comment/{book\_id}/comment\_ created\_time | Datetime | Thời gian tạo |  |
| 5 | /comment/{book\_id}/comment\_ is\_deleted | Text | Tình trạng (đã xóa) | - 0: Chưa xóa  - 1: Đã xóa |

Bảng 3‑6. Bảng mô tả thuộc tính Comment

## Phân tích thiết kế thành phần xử lý

### Sơ đồ Usecase

Hình 3‑1. Sơ đồ usecase

### 3.3.2. Danh sách Usecase

|  |  |  |
| --- | --- | --- |
| **STT** | **Usecase** | **Ý nghĩa** |
| 1 | Usecase đăng kí | Người dùng đăng kí tài khoản ứng dụng |
| 2 | Usecase đăng nhập | Người dùng đăng nhập vào ứng dụng |
| 3 | Usecase đăng xuất | Người dùng đăng xuất khỏi ứng dụng |
| 4 | Usecase tra cứu người dùng | Người dùng tra cứu thông tin của các người dùng khác |
| 5 | Usecase khởi tạo màn hình trang chủ | Khỏi tạo màn hình chính của ứng dụng |
| 6 | Usecase tra cứu sách | Người dùng tra cứu sách theo tên, thể loại,… |
| 7 | Usecase xem bảng tin chung | Người dùng xem tất cả các sách đã được chia sẻ |
| 8 | Usecase thống kê hệ thống | Quản trị hệ thống xem thông tin thống kê |
| 9 | Usecase quản lý thông tin người dùng | Người dùng xem thông tin và thay đổi thông tin của mình |
| 10 | Usecase đổi mật khẩu | Người dùng đổi mật khẩu tài khoản của mình |
| 11 | Usecase đổi thông tin người dùng | Người dùng đổi thông tin tài khoản của mình |
| 12 | Usecase quản lý lịch sử đọc sách | Người dùng quản lý lịch sử đọc sách của mình |
| 13 | Usecase xác nhận đọc xong sách | Người dùng xác nhận quyển sách đã được đọc xong |
| 14 | Usecase xóa lịch sử đọc sách | Người dùng xóa sách khỏi lịch sử đọc sách của mình |
| 15 | Usecase quản lý sách – bài đăng chia sẻ | Người dùng quản lý sách hoặc bài đăng của mình |
| 16 | Usecase thêm sách – bài đăng | Người dùng thêm sách hoặc bài đăng vào hệ thống |
| 17 | Usecase xem sách – bài đăng | Người dùng xem sách – bài đăng của mình |
| 18 | Usecase chỉnh sửa sách – bài đăng | Người dùng chỉnh sửa sách – bài đăng của mình |
| 19 | Usecase xóa sách – bài đăng | Người dùng xóa sách – bài đăng của mình |
| 20 | Usecase xem chi tiết sách | Người dùng xem chi tiết quyển sách đã chọn |
| 21 | Usecase bình luận sách | Người dùng đăng tải bình luận lên trang chi tiết sách |
| 22 | Usecase đánh giá sách | Người dùng thêm đánh giá cho quyển sách đã chọn |
| 23 | Usecase đọc sách | Người dùng đọc nội dung sách trên ứng dụng |
| 24 | Usecase lập lịch nhắc nhở | Người dùng lập lịch đọc sách trên ứng dụng |
| 25 | Usecase điều khiển bằng giọng nói | Người dùng điều khiển ứng dụng bằng giọng nói thay vì thao tác bằng tay |

Bảng 3‑7. Danh sách usecase

### Sơ đồ tuần tự

* A screenshot of a social media post

  Description automatically generatedUsecase đăng kí

Hình 3‑2. Sơ đồ tuần tự usecase đăng kí

* A screenshot of a cell phone on a table

  Description automatically generatedUsecase đăng nhập

Hình 3‑3. Sơ đồ tuần tự usecase đăng nhập

* A picture containing sky, screenshot

  Description automatically generatedUsecase đăng xuất

Hình 3‑4. Sơ đồ tuần tự usecase đăng xuất

* A close up of a map

  Description automatically generatedUsecase tra cứu người dùng

Hình 3‑5. Sơ đồ tuần tự usecase tra cứu người dùng

* A close up of a map

  Description automatically generatedUsecase khởi tạo màn hình trang chủ

Hình 3‑6. Sơ đồ tuần tự usecase khởi tạo màn hình trang chủ

* Usecase tra cứu sách

A close up of a map

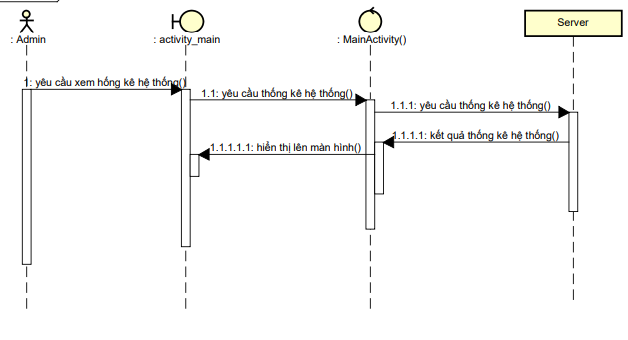
Description automatically generated

Hình 3‑7. Sơ đồ tuần tự usecase tra cứu sách

* A picture containing sky, screenshot, indoor

  Description automatically generatedUsecase xem bảng tin chung

Hình 3‑8. Sơ đồ tuần tự usecase xem bảng tin chung

* Usecase thống kê hệ thống

Hình 3‑9. Sơ đồ tuần tự usecase thống kê hệ thống

* A close up of a map

  Description automatically generatedUsecase quản lý thông tin người dùng, đổi mật khẩu, đổi thông tin người dùng

Hình 3‑10. Sơ đồ tuần tự usecase quản lý thông tin người dùng

* Usecase quản lý lịch sử đọc sách, xác nhận đọc sách, xóa lịch sử đọc sách

A close up of a map

Description automatically generated

Hình 3‑11. Sơ đồ tuần tự usecase quản lý lịch sử đọc sách

* A close up of a map

  Description automatically generatedUsecase thêm sách – bài đăng

Hình 3‑12. Sơ đồ tuần tự usecase thêm sách – bài đăng

* A picture containing screenshot

  Description automatically generatedUsecase xem sách – bài đăng

Hình 3‑13. Sơ đồ tuần tự usecase xem sách – bài đăng

* Usecase chỉnh sửa sách – bài đăng

A close up of a map

Description automatically generated

Hình 3‑14. Sơ đồ tuần tự usecase chỉnh sửa sách – bài đăng

* A screenshot of a social media post

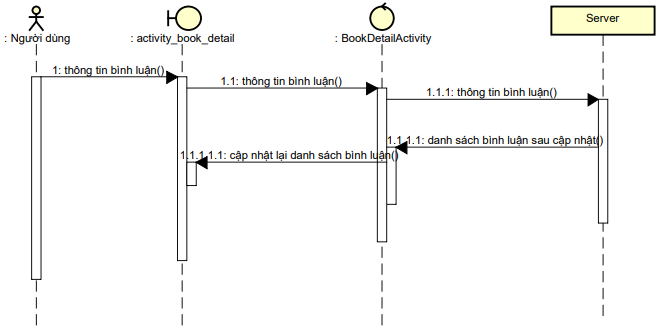
  Description automatically generatedUsecase xóa sách – bài đăng

Hình 3‑15. Sơ đồ tuần tự usecase xóa sách – bài đăng

* A close up of a map

  Description automatically generatedUsecase xem chi tiết sách

Hình 3‑16. Sơ đồ tuần tự usecase xem chi tiết sách

* Usecase bình luận sách

Hình 3‑17. Sơ đồ tuần tự usecase bình luận sách

* Usecase đánh giá sách

A screenshot of a cell phone

Description automatically generated

Hình 3‑18. Sơ đồ tuần tự usecase đánh giá sách

* A picture containing screenshot

  Description automatically generatedUsecase đọc sách

Hình 3‑19. Sơ đồ tuần tự usecase đọc sách

* Usecase điều khiển bằng giọng nói

A screenshot of a cell phone

Description automatically generated

Hình 3‑20. Sơ đồ usecase điều khiển bằng giọng nói

* A close up of a map

  Description automatically generatedUsecase lập lịch nhắc nhở

Hình 3‑21. Sơ đồ usecase lập lịch nhắc nhở

## Thiết kế giao diện

|  |  |  |
| --- | --- | --- |
| **STT** | **Tên giao diện** | **Chức năng và ý nghĩa** |
| 1 | activity\_add\_shared\_book | Màn hình thêm bảng tin, thêm sách |
| 2 | activity\_admin | Màn hình thống kê hệ thống |
| 3 | activity\_all\_shared\_books | Màn hình bảng tin của toàn bộ người dùng |
| 4 | activity\_book\_detail | Màn hình chi tiết sách |
| 5 | activity\_book\_library | Màn hình tra cứu sách |
| 6 | activity\_log\_in | Màn hình đăng nhập |
| 7 | activity\_main | Màn hình trang chủ |
| 8 | activity\_pdf\_viewer | Màn hình đọc sách |
| 9 | activity\_personal\_page | Màn hình thông tin cá nhân |
| 10 | activity\_register | Màn hình đăng kí |
| 11 | activity\_user\_information | Màn hình chỉnh sửa thông tin người dùng |
| 12 | activity\_user\_infomation\_change\_password | Màn hình đổi mật khẩu |
| 13 | activity\_user\_reading\_history | Màn hình lịch sử đọc sách |
| 14 | activity\_users | Màn hình tra cứu người dùng |
| 15 | activity\_user\_shared\_book | Màn hình bảng tin chung |

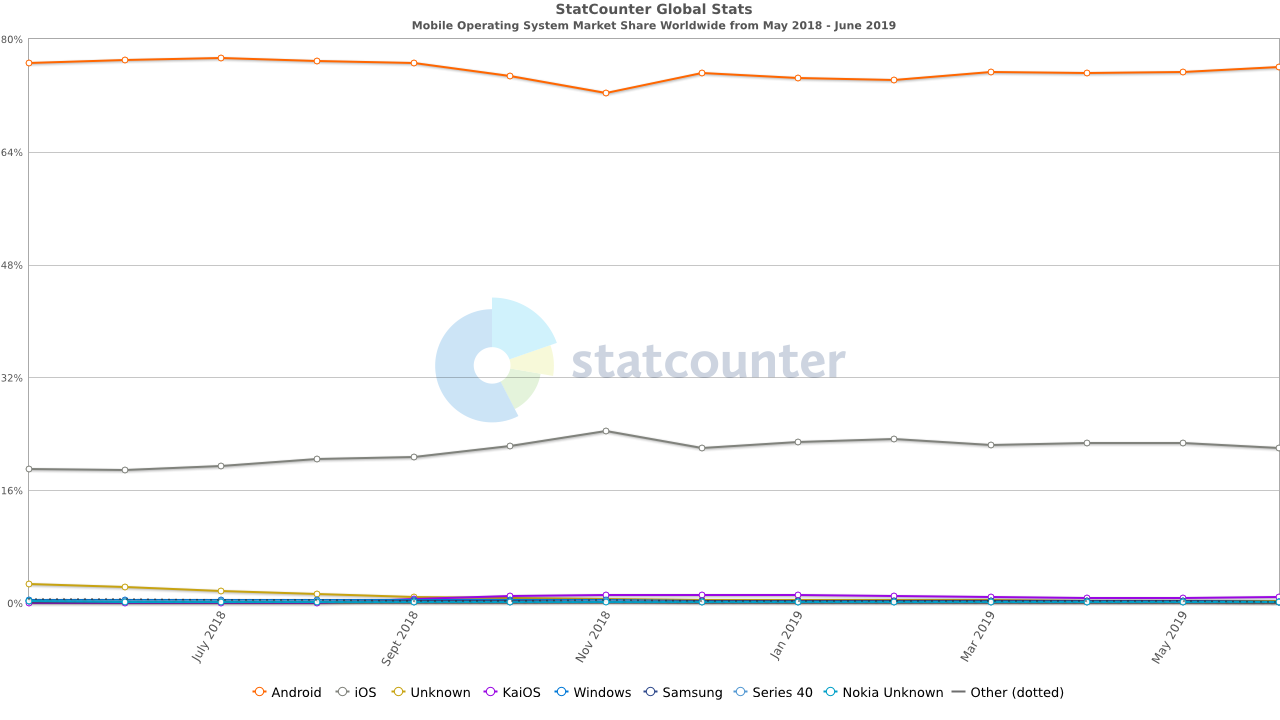
Bảng 3‑8. Danh sách màn hình

## Xây dựng giao diện

|  |  |
| --- | --- |
| A screenshot of a cell phone  Description automatically generated  Hình 3‑22. Màn hình đăng nhập | A screenshot of a cell phone  Description automatically generated  Hình 3‑23. Màn hình đăng kí |
| Hình 3‑24. Màn hình trang chủ | Hình 3‑25. Màn hình quản lý thông tin |
| Hình 3‑26. Màn hình lịch sử đọc sách | Hình 3‑27. Màn hình chi tiết sách |
| Hình 3‑28. Màn hình bảng tin chia sẻ | A screenshot of a cell phone  Description automatically generated  Hình 3‑29. Màn hình tra cứu sách |
| A screenshot of a cell phone  Description automatically generated  Hình 3‑30.Màn hình thêm sách | A screenshot of a cell phone  Description automatically generated  Hình 3‑31. Màn hình tra cứu người dùng |
| Hình 3‑32. Màn hình trang cá nhân | Hình 3‑33. Màn hình thống kê |
| A screenshot of a cell phone  Description automatically generated  Hình 3‑34. Màn hình danh sách nhắc nhở | A screenshot of a cell phone  Description automatically generated  Hình 3‑35. Màn hình đặt lịch nhắc nhở |

# CHƯƠNG 4: CÀI ĐẶT VÀ TRIỂN KHAI HỆ THỐNG

## Môi trường triển khai

Trên thị trường ứng dụng di động hiện nay, 3 hệ điều hành chiếm thị phần cao nhất là: Android, iOS và Window phone.

Hình 4‑1. Sơ đồ thị trường hệ điều hành điện thoại toàn thế giới từ tháng 5 2018 đến tháng 6 2019

Dựa vào biểu đồ trên[[10]](#footnote-11) có thể nhận thấy Android chiếm phần lớn hơn cả.

Có 3 hướng chính để phát triển một dứng dụng di động: Web App, Native App và Hybrid App. Mỗi hướng đi sẽ cần những kĩ năng và kiến thức riêng để triển khai và hoàn thiện từ đó sinh ra những ưu nhược điểm riêng của từng loại.

