Trường Đại Học Ngoại Ngữ - Tin Học Thành Phố Hồ Chí Minh

**Khoa Công Nghệ Thông Tin**



**MÔN HỌC : CÁC HỆ QUẢN TRỊ CƠ SỞ DỮ LIỆU**

**ĐỀ TÀI : PHẤT TRIỂN CƠ SỞ DỮ LIỆU QUẢN LÝ PHÒNG TRỌ**

**Giáo Viên Hướng Dẫn** : **ThS. Lê Thị Minh Nguyện**

**Thành Viên :**

1. Lưu Toàn Nghĩa - 21DH113306
2. Nguyễn Minh Khang – 21DH110778
3. Đặng Mai Nguyên Vũ – 21DH114321
4. Dương Quí On – 21DH112773
5. Trương Quốc Phú – 21DH111428

Tp. Hồ chí minh, Ngày 23 tháng 02 năm 2024

**Mục lục**

[CHƯƠNG 1. KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU 6](#_Toc160919447)

[1. Khảo sát hiện trạng 6](#_Toc160919448)

[2. Mô tả nội dung cần giải quyết của đề tài 9](#_Toc160919449)

[CHƯƠNG 2. CƠ SỞ LÝ THUYẾT 10](#_Toc160919450)

[1. Kiến trúc của Oracle Database: 10](#_Toc160919451)

[2. Quản trị User và Profile: 11](#_Toc160919452)

[3. Ngôn ngữ PL/SQL: 12](#_Toc160919453)

[4. Cơ chế sao lưu và phục hồi dữ liệu sau sự cố: 13](#_Toc160919454)

[5. Quản lý giao tác/ giao dịch 15](#_Toc160919455)

[6. Xử Lý đồng thời: 16](#_Toc160919456)

[CHƯƠNG 3. PHÂN TÍCH HỆ THỐNG 17](#_Toc160919457)

[1. Phân tích chức năng 17](#_Toc160919458)

[2. Phân tích dữ liệu 19](#_Toc160919459)

[CHƯƠNG 4. THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG 20](#_Toc160919460)

[1. Mô hình dữ liệu quan hệ (Relational Data model) 20](#_Toc160919461)

[2. Từ điển dữ liệu (Mô tả các bảng và thuốc tính) 21](#_Toc160919462)

[3. Thiết kế và cài đặt CSDL bằng Oracle 21](#_Toc160919463)

[CHƯƠNG 5. KẾT LUẬN 21](#_Toc160919464)

[CHƯƠNG 6. TÀI LIỆU THAM KHẢO 22](#_Toc160919465)

[CHƯƠNG 7. PHỤ LỤC 22](#_Toc160919466)

**Danh mục hình ảnh**

**Danh mục bảng biểu**

[Bảng 1: Cơ cấu tổ chức của hệ thống 7](#_Toc161691505)

[Bảng 2: Sơ đồ BFD 18](#_Toc161691506)

[Bảng 3: Sơ đồ ERD 21](#_Toc161691507)

[Bảng 4: Mô hình dữ liệu quan hệ (Relational Data model) 22](#_Toc161691508)

**LỜI MỞ ĐẦU**

Mục tiêu của đề tài này là cung cấp các giải pháp và công cụ hữu ích để giúp quản lý phòng trọ hoạt động một cách hiệu quả, tăng cường lợi nhuận và cải thiện trải nghiệm của khách hàng.

Đề tài quản lý phòng trọ giúp chủ sở hữu hiểu rõ về quy trình và phương pháp quản lý hiệu quả. Chủ sở hữu có thể áp dụng các biện pháp như quản lý tài chính, quản lý nguồn nhân lực, quản lý dịch vụ và tiện ích để tăng thu nhập và giảm chi phí. Quản lý phòng trọ sẽ học cách xây dựng và duy trì mối quan hệ tốt với khách hàng, quản lý hợp đồng thuê, xử lý các vấn đề phát sinh và quản lý các hoạt động hàng ngày. Các biện pháp quản lý phòng trọ bao gồm bảo trì, vệ sinh và quản lý an ninh đảm bảo rằng các phòng trọ đáp ứng được nhu cầu và mong muốn của khách hàng. Nếu được quản lý tốt, phòng trọ có thể góp phần vào việc cải thiện chất lượng cuộc sống của cộng đồng bằng cách tạo ra môi trường sống tốt hơn, hạn chế ảnh hưởng tiêu cực đến môi trường và hướng tới sự phát triển bền vững. Tóm lại, đề tài quản lý phòng trọ có mục tiêu là cung cấp các giải pháp và phương pháp quản lý hiệu quả cho hoạt động kinh doanh phòng trọ. Điều này mang lại lợi ích cho chủ sở hữu, nhân viên quản lý, khách hàng và cộng đồng xung quanh.

# KHẢO SÁT HIỆN TRẠNG VÀ XÁC ĐỊNH YÊU CẦU

## Khảo sát hiện trạng

* + Nghiệp vụ doanh nghiệp: Nghiệp vụ doanh nghiệp tập trung vào việc cung cấp một nền tảng trực tuyến để các cá nhân và tổ chức có thể dễ dàng đăng tin và tìm kiếm bất động sản cho thuê. Mục tiêu của bạn là tạo ra một môi trường thuận tiện, tin cậy và an toàn cho chủ nhà và người thuê bất động sản.
  + Lĩnh vực hoạt động: Trang web hoạt động trong lĩnh vực bất động sản, tập trung chủ yếu vào việc cho thuê bất động sản. Các danh mục chính bao gồm phòng trọ, chung cư, nhà ở và đất trống. Bằng cách cung cấp một nền tảng dễ sử dụng và linh hoạt, bạn tạo điều kiện cho chủ nhà có thể quản lý và quảng bá tài sản của họ một cách hiệu quả, đồng thời giúp người thuê dễ dàng tìm kiếm các tùy chọn phù hợp với nhu cầu của họ.
  + Quy trình hoạt động nghiệp vụ liên quan:
    - Thu thập thông tin: Tổ chức việc thu thập thông tin chi tiết từ các chủ nhà về bất động sản họ muốn cho thuê. Điều này bao gồm mô tả, hình ảnh, thông tin về vị trí và các tiện ích kèm theo.
    - Tạo và quản lý danh sách: Xây dựng hệ thống quản lý danh sách cho các bất động sản cho thuê, bao gồm việc phân loại chúng vào các danh mục như phòng trọ, chung cư, nhà ở và đất trống. Đảm bảo các thông tin được cung cấp là chính xác và cập nhật.
    - Phát triển trang web: Phát triển và duy trì trang web đăng tin, tạo giao diện dễ sử dụng và thân thiện với người dùng. Tích hợp các tính năng tìm kiếm nâng cao để người dùng có thể dễ dàng lọc và tìm kiếm các bất động sản phù hợp.
    - Xây dựng cộng đồng: Xây dựng một cộng đồng người dùng tích cực bao gồm cả chủ nhà và người thuê. Cung cấp tính năng đánh giá và nhận xét để tạo ra sự tin cậy và minh bạch trong quá trình giao dịch.
    - Tiếp thị và quảng cáo: Tiếp thị và quảng cáo trang web để thu hút lượng truy cập và đăng ký từ cả hai bên, chủ nhà và người thuê. Sử dụng các kênh tiếp thị hiệu quả như quảng cáo trực tuyến, tối ưu hóa công cụ tìm kiếm và tiếp thị nội dung.
* Cơ cấu tổ chức của hệ thống (yêu cầu: mô tả ngắn gọn và vẽ sơ đồ cơ cấu tổ chức của doanh nghiệp).
  + Giám đốc điều hành (CEO): Người đứng đầu ban quản trị, chịu trách nhiệm chung về việc quản lý và phát triển hệ thống đăng tin.
  + Bộ phận nhân sự: Tuyển nhân sự và đào tạo nhân sự
  + Bộ phận tài chính: Xây dựng và quản lý các hợp đồng với khách hàng và đối tác
  + Bộ phận kinh doanh và phát triển: Nguyên cứu thị trường và đề xuất chiến lược kinh doanh mới
  + Bộ phận hỗ trợ khách hàng: Xây dựng mối quan hệ với đối tác khách hàng
  + Bộ phận kỹ thuật: Định rõ yêu cầu sản phẩm, xây dựng lộ trình phát triển của website, phát triển kiểm thử trang web

