**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH**



**BÁO CÁO BÀI TẬP LỚN**

**ĐỀ TÀI:**

**Nghiên cứu và XÂY DỰNG ứng dụng QUẢN LÝ PHÒNG TRỌ**

Giảng viên hướng dẫn: ThS. Trần Thị Dung

Nhóm sinh viên thực hiện:

- Đỗ Quốc Tuấn MSV: 5951071118

- Trần Quang Trường 5951071114

- Nguyễn Mai Chí Trung 5951071112

- Đặng Văn Phước 6051071096

Lớp: CQ.59.CNTT/CQ.60.CNTT

TP. Hồ Chí Minh, năm 2020

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI**

**PHÂN HIỆU TẠI THÀNH PHỐ HỒ CHÍ MINH**



**BÁO CÁO BÀI TẬP LỚN**

**ĐỀ TÀI:**

**Nghiên cứu và XÂY DỰNG ứng dụng QUẢN LÝ PHÒNG TRỌ**

Giảng viên hướng dẫn: ThS. Trần Thị Dung

Nhóm sinh viên thực hiện:

- Đỗ Quốc Tuấn MSV: 5951071118

- Trần Quang Trường 5951071114

- Nguyễn Mai Chí Trung 5951071112

- Đặng Văn Phước 6051071096

Lớp: CQ.59.CNTT/CQ.60.CNTT

TP. Hồ Chí Minh, năm 2020

**TRƯỜNG ĐẠI HỌC GIAO THÔNG VẬN TẢI CỘNG HÒA XÃ HỘI CHỦ NGHĨA VIỆT NAM**

**PHÂN HIỆU TẠI TP. HỒ CHÍ MINH Độc lập – Tự do – Hạnh phúc**

**GIỚI THIỆU SƠ LƯỢC**

BỘ MÔN: **CÔNG NGHỆ THÔNG TIN**

-------\*\*\*-------

Lớp: CQ.59.CNTT/CQ.60.CNTT

- Đỗ Quốc Tuấn MSV: 5951071118

- Trần Quang Trường 5951071114

- Nguyễn Mai Chí Trung 5951071112

- Đặng Văn Phước 6051071096

1. **Tên đề tài**

NGHIÊN CỨU VÀ XÂY DỰNG ỨNG DỤNG QUẢN LÝ PHÒNG TRỌ

1. **Mục đích, yêu cầu**
   1. **Mục đích:**

Nhằm giúp các chủ nhà trọ, cơ sở cho thuê phòng quản lí hiệu quả công việc. Công nghệ thông tin phát triển nên nhu cầu cần có một phần mềm để tối ưu hóa hiệu suất, nâng cao hiệu quả làm việc, tiết kiệm thời gian, chi phí ghi chép bằng tay thủ công. Hệ thống được xây dựng trên nền tảng ứng dụng Desktop.

* 1. **Yêu cầu:**
* **Yêu cầu công nghệ**
  + Sử dụng ngôn ngữ lập trình Java.
  + Sử dụng công cụ NetBeans.
  + Hệ quản trị cơ sở dữ liệu SQL server.
* **Yêu cầu chức năng**
  + Ứng dụng có đầy đủ các chức năng như quản lý khách thuê, quản lý các dịch vụ cho thuê, cho phép đặt phòng, thanh toán hóa đơn,...
  + Ứng dụng cho phép người dùng: quản lý phòng và khách thuê, thống kê hóa đơn,…
* **Yêu cầu phi chức năng**
  + Tốc độ: Chương trình được lập trình để tối ưu về tốc độ, sử dụng băng thông hiệu quả. Tốc độ hiển thị dữ liệu đc tối ưu hiệu quả trên ứng dụng.
  + Giao diện: Thân thiện với người dùng và dễ dàng thao tác.

1. **Nội dung và phạm vi đề tài**
   1. **Nội dung:**
      * Tổng quan bài toán.
      * Tổng quan về các công nghệ đang sử dụng.
      * Phân tích và thiết kế hệ thống.
      * Lập trình xây dựng.
      * Kiểm thử và chạy thực nghiệm.
      * Kết quả thu được
   2. **Phạm vi:**
      * Nghiên cứu công cụ NetBeans và hệ quản trị cơ sở dữ liệu SQL server.
      * Các thư viện hổ trợ trong việc hiển thị và xử lý dữ liệu: JDBC, JCalendar, JFreeChart,…
2. **Công nghệ, công cụ và ngôn ngữ lập trình**
   * Sử dụng ngôn ngữ lập trình Java.
   * Sử dụng công cụ NetBeans.
   * Hệ quản trị cơ sở dữ liệu SQL server.
3. **Các kết quả chính dự kiến sẽ đạt được và ứng dụng**
   * + Xây dựng thành công ứng dụng quản lý phòng trọ đầy đủ các chức năng đã nêu trên.
4. **Giáo viên và cán bộ hướng dẫn**

Họ tên: ThS. Trần Thị Dung

Đơn vị công tác: Phân hiệu Trường Đại học Giao thông Vận tải.