* Web App (Mobile Web)

Hướng Mobile Web thường được áp dụng khi đã có sẵn một website đang hoạt động. Ta sẽ tạo thêm 1 trang web riêng cho mobile, sử dụng HTML, CSS, một số Framework hỗ trợ mobile và responsive (Bootstrap, jQuery Mobile, Materialize). Người dùng sẽ truy cập vào trang web dành cho mobile để dùng ứng dụng.

|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| * Chỉ cần có kiến thức về web là viết được. * Viết một lần, chạy được trên mọi hệ điều hành. * Người dùng không cần phải cài App, có thể vào thẳng trang web. * Không cần phải thông qua App Store, tiết kiệm tiền. * Dễ nâng cấp (Chỉ việc nâng cấp web là xong). | * Với một số máy đời cũ, Web App sẽ bị bể giao diện, hiển thị sai, hoặc javascript không chạy. * Performance chậm * Không thể tận dụng được các tính năng của di động: Push Notification, chụp hình, nghiêng máy, định vị GPS… |

* Native App

Viết Native App nghĩa là lập trình viên sẽ sử dụng IDE, SDK mà nhà sản xuất cung cấp để lập trình ra một ứng dụng, build ứng dụng đó thành file cài và gửi lên App Store để kiểm duyệt. Người dùng sẽ phải tìm ứng dụng trên App Store, tải về máy và chạy. Đây là hướng phát triển được áp dụng nhiều và với những ứng dụng game, xử lý ảnh, cần tính toán nhiều, Native App là lựa chọn duy nhất.

|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| * Tận dụng được toàn bộ những tính năng của thiết bị: Chụp ảnh, nghiêng máy, rung, GPS, Notification. * Có thể chạy mà không cần dùng mạng. * Xử lý rất nhanh vì code native sẽ được chạy trực tiếp. * Là lựa chọn duy nhất cho các ứng dụng trò chơi, xử lý hình ảnh … | * Lập trình viên phải cài đặt nhiều phần mềm đi kèm, khó tiếp cận. * Với mỗi hệ điều hành, ta phải viết một ứng dụng riêng. Khó đảm bảo sự đồng bộ giữa các ứng dụng. * Cần phải đưa ứng dụng lên chợ ứng dụng của hệ điều hành tương ứng, mỗi lần có cập nhật chỉnh sửa cần phải thông báo người dùng. * Tốn nhiều công sức và thời gian để lập trình. |

* Hybrid App

Hybrid App kết hợp những ưu điểm của Mobile Web và Native App. Ta xây dựng một ứng dụng bằng HTML, CSS, Javascript, chạy trên WebView của mobile. Tuy nhiên, Hybrid App vẫn có thể tận dụng những tính năng của device: chụp hình, GPS, rung, ….

Hybrid App sẽ được viết dựa trên một cross-platform Framework: Cordova, Phonegap, Titanium, …. Ta sẽ gọi những chức năng của mobile thông qua API mà Framework này cung cấp, dưới dạng Javascript. Bạn chỉ cần viết một lần, những Framework này sẽ tự động dịch ứng dụng này ra các file cài đặt cho Android, iOS và Windows Phone.

|  |  |
| --- | --- |
| **Ưu điểm** | **Nhược điểm** |
| * Chỉ cần biết HTML, CSS, JS. * Viết một lần, chạy được trên nhiều hệ điều hành. * Tận dụng được các chức năng của thiết bị. | * Không ổn định, khó debug. Framework sẽ dịch code của bạn thành code native, việc sửa lỗi ứng dụng khá khó vì bạn không biết code sẽ được dịch ra như thế nào. * Xử lý chậm. * Cần cài đặt nhiều phần mềm phụ trợ.thì mới build ứng dụng được). |

* Kết luận: Dựa trên những phân tích trên, nhóm lựa chọn Android Native App để thực hiện đề tài cùng với môi trường lập trình như sau: Ubuntu 18.04 LTS, Window 10 Education, Android Studio 3.4.1, Genymotion 2.12.1.

## Giải pháp công nghệ sử dụng ở phía Client (ứng dụng Android)

### Áp dụng Annotation[[11]](#footnote-12) để giảm thời gian lập trình

Một framework mã nguồn mở dùng để tăng tốc quá trình phát triển phần mềm Android. Annotation sẽ lo việc kết nối để giúp người lập trình dễ dàng tập trung vào những việc thật sự quan trọng. Quan trọng nhất, Annotation sẽ đơn giản hóa lượng lệnh người lập trình phải gõ từ đó giúp việc bảo trì code dễ dàng hơn. Ngoài ra các nhà lập trình có thể tự thay thay đổi cách mỗi Annotation hiện có làm việc theo mong muốn của mình. Phiên bản mới nhất 4.6.0 được vừa được phát hành đầu năm nay.

Tính đến nay đã có hơn 120 ứng dụng Android[[12]](#footnote-13) áp dụng Annotation trong quá trình phát triển.

Ví dụ

@EActivity(R.layout.translate) // Sets content view to R.layout.translate

public class TranslateActivity extends Activity {

@ViewById // Injects R.id.textInput

EditText textInput;

@ViewById(R.id.myTextView) // Injects R.id.myTextView

TextView result;

@AnimationRes // Injects android.R.anim.fade\_in

Animation fadeIn;

@Click // When R.id.doTranslate button is clicked

void doTranslate() {

translateInBackground(textInput.getText().toString());

}

@Background // Executed in a background thread

void translateInBackground(String textToTranslate) {

String translatedText = callGoogleTranslate(textToTranslate);

showResult(translatedText);

}

@UiThread // Executed in the ui thread

void showResult(String translatedText) {

result.setText(translatedText);

result.startAnimation(fadeIn);

}

// [...]

}

* @EActivity: gán một layout đã được định nghĩa trong lớp R cho lớp TranslateActivity, thay thế cho hàm setContentView(). Từ đó ta sẽ không cần Override lại hàm onCreate(). Trong đa số trường hợp chúng ta chỉ làm việc này trong hàm onCreate(), nên việc Override lại hàm sẽ tốn thời gian hơn việc dùng Annotation @EActivity.
* @ViewById: gán một item trên một layout bất kì cho một biến java trong lớp. Thay thế cho hàm FindById(). Hàm này phải được gọi khi khởi tạo màn hình của ứng dụng. Việc sử dụng Annotation sẽ giúp người lập trình không cần quan tâm đến việc không quan trọng này, chỉ cần khi báo như một thuộc tính của lớp.
* @Click: gán sự kiện Click cho một item trên màn hình. Một cách đơn giản và ngắn gọn hơn so với dùng các lệnh thông thường mà không áp dụng Annotation.
* @Background và @Uithread: 2 Annotation đơn giản xác định hàm nào sẽ chạy luồng nào, trên UI hay chạy nền.

### Sử dụng thư viện Android PdfViewer[[13]](#footnote-14) để đọc và hiển thị file pdf từ bộ nhớ điện thoại

Thư viện được phát triển bởi cộng đồng github từ hơn 3 năm trước. Gồm các chức năng cơ bản của một ứng dụng xem file .pdf (có thể mở rộng thông qua việc chỉnh sửa source code). Hỗ trợ các chuyển động (animation), cử chỉ (gestures) … giúp việc đọc file .pdf thuận tiện và thân thiện.

Phiên bản hiện tại là 3.0 đã hỗ trợ trên các nền tảng android từ Android 4.0 trở lên.

### Vẽ biểu đồ với MPAndroidChart[[14]](#footnote-15)

Được phát triển bởi Philipp Jahoda và chia sẻ trên github, thư viện hỗ trợ từ phiên bản Android 4.0 trở lên với nhiều loại biểu đồ khác nhau và được đi kèm với vô số hiệu ứng cũng như chuyển động hiện đại, tạo một cảm giác thoải mái cho người sử dụng.

### Việc tải và sử dụng hình ảnh

Khi thực hiện load ảnh vào trong các view như ImageView,.. sẽ có rất nhiều vấn đề phải đối mặt như: lỗi OutOfMemory, load ảnh chậm, giao diện bị đơ (Unresponsive), việc cuộn các List ImageView không mượt mà,…

Hiện nay, có rất nhiều thư viện mã nguồn mở ra đời nhằm hỗ trợ ứng dụng Android trong việc load ảnh cho hiệu năng cực kì tốt như: Picasso, Universal Image Loader (UIL), Glide, Fresco,… Không có thư viện nào là hoàn hảo, mỗi thư viện đều có cơ chế hoạt động khác nhau nên dẫn đến tốc độ và sự tối ưu trong việc load ảnh là khác nhau. Vì vậy, để tìm ra thư viện phù hợp với dự án, cần xem xét so sánh ưu và nhược điểm của chúng. Tuy nhiên, việc so sánh tất cả các thư viện là điều không khả thi, nên sẽ tiến hành so sánh 2 thư viện được sử dụng phổ biến hiện nay là: Glide và Picasso (dựa trên số sao được đánh giá trên Github: Picasso – 15268 và Glide – 22169 )

|  |  |  |  |
| --- | --- | --- | --- |
| Tiêu chí | Glide | **Picasso** | **Nhận xét** |
| **Cú pháp** | GlideApp  .with(myFragment)  .load(url)  .centerCrop()  .placeholder(R.drawable.image)  .into(myImageView)’ | Picasso.get()  .load(url)  .resize(50, 50)  .centerCrop()  .into(imageView) | Nhìn chung, Glide và Picasso có cú pháp load ảnh vào ImageView khá giống nhau. Tuy nhiên, Glide được thiết kế tốt hơn khi phương thức with() không chỉ cho truyền vào tham số Context như Picasso mà còn cho phép cả Activity và Fragment.  Với Glide, việc tải ảnh được tích hợp với vòng đời của Activity thông qua các phương thức Callbacks onPause() và onResume(). |
| **Bộ nhớ đệm** | Cả hai thư viện đều hỗ trợ caching hình ảnh vào bộ nhớ đệm. | | Với thiết kế của Glide, hình ảnh có thể được tải và hiển thị rất nhanh. Trong khi cách thực hiện của Picasso có thể bị chậm vì với mọi kích thước của ImageView, cache chỉ trả về 1 kích thước duy nhất.  Tuy nhiên, việc lưu trữ hình ảnh giống nhau với kích thước khác nhau trong Cache như Glide cần nhiều không gian bộ nhớ. |
| Ứng với mỗi lần yêu cầu hiển thị ảnh với ImageView có kích thước khác nhau. Glide sẽ tiến hành tải ảnh trước, sau đó resize để phù hợp với kích thước hiển thị với ImageView rồi mới lưu vào trong cache. | Tải hình ảnh một lần và lưu với kích thước full-size trong cache. |  |
| **Dung lượng bộ nhớ sử dụng** | Cùng một hình ảnh và ImageView, Glide có thể sử dụng ít bộ nhớ hơn so với Picasso. | Cùng một hình ảnh và ImageView, Picasso có thể sử dụng nhiều bộ nhớ hơn so với Glide. | Điều này là dễ hiểu vì cơ chế caching của Picasso (cache full-size image). |
| **Thời gian load ảnh** | Về thứ tự thực hiện, cả hai đều kiểm tra trong bộ nhớ cache trước, nếu không có thì mới tiến hành download từ URL. | | Tốc độ tải ảnh của Picasso nhanh hơn thì khi ảnh được tải, nó sẽ đưa thẳng ảnh với full-size và cache trong khi Glide phải điều chỉnh kích thước phù hợp với ImageView trước rồi mới đưa vào cache.  Tốc độ load ảnh của Glide nhanh hơn Picasso vì Picasso có thể phải resize kích thước của ảnh trước khi hiển thị lên ImageView. |
| Khi tải một ảnh từ URL, tốc độ có thể chậm hơn so với Picasso. | Khi tải một ảnh từ URL, tốc độ có thể nhanh hơn so với Glide. |  |
| Tốc độ hiển thị ảnh từ Cache ra ImageView có thể nhanh hơn so với Picasso. | Tốc độ hiển thị ảnh từ Cache ra ImageView có thể chậm hơn so với Glide. |

Bảng 4‑1. So sánh Glide và Picasso

Bên cạnh đó, có một số tính năng mà Glide hỗ trợ còn Picasso thì không như: hiển thị ảnh GIF, load nhiều hình ảnh với các view có kích thước giống nhau một lúc thông qua phương thức thumbnail(), Glide cung cấp nhiều phương thức để cấu hình và tùy biến hơn Picasso.

* Kết luận:đề tài sử dụng thư viện Glide cho dự án vì cả hai thư viện đều cho hiệu năng tương đương nhau. Tuy nhiên, dự án có chức năng load nhiều ảnh từ Gallery vào ListView nên tính năng chức năng Thumbnail của Glide có thể sẽ đem lại hiệu quả.

### Giao tiếp với Server

Trước giờ trên hệ điều hành Android, có nhiều cách để kết nối và lấy dữ liệu từ một Web Service. Có thể dùng các lớp trong gói HTTP Apache (đã loại bỏ trên Android 6.0, phải dùng thư viện ngoài) hoặc dùng một số thư viện để kết nối với Internet và nhận dữ liệu từ Server như: Volley, OkHttp, KSOAP, Retrofit,…

Chính vì có nhiều thư viện hỗ trợ việc liên lạc giữa ứng dụng Android (Client) với Server như vậy, nền chúng cần được tìm hiểu và so sánh dữa trên các yếu tố cơ bản của loại thư viện này.

Dựa theo đánh giá trên Github của từng thư viện, ta có kết quả sau: Retrofit (28409), OkHttp (27239), Volley (1688),… ).

Vì kể từ phiên bản Retrofit 2.0, Retrofit phụ thuộc vào OkHttp ( và OkHttp phụ thuộc vào Okio), vì vậy sử dụng Retrofit 2.0 đồng nghĩa với việc đang sử dụng OkHttp và Okio. Do đó, 2 thư viện Retrofit (của Square) và Volley (của Google) là các đối tượng được đem ra so sánh.

|  |  |  |  |
| --- | --- | --- | --- |
| Tiêu chí so sánh | Retrofit | Volley | Nhận xét |
| Thực hiện gửi Request | Cả hai thư viện đều hỗ trợ thực thi Request ở background, vì hệ điều hành Android không cho phép các xử lý liên quan đến network tương tác với MainThread, điều đó có thể gây ra NetWorkOnMainThreadException. | |  |
| Kiểu dữ liệu trả về được hỗ trợ | Retrofit API Response hỗ trợ chuyển đổi thành các kiểu dữ liệu như: Boolean, Integer, Date (Long Format Date), String, Object (JSON Object), Collections (String Format).  Với sự kết hợp với OkHttp thì Retrofit 2 cũng cho phép trả về Image trong Response. | Retrofit API Response hỗ trợ chuyển đổi thành các kiểu dữ liệu như: Boolean, Integer, Date (Long Format Date), String, Object (JSON Object), Collections (String Format).  Với sự kết hợp với OkHttp thì Retrofit 2 cũng cho phép trả về Image trong Response. | Khi so sánh giữa Retrofit và Volley thì hầu hết các kiểu dữ liệu cần thiết đều được 2 thư viện hỗ trợ.  Retrofit hỗ trợ chuyển đổi trực tiếp một JSON Object thành một POJO (Plain Old Java Object) (Volley thì không có tính năng này). |
| Cơ chế gửi lại | Hỗ trợ khi gửi lại Request khi bị vấn đề TimeOut (sử dụng phương thức setRetryPolicy). | Ở phiên bản Retrofit 2.0 thì đã có hỗ trợ cơ chế retry khi Request Timeout. |  |
| Lưu tạm | Android Volley có hỗ trợ việc cache lại Request. Đây là tính năng rất nổi bật của Volley. Khi thực hiện tạo gửi một Request, Volley đầu tiên sẽ kiểm tra trong cache, nếu response tương ứng đã có sẵn thì sẽ trực tiếp trả về cho MainThread, nếu không thì Request mới được gửi đi. (Hiệu quả khi caching Bitmap). | Không hỗ trợ |  |
| Tải ảnh | Không hỗ trợ tốt việc tải ảnh. Đều cần thư viện khác để làm điều này. (Retrofit 2 được nâng cấp nhiều hơn về việc này nhưng vẫn chưa đủ). | |  |
| Hiệu năng | Cả hai thư viện đều sử dụng Thread Pool và cho phép thực thi nhiều Request đồng thời. Vì vậy, bên cạnh đó, cả hai thư viện đều hỗ trợ Muiltipart Files Upload.  Cả hai thư viện đều không cho phép Downloads/Streaming File có kích thước lớn, có thể gây ra lỗi OutOfMemory. | |  |

Bảng 4‑2. So sánh Retrofit và Volley

* Kết luận: Cả hai thư viện trên đều đáp ứng đủ các nhu cầu cho dự án hiện tại của nhóm. Tuy nhiên, theo khảo sát thì tài liệu của Retrofit đầy đủ hơn so với Volley, cú pháp nhìn rõ ràng và có nhiều lựa chọn để tùy chỉnh, cộng đồng người sử dụng đông hơn nên nhóm quyết định chọn Retrofit làm thư viện thực hiện gửi nhận dữ liệu giữa Client và Server để có được sự hỗ trợ tốt nhất. Bên cạnh sử dụng Retrofit để thực hiện các phương thức HTTP, đề tài còn áp dụng thư viện RxAndroid hỗ trợ gửi request lên Server và lắng nghe kết quả trả về, Gson để đóng gói dữ liệu gửi đi trong request và parse kết quả trả về.