Bảng 1: Cơ cấu tổ chức của hệ thống

* Đánh giá hiện trạng hệ thống cũ và các vấn đề cần phải giải quyết của hệ thống mới (Yêu cầu: chủ yếu nói về các vấn đề liên quan đến lưu trữ và truy xuất dữ liệu)
  + Hiện trạng hệ thống cũ (MySQL):
    - Phức tạp và lỗi thời: Hệ thống lưu trữ dữ liệu hiện tại sử dụng MySQL, có thể trở nên phức tạp và lỗi thời do sự phát triển của dự án và thêm vào đó là sự mở rộng của dữ liệu.
    - Hiệu suất không tối ưu: MySQL có thể không đáp ứng được yêu cầu về hiệu suất và khả năng mở rộng khi dữ liệu và tải công việc tăng lên.
    - Hạn chế về tính linh hoạt: MySQL có thể hạn chế trong việc xử lý một số tính năng và yêu cầu cụ thể của hệ thống mới.
  + Các vấn đề cần giải quyết trong hệ thống mới (Oracle):
    - Tối ưu hiệu suất: Oracle thường được biết đến với khả năng xử lý và hiệu suất cao, giúp cải thiện tốc độ truy xuất dữ liệu và thời gian phản hồi của hệ thống.
    - Khả năng mở rộng: Oracle cung cấp các công nghệ mở rộng linh hoạt cho phép hệ thống dễ dàng mở rộng khi cần thiết, giảm thiểu rủi ro về sự cố và giảm thiểu thời gian chết.
    - Bảo mật và quản lý dữ liệu: Oracle cung cấp các tính năng bảo mật mạnh mẽ và công cụ quản lý dữ liệu để đảm bảo tính bảo mật và toàn vẹn của dữ liệu.
    - Hỗ trợ đa nền tảng: Oracle hỗ trợ đa nền tảng, cho phép tích hợp dễ dàng với các hệ thống và ứng dụng khác.
  + Quá trình chuyển đổi:
    - Phân tích và thiết kế cơ sở dữ liệu mới: Phải tiến hành một quá trình phân tích kỹ lưỡng để đảm bảo rằng cơ sở dữ liệu mới được thiết kế để đáp ứng đầy đủ các yêu cầu của hệ thống và người dùng.
    - Chuyển dữ liệu: Quá trình chuyển dữ liệu từ MySQL sang Oracle cần được thực hiện một cách cẩn thận để đảm bảo tính toàn vẹn và chính xác của dữ liệu.
    - Kiểm thử và triển khai: Sau khi chuyển đổi, hệ thống cần được kiểm thử kỹ lưỡng để đảm bảo rằng mọi tính năng hoạt động đúng và không gặp sự cố nào trước khi triển khai cho người dùng cuối.

## Mô tả nội dung cần giải quyết của đề tài

* Phạm vi, quy trình, nghiệp vụ mà đề tài sẽ giải quyết:
  + Phạm vi: Xây dựng một hệ thống quản lý bất động sản cho thuê, bao gồm việc đăng ký, quản lý thông tin bất động sản, đặt lịch hẹn, thanh toán và đánh giá.
  + Quy trình: Bao gồm thu thập thông tin, xử lý đăng ký, quản lý dữ liệu bất động sản, tạo lịch hẹn và giao dịch thanh toán.
  + Nghiệp vụ: Tạo và quản lý danh sách bất động sản, đăng tin, tìm kiếm và đặt lịch hẹn, xác nhận thanh toán và đánh giá.
* Các chứng từ, báo cáo, số liệu mà hệ thống cần quản lý:
  + Chứng từ: Hồ sơ thuê nhà, hợp đồng thuê, hóa đơn thanh toán.
  + Báo cáo: Báo cáo về lịch sử giao dịch, doanh thu, khách hàng, và bất động sản đang được quản lý.
  + Số liệu: Thông tin về các bất động sản, khách hàng, giao dịch, và thanh toán.
* Các vấn đề liên quan đến hiệu năng xử lý dữ liệu trong hệ thống:
  + Xử lý tải công việc lớn: Đảm bảo hệ thống có khả năng xử lý đồng thời nhiều giao dịch và truy vấn từ người dùng.
  + Tối ưu hóa cơ sở dữ liệu: Cải thiện hiệu năng của cơ sở dữ liệu để đảm bảo thời gian phản hồi nhanh chóng.
* Các vấn đề về an toàn dữ liệu:
  + Bảo mật dữ liệu cá nhân: Đảm bảo dữ liệu khách hàng và thông tin cá nhân được bảo vệ chặt chẽ.
  + Sao lưu và phục hồi: Thiết lập các chiến lược sao lưu định kỳ và phục hồi dữ liệu để đảm bảo tính toàn vẹn của dữ liệu trong trường hợp xảy ra sự cố.
* Các vấn đề liên quan đến xử lý truy xuất đồng thời:
  + Đồng bộ hóa truy cập: Đảm bảo hệ thống có khả năng xử lý nhiều truy vấn cùng lúc mà không gây ra xung đột hoặc sự chậm trễ.
  + Quản lý tài nguyên: Phân bổ tài nguyên hệ thống một cách hiệu quả để đáp ứng các yêu cầu truy cập đồng thời.
* Lý do sử dụng hệ quản trị Oracle sẽ giải quyết các vấn đề nêu trên:
  + Oracle có khả năng xử lý và quản lý dữ liệu lớn với hiệu suất cao và tính linh hoạt mở rộng.
  + Oracle cung cấp các công cụ và tính năng bảo mật mạnh mẽ để bảo vệ dữ liệu và thông tin cá nhân của khách hàng.
  + Oracle có khả năng xử lý truy xuất đồng thời và tối ưu hóa hiệu năng của cơ sở dữ liệu, giúp đảm bảo thời gian phản hồi nhanh chóng và mượt mà cho người dùng.

# CƠ SỞ LÝ THUYẾT

## Kiến trúc của Oracle Database:

* Instance : Mỗi khi một Oracle Database được khởi động, một instance sẽ được tạo ra. Instance là một tập hợp các quá trình và bộ nhớ được sử dụng để quản lý và truy cập dữ liệu trong cơ sở dữ liệu. Instance là một thực thể riêng biệt, hoạt động độc lập với các instance khác.
* Database: Là nơi lưu trữ tất cả dữ liệu và các đối tượng dữ liệu của hệ thống. Cơ sở dữ liệu bao gồm các tệp dữ liệu , tệp nhật ký, tệp kiểm soát, và các tệp khác. Tất cả các dữ liệu được quản lý bởi Oracle Database đều được lưu trữ trong các tệp dữ liệu.
* Memory Structures: Cung cấp các cấu trúc bộ nhớ để lưu trữ dữ liệu tạm thời và thông tin quản lý của hệ thống. Các cấu trúc bộ nhớ quan trọng bao gồm:
* System Global Area (SGA): Là một vùng bộ nhớ chia sẻ được sử dụng cho toàn bộ instance. SGA chứa dữ liệu tạm thời và thông tin quản lý của hệ thống như bộ đệm bộ dữ liệu, bộ đệm dịch vụ, và bộ nhớ phân vùng.
* Program Global Area (PGA): Là bộ nhớ dành riêng cho mỗi quá trình trong instance. PGA chứa dữ liệu tạm thời và các biến thực thi của các chương trình trong quá trình.
* Process Architecture: Oracle Database sử dụng một số quá trình để thực hiện các nhiệm vụ quản lý, truy vấn và xử lý dữ liệu. Các quá trình quan trọng bao gồm:
* Quá trình chính: Thực hiện các yêu cầu từ ứng dụng và người dùng, và tương tác trực tiếp với người dùng.
* Quá trình phía sau: Thực hiện các nhiệm vụ quản lý và bảo trì hệ thống như quản lý bộ nhớ, ghi nhật ký, phục hồi dữ liệu, và điều phối truy vấn.
* Client/Server Architecture: Cho phép nhiều người dùng và ứng dụng truy cập cùng một cơ sở dữ liệu từ xa thông qua mạng. Trong kiến trúc này, các ứng dụng khách gửi yêu cầu tới máy chủ Oracle, và máy chủ Oracle xử lý yêu cầu và trả lại kết quả cho ứng dụng khách. Điều này cho phép nhiều người dùng cùng truy cập và làm việc với dữ liệu trong cùng một thời điểm.