# **LỜI CẢM ƠN**

Lời đầu tiên chúng em xin gửi lời cảm ơn chân thành đến cô Dung trong **Bộ**

**môn Công nghệ thông tin – Phân hiệu Trường Đại học Giao thông vận tải.**

Những người đã truyền dạy, đã trang bị cho chúng em kho tàng kiến thức về bầu trời công nghệ thông tin rộng lớn.

Ở đây, chúng em không chỉ học được kiến thức về sách vở mà chúng em còn học được các bài học, kỷ năng sống trước khi tạm biệt mái trường đại học thân yêu này và tiến ra biển đời mênh mông rộng lớn. Đặc biệt, chúng em xin gửi lời cảm ơn chân thành và sâu sắc đến cô Trần Thị Dung, người đã đồng hành cùng chúng em trong suốt quá trình làm đồ án, người đã bỏ thời gian quý báu, thậm chí là thời gian nghỉ ngơi để hướng dẫn, để định hướng đường đi nước bước cho chúng em. Chúng em thật chẳng biết dùng lời nào để diễn tả được công lao của cô. “**CẢM ƠN CÔ**. **CẢM ƠN CÔ VỀ TẤT CẢ**”.

Trong quá trình học tập và tìm hiểu chúng em đã nổ lực rất nhiều với mong muốn hoàn thành bài tập lớn một cách tốt nhất, nhưng đời người sẽ có những thiếu sót không thể tránh khỏi, và với những người chưa chững chạc và trưởng thành như chúng em thì sai lầm là không thể không mắc phải. Chúng em mong cô có thể thông cảm và cho chúng em những ý kiến, đóng góp để chúng em có thể hoàn thành Bài tập lớn của mình một cách trọn vẹn nhất.

Sau cùng, chúng em xin kính chúc Cô lời chúc sức khoẻ, luôn hạnh phúc và thành công hơn nữa trong công việc cũng như trong cuộc sống.

Chúng em xin chân thành cảm ơn!