### Xử lý kết quả trả về từ Server

* GSON

GSON (tên đầy đủ là Google - GSON) là một thư viện Java được phát triển bởi Google, dùng để chuyển đổi từ một Java Object (POJO) sang dạng JSON và từ JSON String về Java Object tương ứng mà không cần đặt các Annotation trước mỗi class Object. Ngoài ra, GSON cũng hỗ trợ chuyển đổi các Object phức tạp, Java Generics,…

Bên cạnh GSON cũng còn rất nhiều các thư viện hữu ích khác hỗ trợ việc chuyển đổi này như: Jackson, JSON-Simple, FlexJSON, JSON-Lib,..

GSON được chọn để xây dựng đề tài thay vì các thư viện khác vì rất dễ để tìm những bài viết hướng dẫn kĩ càng về GSON trên các trang về lập trình, cùng với đó là lượng người đánh cao GSON là cực kì đông đảo. Bên cạnh đó, bản thân thư viện GSON cũng có tài liệu rất rõ ràng và chi tiết.

### Xây dựng giao diện

Trong quá trình thực hiện, đề tài con sử dụng các thư viện sau để phục vụ cho quá trình lập trình client.

* MPAndroidChart: vẽ biểu đồ tròn.
* Toasty: làm pop-up đơn giản, thay thế cho thư viện Toast mặc định.
* Fab – Floating action button: tạo nút hoặc tập các nút hành động.
* Lombok: hỗ trợ cho việc xây dựng và khởi tạo các model.

### Giải pháp công nghệ sử dụng ở phía Server

### Ngôn ngữ Python

### Tổng quan

Là một ngôn ngữ lập trình bậc cao, thông dịch, hướng đối tượng, đa mục đích và cũng là một ngôn ngữ lập trình động. Cú pháp của Python thường khá dễ học và ngôn ngữ này cũng mạnh mẽ và linh hoạt không kém các ngôn ngữ khác trong việc phát triển các ứng dụng.

Python được phát triển bởi Guido van Rossum vào cuối những năm 80 và đầu những năm 90 tại Viện toán-tin ở Hà Lan. Python kế thừa từ nhiều ngôn ngữ như ABC, Module-3, C, C++, Unix Shell, …

### Lý do sử dụng

* Cú pháp đơn giản, ngôn ngữ dễ đọc, mã nguồn dễ đọc và dễ tìm hiểu.
* Tốc độ xử lý của Python nhanh hơn so với PHP.
* Python được biên dịch và chạy trên tất cả nên tảng lớn hiện nay.
* Cộng đồng sử dụng và hỗ trợ lớn.

### Nghiên cứu và triển khai thư viện hỗ trợ truy xuất dữ liệu trên cơ sở dữ liệu Etcd

Hình 4‑2. Sơ đồ thư viện hỗ trợ truy xuất dữ liệu trên cơ sở dữ liệu Etcd

* Class Client: Là lớp bao bọc cho Etcd3Client để giữ thông tin kết nối và duy trì trạng thái kế nối.
* Class EtcdHub: Là container để quản lý Etcd context, quản lý kết nối của các node trong etcd cluster.
* Class EtcdRepo: Hỗ trợ các phương thức truy xuất.
* Thông tin các node được lưu trữ trong file yaml theo dạng:

config:

cluster:

nodes:

-

name: n1

ip: 192.168.136.128

port: 2379

-

name: n2

ip: 192.168.136.129

port: 2379

-

name: n3

ip: 192.168.136.130

port: 2379

version: ""

* Cách quản lý kết nối: thiết lập các kết nối bằng phương thức setupConnections() để quản lý các client và các kết nối của chúng trong client\_pools.
* Cách tạo kết nối: kết nối bằng phương thức tryAConnection(), nếu có bất kỳ kết nối nào bị mất, chúng sẽ được quản lý.
* Cách lấy metadata từ các model: trong mỗi class model phải xác định các thuộc tính cần thiết: context (các tiền tố bắt buộc của khóa), getters (hàm lấy giá trị thuộc tính), setters (hàm đặt giá trí thuộc tính).