## Quản trị User và Profile:

* Trong Oracle, quản trị user và profile là quá trình quan trọng để đảm bảo an ninh, quản lý quyền truy cập và tối ưu hóa hiệu suất của cơ sở dữ liệu.
  1. **Quản trị User:**
* Tạo User: Quản trị viên cơ sở dữ liệu có thể tạo mới user bằng cách sử dụng câu lệnh SQL CREATE USER. Khi tạo user, quản trị viên có thể chỉ định tên user, mật khẩu, và các thuộc tính khác như tên bảng không gian mặc định, quyền truy cập và giới hạn tài nguyên.
* Phân quyền User: Sau khi tạo user, quản trị viên có thể gán các quyền truy cập cụ thể cho user đó bằng cách sử dụng câu lệnh GRANT. Quyền truy cập có thể bao gồm quyền truy cập vào bảng, thực thi thủ tục lưu trữ, hoặc thậm chí là quyền quản trị.
* Quản lý User: Quản trị viên có thể quản lý user bằng cách thực hiện các thao tác như khóa, mở khóa, xóa user hoặc thay đổi các thuộc tính của user.
* Kiểm soát Quyền Truy Cập: Oracle cung cấp các cơ chế kiểm soát quyền truy cập như role, giúp quản trị viên dễ dàng quản lý và điều chỉnh quyền truy cập của các nhóm người dùng.
  1. **Quản trị Profile:**
* Tạo Profile: Profile là tập hợp các giới hạn và tùy chọn được áp dụng cho user hoặc role. Quản trị viên có thể tạo profile bằng câu lệnh CREATE PROFILE, và chỉ định các giới hạn như tài nguyên hệ thống, số lần đăng nhập không thành công, thời gian khóa, vv.
* Áp dụng Profile cho User: Sau khi tạo profile, quản trị viên có thể áp dụng profile cho user bằng cách sử dụng câu lệnh ALTER USER. Bằng cách này, user sẽ thừa kế các giới hạn và tùy chọn được xác định trong profile.
* Quản lý Profile: Quản trị viên có thể quản lý profile bằng cách thay đổi các giới hạn và tùy chọn của nó hoặc thậm chí xóa profile không cần thiết.
* Sử dụng Profile để Tối Ưu Hóa Hiệu Suất: Profile có thể được sử dụng để tối ưu hóa hiệu suất của cơ sở dữ liệu bằng cách giới hạn tài nguyên và số lượng session mà mỗi user có thể sử dụng.

## Ngôn ngữ PL/SQL:

PL/SQL là một ngôn ngữ lập trình procedural được tích hợp vào Oracle Database.

* 1. **Tính chất của PL/SQL:**
* Procedural: PL/SQL là một ngôn ngữ lập trình procedural, cho phép bạn viết các chương trình có cấu trúc và tuần tự, với khả năng điều khiển luồng thực thi.
* Embedded: PL/SQL có thể được nhúng vào các câu lệnh SQL, cho phép bạn tạo ra các chương trình hoàn chỉnh để thực hiện các thao tác dữ liệu phức tạp.
* Tích hợp với Oracle Database: PL/SQL được tích hợp sâu vào Oracle Database, cho phép tương tác trực tiếp với dữ liệu và các đối tượng trong cơ sở dữ liệu.
  1. **Cú pháp cơ bản:**
* PL/SQL sử dụng cú pháp tương tự như các ngôn ngữ lập trình procedural khác như C hoặc Pascal.
* Câu lệnh PL/SQL thường kết thúc bằng dấu chấm phẩy (;).
* PL/SQL sử dụng các khối code như DECLARE, BEGIN, EXCEPTION và END để định nghĩa và thực thi chương trình.
  1. **Cấu trúc chương trình PL/SQL:**
* DECLARE: Khai báo các biến, hằng số và các đối tượng khác mà chương trình sẽ sử dụng.
* BEGIN: Bắt đầu một khối code chính của chương trình, nơi thực hiện các lệnh thực thi.
* EXCEPTION: Xử lý các ngoại lệ (exceptions) có thể xảy ra trong quá trình thực thi chương trình.
* END: Kết thúc khối code.
  1. **Các tính năng chính:**
* Biến và hằng số: PL/SQL cho phép khai báo và sử dụng biến và hằng số để lưu trữ dữ liệu.
* Cấu trúc điều khiển: PL/SQL hỗ trợ các cấu trúc điều khiển như IF-ELSE, LOOP, WHILE để kiểm soát luồng thực thi chương trình.
* Thủ tục và hàm: PL/SQL cho phép bạn định nghĩa và sử dụng thủ tục (procedures) và hàm (functions) để tái sử dụng mã nguồn.
* Các câu lệnh SQL nhúng: Bạn có thể sử dụng các câu lệnh SQL nhúng trong mã PL/SQL để thực hiện các thao tác trên dữ liệu trong cơ sở dữ liệu.

## Cơ chế sao lưu và phục hồi dữ liệu sau sự cố:

Cơ chế sao lưu và phục hồi dữ liệu là một phần quan trọng của bất kỳ hệ thống quản lý cơ sở dữ liệu nào, bao gồm cả Oracle.

* 1. **Sao lưu dữ liệu trong Oracle:**
* Sao lưu toàn bộ cơ sở dữ liệu (full database backup): Oracle cho phép bạn tạo ra một bản sao lưu toàn bộ cơ sở dữ liệu, bao gồm tất cả các đối tượng và dữ liệu trong database. Bạn có thể sử dụng công cụ như Oracle Recovery Manager (RMAN) để tạo và quản lý các bản sao lưu này.
* Sao lưu Incremental (incremental backup): Oracle cũng hỗ trợ sao lưu tăng dần, nơi chỉ các thay đổi từ thời điểm sao lưu trước đó được sao lưu. Điều này giúp giảm thời gian và tài nguyên cần thiết cho quá trình sao lưu.
* Sao lưu dữ liệu nhất quán (consistent backup): Oracle đảm bảo rằng các bản sao lưu được tạo ra ở một thời điểm nhất quán, đảm bảo tính toàn vẹn của dữ liệu khi phục hồi.
  1. **Phục hồi dữ liệu trong Oracle:**
* Phục hồi toàn bộ cơ sở dữ liệu (complete database recovery): Khi xảy ra sự cố, như mất dữ liệu hoặc hỏng hóc, bạn có thể sử dụng các bản sao lưu để phục hồi toàn bộ cơ sở dữ liệu về một trạng thái nhất định trước khi sự cố xảy ra.
* Phục hồi tới một thời điểm cụ thể (point-in-time recovery): Oracle cho phép bạn phục hồi cơ sở dữ liệu tới một thời điểm cụ thể trong quá khứ. Điều này giúp bạn khôi phục dữ liệu đến một trạng thái nhất định trước khi một sự kiện không mong muốn xảy ra.
* Phục hồi tập tin nhật ký (log file recovery): Nếu mất dữ liệu gần đây do một lỗi ghi hoặc mất điểm nhật ký, bạn có thể sử dụng tập tin nhật ký để phục hồi các thay đổi đã mất.
  1. **Các công cụ và tiện ích hỗ trợ:**
* Oracle Recovery Manager (RMAN): Là một công cụ quản lý sao lưu và phục hồi được tích hợp sẵn trong Oracle Database. RMAN cung cấp các tính năng mạnh mẽ để tự động hóa quy trình sao lưu và phục hồi.
* Enterprise Manager (EM): Oracle Enterprise Manager cung cấp một giao diện đồ họa dễ sử dụng để quản lý và giám sát quá trình sao lưu và phục hồi.
* Các lệnh SQL và công cụ dòng lệnh: Oracle cung cấp một loạt các lệnh SQL và công cụ dòng lệnh cho phép bạn thực hiện các tác vụ sao lưu và phục hồi dữ liệu trong môi trường dòng lệnh.