![A close up of text on a white background

Description automatically generated](data:image/jpeg;base64,/9j/4AAQSkZJRgABAQEAYABgAAD/4RDuRXhpZgAATU0AKgAAAAgABAE7AAIAAAAMAAAISodpAAQAAAABAAAIVpydAAEAAAAYAAAQzuocAAcAAAgMAAAAPgAAAAAc6gAAAAgAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAExvbmcgTmd1eWVuAAAFkAMAAgAAABQAABCkkAQAAgAAABQAABC4kpEAAgAAAAMyMAAAkpIAAgAAAAMyMAAA6hwABwAACAwAAAiYAAAAABzqAAAACAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAMjAxOTowNjoyNCAyMjoxMzoxNwAyMDE5OjA2OjI0IDIyOjEzOjE3AAAATABvAG4AZwAgAE4AZwB1AHkAZQBuAAAA/+ELHmh0dHA6Ly9ucy5hZG9iZS5jb20veGFwLzEuMC8APD94cGFja2V0IGJlZ2luPSfvu78nIGlkPSdXNU0wTXBDZWhpSHpyZVN6TlRjemtjOWQnPz4NCjx4OnhtcG1ldGEgeG1sbnM6eD0iYWRvYmU6bnM6bWV0YS8iPjxyZGY6UkRGIHhtbG5zOnJkZj0iaHR0cDovL3d3dy53My5vcmcvMTk5OS8wMi8yMi1yZGYtc3ludGF4LW5zIyI+PHJkZjpEZXNjcmlwdGlvbiByZGY6YWJvdXQ9InV1aWQ6ZmFmNWJkZDUtYmEzZC0xMWRhLWFkMzEtZDMzZDc1MTgyZjFiIiB4bWxuczpkYz0iaHR0cDovL3B1cmwub3JnL2RjL2VsZW1lbnRzLzEuMS8iLz48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOnhtcD0iaHR0cDovL25zLmFkb2JlLmNvbS94YXAvMS4wLyI+PHhtcDpDcmVhdGVEYXRlPjIwMTktMDYtMjRUMjI6MTM6MTcuMjAxPC94bXA6Q3JlYXRlRGF0ZT48L3JkZjpEZXNjcmlwdGlvbj48cmRmOkRlc2NyaXB0aW9uIHJkZjphYm91dD0idXVpZDpmYWY1YmRkNS1iYTNkLTExZGEtYWQzMS1kMzNkNzUxODJmMWIiIHhtbG5zOmRjPSJodHRwOi8vcHVybC5vcmcvZGMvZWxlbWVudHMvMS4xLyI+PGRjOmNyZWF0b3I+PHJkZjpTZXEgeG1sbnM6cmRmPSJodHRwOi8vd3d3LnczLm9yZy8xOTk5LzAyLzIyLXJkZi1zeW50YXgtbnMjIj48cmRmOmxpPkxvbmcgTmd1eWVuPC9yZGY6bGk+PC9yZGY6U2VxPg0KCQkJPC9kYzpjcmVhdG9yPjwvcmRmOkRlc2NyaXB0aW9uPjwvcmRmOlJERj48L3g6eG1wbWV0YT4NCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgCiAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAKICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgICAgIAogICAgICAgICAgICAgICAgICAgICAgICAgICAgPD94cGFja2V0IGVuZD0ndyc/Pv/bAEMABwUFBgUEBwYFBggHBwgKEQsKCQkKFQ8QDBEYFRoZGBUYFxseJyEbHSUdFxgiLiIlKCkrLCsaIC8zLyoyJyorKv/bAEMBBwgICgkKFAsLFCocGBwqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKioqKv/AABEIAU8BbQMBIgACEQEDEQH/xAAfAAABBQEBAQEBAQAAAAAAAAAAAQIDBAUGBwgJCgv/xAC1EAACAQMDAgQDBQUEBAAAAX0BAgMABBEFEiExQQYTUWEHInEUMoGRoQgjQrHBFVLR8CQzYnKCCQoWFxgZGiUmJygpKjQ1Njc4OTpDREVGR0hJSlNUVVZXWFlaY2RlZmdoaWpzdHV2d3h5eoOEhYaHiImKkpOUlZaXmJmaoqOkpaanqKmqsrO0tba3uLm6wsPExcbHyMnK0tPU1dbX2Nna4eLj5OXm5+jp6vHy8/T19vf4+fr/xAAfAQADAQEBAQEBAQEBAAAAAAAAAQIDBAUGBwgJCgv/xAC1EQACAQIEBAMEBwUEBAABAncAAQIDEQQFITEGEkFRB2FxEyIygQgUQpGhscEJIzNS8BVictEKFiQ04SXxFxgZGiYnKCkqNTY3ODk6Q0RFRkdISUpTVFVWV1hZWmNkZWZnaGlqc3R1dnd4eXqCg4SFhoeIiYqSk5SVlpeYmZqio6Slpqeoqaqys7S1tre4ubrCw8TFxsfIycrS09TV1tfY2dri4+Tl5ufo6ery8/T19vf4+fr/2gAMAwEAAhEDEQA/APpGis++13TdNN2Ly58s2Vob24wjN5cI3fMcA/3WwOp2nA4q+pDKCOhGRQAtFFV5L+2h1CCxkkxc3CPJEm0/Mqbdxz0GNy9fWgCxRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAHFz6Rf6r4B8SzfZZP7U161uPKt5MIyKYykER3Y2nbgkHozNWEnhK9ay1KXSvC76TZTJaJNpEk8Kyaj5cpeYuY3ZCXQ7Mu+X5D4HJ9Roo2Y27q3r+P/DHmH/CK3S2RaPwiBoTamLhvC4kt+Y/IKZ8vf5H+txJs37eN33uKjn8E+IZtO0yPTY/7MigN6/2AXChUgkkjZLNnXJjDIrLuiyEPAJAwfU6KP6/r+vLbQX9fn/n+u5T0iOOLR7SOHTf7LjSJVWywg+zgD7mIyV4/2SRVyiim9XcSVlYKKKKQwooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooqC+tftthPa+dLB50bR+bCQHTIxlSQcH8KTvbQDE0/XJpNJ1LXLlpJLAysLGKG3Mh8pfkD4UFm3sGbPQKVPHJOPLqniPTdA8KPDcDVtR1S7UXCSyQpG4e2lk2CSNCBGrKCGUMxC9Wzz2cFlBbafHZW0Yit4ohFGicBVAwAPwrAtPBkT+F9N0bXbqS8/stl+y3Nm81jIgVDGpLRybt2xiCQQDk/KOlPq/kHTXz/ABWn3FSDxnqmoCGz0vQ7eTV91yLiC4v2igjEEgjYrKImZ8sy7fkHGc7cYMbePry6sDeaLoa3MVtYJfXy3N4IXiVt/wAkYCMJHHlv1KL935uTjWm8EaBNY2lqLWa3WzDiGS1vJoJhvOX3SxuHbcfmbcx3Hk5PNJd+BvDt75Ak08xJDAtsIra4kgjkhXpFIkbBZEGT8rhhy3HJydP6/r+tx9f6/r+tupkWXjae68a3uhWUJvLh3ingjlPkxwWphiZ3L7TuIaT7oycsM7V5Hb1k3HhfSLqczyWm2b7QlyJY5XjdJEQICrKQV+QbSBgEEg5BOa9rZ+LVvY2vdb0Wa1D5kih0aWN2X0Dm6YA++0/Snpt/X9f15Kdf6/r+vxMrS9bn02Hxfqmtxzh9PusyW0d8bqNQttG4EO6OPZuDDKnjcSc4NWvD+ra/d+MNTsdftbSySGwtZobe0ujcKC7zhmLmKM5OxRjkDbkHk1ZsvA+h2P8AaHlxXk66khS7jvNSublJgVCnKyyMM7VC5xnAxnFWdG8LaToF1cXOmwTC5ukRJ57i6luJJVQttDPIzE43kDnpgdAABW/BFPZ2/rVGvRRRSEFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABWffa7pumm7F5c+WbK0N7cYRm8uEbvmOAf7rYHU7TgcVoE4Uk9vQZri59Iv8AVfAPiWb7LJ/amvWtx5VvJhGRTGUgiO7G07cEg9GZqRUUm0mdmpDKCOhGRS15cnhK9ay1KXSvC76TZTJaJNpEk8Kyaj5cpeYuY3ZCXQ7Mu+X5D4HJd/wit0tkWj8IgaE2pi4bwuJLfmPyCmfL3+R/rcSbN+3jd97iq/r+v69bEK9tf6/r/hrno8l/bQ6hBYySYubhHkiTafmVNu456DG5evrVivLJ/BPiGbTtMj02P+zIoDev9gFwoVIJJI2SzZ1yYwyKy7oshDwCQMHrIfGHhLQrWDTr/U9I8OzQRIDpV1fW8MlqMAhCiuVHGOhIx0pdP6/r+vS7/r+v6/W3T0VxK6Yk/jt9WvPD39vQXLQS6ZqyNbyx6fHsAIXzHDr8wL7ow27ePSsG38H6vL8T49XudEMMTahci8nSOxSCe1aKRYx8i/aJCf3e8SNjdyAQBtFfb+v6/rUOlz0ux1C21KB5rKTzI0mkgY7SMPG5Rxz6MpGenHFWa850TwfJo+keI9M0Tw3BoepTi8FnrNulvGkiySO0KBkJkXaGUYZAF28ZwK5260JfC3w812eSz1PSFuXs0li1BNOS3f8AfDd8lqUQhg21zKyblIBYDkF/0HbW3me0UV4v4X03T9W8NGTR/DdvqdomrmbVdOsobOG3vAYSqCJUnkhdUJRijynDAng7RXS6d4Ia6TR7fWdFhOlQy3sh02d0ljto5GUwwsmSrbcfdG5VIGD8qmm9P69P6+RN/wCvvPQ6K8+1bSbKHxRY6D4fkhAurAWOpWiS75IbRW3o7AkkLt82IE95Rj7tdTPYKPFem3EekB0t7OeJb5Z9otgTH+7EX8W7bnP8Oz3pLX+v6/rzH/X9f159TYrIsPFOj6nPLDZXZkeGN5XBhdcKkjRMclR/GjD3xkcc1r1Ug0rT7V3a2sLWFnUoxjhVSylixBwOQWZj9ST3pO/QehX0bxFpfiCN30i5M6okUjExOmFkQOn3gOqkHHbPOK06rWenWWnqVsLO3tQwVSIYlTIVQqjgdlAA9AMVZqna+hKvbUKKKKQwooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKK47W9d1JdI8W6lptx5VvpNjLHakRq265SNneTJByFO1MdMq+c9gaV3Y7GiuNPjXUrQz2uqaHHFqHl2z2kMV75iymeQxosj7AI2DDLYDgD7pbpTx4x1MSHTH0S3GvfaxbLbC/JtmBiM3med5W4LsBGPLzu4xj5qCU7nX0V59ffESWxvNJa9tJIJJ5LqylsISJRJdpJHHGolIAClmOGbYPmGQDgV3lo1w9nE19FHDcFAZY4pTIit3AYqpYe+B9KN1cfWxLRXJQ6v4nf4k32mfYtObSobW2lBN8wkRXkmBkC+Ryx2AbC2BtGG+Y4t2nimS71L+zhYql7HezQTxGf/VRRqGEudvO5ZITt45kxngmjpcHo7f1tc6KiuEsPiNcnS7bVNc0MWNjeaXLqVube7+0SlI1VmV02KFJDjbhmz32nitPwn4svvELXsd9odxp726o8btFcpHMG3fKGnghO4beQFIwy8nkAeg7WOooqrplxd3WlW0+pWX2C7kjVprXzRL5LEcrvHDY9RVDxFa6/dR248OX8Nm6+b5pmUHfmFxH1VukhRj04U9ehHoJamzRXEeIPHN74cvINMg0i51q9htI57w28FwxO7IGzyYJFLEo/DmMdOeu1reL9Zstd1oy6cl1pFnqNtbmVpxHLCssUHypGEO/DyFm3MvDYGcYDtd2X9a2DodzRXJR+NLo3sM82kxpolzeS2cN4t3um3pvBZodgAQmNgCHLfdyoycUbnxvqraHPcy6XHp6Xuk3F/pdxHdCd/kjDjzYygCNhlOAXHBBPTM3Vm+w0rtLud3RXN+G/Fi67o82rmJYNIhQ7byZyjysmRKTEVBjVWUj5juyG+UDBNQ+I9astE0idtPF9f65eskFtNMLdbVGjklQM21j8qRgNwTksQDwtU007Ep3V/wCtN/uOvorhr74g3lro9vNDo8Muovc3FtLYia5kOYW2u0fk20jOmcfMUQDcucE4psnxEvZ7MXmkaAtxax6LBrM73F75LJFJ5h8tVCNufEZwDhTzll4yulyrP+vW35nd0VxNx8Q5B4nTTdO0K8vbVZoYbi4S3uS0bSBWyNsDRbVWRS26VCMNwcDdp6f4h1fUtXljt9Di/syC8ks5bo3w80FM/OItmCnQffDZJ+UgZJ/X5f5oXS50dFc/FrM0l9rd808UelaZGYU80hI3lQFpXZ8EhVyE46FX4PbC0n4lT3sN2L3RGtZ7eayULm4RJEuZ/JDAzwRP8pDHhCDgANnOBa6L+r7Bsd7RXL3ni64XUrrTNN0yO5v01AWNuk10Yo5P9GW4Z3cIxQBWIwFYkgepxhTazrt94Ltp9QGx2142t3JZaiYZLdRqHlIsbCH96vRW3BCyg9zTSvbzt+I2rL+vP/I9FormNe8U3+nX15DpOkRagmm2q3d80t55DKjbsLENjB3xGxwxQfd+bk4kkh8RXutQX2nalDHo8ptpBbug3lAshlH3CcsWh/i/hPTutxPQ6OiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAAjKkA49x2rLh8PWEfhl9ClR57KaF4Z/Mch5g+d7My4O5izEkY5J6VqVn32u6bppuxeXPlmytDe3GEZvLhG75jgH+62B1O04HFA1e+hQi8EaDHp93ZtazzpebPOlubyaac7DlMTO5kXaeVww2kkjBJNB8EaCdN+xm2uMef8AaftP22f7T5uNu/7Rv83dt+XO77vy9OK3lIZQR0IyKWgS20MVPCGgrZR2h06OSCOKWHZKzSb1lIMm/cTvLEAlmyScnPJqCTSvEtqVt9C1rTbewiRUijv9PnvJgAOd0xulLnOeSM/XrWzJf20OoQWMkmLm4R5Ik2n5lTbuOegxuXr61YoAyZPDtndapaareeb/AGnbxLG01rcTW6ShSTho1fDruLEK+7GT6mo9K0F7PV9Q1W/uIbi+vQsZeC3MKJEmdi7SzEtzy2ecLwAAK0U1C2k1SbTkkzdQwpPJHtPyo5YKc9OTG3HXj6VGmpb9cm037Hdr5UCT/amixA+5mGxXzy425IxwCPWh66f15h6+X/AMPw58P9H0HR4rSZJNSlFiLKaa8mkmV48AOqJI7CJGwMomAcDrgVq6R4dsNEglhsXvnjlAUrd6hPchQBgBfNdto5/hx+lWLfVbS6sZ7y2d5oYJJY3KROzbomZHUKBliGUjgHOOM0mkaxZ67pq32mvI8DO8f72B4XVkYoylHAZSGUjBA6U3re43e93/AEx+mabaaNpVtpumxeTaWsaxQx7i21QMAZJJP4mrVFFK99WIyNU8L6Xq+oR310t1Fdxp5fnWd9PauyZyFYxOu8AkkBsgZOOpqKXwbok+uPq01vPJcvMk7o15MYGlRVVHMG/yywCrhiuQVB6gGp9V8TaTolylvqd0YZXiMyqIXfK+YkeflB/ikQY68+gOC48TaTa6wNKnuit6XiQRiFzzLvKDcBjnyn78Y5xkZFpa39a/5/iD63IYPB2h22rnUorR/PMjzLG1zK0Ecj53OkJYxo5y2WVQTubn5jmK08C+HrJJ0gspCk8D22yW7mkWGJ/vRxBnIhU8fLHtHC+gx0FFA7u9zEufCOlTreKkctut95Yu1hlYJOEI4ZDlcsqhGbG4rgE8DC+I/DUPiT+zUuLm4t4rG7+0kW00kLv+6kj2iSNlZP8AWZyDzjHetqigS02Ofn8DeH7izsrVrOWOOyEixeTdzRMRIQZA7K4MgcgFt5bceTk1Sb4caJJqSSSrOdPh0+3sIdOS5mSLy4mkIEih8TKQ4G2QMMA9dxrraKd+odLGPL4W0qTXP7WVLqC7JRnNtfTwRylRhTJGjhJCAAPmB4AHQAVftLC3sUnWzTyhPK80hDE5djlm5z37dKs0UgM220Cxg8Otojo1xZyRPFMJjlpw+d5YjGSxZiSMck1nQ+AfDsFreQLaXEgvkijuJZr6eWVxExaP94zlwVJyCCCOMHgY6OigDnU8B+Ho7F7WO1uEElz9raZL+cT+dsEZk84P5gYqMEhuQTnOTTZvAHh2bR7fS1trq3sbaZp4oLTULi3UO0nmbj5ci5If5hnO09MV0lFAGDf+C9D1OSKS+t7iRo4Vt2IvZ1+0RDok2HHnLyeJN33m9TndVQihVAVQMAAcAUtFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFAAThST29BmuLn0i/1XwD4lm+yyf2pr1rceVbyYRkUxlIIjuxtO3BIPRmau0opDTs010PLk8JXrWWpS6V4XfSbKZLRJtIknhWTUfLlLzFzG7IS6HZl3y/IfA5Lv8AhFbpbItH4RA0JtTFw3hcSW/MfkFM+Xv8j/W4k2b9vG773Fen0U/6/r+vQlK39fL+vxPLJ/BPiGbTtMj02P8AsyKA3r/YBcKFSCSSNks2dcmMMisu6LIQ8AkDB6yHxh4S0K1g06/1PSPDs0ESA6VdX1vDJajAIQorlRxjoSMdK6eijpb+v6/ruP8Ar+v6/Q4T/hGtNX4kr4htvClnqdvqFtbtFqdtFbE28yvIzTEswY7leP5k3E7fYVLpenSQfEbW5h4Qms9O1S2jhmu8WgjuJEaYtI6rKXYOJEAJXJ/iAxXbUUeQNt387fgcN4M8PQeE5NWjtvBiWk6zXMsV7Zx2qC8iaZnjhUhw4IUquHCqNvXGKorpniHUfh5rGl/2Nf6XdSahNciOWa0ZrmGS7aVo1IeRNxjJUiQbSTg5XJr0eijdWY76387nBaF4XXTfh3rWnJoepTpdiUjSNQls7bzcxhdiG0xFErY6jByWY9cnuYF2W0S7PL2oBsznbx0z3qSim3cRVutMsL2QSXtlb3DquwNLErkLuDYyR03Kp+qg9qJNMsJbwXctlbvcgqwmaJS4K7gp3YzxvbHpuPqatUUgCiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArjtb13Ul0jxbqWm3HlW+k2MsdqRGrbrlI2d5MkHIU7Ux0yr5z27EjKkA49x2rLh8PWEfhl9ClR57KaF4Z/Mch5g+d7My4O5izEkY5J6UiotJq5gnxrqVoZ7XVNDji1Dy7Z7SGK98xZTPIY0WR9gEbBhlsBwB90t0p48Y6mJDpj6JbjXvtYtlthfk2zAxGbzPO8rcF2AjHl53cYx81aEXgjQY9Pu7NrWedLzZ50tzeTTTnYcpiZ3Mi7TyuGG0kkYJJoPgjQTpv2M21xjz/tP2n7bP8AafNxt3/aN/m7tvy53fd+XpxVf1/X9fcQtF/X9f11OdvviJLY3mkte2kkEk8l1ZS2EJEoku0kjjjUSkABSzHDNsHzDIBwK7y0a4ezia+ijhuCgMscUpkRW7gMVUsPfA+lZSeENBWyjtDp0ckEcUsOyVmk3rKQZN+4neWIBLNkk5OeTUEmleJbUrb6FrWm29hEipFHf6fPeTAAc7pjdKXOc8kZ+vWl0/r+v6+9/wBf1/X/AAM3W/H8mjeLItLGmrd2huLe3muIGuGeB5mVVD4gMK8upw0wO0g45AN5PFsk0NkYbBTJeardaaivPgKYfPw5O08N5HTHG7vjmSfwRo1/qK6lqdu098ZIZ5WjuJo4Xni27JfJ8woGGwYJBOBgkinxeCtBh1xNXjtJftkc73MZN1KY4pXDB2SMtsUsGbOFGScnnmmtrP8Arb/gj0tp/Wj/AFsc1aeJfFjfD/xLqepQWFvLYx6iba4trszOskUkqquxoUXau3AYkltoJAycaY8aahbR3VvqeixRaki2ptraK98xZftEjRxh32ARkMjbsBwAMgt0rVHhDRh/aaiG48rVUkS7t/ts3kuJPvlY9+xC2SSygHJJzzU1/wCGtJ1Q3JvrTzGuoYoJGEjq22NmePaQQVZWdiGXDA4OeBgB21t5/pb9TFfxlqcMb2txocI1dL+OyNvHf7oCZIy6OJTGDtx1ygIwcBuM666hrg0VJ5tCj/tD7QsclpFfKyBDIFaRZCq5AQl8FVJxjFFn4V0ewhjjgtpGMdyLvzZriSWV5QNod5HYs5C8DcTgADoBV3U9Ls9YtFttRh86FZY5gu4r88bh0OQQeGUH8OaS8+/4af8AB/4HRf1+f/ALdYF5Z+JZPEiz2epQRaV5luWtyo37VEvmjOwnLExY5/hPTvv0UB0sYFtZ+JV8SvPc6lA+kmeVlt1UbhGY4xGudmeHErH5v4hyeg36KKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACs++13TdNN2Ly58s2Vob24wjN5cI3fMcA/wB1sDqdpwOK0CcKSe3oM1xc+kX+q+AfEs32WT+1NetbjyreTCMimMpBEd2Np24JB6MzUiopNpM7NSGUEdCMilry5PCV61lqUuleF30mymS0SbSJJ4Vk1Hy5S8xcxuyEuh2Zd8vyHwOS7/hFbpbItH4RA0JtTFw3hcSW/MfkFM+Xv8j/AFuJNm/bxu+9xVf1/X9etiFe2v8AX9f8Nc9Hkv7aHUILGSTFzcI8kSbT8ypt3HPQY3L19asV5ZP4J8Qzadpkemx/2ZFAb1/sAuFCpBJJGyWbOuTGGRWXdFkIeASBg9ZD4w8JaFawadf6npHh2aCJAdKur63hktRgEIUVyo4x0JGOlLp/X9f16Xf9f1/X626egnA5ryiTSU8QeNdbm0vRPtVydWspoNfEsPl2kSwWztsO8yAsoI+RMNuUM2OmtZ6Vrya1ptk+iSpaWWu3t/JqBuIvLeOZbkpsUMXJzMoOVXB6ZHIHflv1/wCGG7Wv/Wz/AMrfM7a11S0vdFi1W0d5rOaAXEbpExZ0K7gQmNxJHbGfao9K1mz1vSxqGmtLJAWdMSQPE4ZGKspRwGBDKRggdK8/8C+GL3Qm026XwrPpU+naU8F83nW5l1SYhNoBSRtwXY2DIybdygADONfRILxvB+q2ev8Ag28nWS9uJxp9w1nL9qSa4eRQB5xTKhgTuI5HGaJ6J2Cy/E6zSdQ/tXSba/8Asl1ZfaIw/wBnvIvLmjz2dcnB9quV5zb6Nq0/wx8P6bf+Hbpv7La3iv8AR5Jrcm+iSPaVBEhjZN5VtrsuQhBHIz0EHiXwt4Z020sdRuNL8Jkxb4tLvLq2t2iTcRwqOVwSD90kfjmnKybt3Eb1/fW+l6bc399J5VtaxNNNJtLbUUZJwOTwO1TqQygjoRkV5h410ZvF8Wo3emaFZ+LLa/0rydIvBcW7w2co8wOwLt8pJKYZASSgDFQAaj8R+E9b1Hx/Dqdto7h7a8smgvrdLFF+zo0Zm3yMDcl/9YNqlVK4HPIYSu0n3B2tp/Wx6XbahbXlzdwW8m+WylEM67SNjlFcDnr8rqePWrNcNofhOz0Px3q9xbeEbaD7ZN59prFtBbKsCmBEZD8wkBLq5wqkHfknk1n/AA78IXmh6495qtpq0OoG2aO6upP7P+zXkhZSX3Qqs0hJGVaYZALZ5Jylrb0B6HpNVNR1O00mCOa/lMUcs8duhCM2ZJGCIMAHqxAz0Heqfhuz+w6bPH/Zf9l7r24k8n7R52/dKzebu7b87tv8O7HatK4tbe7RUuoI5lR1kVZEDBWU5VhnuCAQexo7B3MmTxjoUOlWepSXpFperI1vJ5EnzhEaRjjbkYVGPIGccdRWvbzx3NtFPCS0cqB0JBGQRkcHkVWbRtLe1htX020a3gDLFEYFKRhlKsFGMDKsQcdQSO9XERY41SNQiKAFVRgAegoAWiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAoorjtb13Ul0jxbqWm3HlW+k2MsdqRGrbrlI2d5MkHIU7Ux0yr5z2BpXdjsaK40+NdStDPa6poccWoeXbPaQxXvmLKZ5DGiyPsAjYMMtgOAPululPHjHUxIdMfRLca99rFstsL8m2YGIzeZ53lbguwEY8vO7jGPmoJTudfRXn198RJbG80lr20kgknkurKWwhIlEl2kkccaiUgAKWY4Ztg+YZAOBXeWjXD2cTX0UcNwUBljilMiK3cBiqlh74H0o3Vx9bEtFc5N4h1eTxDd6fpWhxXcFjJEtzNJfCJysihsxpsIYqCSQzKOBgkkgZg+IE7eNBocelxzQzXEtrb3sUlwUMyRs+13a3EQ+4ykJI7AjGDg4FrsHmdtRXI+DdX8Uappt9LrFjpuYrq7igeK/dizRzuixsPIUBQF27+SQASuSRVTTPEF5pHgO41SW0uL69Ory2zWkuoebiRr0wbUlZF+QE5UFRgYHvR1Hb87Hc0VxF/8QJ9J0q4fVrDT7O+g1JdPk83UytnGzQiYO9w0QKqVIH+r+8QO+a09H8Ratq3hK31ePRYHuLicKlvb3+9Gh83aZhIyLuGzMgAHzDGDzRv/AF3E9DpKKKwLyz8SyeJFns9Sgi0rzLctblRv2qJfNGdhOWJixz/CencDob9FYFtZ+JV8SvPc6lA+kmeVlt1UbhGY4xGudmeHErH5v4hyeg36ACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAAjKkA49x2rLh8PWEfhl9ClR57KaF4Z/Mch5g+d7My4O5izEkY5J6VqVn32u6bppuxeXPlmytDe3GEZvLhG75jgH+62B1O04HFA1e+hQi8EaDHp93ZtazzpebPOlubyaac7DlMTO5kXaeVww2kkjBJNB8EaCdN+xm2uMef8AaftP22f7T5uNu/7Rv83dt+XO77vy9OK3lIZQR0IyKWgS20MVPCGgrZR2h06OSCOKWHZKzSb1lIMm/cTvLEAlmyScnPJqCTSvEtqVt9C1rTbewiRUijv9PnvJgAOd0xulLnOeSM/XrWzJf20OoQWMkmLm4R5Ik2n5lTbuOegxuXr61YoAqWWnx2ss1ywQ3l0EN1KgZVkZVC5CknaMDpn8T1rMh8E6Db64mrRWkv2uO4kuYy13M0cUsgYOyRl9ilt7ZwBknJ5xW9RR1uHSxkW/hfTLS4v5rUXkJ1Df56R386xgucsyIH2xsTk7kCtkk5yagsPBWiabYy2dvFePby3CXTJc6jcT/vUk80OPMdsHf8xxjceua1bHULbUoHmspPMjSaSBjtIw8blHHPoykZ6ccVZo2D+vmcr4o8Ff21GH0m+Gl3TXwvJ5iJm81hD5PWOWN0+UL9x1zjBByasaZ4Sji8MxaRr15NrGy4W5MkssvDrIJECl5Hk2qwGAzseOTjitm21C2vLm7gt5N8tlKIZ12kbHKK4HPX5XU8etF1qFtZT2kNzJskvJjBANpO99jPjjp8qMcnjihabf12B67lmiqV/qX2C4sYvsd3c/bJ/I328W9YPlZt8hz8qfLjPqQO9XaACiiigAoqrqepWmj6Xc6jqMpitbWMyzSBGbaoGScKCT9AKpnxPpAtGuTdN5K3osC3kv/ry4jCY25+8QM9PfFHkHma1FVdM1K01jTYb/AE6Uy2067o3KMu4ZxnDAGrVABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAE4Uk9vQZri59Iv9V8A+JZvssn9qa9a3HlW8mEZFMZSCI7sbTtwSD0ZmrtKKQ07NNdDy5PCV61lqUuleF30mymS0SbSJJ4Vk1Hy5S8xcxuyEuh2Zd8vyHwOS7/hFbpbItH4RA0JtTFw3hcSW/MfkFM+Xv8AI/1uJNm/bxu+9xXp9FP+v6/r0JSt/Xy/r8TyyfwT4hm07TI9Nj/syKA3r/YBcKFSCSSNks2dcmMMisu6LIQ8AkDB6yHxh4S0K1g06/1PSPDs0ESA6VdX1vDJajAIQorlRxjoSMdK6eijpb+v6/ruP+v6/r9DkIPCema14m1HWdTsLe9jle2m0273hygWNTviYHKZIGSMbgB1AFc1b+D9Xl+J8er3OiGGJtQuReTpHYpBPatFIsY+RftEhP7veJGxu5AIA2+qUUKyd/X8Q1tY850TwfJo+keI9M0Tw3BoepTi8FnrNulvGkiySO0KBkJkXaGUYZAF28ZwKztK8FXVl4b1KOz0XV4TcS2rXWmXZ02GO8jSTdKiC12oWZSVJk27htUnHT1eij/gfgD1/P7zgPDNzpHgdNWbXodJ8GWupah59hZT3dvCDGIIVJAVtobcDuAzgnqQQSvinT9C8axaPrmlaVpfjCKwviJltzbTtJEYpFKK0jBDh3RipYdM9cV31FG9r9P0GcN4hsJR4h8L6hpfgq4uJNNcO9xbizRreEwyx+QC0qnhnUlVymBwSRisHXfh5eaj4o1LVBo8c0krzyQzNIm4n/RPJ6txgpOR/dOTwWBPq9FNOzv6/iJ6q39df8zzXRPB99b/ABGk1bVbbVmuReTyrqMQ0/7PLC27y43cKLoqFZV2HKhlXHABHZ6PZ/ZtW1uX+y/sX2m7WT7R9o8z7ZiJF8zb/wAs8Y2bf9nPeteiktFbysD1bZFc20F5ayW15DHPBKpWSKVAyuD1BB4IqL+zLDyjF9htvLM32gp5K483du8zGPvbuc9c81aooAitraCztkt7SGOCCMYSKJAqqPQAcCpaKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigArjtb13Ul0jxbqWm3HlW+k2MsdqRGrbrlI2d5MkHIU7Ux0yr5z27EjKkA49x2rLh8PWEfhl9ClR57KaF4Z/Mch5g+d7My4O5izEkY5J6UiotJq5gnxrqVoZ7XVNDji1Dy7Z7SGK98xZTPIY0WR9gEbBhlsBwB90t0p48Y6mJDpj6JbjXvtYtlthfk2zAxGbzPO8rcF2AjHl53cYx81aEXgjQY9Pu7NrWedLzZ50tzeTTTnYcpiZ3Mi7TyuGG0kkYJJoPgjQTpv2M21xjz/tP2n7bP9p83G3f9o3+bu2/Lnd935enFV/X9f19xC0X9f1/XU52++IktjeaS17aSQSTyXVlLYQkSiS7SSOONRKQAFLMcM2wfMMgHArvLRrh7OJr6KOG4KAyxxSmRFbuAxVSw98D6VlJ4Q0FbKO0OnRyQRxSw7JWaTespBk37id5YgEs2STk55NQSaV4ltStvoWtabb2ESKkUd/p895MABzumN0pc5zyRn69aXT+v6/r73/X9f1/wMrxD4q1lLiVNEs7dLKz1Wzsbq9mucS7pJod6pD5bBl2yhdxdTkkgcAmceNL83Mc40aI6Tc3UtnbXIvf3zSpvGXi2YVC0bAEOx5UlRk4vT+CdFv79NQ1O1M975kVxIY55o4Wnj27ZRCHKhxsA3HLbRtJIqa38H6Ja6wdThtJBP5rTKjXMrQxyNndIkJby0c5OWVQTubn5jk15bdf+G/4Pp5jv/X9f16GA/xPtX0qW7s7B5ZINPkvpInmCgqkCy4VgDuG5thPGCrcHHOnY6tqmtyajpN/AdBvY7aG5imsrlLl1jkLhc+ZFsDgxMCMOOeCetTQ+BPDlvafZodO2w/YZNP2efIf9HkYsyfe7k9eo6AgVfu/D+n3r3Mkq3Ect1BHbyy211LA5jjZmQBkYFcF25BBOcHIolZp2EtEv67f8EwdM8T6kfA/heZYE1PWdYt4tomlFvGz+SZHd2VG2jCtwqHkgYA5FbUPiDeWmk28sWjwyak1xPbzWHnXErBojh2j8i3lZ0yR8xRANy5wTitiLwNoMOhQaRHDeC0tpBJbE6jcGW3YDaPLlMnmRjbkYVgMEjoTRN4F8Pz2lpbtZzIlp5nlmK8mjdvMIMm91cNJvIBbeW3Hk5pz1baHp/Xz/wCAc9rXj7VL3wfd6h4T06JXj0NdTlnvLnymthJG7IEQRuJHGxiQ21egzycdp9qvRqVpAlj5lpLA7zXnnKPKcbdqbOp3ZY5HA2+9Y938O/DF7ZQ2dxpzm1htFshCl1MiPCoIRHVXAk25JUtkqTkYPNbT6TZSatbak8Obu1heCGTe3yo5UsMZwc7F5IzxTdr6ef8AwP6/MXRf12/4JcrndJsfFMN1cNq2qW9xE0MohVFA2yGaRkJwg4ERjXvyp69T0VFQ1cZh+GrTxDaQyDxNqMN7IYoAhiUDDiJRKeEX70m4j2I4HStyiiqbu7krRWCiiikMKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACs++13TdNN2Ly58s2Vob24wjN5cI3fMcA/3WwOp2nA4rQJwpJ7egzXFz6Rf6r4B8SzfZZP7U161uPKt5MIyKYykER3Y2nbgkHozNSKik2kzs1IZQR0IyKWvLk8JXrWWpS6V4XfSbKZLRJtIknhWTUfLlLzFzG7IS6HZl3y/IfA5Lv8AhFbpbItH4RA0JtTFw3hcSW/MfkFM+Xv8j/W4k2b9vG773FV/X9f162IV7a/1/X/DXPR5L+2h1CCxkkxc3CPJEm0/Mqbdxz0GNy9fWrFeWT+CfEM2naZHpsf9mRQG9f7ALhQqQSSRslmzrkxhkVl3RZCHgEgYPWQ+MPCWhWsGnX+p6R4dmgiQHSrq+t4ZLUYBCFFcqOMdCRjpS6f1/X9el3/X9f1+tunoryiTSU8QeNdbm0vRPtVydWspoNfEsPl2kSwWztsO8yAsoI+RMNuUM2Oj7bwtrUfxQj19tHktQmoXLXN1CllHFNbGKRY8GNftMjE+WWEhxu5CnA2myu+1/wAhtaaHqlFeA+AVtNZmu00gWn/CRTaNPFa6lZw2wdXYAmS4ljnll8wnbh3jiP3srk7R6F8PPDH9gxak39matYG4jjD216unxxOw3ZZFswBnnBZwCRt9OB6L5C0v8/8AI7ys7V9f03QUhbVbgwLN5mwiNnzsjaRvug4wiMefTHUgUzwzafYfCum2v9m/2V5NsifYfP8AP+z4H3PM/ix0z3q5d6fZ36qL60guQm7aJog+3cpVsZHdSQfUEiiStdIF5mdfeLdF07yPtl20f2i3FzEPIkbdGXRAeF4+aVBg889ODjZqpPpOnXJQ3On2spjQRp5kKttUMrbRkcDKqceqg9hVunoJX6hRRRSGFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFcdreu6kukeLdS0248q30mxljtSI1bdcpGzvJkg5CnamOmVfOewNK7sdjRXGnxrqVoZ7XVNDji1Dy7Z7SGK98xZTPIY0WR9gEbBhlsBwB90t0p48Y6mJDpj6JbjXvtYtlthfk2zAxGbzPO8rcF2AjHl53cYx81BKdzr6K8+vviJLY3mkte2kkEk8l1ZS2EJEoku0kjjjUSkABSzHDNsHzDIBwK7y0a4ezia+ijhuCgMscUpkRW7gMVUsPfA+lG6uPrYlorifEPirWUuJU0Szt0srPVbOxur2a5xLukmh3qkPlsGXbKF3F1OSSBwCX6d4+m1Pxd/Zltod22nm4lthfiC5wrx7gzMTAIQm5CoIlY8r8oydotVf8Aro/1B6K52dFcdp3jzzY9Xn1e0ttPj0y3kumtPtTteiJM5Z4HiTaCBwys6k9GI5q5L4j1jT/DNzq+saHbwlPKMFrbX5lkcOwXDlo0VGG4cAsM5+bvR0uHWx0tFcZ4l1PxFH4A1ye90mGznt4JGZrLWGUmERlmkim8jcsg5ABQDIzmr+qa3qWm6hpWj6NpkepXN3aSzebeXxgVFiMQJdhG5JPmdlPPbBJB/X6j7fP8DpKK4KLxb4k1XX9AfRtMsf7O1LTJrpobu/aJ9ytCCTtgfBXeQADhtxJxgZ17me8PxN0+2nEkVp/Z9zJAYb47JirQhvNgMYGRvG1g5/i45p22+f4X/wAhHTUVylv4wvZtRt3fSI10e7vZLG3uxeZmMiFxuaHYAqFo2AIcn7pKjJwzwt4yv9en00ajo0WnQ6rp7X1mUvfOcqpjDB18tQv+tUjDNkdQp4pLXb+uv5AddRVTTbi8uraR9RsfsMqzSIsfnCTciuQj5HTcoDY6jOD0qLWYdTntYV0a5jtpluoWldwDuhDgyKMqeSmQPr1HWgO5oUVy8+n+MW0DTooNXtV1ONJfts5RdkrGJxHgeX0EhjPQcKevQ9HaLMlnCl1IJJ1jUSOB95scnoO/sKAJaKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAAjKkA49x2rLh8PWEfhl9ClR57KaF4Z/Mch5g+d7My4O5izEkY5J6VqVn32u6bppuxeXPlmytDe3GEZvLhG75jgH+62B1O04HFA1e+hQi8EaDHp93ZtazzpebPOlubyaac7DlMTO5kXaeVww2kkjBJNB8EaCdN+xm2uMef8AaftP22f7T5uNu/7Rv83dt+XO77vy9OK3lIZQR0IyKWgS20MVPCGgrZR2h06OSCOKWHZKzSb1lIMm/cTvLEAlmyScnPJqCTSvEtqVt9C1rTbewiRUijv9PnvJgAOd0xulLnOeSM/XrWzJf20OoQWMkmLm4R5Ik2n5lTbuOegxuXr61YoA56fwTot/fpqGp2pnvfMiuJDHPNHC08e3bKIQ5UONgG45baNpJFWbXwtpdlrEmp2a3cE0sjSPFHfTrAzt95jAH8vJ6k7ck89eatzal5Os2un/AGO7k+0xSSfaY4swxbNvyu2eGO7gd8H0q7QHkY1h4T0nTpppY0url54jC5v7+e7wh+8o8522g4GQMZwM9BTbTwjpNlpkunRreS2kpQmK51C4nC7CCoXzHYoAQOFwOK26KAM7XNBsfEelvp+qi4a1kz5iQXctvvBBBVjGykqQTlScH0rIvvAWn393phlu9SW10+2ngWNdSuVlk81ozlpxKHYDyyNrEg5H90V1FFA7sx7zwpo97HYIbaS2GnLstPsNzJamJMAFAYmU7PlX5enyjjim3PhPS7vxFFrkxv8A7fDxGyancpGo+XKiISBNp2LkbcNgZBqXVfE2k6Jcpb6ndGGV4jMqiF3yvmJHn5Qf4pEGOvPoDguPE2k2usDSp7orel4kEYhc8y7yg3AY58p+/GOcZGS+qYtkQ2/g/RLXWDqcNpIJ/MaZUa5laGORs7pEhLeWjnJyyqCdzc/Mc2rPQNN086f9ktvL/s21a0tf3jHy4jsyvJ5/1acnJ468mtGijYCppul2ekW0kGnQ+TFJNJOy7i2Xkcu55J6sxOOnpVuiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKAAnCknt6DNcXPpF/qvgHxLN9lk/tTXrW48q3kwjIpjKQRHdjaduCQejM1dpRSGnZproeXJ4SvWstSl0rwu+k2UyWiTaRJPCsmo+XKXmLmN2Ql0OzLvl+Q+ByXf8ACK3S2RaPwiBoTamLhvC4kt+Y/IKZ8vf5H+txJs37eN33uK9Pop/1/X9ehKVv6+X9fieWT+CfEM2naZHpsf8AZkUBvX+wC4UKkEkkbJZs65MYZFZd0WQh4BIGD1kPjDwloVrBp1/qekeHZoIkB0q6vreGS1GAQhRXKjjHQkY6V09FHS39f1/Xcf8AX9f1+hxEsLt8T7DXNK8Lyz20ljJbz6xbNaBZxIYGjcnzRI6KEYfdzz8oINc/pXw8vLPxPHqraRGk63cU3nmRC2ftV0zt97OfKeEZ67cKOhA9Xoo2tbp/ncHqmn1VjxOz8N63oEOt61d6bPoyvoV6L6WAWcEf2glGVohbL5hUDeVeUsw5zjJLVvCtha65oHiGDwxZWkWpyxWombSLaxhhlgSTLwgxz3EfmOu/IlIVsrlcZNe60ULa39bt/qNu7v53/I4LQvC66b8O9a05ND1KdLsSkaRqEtnbebmMLsQ2mIolbHUYOSzHrk9zAuy2iXZ5e1ANmc7eOme9SUU27iKt1plheyCS9sre4dV2BpYlchdwbGSOm5VP1UHtRJplhLeC7lsrd7kFWEzRKXBXcFO7GeN7Y9Nx9TVqikAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFcdreu6kukeLdS0248q30mxljtSI1bdcpGzvJkg5CnamOmVfOe3YkZUgHHuO1ZcPh6wj8MvoUqPPZTQvDP5jkPMHzvZmXB3MWYkjHJPSkVFpNXME+NdStDPa6poccWoeXbPaQxXvmLKZ5DGiyPsAjYMMtgOAPululPHjHUxIdMfRLca99rFstsL8m2YGIzeZ53lbguwEY8vO7jGPmrQi8EaDHp93ZtazzpebPOlubyaac7DlMTO5kXaeVww2kkjBJNB8EaCdN+xm2uMef9p+0/bZ/tPm427/ALRv83dt+XO77vy9OKr+v6/r7iFov6/r+upzt98RJbG80lr20kgknkurKWwhIlEl2kkccaiUgAKWY4Ztg+YZAOBXeWjXD2cTX0UcNwUBljilMiK3cBiqlh74H0rKTwhoK2UdodOjkgjilh2Ss0m9ZSDJv3E7yxAJZsknJzyagk0rxLalbfQta023sIkVIo7/AE+e8mAA53TG6Uuc55Iz9etLp/X9f197/r+v6/4GPres69deO7bRLCz1yxtI4TLJc2P2BhNl1XcfOZiIwM5AUOSeBxWnpc95L8RdZhvhJEsNjbtBHHfGWF42kmAfyjGvlyfIc4ZgRt54rbh0+MXcd9cBJL9bfyHmQMqkZBOFJOBuGepI9aoW3hPS7TxFLrkJv/t8wxIz6ncvGy5YhfLaQptBdiF24XJwBQtEk/P+vuB63+X6XMvTNan0a18WTa3qF1qVvot0WSSSKIS+V9milKARogPLtjjPTJqbwn4svvELXsd9odxp726o8btFcpHMG3fKGnghO4beQFIwy8nkC1p/grRNNkvmhjvJxqKFLuO91G4ukmBUKdySuyk7VC5xnAx04q1pHh2w0SCWGxe+eOUBSt3qE9yFAGAF8122jn+HH6UdPkvvG7dC1plxd3WlW0+pWX2C7kjVprXzRL5LEcrvHDY9RVDxFa6/dR248OX8Nm6+b5pmUHfmFxH1VukhRj04U9ehv6Zptpo2lW2m6bF5NpaxrFDHuLbVAwBkkk/iatUSs72EtDm9UsPFcxtP7K1a3gCWgS43opLzeZES4/dnjYso7DLDjuOkoop3ElYKKKKQwooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAooooAKKKKACiiigAquL+2bU209Zc3SQid4wp+VCSoJPQZKtgdTg+hqwThST29Bmuc0a0kk0fUtU1LT5Li41UvK9i6KHMIXbFCVcgAlAMhiBudskCgZraXqX9p28sv2O7s/Knkg2XcXls+xiu9RnlGxlT3BBq7XnekaPdD4b6roV14V1TTYTcztBZ2r2QZ4pbh3VYx5rRAKrAMr4UjIww4q74a1PT/BvhxLfxUdE8KrLcSfZLeQ29kZEG35mRJGj35PJRiDlThc7QeQvTuztycDmq+n39tqum21/YSeba3USzQybSu5GGQcHBHB71xWviy8XXdhqGm6bYeNdGjimhNvFc28sEVySm2R97bSAoYEjc67uFOTXM3ngHVrnTvD9vq2n6nPBaaLbWqxaWdPd7G4QESMGuR8hOUw8LZ+T2U0LXf8Arf8AyX3j0PUhrtifEJ0QtOt95JnVXtZVjdAVDFZCuxsF1yAxIzWjXJXraqPiJpc0WgahcWNraT2sl8JbYITK0DBtplD4Hltu+XOegNc/pPgq6X4gz6jrdlqk0rXVw325F082s8DhgkUjbRdMoRguxsqCoxwAQLW3z/MXS56JY6hbalA81lJ5kaTSQMdpGHjco459GUjPTjirNec6J4Pk0fSPEemaJ4bg0PUpxeCz1m3S3jSRZJHaFAyEyLtDKMMgC7eM4FZ2leCrqy8N6lHZ6Lq8JuJbVrrTLs6bDHeRpJulRBa7ULMpKkybdw2qTjof8Ab0v6s9OttQtry5u4LeTfLZSiGddpGxyiuBz1+V1PHrVmvP9G8NyWnhXxdbf8IfDZ2upTPJa6HDcRQiaM20SbC0R2xszKwODwckEjk95Auy2iXZ5e1ANmc7eOme9P8A4AjO1XxNpOiXKW+p3RhleIzKohd8r5iR5+UH+KRBjrz6A4jufFui2mt/2TPdSC63pG5W3laKJ3+4jyhfLRmyMKzAncuByM37rTLC9kEl7ZW9w6rsDSxK5C7g2MkdNyqfqoPavO9X8G6te+JL+2ktJ7myvNShv7a5E0a21tjy97SpvEjyjyyqAKUGUPB3EJbpP+tf8geza/rT/M9Glv7aDULeyklxc3Ku8UYUksqY3HjoBuUZPcgd6p6Z4l0nWFvW06685bF9s7eU6qPl3AqSAHUqQQy5B7Gq2j2811rup6xeQyRbm+x2iSKVIhjJy2D/AH3LHPdVQ1m+Ghqb+LPEMupeHb6wtNQeOSKa5ltnVgkSxlSI5WbJwSOMY6kHilrbTsPS+puaN4i0vxBG76RcmdUSKRiYnTCyIHT7wHVSDjtnnFadVrPTrLT1K2Fnb2oYKpEMSpkKoVRwOygAegGKs1TtfQlXtqFFFFIYUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFRq7G6kQn5QikD3Jb/AUASUUUUAFFFFABRRRQAUUUUAFFRs7C6jQH5SjEj3BX/E1JQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQAUUUUAFFFFABRRRQB//2Q==)Ví dụ: để quản lý model Book có cấu trúc dữ liệu như sau:

Ta xây dựng class BOOK:

|  |
| --- |
| class BOOK(EntityObject): |
|  | context = '/book' |
|  | getters = ('book\_id', 'book\_category', 'book\_type', 'book\_title', 'book\_author', |
|  | 'book\_public\_date', 'book\_rating', 'book\_rated\_time', 'book\_download', |
|  | 'book\_read\_time', 'book\_page', 'book\_description', 'book\_image', 'book\_file', 'book\_is\_deleted', |
|  | 'book\_created\_time') |
|  |  |
|  | setters = ('set\_book\_id', 'set\_book\_category', 'set\_book\_type', 'set\_book\_title', 'set\_book\_author', |
|  | 'set\_book\_public\_date', 'set\_book\_rating', 'set\_book\_rated\_time', 'set\_book\_download', |
|  | 'set\_book\_read\_time', 'set\_book\_page', 'set\_book\_description', 'set\_book\_image', |
|  | 'set\_book\_file', 'set\_book\_is\_deleted', 'set\_book\_created\_time') |
|  |  |
|  | def \_\_init\_\_(self, book\_id='', book\_category='', book\_type='', book\_title='', book\_author='', book\_public\_date='', |
|  | book\_rating='0', book\_rated\_time='0', book\_download='0', book\_read\_time='0', book\_page='', book\_description='', |
|  | book\_image='', book\_file='', book\_is\_deleted='0', book\_created\_time=''): |
|  | super(BOOK, self).\_\_init\_\_() |
|  | self.\_book\_id = book\_id |
|  | self.\_book\_category = book\_category |
|  | self.\_book\_type = book\_type |
|  | self.\_book\_title = book\_title |
|  | self.\_book\_author = book\_author |
|  | self.\_book\_public\_date = book\_public\_date |
|  | self.\_book\_rating = book\_rating |
|  | self.\_book\_rated\_time = book\_rated\_time |
|  | self.\_book\_download = book\_download |
|  | self.\_book\_read\_time = book\_read\_time |
|  | self.\_book\_page = book\_page |
|  | self.\_book\_description = book\_description |
|  | self.\_book\_image = book\_image |
|  | self.\_book\_file = book\_file |
|  | self.\_book\_is\_deleted = book\_is\_deleted |
|  | self.\_book\_created\_time = book\_created\_time |
|  |  |
|  | def book\_id(self): |
|  | if not hasattr(self.book\_id.\_\_func\_\_, 'id'): |
|  | self.book\_id.\_\_func\_\_.id = 'book\_id' |
|  | if not hasattr(self.book\_id.\_\_func\_\_, 'key'): |
|  | self.book\_id.\_\_func\_\_.key = '/book/{book\_id}' |
|  | return self.\_book\_id |
|  |  |
|  | def set\_book\_id(self, string): |
|  | self.\_book\_id = string |
|  |  |
|  | def book\_category(self): |
|  | if not hasattr(self.book\_category.\_\_func\_\_, 'key'): |
|  | self.book\_category.\_\_func\_\_.key = '/book/{book\_id}/book\_category' |
|  | return self.\_book\_category |
|  |  |
|  | def set\_book\_category(self, string): |
|  | self.\_book\_category = string |
|  |  |
|  | def book\_type(self): |
|  | if not hasattr(self.book\_type.\_\_func\_\_, 'key'): |
|  | self.book\_type.\_\_func\_\_.key = '/book/{book\_id}/book\_type' |
|  | return self.\_book\_type |
|  |  |
|  | def set\_book\_type(self, string): |
|  | self.\_book\_type = string |
|  |  |
|  | def book\_title(self): |
|  | if not hasattr(self.book\_title.\_\_func\_\_, 'key'): |
|  | self.book\_title.\_\_func\_\_.key = '/book/{book\_id}/book\_title' |
|  | return self.\_book\_title |
|  |  |
|  | def set\_book\_title(self, string): |
|  | self.\_book\_title = string |
|  |  |
|  | def book\_author(self): |
|  | if not hasattr(self.book\_author.\_\_func\_\_, 'key'): |
|  | self.book\_author.\_\_func\_\_.key = '/book/{book\_id}/book\_author' |
|  | return self.\_book\_author |
|  |  |
|  | def set\_book\_author(self, string): |
|  | self.\_book\_author = string |
|  |  |
|  | def book\_public\_date(self): |
|  | if not hasattr(self.book\_public\_date.\_\_func\_\_, 'key'): |
|  | self.book\_public\_date.\_\_func\_\_.key = '/book/{book\_id}/book\_public\_date' |
|  | return self.\_book\_public\_date |
|  |  |
|  | def set\_book\_public\_date(self, string): |
|  | self.\_book\_public\_date = string |
|  |  |
|  | def book\_rating(self): |
|  | if not hasattr(self.book\_rating.\_\_func\_\_, 'key'): |
|  | self.book\_rating.\_\_func\_\_.key = '/book/{book\_id}/book\_rating' |
|  | return self.\_book\_rating |
|  |  |
|  | def set\_book\_rating(self, string): |
|  | self.\_book\_rating = string |
|  |  |
|  | def book\_rated\_time(self): |
|  | if not hasattr(self.book\_rated\_time.\_\_func\_\_, 'key'): |
|  | self.book\_rated\_time.\_\_func\_\_.key = '/book/{book\_id}/book\_rated\_time' |
|  | return self.\_book\_rated\_time |
|  |  |
|  | def set\_book\_rated\_time(self, string): |
|  | self.\_book\_rated\_time = string |
|  |  |
|  | def book\_download(self): |
|  | if not hasattr(self.book\_download.\_\_func\_\_, 'key'): |
|  | self.book\_download.\_\_func\_\_.key = '/book/{book\_id}/book\_download' |
|  | return self.\_book\_download |
|  |  |
|  | def set\_book\_download(self, string): |
|  | self.\_book\_download = string |
|  |  |
|  | def book\_read\_time(self): |
|  | if not hasattr(self.book\_read\_time.\_\_func\_\_, 'key'): |
|  | self.book\_read\_time.\_\_func\_\_.key = '/book/{book\_id}/book\_read\_time' |
|  | return self.\_book\_read\_time |
|  |  |
|  | def set\_book\_read\_time(self, string): |
|  | self.\_book\_read\_time = string |
|  |  |
|  | def book\_page(self): |
|  | if not hasattr(self.book\_page.\_\_func\_\_, 'key'): |
|  | self.book\_page.\_\_func\_\_.key = '/book/{book\_id}/book\_page' |
|  | return self.\_book\_page |
|  |  |
|  | def set\_book\_page(self, string): |
|  | self.\_book\_page = string |
|  |  |
|  | def book\_description(self): |
|  | if not hasattr(self.book\_description.\_\_func\_\_, 'key'): |
|  | self.book\_description.\_\_func\_\_.key = '/book/{book\_id}/book\_description' |
|  | return self.\_book\_description |
|  |  |
|  | def set\_book\_description(self, string): |
|  | self.\_book\_description = string |
|  |  |
|  | def book\_image(self): |
|  | if not hasattr(self.book\_image.\_\_func\_\_, 'key'): |
|  | self.book\_image.\_\_func\_\_.key = '/book/{book\_id}/book\_image' |
|  | return self.\_book\_image |
|  |  |
|  | def set\_book\_image(self, string): |
|  | self.\_book\_image = string |
|  |  |
|  | def book\_file(self): |
|  | if not hasattr(self.book\_file.\_\_func\_\_, 'key'): |
|  | self.book\_file.\_\_func\_\_.key = '/book/{book\_id}/book\_file' |
|  | return self.\_book\_file |
|  |  |
|  | def set\_book\_file(self, string): |
|  | self.\_book\_file = string |
|  |  |
|  | def book\_is\_deleted(self): |
|  | if not hasattr(self.book\_is\_deleted.\_\_func\_\_, 'key'): |
|  | self.book\_is\_deleted.\_\_func\_\_.key = '/book/{book\_id}/book\_is\_deleted' |
|  | return self.\_book\_is\_deleted |
|  |  |
|  | def set\_book\_is\_deleted(self, string): |
|  | self.\_book\_is\_deleted = string |
|  |  |
|  | def book\_created\_time(self): |
|  | if not hasattr(self.book\_created\_time.\_\_func\_\_, 'key'): |
|  | self.book\_created\_time.\_\_func\_\_.key = '/book/{book\_id}/book\_created\_time' |
|  | return self.\_book\_created\_time |
|  |  |
|  | def set\_book\_created\_time(self, string): |
|  | self.\_book\_created\_time = string |
|  |  |
|  | def to\_dict(self): |
|  | return { |
|  | 'book\_id': self.book\_id(), |
|  | 'book\_category': self.book\_category(), |
|  | 'book\_type': self.book\_type(), |
|  | 'book\_title': self.book\_title(), |
|  | 'book\_author': self.book\_author(), |
|  | 'book\_public\_date': self.book\_public\_date(), |
|  | 'book\_rating': self.book\_rating(), |
|  | 'book\_rated\_time': self.book\_rated\_time(), |
|  | 'book\_download': self.book\_download(), |
|  | 'book\_read\_time': self.book\_read\_time(), |
|  | 'book\_page': self.book\_page(), |
|  | 'book\_description': self.book\_description(), |
|  | 'book\_image': self.book\_image(), |
|  | 'book\_file': self.book\_file(), |
|  | 'book\_is\_deleted': self.book\_is\_deleted(), |
|  | 'book\_created\_time': self.book\_created\_time() |
|  | } |

Ví dụ truy xuất dữ liệu trên bảng BOOK:

- Bước 1: Tạo kết nối

hub = EtcdHub(os.path.dirname(\_\_file\_\_), Client)

etcd\_config\_file\_path = os.path.join(os.path.dirname(\_\_file\_\_), 'db\_config.yaml')

hub.loadFromFile(etcd\_config\_file\_path)

hub.establishConnections()

connection = hub.tryAConnection()

- Bước 2: Thực hiện truy xuất dữ liệu

* Ghi dữ liệu:

repo = etcd.EntityRepo(connection)

repo.put(BOOK(book\_id=book\_id, book\_category=book\_category, book\_type=book\_type, book\_title=book\_title, book\_author=book\_author, book\_public\_date=book\_public\_date, book\_page=book\_page, book\_description=book\_description, book\_image=book\_image\_data, book\_file=book\_file\_data, book\_created\_time=created\_time))

* Đọc dữ liệu:
  + Lấy BOOK có book\_id= "102345245566":

book = repo.get(BOOK, book\_id=book\_id)

if book is not None:

print(book.to\_dict())

* + Lấy danh sách BOOK:

books = repo.list(BOOK)

res = []

for book in books:

info = book.to\_dict()

if info.get('book\_is\_deleted') == "0" and info.get('book\_type')=="ebook":

res.append(info)

print(info

### Ứng dụng thuật toán phân lớp Naive Bayes

### Chức năng phân lớp thể loại sách dựa vào tiêu đề

* Bước 1: Tạo dữ liệu huấn luyện (580 mẫu):

|  |  |  |
| --- | --- | --- |
| STT | Target (Thể loại) | Feature (Tiêu đề) |
| 1 | Văn học | Chuyện Li Kì Ở Xóm Thần Tiên (Bản Thường) |
| 2 | Văn học | Con Đường Hồi Giáo (Tái Bản 2017) |
| 3 | Văn học | Điều Kỳ Diệu Của Tiệm Tạp Hóa Namiya (Tái Bản 2018) |
| 4 | Văn học | Harry Potter Và Phòng Chứa Bí Mật - Tập 2 (Tái Bản 2017) |
| 5 | Văn học | Thế Giới Rộng Lớn Đừng Đi Một Mìn |
| .. | … | … |
| 97 | Kinh tế | Combo 101 - Những Điều Nhà Lãnh Đạo Cần Biết (Bộ 6 Cuốn) |
| 98 | Kinh tế | Combo Tứ Thư Lãnh Đạo (Tái Bản) - Bộ 4 Cuốn |
| 99 | Kinh tế | 10 Điều Khác Biệt Nhất Giữa Kẻ Giàu & Người Nghèo (Tái Bản 2018) |
| 100 | Kinh tế | 10 Thói Quen Của Triệu Phú |
| 101 | Kinh tế | 13 Nguyên Tắc Nghĩ Giàu Làm Giàu - Think And Grow Rich (Tái Bản 2018) |
| … | … | … |
| 193 | Tâm lý - Kỹ năng sống | Bạn Đang Nghịch Gì Với Đời Mình? |
| 194 | Tâm lý - Kỹ năng sống | Bạn Đắt Giá Bao Nhiêu? |
| 195 | Tâm lý - Kỹ năng sống | Bạn Đỡ Ngu Ngơ Rồi Đấy |
| 196 | Tâm lý - Kỹ năng sống | Ba Người Thầy Vĩ Đại |
| 197 | Tâm lý - Kỹ năng sống | Bốn Mùa Cuộc Sống - Châm Ngôn Ngày Mới (Tái Bản 2018) |
| … | … | … |
| 278 | Nuôi dạy con | Combo Vì Chơi Là Học |
| 279 | Nuôi dạy con | 365 Ngày Chuẩn Bị Trước Khi Mang Thai |
| 280 | Nuôi dạy con | 3 Phút Cho Ông Bố Bận Rộn |
| 281 | Nuôi dạy con | 70 Thói Quen Tốt Trong Việc Dưỡng Con Theo Phương Pháp Shichida |
| 282 | Nuôi dạy con | Ăn Dặm Kiểu Nhật (Tái Bản 2018) |
| … | … | … |
| 379 | Sách thiếu nhi | Bách Khoa Toàn Thư Pokémon - Best Wishes Series |
| 380 | Sách thiếu nhi | Chuyện Con Mèo Dạy Hải Âu Bay (Tái Bản 2017) |
| 381 | Sách thiếu nhi | Combo Ai Ở Sau Lưng Bạn Thế? (Bộ 6 Cuốn) |
| 382 | Sách thiếu nhi | Cùng Chơi Với Bé - Đi Tắm Thật Thích |
| 383 | Sách thiếu nhi | Cùng Chơi Với Bé - Ú Òa! Ú Òa! |
| … |  |  |
| 575 | Tiểu sử - Hồi ký | Bản Lĩnh Putin (Tái Bản 2018) |
| 576 | Tiểu sử - Hồi ký | Bảo Đại - Hoàng Đế Cuối Cùng |
| 578 | Tiểu sử - Hồi ký | Bảo Đại - Hoàng Đế Cuối Cùng (Tái Bản 2018) |
| 579 | Tiểu sử - Hồi ký | Benjamin Franklin - Cuộc Đời Một Người Mỹ |
| 580 | Tiểu sử - Hồi ký | Bên Kia Đèo Mã Phục |

* Bước 2: Client gửi tiêu đề sách lên server và yêu cầu phân loại thể loại sách

VD:

{

"book\_title": "Dạy con nên người - Cách trở thành bố mẹ tốt"

}

* Bước 3: Server nhận dữ liệu và bắt đầu xử lý theo những bước sau

Bags of

Words

TF-IDF

Multinomial

NB

Trích xuất

đặc trưng

Các lớp thể loại được phân tích dựa trên dữ liệu mẫu:

|  |  |
| --- | --- |
|  | Các thể loại |
| 1 | Văn học |
| 2 | Nuôi dạy con |
| 3 | Kinh tế |
| 4 | Tâm lý - Kỹ năng sống |
| 5 | Sách thiếu nhi |
| 6 | Tiểu sử - Hồi ký |

- Trích xuất đặc trưng: Trong tiếng anh, từ đất nước là “country” - chỉ gồm 1 từ duy nhất, tuy nhiên trong tiếng việt gồm 2 chữ “đất” và “nước”, do đó bước này sẽ phân tích ra được từ nào là từ đơn và từ nào là từ ghép trong câu dữ liệu.

- Bags of words: Vector hóa từng câu sau khi được tách từ. Bước này sẽ thực hiện việc đếm số lượng xuất hiện của mỗi từ trong câu. Tuy nhiên có một hạn chế là trong câu sẽ có nhiều từ như “ở, thì, là, nên, vì,...”, những từ này tuy số lần xuất hiện nhiều nhựng lại không có tác dụng trong việc phân lớp dữ liệu. Chính vì vậy nếu chỉ xét theo tần số xuất hiện của từng từ thì việc phân loại văn bản rất có thể cho kết quả sai dẫn tỷ lệ chính xác sẽ thấp.

- TF-IDF: Để giải quyết hạn chế trên, bước này sẽ tính giá trị TF-IDF của một từ là một con số thu được qua thống kê thể hiện mức độ quan trọng của từ này trong một văn bản, mà bản thân văn bản đang xét nằm trong một tập hợp các văn bản.

- Multinomial Naive Bayes: dùng mô hình đa thức với các giá trị đầu vào từ bước phía trên để cho ra kết quả.

* Bước 4: Kết quả trả về

- Kết quả dự đoán: Nuôi dạy con

- Xác xuất của từng lớp thể loại:

[0.15232865 0.55035325 0.07724884 0.05108834 0.07669691 0.09228402]

- Thời gian thực hiện: 0.339988s

### Chức năng điều khiển ứng dụng bằng giọng nói

* Bước 1: Tạo dữ liệu huấn luyện (54 mẫu)

|  |  |  |
| --- | --- | --- |
| STT | Target (Phân lớp chức năng của ứng dụng) | Feature (Các câu nói điều khiển mà người dùng truyền vào) |
| 1 | manage\_account | quản lý tài khoản |
| 2 | manage\_account | đổi mật khẩu |
| 3 | manage\_account | chỉnh sửa thông tin tài khoản |
| 4 | manage\_account | tới trang quản lý tài khoản |
| 5 | manage\_account | hello quản lý tài khoản của tôi |
| 6 | manage\_account | tôi muốn đổi mật khẩu |
| 7 | manage\_account | tôi muốn xem thông tin cá nhân của tôi |
| 8 | manage\_account | xem thông tin cá nhân |
| 9 | manage\_account | thông tin tài khoản cá nhân |
| 10 | share | các sách được chia sẻ |
| 11 | share | sách chia sẻ |
| 12 | share | chia sẻ sách mới |
| 13 | share | tôi muốn chia sẻ sách |
| 14 | share | danh sách các sách được chia sẻ |
| 15 | share | bản tin chia sẻ sách mới |
| 16 | share | bản tin chia sẻ sách |
| 17 | share | chia sẻ sách |
| 18 | share | chia sẻ |
| 19 | share | trao đổi sách giấy |
| 20 | share | chia sẻ sách giấy |
| 21 | manage\_reading\_history | Xem lịch sử đọc sách của tôi |
| 22 | manage\_reading\_history | tôi đã đọc những quyển sách nào? |
| 23 | manage\_reading\_history | tôi muốn xem những quyển sách tôi đã đọc |
| 24 | manage\_reading\_history | lịch sử đọc sách |
| 25 | manage\_reading\_history | sách đã đọc |
| 26 | manage\_reading\_history | quản lý lịch sử đọc sách |
| 27 | search\_user | tìm kiếm người dùng |
| 28 | search\_user | tra cứu người dùng |
| 29 | search\_user | tìm kiếm user |
| 30 | search\_user | tra cứu user |
| 31 | search\_user | có những ai đang sài app đọc sách |
| 32 | search\_user | tìm người dùng |
| 33 | search\_user | thông tin người dùng khác |
| 34 | logout | đăng xuất |
| 35 | logout | đăng xuất cho tôi |
| 36 | logout | tôi muốn thoát |
| 37 | logout | tôi muốn logout |
| 38 | logout | tôi muốn đăng xuất |
| 39 | logout | logout |
| 40 | logout | không sài app nữa |
| 41 | logout | thoát tài khoản |
| 42 | search\_book | tra cứu sách |
| 43 | search\_book | thư viện sách |
| 44 | search\_book | tới trang tra cứu sách |
| 45 | search\_book | tới trang thư viện sách |
| 46 | search\_book | màn hình tra cứu sách |
| 47 | search\_book | tất cả các quyển sách |
| 48 | search\_book | tìm sách |
| 49 | search\_book | cho tôi hỏi có quyển sách nào có tên là abc không? |
| 50 | search\_book | hãy tìm kiếm cho tôi quyển sách có tên xyz |
| 51 | search\_book | kiếm sách có tên xyz |
| 52 | search\_book | có bao nhiêu quyển sách khoa học |
| 53 | search\_book | có bao nhiêu quyển sách thuộc thể loại văn học |
| 54 | search\_book | tìm cho tôi quyển sách ddd |

* Bước 2: Client gửi câu nói đã chuyển sang thành dạng văn bản lên server và yêu cầu phân loại chức năng của ứng dụng

VD:

{

"speech\_to\_text": "xin chào, tôi muốn tìm kiếm sách"

}

* Bước 3: Server nhận dữ liệu và bắt đầu xử lý theo mô hình đã trình bày ở phần 4.3.3.1

Các lớp chức năng lấy được từ dữ liệu huấn luyện:

|  |  |  |
| --- | --- | --- |
| STT | Phân lớp chức năng | Ý nghĩa |
| 1 | manage\_account | Quản lý tài khoản |
| 2 | share | Thông tin các bài chia sẻ sách |
| 3 | manage\_reading\_history | Quản lý lịch sử đọc |
| 4 | search\_user | Tra cứu người dùng |
| 5 | search\_book | Tra cứu sách |
| 6 | logout | Đăng xuất |

* Bước 4: Kết quả sau khi chạy thuật toán

- Kết quả dự đoán: search\_book

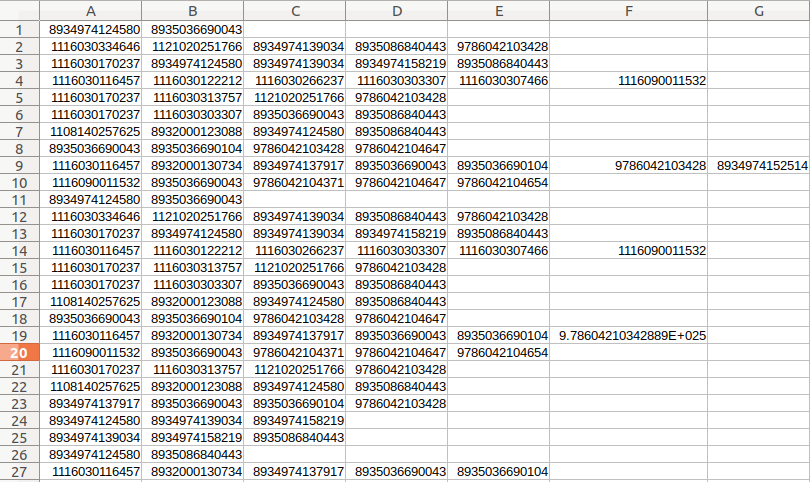
- Xác xuất của từng lớp:

[0.19253426 0.14995994 0.11643847 0.21191874 0.1211586 0.20798999]

- Thời gian thực hiện: 0.032429

### Ứng dụng thuật toán Apriori trong việc tư vấn sách cho từng người dùng

Ý tưởng của bài toán là dựa vào lịch sử đọc sách của mỗi người dùng mà đề xuất các cuốn sách liên quan người dùng có thể thích đọc.