## Quản lý giao tác/ giao dịch

* **Khái niệm**: Một giao tác là một chuỗi các hành động thao tác dữ liệu được thực hiện trong cơ sở dữ liệu. Một giao tác có thể bao gồm nhiều lệnh SQL như INSERT, UPDATE, DELETE, vv.
  1. **Quản lý giao tác trong Oracle:**
* Bắt đầu và Kết thúc Giao tác: Giao tác bắt đầu khi bạn thực hiện một lệnh BEGIN hoặc START TRANSACTION, và kết thúc khi bạn thực hiện một lệnh COMMIT hoặc ROLLBACK.
* Lệnh COMMIT: Lệnh COMMIT được sử dụng để lưu trữ các thay đổi trong giao tác vào cơ sở dữ liệu và kết thúc giao tác.
* Lệnh ROLLBACK: Lệnh ROLLBACK được sử dụng để hủy bỏ các thay đổi trong giao tác và quay trở lại trạng thái trước khi giao tác được thực hiện.
* Lệnh SAVEPOINT: SAVEPOINT được sử dụng để đánh dấu một điểm trong giao tác để có thể rollback đến điểm đó nếu cần thiết.
* Kiểm soát đồng thời: Oracle sử dụng các phương pháp kiểm soát đồng thời như locking và MVCC để đảm bảo tính độc lập và nhất quán của các giao tác đang thực hiện đồng thời.
  1. **Quản lý Lỗi và Giao tác Phục hồi:**
* Xử lý Lỗi: Trong trường hợp có lỗi xảy ra trong quá trình thực hiện giao tác, Oracle cung cấp cơ chế xử lý lỗi và rollback tự động để đảm bảo tính nhất quán của dữ liệu.
* Phục hồi Giao tác: Nếu hệ thống gặp sự cố, Oracle cung cấp các công cụ và cơ chế phục hồi để khôi phục giao tác đã được bắt đầu hoặc hoàn thành một cách an toàn và nhất quán.

## Xử Lý đồng thời:

* 1. **Locking Mechanisms (Cơ chế khóa):**
* Oracle sử dụng cơ chế locking để đảm bảo tính nhất quán của dữ liệu trong quá trình thực thi giao dịch đồng thời.
* Các loại khóa bao gồm:
  + **Shared Locks (Khóa chia sẻ)**: Cho phép nhiều giao dịch đọc dữ liệu cùng một lúc.
  + **Exclusive Locks (Khóa độc quyền)**: Được sử dụng khi một giao dịch muốn cập nhật hoặc xóa dữ liệu, không cho phép giao dịch khác đọc hoặc cập nhật dữ liệu đó.
  1. **Multiversion Concurrency Control (MVCC):**
* Oracle sử dụng MVCC để đảm bảo tính độc lập của các giao dịch đồng thời.
* MVCC cho phép giao dịch đọc các phiên bản trước của dữ liệu mà không bị chặn bởi giao dịch cập nhật dữ liệu hiện tại.
  1. **imestamp-Based Concurrency Control:**
* Oracle sử dụng timestamp-based concurrency control để quản lý thứ tự thực thi các giao dịch và đảm bảo tính nhất quán của dữ liệu.
* Mỗi giao dịch được gán một timestamp và quyết định về việc đọc hoặc cập nhật dữ liệu được dựa trên thời gian của các giao dịch.
  1. **Read Consistency:**
* Oracle cung cấp tính năng đọc nhất quán (read consistency) để đảm bảo rằng các giao dịch đọc dữ liệu nhìn thấy một trạng thái nhất quán của dữ liệu, ngay cả khi dữ liệu đang được cập nhật bởi các giao dịch khác.
* Điều này đảm bảo rằng kết quả của mỗi truy vấn là nhất quán và không bị ảnh hưởng bởi các thay đổi đang xảy ra.
  1. **Deadlock Detection and Resolution:**
* Oracle sử dụng các cơ chế phát hiện và giải quyết deadlock để tránh tình trạng deadlock, nơi hai hoặc nhiều giao dịch đang chờ đợi lẫn nhau để giải phóng khóa mà họ cần.

# PHÂN TÍCH HỆ THỐNG

## Phân tích chức năng

# Mô hình phân cấp chức năng của hệ thống (sơ đồ BFD)

* **Về phía Admin (Quản trị viên)**

Bảng 2: Sơ đồ BFD

*Bảng chức năng và các quy định nghiệp vụ/các ràng buộc toàn vẹn dữ liệu phía Admin*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Nghiệp vụ** | **Chức năng** | **Quy định nghiệp vụ** | **Ràng buộc toàn vẹn dữ liệu** |
| 1 | Quản lý người dùng | * Thêm, xóa, sửa người dùng * Phân quyền người dùng | QĐ4, QĐ5, QĐ8 | Sử dụng ràng buộc khóa ngoại để liên kết thông tin người dùng với các bảng khác như tin đăng, lịch sử giao dịch, |
| 2 | Quản lý tin đăng | * Thêm, xóa, sửa tin * Duyệt tin * Đăng tin | QĐ1, QĐ2, QĐ3 | Sử dụng ràng buộc khóa ngoại để liên kết tin đăng với người dùng tạo ra nó và các thông tin khác như loại hình, tình trạng, |
| 3 | Quản lý thanh toán | * Quản lý giao dịch * Xem lịch sử giao dịch | QĐ10 | Sử dụng ràng buộc khóa ngoại để liên kết thông tin thanh toán với người dùng thực hiện và các bảng khác như lịch sử giao dịch |
| 4 | Quản lý danh mục | * Thêm, xóa, sửa danh mục (vd: loại hình, tình trạng, …) | QĐ7 | Sử dụng ràng buộc khóa ngoại để liên kết các danh mục với các bảng khác như tin đăng, người dùng, v.v. |
| 5 | Thống kê, báo cáo | * Thống kê số lượng bài đăng * Báo cáo hoạt động của hệ thống |  | Sử dụng dữ liệu từ các bảng khác để tạo ra thông tin thống kê và báo cáo chính xác. |

* **Về phía Người dùng ( Khách hàng)**

*Bảng chức năng và các quy định nghiệp vụ/các ràng buộc toàn vẹn dữ liệu phía Người dùng*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Nghiệp vụ** | **Chức năng** | **Quy định nghiệp vụ** | **Ràng buộc toàn vẹn dữ liệu** |
| 1 | Xem tin | * Tìm kiếm tin * Lọc tin * Xem chi tiết tin |  | Đảm bảo dữ liệu trong các bảng tin đăng phản ánh chính xác thông tin mà người dùng tìm kiếm. |
| 2 | Quản lý đăng tin mới | * Thêm, xóa, sửa tin mới * Đăng tin |  | Sử dụng dữ liệu của các bảng khác để tạo ra một biểu mẫu cho người dùng nhập các dữ liệu để có thể đăng tin mới. |
| 3 | Quản lý tài khoản | * Đổi mật khẩu * Thêm thông tin tài khoản * Đăng ký, đăng nhập tài khoản | QĐ6, QĐ9 | Đảm bảo thông tin cá nhân của người dùng được lưu trữ an toàn và không bị rò rỉ thông tin bằng cách chuyển đổi mật khẩu sang dạng MD5 hash. |
| 4 | Bình luận và yêu thích tin | * Người dùng có thể thêm tin đăng vào danh sách yêu thích. * Người dùng có thể bình luận về tin đăng. |  | Sử dụng ràng buộc khóa ngoại để liên kết danh sách yêu thích và bình luận với người dùng và tin đăng tương ứng. |

**Bảng qui định:**

|  |  |  |
| --- | --- | --- |
| **STT** | **Số qui định** | **Qui định nghiệp vụ** |
| **1** | **QĐ1** | Mỗi tin đăng phải có một tiêu đề và mô tả |
| **2** | **QĐ2** | Ngày đăng của mỗi tin phải được ghi lại và không thể thay đổi sau khi tin được đăng |
| **3** | **QĐ3** | Tin đăng phải thuộc vào một loại hình nhà trọ hoặc bất động sản cụ thể |
| **4** | **QĐ4** | Mỗi người dùng phải có một tài khoản duy nhất được đăng ký bằng số điện thoại |
| **5** | **QĐ5** | Mỗi người dùng phải có một tài khoản duy nhất được đăng ký bằng số điện thoại |
| **6** | **QĐ6** | Thông tin cá nhân của người dùng như họ tên, địa chỉ, và hình ảnh cá nhân có thể được cập nhật và quản lý |
| **7** | **QĐ7** | Đảm bảo rằng trường "MaLoaiHinh" trong bảng "LOAIHINH" là khóa chính và không chứa giá trị trùng lặp. |
| **8** | **QĐ8** | Đảm bảo rằng trường "MaCV" trong bảng "QUANTRIVIEN" là khóa ngoại tham chiếu đến trường "MaCV" trong bảng "CHUCVU |
| **9** | **QĐ9** | Đảm bảo rằng trường "MatKhau" trong bảng "KHACHHANG" không chấp nhận giá trị null, tức là phải có giá trị |
| **10** | **QĐ10** | Đảm bảo rằng trường "SoTien" trong bảng "LICHSUGD" phải có giá trị không âm (lớn hơn hoặc bằng 0). |

**Ràng buộc toàn vẹn**

**Ràng buộc R1**: “Mỗi tin đăng phải có một tiêu đề và mô tả.”

· Bối cảnh: TIN

· Ngôn ngữ hình thức: ∀t ∈ TIN (t.TIEUDE ≠ null ∧ t.MOTA ≠ null)

· Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R1 | Thêm | Xóa | Sửa |
| TIN | + | - | +(TIEUDE & MOTA) |

**Ràng buộc R2**: “Ngày đăng của mỗi tin phải được ghi lại và không thể thay đổi sau khi tin được đăng.”

· Bối cảnh: TIN

· Ngôn ngữ hình thức: ∀t ∈ TIN (t.NGAYDANG ≠ null )

· Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R2 | Thêm | Xóa | Sửa |
| TIN | + | - | - |

**Ràng buộc R3**: “Tin đăng phải thuộc vào một loại hình nhà trọ hoặc bất động sản cụ thể.”

· Bối cảnh: TIN, LOAIHINH

· Ngôn ngữ hình thức: ∀t ∈ TINDANG (∃l ∈ LOAIHINH (t.LOAIHINH = l))

· Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R3 | Thêm | Xóa | Sửa |
| TIN | + | - | +(LOAIHINH) |

**Ràng buộc R4**: “Mỗi người dùng phải có một tài khoản duy nhất được đăng ký bằng số điện thoại.”

· Bối cảnh: KHACHHANG

· Ngôn ngữ hình thức: ∀u1, u2 ∈ KHACHHANG ((u1.SDT = u2.SDT) → (u1 = u2))

Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R4 | Thêm | Xóa | Sửa |
| KHACHHANG | + | - | +(SDT) |

**Ràng buộc R5**: “Mỗi người dùng phải có một tài khoản duy nhất được đăng ký bằng số điện thoại.”

· Bối cảnh: KHACHHANG

· Ngôn ngữ hình thức: ∀u1, u2 ∈ KHACHHANG ((u1.SDT = u2.SDT) → (u1 = u2))

Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R5 | Thêm | Xóa | Sửa |
| TIN | + | - | +(SDT) |

**Ràng buộc R6**: “Thông tin cá nhân của người dùng như họ tên, địa chỉ, và hình ảnh cá nhân có thể được cập nhật và quản lý.”

· Bối cảnh: KHACHHANG

· Ngôn ngữ hình thức: ∀u ∈ KHACHHANG (u.HOTEN ≠ null ∧ u.DIACHI ≠ null ∧ u.HINHANH ≠ null)

Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R6 | Thêm | Xóa | Sửa |
| KHACHHANG | - | - | +(HOTEN & DIACHI & HINHANH) |

**Ràng buộc R7:** “Đảm bảo rằng trường "MaLoaiHinh" trong bảng "LOAIHINH" là khóa chính và không chứa giá trị trùng lặp.”

· Bối cảnh: LOAIHINH

· Ngôn ngữ hình thức: ∀l1, l2 ∈ LOAIHINH (l1.MALOAIHINH = l2. MALOAIHINH → l1 = l2)

Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R7 | Thêm | Xóa | Sửa |
| LOAIHINH | + | - | +( MALOAIHINH) |

**Ràng buộc R8**: “Đảm bảo rằng trường "MaCV" trong bảng "QUANTRIVIEN" là khóa ngoại tham chiếu đến trường "MaCV" trong bảng "CHUCVU".”

· Bối cảnh: QUANTRIVIEN, CHUCVU

· Ngôn ngữ hình thức: ∀q ∈ QUANTRIVIEN (∃c ∈ CHUCVU (q.MaCV = c.MaCV))

Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R8 | Thêm | Xóa | Sửa |
| CHUCVU | + | - | +( MACV) |

**Ràng buộc R9**: “Đảm bảo rằng trường "MatKhau" trong bảng "KHACHHANG" không chấp nhận giá trị null, tức là phải có giá trị.”

· Bối cảnh: KHACHHANG

· Ngôn ngữ hình thức: ∀k ∈ KHACHHANG (k.MatKhau ≠ NULL)

Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R9 | Thêm | Xóa | Sửa |
| KHACHHANG | + | - | +( MatKhau) |

**Ràng buộc R10**: “Đảm bảo rằng trường "SoTien" trong bảng "LICHSUGD" phải có giá trị không âm (lớn hơn hoặc bằng 0).”

· Bối cảnh: LICHSUGD

· Ngôn ngữ hình thức: ∀l ∈ LICHSUGD (l.SoTien >= 0)

Bảng tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R10 | Thêm | Xóa | Sửa |
| LICHSUGD | + | - | +( SoTien) |

## 2.Phân tích dữ liệu

# A diagram of a flowchart Description automatically generated

Bảng 3: Sơ đồ ERD

**Sơ đồ luồng dữ liệu DFD (Data Flow Diagram)**

# A diagram of a flowchart Description automatically generated

# THIẾT KẾ VÀ CÀI ĐẶT HỆ THỐNG

## Mô hình dữ liệu quan hệ (Relational Data model)

A diagram of a computer program

Description automatically generated with medium confidence

Bảng 4: Mô hình dữ liệu quan hệ (Relational Data model)

## Từ điển dữ liệu (Mô tả các bảng và thuốc tính)

**Bảng Loại Hình:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaLoaiHinh | NUMBER |  | PK |
| 2 | TenLoaiHinh | NVARCHAR | 250 |  |

**Bảng Ví Tiền:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaTT | NUMBER |  | PK |
| 2 | TongTien | NUMBER |  |  |
| 3 | TrangThai | NUMBER | 1 |  |
| 4 | NgayTT | DATE |  |  |

**Bảng Tình Trạng:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaTT | NUMBER |  | PK |
| 2 | TenTT | NVARCHAR | 250 |  |

**Bảng Chức Vụ:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaCV | NUMBER |  | PK |
| 2 | TenCV | NVARCHAR | 250 |  |
| 3 | TrangThai | NUMBER | 1 |  |

**Bảng Quản Trị Viên:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaQTV | NUMBER |  | PK |
| 2 | TaiKhoan | VARCHAR | 250 |  |
| 3 | MatKhau | VARCHAR | 250 |  |
| 4 | HinhAnh | VARCHAR | 250 |  |
| 5 | Email | VARCHAR | 250 |  |
| 6 | MaCV | NUMBER |  | FK |

**Bảng Khách Hàng:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaKH | NUMBER |  | PK |
| 2 | SDT | VARCHAR | 20 |  |
| 3 | MatKhau | VARCHAR | 250 |  |
| 4 | HinhAnh | VARCHAR | 250 |  |
| 5 | HoTen | VARCHAR | 250 |  |
| 6 | MaCV | NUMBER |  | FK |
| 7 | TrangThai | NUMBER |  |  |
| 8 | NgayTao | DATE |  |  |
| 9 | DiaChi | VARCHAR | 250 |  |
| 10 | MaTT | NUMBER |  | FK |
| 11 | NgaySinh | DATE |  |  |

**Bảng Thành Phố:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaTP | NUMBER |  | PK |
| 2 | TenTP | NVARCHAR | 250 |  |

**Bảng Quận:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaQuan | NUMBER |  | PK |
| 2 | TenQuan | NVARCHAR | 250 |  |
| 3 | MaTP | NUMBER |  | FK |

**Bảng Phường:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaPhuong | NUMBER |  | PK |
| 2 | TenPhuong | NVARCHAR | 250 |  |
| 3 | MaQuan | NUMBER |  | FK |

**Bảng Tin:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaTin | NUMBER |  | PK |
| 2 | TieuDe | NVARCHAR | 250 |  |
| 3 | MoTa | NVARCHAR | 1000 |  |
| 4 | NgayDang | DATE |  |  |
| 5 | TrangThai | NUMBER |  |  |
| 6 | DienTich | NUMBER |  |  |
| 7 | HinhAnh | NVARCHAR | 250 |  |
| 8 | TienThue | NUMBER |  |  |
| 9 | TienCoc | NUMBER |  |  |
| 10 | MaLoaiHinh | NUMBER |  | FK |
| 11 | MaTT | NUMBER |  | FK |
| 12 | MaTP | NUMBER |  | FK |
| 13 | MaQuan | NUMBER |  | FK |
| 14 | MaPhuong | NUMBER |  | FK |

**Bảng Bình Luận:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaBl | NUMBER |  | PK |
| 2 | NgayBinhLuan | DATE |  |  |
| 3 | NoiDung | NVARCHAR | 1000 |  |
| 4 | TrangThai | NUMBER |  |  |
| 5 | MaTin | NUMBER |  | FK |
| 6 | MaKH | NUMBER |  | FK |

**Bảng Yêu Thích:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaYT | NUMBER |  | PK |
| 2 | NgayYeuThich | DATE |  | NOTNULL |
| 3 | MaKH | NUMBER |  | FK |
| 4 | MaTin | NUMBER |  | FK |

**Bảng Lịch Sử Giao Dịch:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **DỮ LIỆU** | **KIỂU DỮ LIỆU** | **KÍCH THƯỚC** | **RÀNG BUỘC** |
| 1 | MaLSGD | NUMBER |  | PK |
| 2 | SoTien | NUMBER |  | NOTNULL |
| 3 | MaTT | NUMBER |  | FK |
| 4 | MaTin | NUMBER |  | FK |

## Thiết kế và cài đặt CSDL bằng Oracle

* 1. **Quản lý lưu trữ dữ liệu**
* Mô tả cách tổ chức lưu trữ dữ liệu (Datafile, Tablespace, Index, …)
* Các script tạo Database, Table, Index,
* Dữ liệu demo, …
* Các script backup/restore dữ liệu
  1. **Thiết kế và cài đặt các Stored Procedure/Function cần thiết cho hệ thống**

**Các Stored Procedure dùng để Insert, Update, Delete:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94  95  96  97  98  99  100  101  102  103  104  105  106  107  108  109  110  111  112  113  114  115  116  117  118  119  120  121  122  123  124  125  126  127 | *--THÊM KHÁCH HÀNG VÀ KIỂM TRA RÀNG BUỘC VỚI CHỨC VỤ, TÌNH TRẠNG:*  **CREATE** **OR** **REPLACE** **PROCEDURE** Them\_KhachHang (  p\_MaKH **IN** NUMBER,  p\_SDT **IN** VARCHAR2,  p\_MatKhau **IN** VARCHAR2,  p\_HoTen **IN** VARCHAR2,  p\_HinhAnh **IN** VARCHAR2,  P\_TrangThai **IN** NUMBER,  p\_Ngaytao **IN** DATE,  p\_DiaChi **IN** VARCHAR2,  p\_MaCV **IN** NUMBER,  p\_MaTT **IN** NUMBER,  p\_NgaySinh DATE  ) **IS**  v\_ChucVu NUMBER;  v\_ThanhToan NUMBER;  **BEGIN**  **SELECT** **COUNT**(\*) **INTO** v\_ChucVu **FROM** CHUCVU **WHERE** MaCV = p\_MaCV;  IF v\_ChucVu = 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20001, 'Ch?c v? không t?n t?i.');  **END** IF;  **SELECT** **COUNT**(\*) **INTO** v\_ThanhToan **FROM** VITIEN **WHERE** MaTT = p\_MaTT;  IF v\_ThanhToan = 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20002, 'B?n ch?a có ti?n trong tài kho?n!.');  **END** IF;  **INSERT** **INTO** KHACHHANG (MaKH, SDT, MatKhau, HoTen, HinhAnh, TrangThai, Ngaytao, DiaChi, MaCV, MaTT, NgaySinh)  **VALUES** (p\_MaKH, p\_SDT, p\_MatKhau, p\_HoTen, p\_HinhAnh, P\_TrangThai, p\_Ngaytao, p\_DiaChi, p\_MaCV, p\_MaTT, p\_NgaySinh);    **COMMIT**;  DBMS\_OUTPUT.PUT\_LINE('D? li?u khách hàng ?ã ???c chèn thành công.');  EXCEPTION  **WHEN** DUP\_VAL\_ON\_INDEX **THEN**  RAISE\_APPLICATION\_ERROR(-20003, 'Mã khách hàng b? trùng.');  **WHEN** OTHERS **THEN**  RAISE\_APPLICATION\_ERROR(-20004, 'L?i: ' || SQLERRM);  **END** Them\_KhachHang;  /  *--------------------------------------------------------------*  *--THÊM KHÁCH HÀNG VÀ KIỂM TRA KHÓA CHÍNH CÓ TRÙNG LẶP HAY KHÔNG:*  **CREATE** **OR** **REPLACE** **PROCEDURE** KiemTraKhoa\_KhachHang (  p\_MaKH **IN** NUMBER,  p\_SDT **IN** VARCHAR2,  p\_MatKhau **IN** VARCHAR2,  p\_HoTen **IN** VARCHAR2,  p\_HinhAnh **IN** VARCHAR2,  P\_TrangThai **IN** NUMBER,  p\_Ngaytao **IN** DATE,  p\_DiaChi **IN** VARCHAR2,  p\_MaCV **IN** NUMBER,  p\_MaTT **IN** NUMBER,  p\_NgaySinh DATE  ) **IS**  v\_Count NUMBER;  **BEGIN**  **SELECT** **COUNT**(\*) **INTO** v\_Count **FROM** KHACHHANG **WHERE** MaKH = p\_MaKH;  IF v\_Count > 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20005, 'Mã khách hàng ?ã t?n t?i.');  **ELSE**  **INSERT** **INTO** KHACHHANG (MaKH, SDT, MatKhau, HoTen, HinhAnh, TrangThai, Ngaytao, DiaChi, MaCV, MaTT, NgaySinh)  **VALUES** (p\_MaKH, p\_SDT, p\_MatKhau, p\_HoTen, p\_HinhAnh, P\_TrangThai, p\_Ngaytao, p\_DiaChi, p\_MaCV, p\_MaTT, p\_NgaySinh);  **COMMIT**;  DBMS\_OUTPUT.PUT\_LINE('D? li?u khách hàng ?ã ???c chèn thành công.');  **END** IF;  EXCEPTION  **WHEN** DUP\_VAL\_ON\_INDEX **THEN**  RAISE\_APPLICATION\_ERROR(-20006, 'Mã khách hàng b? trùng l?p.');  **WHEN** OTHERS **THEN**  RAISE\_APPLICATION\_ERROR(-20007, 'L?i: ' || SQLERRM);  **END** KiemTraKhoa\_KhachHang;  /  *--------------------------------------------------------------*  *--CẬP NHẬT MẬT KHẨU CHO KHÁCH HÀNG--*  **CREATE** **OR** **REPLACE** **PROCEDURE** CapNhat\_KhachHang (  P\_MaKH NUMBER,  p\_MatKhau **IN** VARCHAR2  ) **IS**  v\_Count NUMBER;  **BEGIN**  **SELECT** **COUNT**(\*) **INTO** v\_Count **FROM** KHACHHANG **WHERE** MaKH = P\_MaKH;    IF v\_Count = 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20008, 'Không tìm th?y khách hàng có mã ' || P\_MaKH);  **ELSE**    **UPDATE** KHACHHANG  **SET** MatKhau = p\_MatKhau  **WHERE** MaKH = P\_MaKH;  **COMMIT**;  DBMS\_OUTPUT.PUT\_LINE('Thông tin khách hàng ?ã ???c c?p nh?t thành công.');  **END** IF;  EXCEPTION  **WHEN** OTHERS **THEN**  RAISE\_APPLICATION\_ERROR(-20009, 'L?i: ' || SQLERRM);  **END** CapNhat\_KhachHang;  /  *--------------------------------------------------------------*  *--XÓA KHÁCH HÀNG--*  **CREATE** **OR** **REPLACE** **PROCEDURE** Xoa\_KhachHang (  p\_MaKH **IN** NUMBER  ) **IS**  **BEGIN**  **DELETE** **FROM** KHACHHANG **WHERE** MaKH = p\_MaKH;    **COMMIT**;  DBMS\_OUTPUT.PUT\_LINE('Tin ??ng ?ã ???c xóa thành công.');  EXCEPTION  **WHEN** NO\_DATA\_FOUND **THEN**  RAISE\_APPLICATION\_ERROR(-20010, 'Không tìm th?y tin ??ng có mã ' || p\_MaKH);  **WHEN** OTHERS **THEN**  RAISE\_APPLICATION\_ERROR(-20011, 'L?i: ' || SQLERRM);  **END** Xoa\_KhachHang;  / |

**Stored Procedure thủ tục dùng để tính toán:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22 | *--TÍNH TỔNG NGÀY ĐÃ ĐĂNG TIN--*  **CREATE** **OR** **REPLACE** **PROCEDURE** DemTongSoNgayDang (  p\_MaTin **IN** NUMBER,  p\_TongNgay **OUT** NUMBER  ) **IS**  **BEGIN**  **DECLARE**  v\_NgayDang DATE;  **BEGIN**  **SELECT** NgayDang **INTO** v\_NgayDang **FROM** TIN **WHERE** MaTin = p\_MaTin;  p\_TongNgay := TRUNC(SYSDATE) - TRUNC(v\_NgayDang);  EXCEPTION  **WHEN** NO\_DATA\_FOUND **THEN**  RAISE\_APPLICATION\_ERROR(-20001, 'Không tìm th?y tin v?i mã tin ' || p\_MaTin);  **END**;  **END** DemTongSoNgayDang;  / |