* Bước 1: Truy xuất cơ sở dữ liệu và lấy hết tất cả lịch sử đọc của tất cả người dùng hiện có trong hệ thống, sau đó lưu vào file csv dưới dạng transaction như sau:

Các mã sách sẽ được lưu trong 1 cột và mỗi dòng sẽ tương ứng với lịch sử đọc của mỗi người dùng.

* Bước 2: Truy xuất cơ sở dữ liệu và lấy ra lịch sử đọc của người dùng cần được tư vấn và cũng lưu thành dạng transaction như trên.
* Bước 3: Xác định giá trị của min\_support và min\_confidence. Giá trị này sau khi thực nghiệm nhiều lần và chọn ra kết quả như sau:

- Min\_support: 0.05

- Min\_confidence: 0.15

* Bước 4: Các thông tin đầu vào để có thể chạy thuật toán khuyến nghị Apriori: Dữ liệu lịch sử đọc của tất cả người dùng ở B1, dữ liệu lịch sử đọc của người dùng cần được tư vấn ở B2, giá trị min\_support và min\_confidence được tìm ra ở B3.

Bắt đầu thực hiện thuật toán:

- Đầu tiên sẽ dựa vào dữ liệu lịch sử đọc của tất cả người dùng ở B1 để tìm các danh sách mã sách không bị trùng lắp (gồm n phần tử), dựa vào danh sách này để tạo ra các tổ hợp (combinations), các tổ hợp này sẽ bắt đầu từ 1 phần tử cho tới n phần tử. Tính toán số lần xuất hiện của các tổ hợp này trên tổng số transaction và so sánh với giá trị min\_support, nếu lớn hơn hoặc bằng min\_support thì các tổ hợp này được gọi là tập phổ biến.

- Tiếp theo là dựa vào min\_confidence để tìm ra luật kết hợp.

VD: Ta có tập phổ biến “Truyện conan”, “Truyện thần đồng đất việt”. Phải xác định xem trường hợp nào có thể xảy ra trong các trường hợp sau đây:

+ Nếu người dùng đã đọc “Truyện conan” thì sẽ đọc “Truyện thần đồng đất việt”.

+ Nếu người dùng đã đọc “Truyện thần đồng đất việt” thì sẽ đọc “Truyện conan ”.

+ Nếu người dùng đã đọc “Truyện thần đồng đất việt” thì sẽ đọc “Truyện conan ” và ngược lại.

+ Nếu người dùng đã đọc “Truyện thần đồng đất việt” thì sẽ đọc “Truyện conan ”, nhưng người dùng đã đọc “Truyện conan” thì không chắc sẽ đọc “Truyện thần đồng đất việt ”.

+ Nếu người dùng đã đọc “Truyện tconan ” thì sẽ đọc “Truyện thần đồng đất việt ”, nhưng người dùng đã đọc “Truyện thần đồng đất việt” thì không chắc sẽ đọc “Truyện conan”.

Do đó nếu muốn biết người dùng đã đọc “Truyện conan” thì có đọc “Truyện thần đồng đất việt” hay không thì ta lấy xác xuất xuất hiện của tập (“Truyện conan”, “Truyện thần đồng đất việt”) trên tổng số transaction chia cho xác xuất của tập (“Truyện conan”) trên tổng số transaction. Nếu kết quả lớn hơn hoặc bằng ngưỡng min\_confidence thì ta sẽ có luật kết hợp “Truyện conan” → “Truyện thần đồng đất việt”. Kết luận: Nếu người dùng đã đọc “Truyện conan” thì sẽ đọc “Truyện thần đồng đất việt”.

* Bước 5: Khi tất cả các luật kết hợp đã được tìm ra thì dựa vào sẽ tiến hành so sánh:

- Dựa vào lịch sử đọc sách của người dùng cần được tư vấn để tìm ra các tổ hợp.

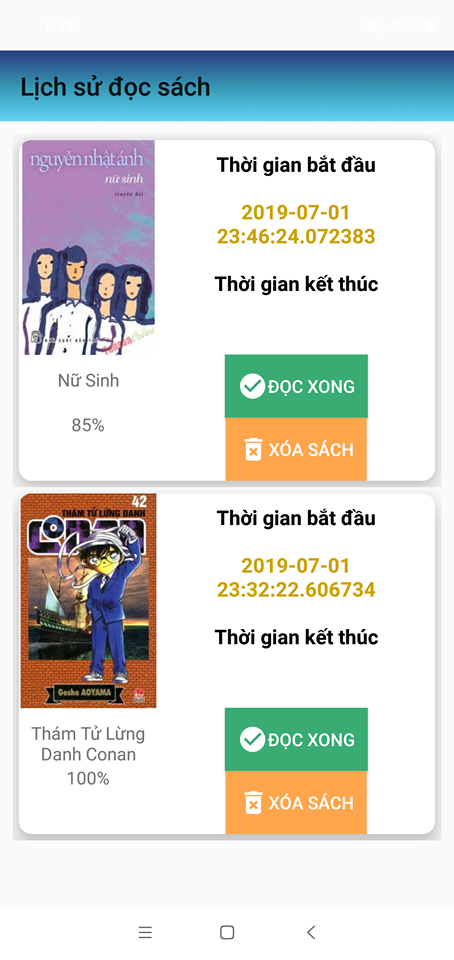
- Ứng với mỗi tổ hợp, so sánh nếu vế trái của các luật các kết hợp bằng tới tổ hợp thì vế phải của luật kết hợp đó sẽ là dữ liệu để tư vấn đọc sách cho người dùng.

- Lượt bỏ các dữ liệu bị trùng lắp và gửi danh sách kết quả về cho người dùng.

* Kết quả:

Người dùng tên “Trương Thị Bích Chi đã đọc 2 quyển sách:

- Nữ sinh (tác giả Nguyễn Nhật Ánh)

- Truyện conan (tập 42)

Hình 4‑3. Màn hình lịch sử đọc sách

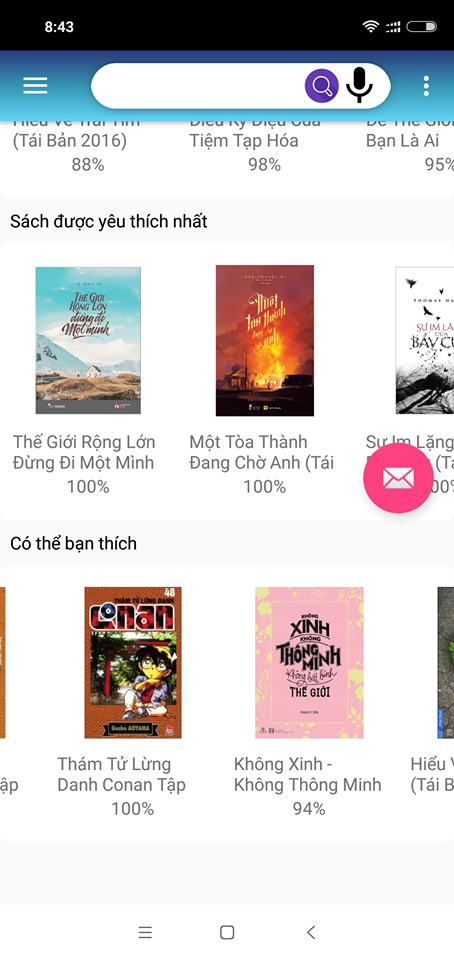
Những quyển sách được tư vấn bao gồm:

- Truyện conan (tập 94), Truyện conan(tập 48), Truyện conan(tập 45)

- Truyện nhật ký cậu bé nhút nhát

- Những sách của Nguyễn Nhật Ánh: Bàn có 5 chỗ ngồi, Con chó nhỏ mang giỏ hoa

- Sách trong thể loại văn học: Hiểu về trái tim, Không xinh - không thông minh,…



Hình 4‑4. Màn hình chính với danh sách sách gợi ý

### Dùng Google Cloud Platform để triển khai Server

Hiện tại cấu trúc của Server gồm 3 thành phần: Java Server (chứa các API phục vụ các business request từ Client), Python Server (xử lý dữ liệu và respone về Client), và Etcd Database. Từ những thành phần trên, các Hosting có sẵn sẽ được sử dụng để deploy từng thành phần (ví dụ dùng Heroku để deploy Server Java và NodeJS, dùng Mysqlfreehosting.com để deploy MySQL). Tuy nhiên nếu làm nhưng vậy thì các thành phần sẽ giao tiếp với nhau kém hiệu quả vì phải phụ thuộc vào kết nối giữa các Server đó với nhau.Vì vậy, để có một giải phải giúp tiết kiệm chi phí khi các thành phần giao tiếp với nhau đề tài sẽ sử dụng Google cloud là nơi để deploy các thành phần của Server. Đặc điểm của mô hình này là các thành phần sẽ giao tiếp local với nhau thông qua các port mà chúng chạy trên Server Google.

# CHƯƠNG 5: KẾT LUẬN

## Kết quả đạt được

Trong quá trình thực hiện đề tài khóa luận, nhóm tác giả đã trau dồi thêm được kiến thức về các thuật toán khai thác dữ liệu cũng như các công nghệ mới phục vụ cho việc xây dựng một ứng dụng Android.

Về mặt chức năng, ứng dụng cơ bản đã hoàn thành các mục tiêu đề ra:

* Ứng dụng có chức năng đọc sách online.
* Cho phép người dùng chia sẻ sách.
* Tự động phân loại sách.
* Liệt kê các sách đang được yêu thích và đề xuất các sách người dùng có thể thích.
* Các chức năng bổ sung: Bình luận, đánh giá sách, tìm kiếm sách, quản lý thông tin cá nhân.
* Quản lý lịch sử đọc sách của người dùng.
* Quản lý thông tin người dùng, thông tin sách, lập các biểu mẫu thống kê.
* Điều khiển ứng dụng bằng giọng nói.
* Lập lịch đọc sách cho người dùng.

## Ưu điểm của hệ thống

* Ứng dụng có giao diện thân thiện, dễ sử dụng.
* Nguồn sách phong phú, đa dạng về thể loại.
* Áp dụng thuật toán và dữ liệu có được để đề xuất sách cho người đọc.
* Sử dụng Java 8 và các thư viện như Annotaion, Lombok để giảm thiểu code cũng như tăng tốc độ xử lý các nhu cầu cơ bản.

## Hạn chế của hệ thống

* Tốc độ xử lý còn hạn chế, phụ thuộc vào tình trạng mạng và server.
* Chưa mở rộng được đa nền tảng và các loại thiết bị khác, hệ thống chỉ chạy được trên điện thoại Android.
* Hiện tại chỉ hỗ trợ file pdf
* Chức năng đọc sách chưa có các phụ trợ như đánh dấu trang, thay đổi font chữ, độ giãn dòng …

## Hướng phát triển

Để hoàn thiện hệ thống hơn, trong tương lai nhóm tác giả đề xuất một số hướng phát triển như sau:

* Cải thiện tốc độ xử lý, nâng cấp khả năng tiếp nhận số lượng yêu cầu trong một thời gian của server.
* Hỗ trợ chia sẻ lên facebook.
* Hỗ trợ đọc sách trên website và IOS.
* Hoàn thiện các chức năng trên ứng dụng giúp người đọc thấy tiện ích hơn.
* Đa dạng hóa ngôn ngữ.
* Đa dạng hóa loại file sách, thêm sách nghe.
* Nghiên cứu và áp dụng cách Spotify mang âm nhạc đến cho người dùng.
* Mở rộng tính năng điều khiển bằng giọng nói.
* Cho phép người quản trị có nhiều góc nhìn hơn về hệ thống bằng việc áp dụng nhiều kiểu biểu đồ.

# TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Nghiên cứu lấy dữ liệu tự động đối với những website load bằng javascripts

<https://viblo.asia/p/scrapy-voi-nhung-website-load-bang-javascripts-4dbZNomQlYM>

Tiếng Anh

[1] A Simple Trending Products Recommendation Engine in Python

<http://blog.yhat.com/posts/trending-products-recommender-engine.html>

[2] Androidannotations

<https://github.com/androidannotations/androidannotations/wiki>

[3] Association Rule Mining via Apriori Algorithm in Python

<https://stackabuse.com/association-rule-mining-via-apriori-algorithm-in-python/?fbclid=IwAR2EzIbvZRieEvsYUV0aoOviKcFPhmBg9MY9n8w_rTjGLYi2ZKDHQmwgQQ4>

[4] Book Recommender: Collaborative Filtering, Shiny

<https://www.kaggle.com/philippsp/book-recommender-collaborative-filtering-shiny?fbclid=IwAR3oSKCkV2-Pj5Dkdmx8PiEPTXfTKpqtrABZ2EyfLzHebUSXbecVUrPKM9U>

[5] FloatingActionButton Library

<https://github.com/Clans/FloatingActionButton>

[6] Framework flask:

<https://www.freecodecamp.org/news/structuring-a-flask-restplus-web-service-for-production-builds-c2ec676de563/?fbclid=IwAR0BC-X_pyuPfhuV0J7bXWFLS59xjayR3blLLUXSsPBGXRBjGHnMFKTCiTk>

[7] Lombok:

<https://projectlombok.org/>

[8] Mkyong Java 8:

<http://www.mkyong.com/tutorials/java-8-tutorials/>

[9] MPAndroidChart:

<https://github.com/PhilJay/MPAndroidChart>

[10] Retrofit:

<https://square.github.io/retrofit/>

[11] Scikit learn

<https://scikit-learn.org/stable/>

[12] Toasty Library

<https://github.com/GrenderG/Toasty>

1. Trang chủ: [https://waka.vn/](https://waka.vn/gioi-thieu) [↑](#footnote-ref-2)
2. Trang chủ: <http://sachtot.vn/> [↑](#footnote-ref-3)
3. Trang chủ: [https:/gacsach.com/](https://waka.vn/gioi-thieu) [↑](#footnote-ref-4)
4. Tham khảo tại: <https://etcd.io/> [↑](#footnote-ref-5)
5. nguồn: https://medium.com/@uzzal2k5/etcd-etcd-cluster-configuration-for-kubernetes-779455337db6 [↑](#footnote-ref-6)
6. Tài liệu: <https://etcd.io/docs/v3.3.12/dev-guide/interacting_v3/> [↑](#footnote-ref-7)
7. Tài liệu tham khảo: <https://en.wikipedia.org/wiki/Bag-of-words_model> [↑](#footnote-ref-8)
8. <http://viet.jnlp.org/tai-nguyen-ngon-ngu-tieng-viet/khai-yeu-ve-corpus> [↑](#footnote-ref-9)
9. <https://www.java.com/en/download/faq/release_dates.xml> [↑](#footnote-ref-10)
10. Nguồn: (nguồn: http://gs.statcounter.com/os-market-share/mobile/worldwide) [↑](#footnote-ref-11)
11. Annotation: <http://androidannotations.org/> [↑](#footnote-ref-12)
12. Danh sách ứng dụng: <https://github.com/androidannotations/androidannotations/wiki/Apps-using-AndroidAnnotations> [↑](#footnote-ref-13)
13. Link github: <https://github.com/barteksc/AndroidPdfViewer> [↑](#footnote-ref-14)
14. Link github: <https://github.com/PhilJay/MPAndroidChart> [↑](#footnote-ref-15)