**Stored Procedure thủ tục dùng để tìm kiếm:**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19 | *--TÌM KIẾM TIN THEO TIÊU ĐỀ--*  **CREATE** **OR** **REPLACE** **PROCEDURE** TimKiem\_TinDang\_Theo\_TieuDe (  p\_TieuDe **IN** VARCHAR2  ) **IS**  v\_MaTin TIN.MaTin%**TYPE**;  **BEGIN**  *-- Tìm ki?m tin ??ng t? b?ng TIN*  **SELECT** MaTin **INTO** v\_MaTin  **FROM** TIN  **WHERE** TieuDe **LIKE** '%' || p\_TieuDe || '%';  DBMS\_OUTPUT.PUT\_LINE('Tìm th?y tin ??ng có tiêu ?? ch?a: ' || p\_TieuDe || '. Mã tin: ' || v\_MaTin);  EXCEPTION  **WHEN** NO\_DATA\_FOUND **THEN**  RAISE\_APPLICATION\_ERROR(-20014, 'Không tìm th?y tin ??ng v?i tiêu ?? ch?a: ' || p\_TieuDe);  **WHEN** OTHERS **THEN**  RAISE\_APPLICATION\_ERROR(-20015, 'L?i: ' || SQLERRM);  **END** TimKiem\_TinDang\_Theo\_TieuDe;  / |

**Cài đặt các Function cần thiết cho hệ thống:**

* Cập nhật lại tin tức khi người dùng muốn thay đổi:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36 | *--CẬP NHẬT LẠI TIN TỨC--*  **CREATE** **OR** **REPLACE** **FUNCTION** CapNhatTin(  p\_MaTin **IN** NUMBER,  p\_TieuDe **IN** VARCHAR2,  p\_MoTa **IN** VARCHAR2,  p\_NgayDang **IN** DATE,  p\_DienTich **IN** NUMBER,  p\_HinhAnh **IN** VARCHAR2,  p\_TienThue **IN** NUMBER,  p\_TienCoc **IN** NUMBER,  p\_MaLoaiHinh **IN** NUMBER,  p\_MaTT **IN** NUMBER,  p\_MaTP **IN** NUMBER,  p\_MaQuan **IN** NUMBER,  p\_MaPhuong **IN** NUMBER  ) **RETURN** NUMBER  **IS**  **BEGIN**  **UPDATE** TIN  **SET** TieuDe = p\_TieuDe,  MoTa = p\_MoTa,  NgayDang = p\_NgayDang,  DienTich = p\_DienTich,  HinhAnh = p\_HinhAnh,  TienThue = p\_TienThue,  TienCoc = p\_TienCoc,  MaLoaiHinh = p\_MaLoaiHinh,  MaTT = p\_MaTT,  MaTP = p\_MaTP,  MaQuan = p\_MaQuan,  MaPhuong = p\_MaPhuong  **WHERE** MaTin = p\_MaTin;    **RETURN** **SQL**%ROWCOUNT; *-- Tr? v? s? l??ng dòng ?ã ???c c?p nh?t*  **END** CapNhatTin;  / |

* Thêm tin mới khi khách hàng đã điền đủ thông tin:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29 | **CREATE** **OR** **REPLACE** **FUNCTION** ThemTin(  p\_TieuDe **IN** VARCHAR2,  p\_MoTa **IN** VARCHAR2,  p\_NgayDang **IN** DATE,  p\_TrangThai **IN** NUMBER,  p\_DienTich **IN** NUMBER,  p\_HinhAnh **IN** VARCHAR2,  p\_TienThue **IN** NUMBER,  p\_TienCoc **IN** NUMBER,  p\_MaLoaiHinh **IN** NUMBER,  p\_MaTT **IN** NUMBER,  p\_MaTP **IN** NUMBER,  p\_MaQuan **IN** NUMBER,  p\_MaPhuong **IN** NUMBER  ) **RETURN** NUMBER  **IS**  v\_MaTin NUMBER;  **BEGIN**  *-- T?o mã tin m?i*  **SELECT** NVL(**MAX**(MaTin), 0) + 1 **INTO** v\_MaTin **FROM** TIN;  *-- Chèn d? li?u vào b?ng TIN*  **INSERT** **INTO** TIN (MaTin, TieuDe, MoTa, NgayDang, TrangThai, DienTich, HinhAnh, TienThue, TienCoc, MaLoaiHinh, MaTT, MaTP, MaQuan, MaPhuong)  **VALUES** (v\_MaTin, p\_TieuDe, p\_MoTa, p\_NgayDang, p\_TrangThai, p\_DienTich, p\_HinhAnh, p\_TienThue, p\_TienCoc, p\_MaLoaiHinh, p\_MaTT, p\_MaTP, p\_MaQuan, p\_MaPhuong);  *-- Tr? v? mã tin m?i ???c chèn*  **RETURN** v\_MaTin;  **END** ThemTin; |

* Xóa tin

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | **CREATE** **OR** **REPLACE** **FUNCTION** Xoa\_Tin (  p\_MaTin **IN** NUMBER  )**RETURN** NUMBER  **IS**  **BEGIN**  *-- Xóa tin ??ng t? b?ng TIN*  **DELETE** **FROM** TIN **WHERE** MaTin = p\_MaTin;    **COMMIT**;  DBMS\_OUTPUT.PUT\_LINE('Tin ?ã ???c xóa thành công.');  EXCEPTION  **WHEN** NO\_DATA\_FOUND **THEN**  RAISE\_APPLICATION\_ERROR(-20016, 'Không tìm th?y tin ??ng có mã ' || p\_MaTin);  **WHEN** OTHERS **THEN**  RAISE\_APPLICATION\_ERROR(-20017, 'L?i: ' || SQLERRM);  **END** Xoa\_Tin;  / |

* 1. **Thiết kế các Stored Procedure hoặc Trigger**

Các Trigger được thiết kế trong hệ thống theo qui định:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77  78  79  80  81  82  83  84  85  86  87  88  89  90  91  92  93  94 | *--Mọi tin đăng phải có ít nhất một tiêu đề và mô tả--*  **CREATE** **OR** **REPLACE** **TRIGGER** KiemTraTieuDeMoTa  **BEFORE** **INSERT** **OR** **UPDATE** **ON** TIN  **FOR** **EACH** **ROW**  **BEGIN**  IF :**NEW**.TieuDe **IS** **NULL** **OR** :**NEW**.MoTa **IS** **NULL** **THEN**  RAISE\_APPLICATION\_ERROR(-20001, 'Mọi tin đăng phải có ít nhất một tiêu đề và mô tả!');  **END** IF;  **END**;  /  *--Ngày đăng của mọi tin phải được ghi lại và không thể thay đổi sau khi tin được đăng.--*  **CREATE** **OR** **REPLACE** **TRIGGER** KiemTraNgayDang  **BEFORE** **INSERT** **OR** **UPDATE** **ON** TIN  **FOR** **EACH** **ROW**  **BEGIN**  IF INSERTING **AND** :**NEW**.NgayDang **IS** **NULL** **THEN**  RAISE\_APPLICATION\_ERROR(-20001, 'Ngày đăng của mọi tin phải được ghi lại1');  **END** IF;  IF UPDATING **AND** :**OLD**.NgayDang **IS** **NOT** **NULL** **AND** :**NEW**.NgayDang != :**OLD**.NgayDang **THEN**  RAISE\_APPLICATION\_ERROR(-20002, 'Ngày đăng không thể thay đổi sau khi tin được đăng');  **END** IF;  **END**;  /  *--Tin ??ng ph?i thu?c vào m?t lo?i hình nhà tr? ho?c b?t ??ng s?n c? th?--*  **CREATE** **OR** **REPLACE** **TRIGGER** KiemTraLoaiHinh  **BEFORE** **INSERT** **OR** **UPDATE** **ON** TIN  **FOR** **EACH** **ROW**  **BEGIN**  IF :**NEW**.MaLoaiHinh **IS** **NULL** **THEN**  RAISE\_APPLICATION\_ERROR(-20001, 'Tin ??ng ph?i thu?c vào m?t lo?i hình nhà tr? ho?c b?t ??ng s?n c? th?.');  **END** IF;  **END**;  /  *--M?i ng??i dùng ph?i có m?t tài kho?n duy nh?t ???c ??ng ký b?ng s? ?i?n tho?i.--*  **CREATE** **OR** **REPLACE** **TRIGGER** KiemTraTaiKhoan  **BEFORE** **INSERT** **OR** **UPDATE** **ON** QUANTRIVIEN  **FOR** **EACH** **ROW**  **DECLARE**  v\_Count NUMBER;  **BEGIN**  *-- Ki?m tra xem s? ?i?n tho?i m?i (:NEW.SDT) ?ã t?n t?i trong b?ng QUANTRIVIEN hay không*  **SELECT** **COUNT**(\*) **INTO** v\_Count **FROM** QUANTRIVIEN **WHERE** SDT = :**NEW**.SDT;  *-- N?u s? ?i?n tho?i ?ã t?n t?i, kích ho?t m?t l?i ?ng d?ng*  IF v\_Count > 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20001, 'M?i ng??i dùng ph?i có m?t tài kho?n duy nh?t ???c ??ng ký b?ng s? ?i?n tho?i.');  **END** IF;  **END**;  /  *--??m b?o r?ng tr??ng "MaCV" trong b?ng "QUANTRIVIEN" là khóa ngo?i tham chi?u ??n tr??ng "MaCV" trong b?ng "CHUCVU".--*  **CREATE** **OR** **REPLACE** **TRIGGER** KiemTraKhaoNgoai  **BEFORE** **INSERT** **OR** **UPDATE** **ON** QUANTRIVIEN  **FOR** **EACH** **ROW**  **DECLARE**  v\_ChucVu NUMBER;  **BEGIN**  *-- Ki?m tra xem MaCV m?i (:NEW.MaCV) có t?n t?i trong b?ng CHUCVU hay không*  **SELECT** **COUNT**(\*) **INTO** v\_ChucVu **FROM** CHUCVU **WHERE** MaCV = :**NEW**.MaCV;  *-- N?u không t?n t?i, kích ho?t m?t l?i ?ng d?ng*  IF v\_ChucVu = 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20002, 'Tr??ng "MaCV" ph?i là m?t khóa ngo?i tham chi?u ??n tr??ng "MaCV" trong b?ng "CHUCVU".');  **END** IF;  **END**;  /  *--??m b?o r?ng tr??ng "MatKhau" trong b?ng "KHACHHANG" không ch?p nh?n giá tr? null, t?c là ph?i có giá tr?--*  **CREATE** **OR** **REPLACE** **TRIGGER** Trg\_KiemTraMatKhau  **BEFORE** **INSERT** **OR** **UPDATE** **ON** KHACHHANG  **FOR** **EACH** **ROW**  **BEGIN**  *-- Ki?m tra giá tr? c?a tr??ng MatKhau không ???c null*  IF :**NEW**.MatKhau **IS** **NULL** **THEN**  RAISE\_APPLICATION\_ERROR(-20001, 'Tr??ng MatKhau không ???c null. Vui lòng nh?p m?t kh?u.');  **END** IF;  **END**;  /  *--??m b?o r?ng tr??ng "SoTien" trong b?ng "LICHSUGD" ph?i có giá tr? không âm (l?n h?n ho?c b?ng 0).--*  **CREATE** **OR** **REPLACE** **TRIGGER** Trg\_KiemTraSoTien  **BEFORE** **INSERT** **OR** **UPDATE** **ON** LICHSUGD  **FOR** **EACH** **ROW**  **BEGIN**  *-- Ki?m tra giá tr? c?a tr??ng SoTien không âm*  IF :**NEW**.SoTien < 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20002, 'Giá tr? c?a tr??ng SoTien ph?i l?n h?n ho?c b?ng 0.');  **END** IF;  **END**;  / |

Các thủ tục được thiết kế trong hệ thống:

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60 | *--M?i ng??i dùng ph?i có m?t tài kho?n duy nh?t ???c ??ng ký b?ng s? ?i?n tho?i.--*  **CREATE** **OR** **REPLACE** **PROCEDURE** KiemTraTaiKhoan (  p\_SDT **IN** VARCHAR2  ) **IS**  v\_Count NUMBER;  **BEGIN**  *-- Ki?m tra xem s? ?i?n tho?i ?ã t?n t?i trong b?ng KHACHHANG hay không*  **SELECT** **COUNT**(\*) **INTO** v\_Count **FROM** KHACHHANG **WHERE** SDT = p\_SDT;    *-- N?u s? ?i?n tho?i ?ã t?n t?i, hi?n th? thông báo l?i*  IF v\_Count > 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20003, 'S? ?i?n tho?i ?ã ???c s? d?ng ?? ??ng ký tài kho?n khác.');  **END** IF;  **END** KiemTraTaiKhoan;  /  *--Thông tin cá nhân c?a ng??i dùng nh? h? tên, ??a ch?, và hình ?nh cá nhân có th? ???c c?p nh?t và qu?n lý--*  **CREATE** **OR** **REPLACE** **PROCEDURE** CapNhatThongTinCaNhan (  p\_MaKH **IN** NUMBER,  p\_HoTen **IN** VARCHAR2,  p\_DiaChi **IN** VARCHAR2,  p\_HinhAnh **IN** VARCHAR2  ) **IS**  **BEGIN**  *-- C?p nh?t thông tin cá nhân c?a ng??i dùng*  **UPDATE** KHACHHANG  **SET** HoTen = p\_HoTen,  DiaChi = p\_DiaChi,  HinhAnh = p\_HinhAnh  **WHERE** MaKH = p\_MaKH;    **COMMIT**;  DBMS\_OUTPUT.PUT\_LINE('Thông tin cá nhân ?ã ???c c?p nh?t thành công.');  EXCEPTION  **WHEN** NO\_DATA\_FOUND **THEN**  RAISE\_APPLICATION\_ERROR(-20010, 'Không tìm th?y ng??i dùng có mã ' || p\_MaKH);  **WHEN** OTHERS **THEN**  RAISE\_APPLICATION\_ERROR(-20011, 'L?i: ' || SQLERRM);  **END** CapNhatThongTinCaNhan;  /  *--??m b?o r?ng tr??ng "MaLoaiHinh" trong b?ng "LOAIHINH" là khóa chính và không ch?a giá tr? trùng l?p.--*  **CREATE** **OR** **REPLACE** **TRIGGER** Trg\_KiemTraMaLoaiHinh  **BEFORE** **INSERT** **OR** **UPDATE** **ON** LOAIHINH  **FOR** **EACH** **ROW**  **DECLARE**  v\_Count NUMBER;  **BEGIN**  *-- Ki?m tra xem giá tr? MaLoaiHinh m?i ???c chèn ho?c c?p nh?t có t?n t?i trong b?ng không*  **SELECT** **COUNT**(\*)  **INTO** v\_Count  **FROM** LOAIHINH  **WHERE** MaLoaiHinh = :**NEW**.MaLoaiHinh;  *-- N?u t?n t?i giá tr? MaLoaiHinh trùng l?p, báo l?i*  IF v\_Count > 0 **THEN**  RAISE\_APPLICATION\_ERROR(-20001, 'Giá tr? MaLoaiHinh ?ã t?n t?i và không th? trùng l?p.');  **END** IF;  **END**;  / |

* 1. **Thiết kế và cài đặt các giao tác (Transaction)**
* Mô tả các tình huống cập nhật dữ liệu có thể gây rủi ro về tính nhất quán dữ liệu
* Mô tả cách ngăn ngừa bằng các Transaction
  1. **Minh họa vấn đề xử lý đồng thời**
* Mô tả các tình huống tranh chấp dữ liệu có thể xảy ra gữa các thao tác
* Cài đặt mô phỏng các tình huống này (tại Server, hoặc từ phía Client)
  1. **Thiết kế và cài đặt mô hình phân quyền của hệ thống**
* Thiết kế phân quyền cụ thể cho user, áp dụng dụng các profile , …

# KẾT LUẬN

* Kết quả đạt được và chưa đạt được của đề tài
* Hướng phát triển mở rộng ứng dụng trong tương lai.

# TÀI LIỆU THAM KHẢO

* Sử dụng chức năng tạo danh mục tài liệu tham khảo của MS Word và dùng Style tham chiếu là IEEE.

# PHỤ LỤC

* Bảng mô tả công việc được phân công của từng thành viên trong nhóm