# **NHẬN XÉT CỦA GIÁO VIÊN**

*Tp. Hồ Chí Minh, ngày … tháng … năm …*

Giáo viên hướng dẫn

**MỤC LỤC**

**[LỜI CẢM ƠN 1](#_Toc1179)**

**[NHẬN XÉT CỦA GIÁO VIÊN 2](#_Toc31586)**

**[MỞ ĐẦU 1](#_Toc30150)**

**[1. Lý do chọn đề tài. 1](#_Toc17058)**

**[2. Mục tiêu nghiên cứu. 1](#_Toc24425)**

**[3. Đối tượng và phạm vi nghiên cứu. 1](#_Toc6324)**

**[4. Phương pháp nghiên cứu. 1](#_Toc12515)**

**[CHƯƠNG 1: CƠ SỞ LÝ THUYẾT 3](#_Toc852)**

**[1.1 SQL SERVER: 3](#_Toc17778)**

**[1.1.1 Giới thiệu. 3](#_Toc29559)**

**[1.1.2 Khái niệm. 3](#_Toc27194)**

**[1.1.3 Các thành phần cơ bản trong SQL Server. 3](#_Toc10662)**

**[1.2 Ngôn ngữ lập trình Java. 4](#_Toc19892)**

**[CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ 8](#_Toc4677)**

**[2.1 Đặc tả bài toán. 8](#_Toc27478)**

**[2.2 Mô hình ERD. 9](#_Toc264)**

**[2.3 Lược đồ quan hệ. 9](#_Toc23984)**

**[2.4 Mô hình quan hệ. 10](#_Toc11561)**

**[2.5 Từ điển dòng dữ liệu. 11](#_Toc4702)**

**[2.6 Ràng buộc toàn vẹn. 15](#_Toc7186)**

**[2.6.1 Ràng buộc về miền giá trị của thuộc tính 15](#_Toc30319)**

**[2.6.2 Ràng buộc toàn vẹn liên thuộc tính: 16](#_Toc22172)**

**[2.6.3 Ràng buộc toàn vẹn liên bộ: 17](#_Toc10043)**

**[2.6.4 Ràng buộc phụ thuộc tồn tại: 19](#_Toc25880)**

**[2.6.5 Ràng buộc liên bộ - liên quan hệ: 21](#_Toc21187)**

**[2.6.6 Ràng buộc liên thuộc tính - liên quan hệ: 21](#_Toc12467)**

**[2.7 Phụ thuộc hàm 22](#_Toc3069)**

**[2.8 Truy vấn SQL và đại số quan hệ. 23](#_Toc7421)**

**[2.9 Chuẩn hóa quan hệ 28](#_Toc15055)**

**[2.10 Cài đặt 3](#_Toc662)0**

**[2.10.1 Tạo cơ sở dữ liệu và tạo bảng 3](#_Toc19579)0**

**[2.10.2 Thêm dữ liệu vào các bảng 32](#_Toc15436)**

**[2.10.3 Tạo khóa ngoại cho các bảng 34](#_Toc12489)**

**[2.10.4 Kiểm tra ràng buộc toàn vẹn cho các bảng 35](#_Toc16948)**

**[CHƯƠNG 3: THỬ NGHIỆM ỨNG DỤNG 35](#_Toc20794)**

**[3.1 Môi trường cài đặt 35](#_Toc27268)**

**[3.2 Giao diện đăng nhập 35](#_Toc23295)**

**[3.3 Giao diện đăng ký hợp đồng 36](#_Toc26165)**

**[3.4 Giao diện thêm khách hàng mới 38](#_Toc14737)**

**[3.5 Giao diện thêm mới phòng và hiện thị danh sách phòng 39](#_Toc32692)**

**[3.6 Giao diện thêm dịch vụ và thiết bị 39](#_Toc20248)**

**[3.7 Giao diện thống kê 40](#_Toc5168)**

**[3.8 Giao diện nếu người dùng muốn đổi mật khẩu 41](#_Toc8458)**

**[3.9 Giao diện khách hàng 42](#_Toc17480)**

**[3.10 Giao diện danh sách thuê phòng 42](#_Toc8861)**

**[3.11 Kết quả và đánh giá 43](#_Toc32543)**

**[TÀI LIỆU THAM KHẢO 44](#_Toc21514)**

# **MỞ ĐẦU**

1. **Lý do chọn đề tài.**

Chúng ta, thế hệ đang sống trong thời đại công nghiệp hóa, hiện đại hóa. Thời đại mà công nghệ thông tin đang ngày một phát triển như vũ bão. Nó có mặt hầu hết trong tất cả các lĩnh vực mà chúng ta đang sinh hoạt, làm việc và cũng như giải trí….Nó giúp đỡ cho chúng ta từ nông nghiệp, công nghiệp, kinh tế, giáo dục, y tế.

Chính vì thế ứng dụng quản lý nhà trọ là một giải pháp tối ưu nhằm :

* + Tiết kiệm thời gian tối đa
  + Tiết kiệm chi phí
  + Quản lý hiệu quả, khoa học
  + Tạo ra sự tiện ích cho chủ trọ

**Hướng tiếp cận của đề tài.**

* Nghiên cứu và hiểu được cách thức hoạt động của database.
* Thiết kế cơ sở dữ liệu sao cho phù hợp với ứng dụng đang phát triển.
* Nắm vững kiến thức nền tảng của ngôn ngữ lập trình java, tiến đến phát triển ứng dụng.
* Tiến hành kiểm tra và chạy thử ứng dụng.

1. **Mục tiêu nghiên cứu.**

* Phát triển thành công ứng dụng quản lý phòng trọ.
* Hiểu được cách thức hoạt động của cơ sở dữ liệu.
* Khiến ứng dụng trơ nên có ích hơn cho xã hội.

1. **Đối tượng và phạm vi nghiên cứu.**

* Đối tượng: Các chủ phòng trọ, dãy trọ.
* Phạm vi ứng dụng: Mong muốn ứng dụng có thể có mặt hầu hết trong các phòng trọ, dãy trọ.

1. **Phương pháp nghiên cứu.**

* Tài liệu: Tìm hiểu và khai thác các tài liệu liên quan đến cơ sở dữ liệu SQL Server, các nền tảng kiến thức về Java cũng như công cụ NetBeans.
* Thực hành: Tiến hành kết nối cơ sở dữ liệu SQL Server với NetBeans. Tiến hành thao tác với dữ liệu.

# **CHƯƠNG 1: CƠ SỞ LÝ THUYẾT**

1. **SQL SERVER:**
2. **Giới thiệu.**

SQL server chính là cụm từ viết tắt của Structure Query Language được sử dụng nhiều trong các lĩnh vực bởi chức năng quản lý dữ liệu. Các ngôn ngữ cấp cao như: Visual C, Oracle, Visual Basic,… đều có trình hỗ trợ là SQL. Những ứng dụng khi chạy phải sử dụng SQL khi người dùng truy cập tới cơ sở dữ liệu thì không cần sử dụng trực tiếp SQL.

1. **Khái niệm.**

SQL Server chính là một hệ quản trị dữ liệu quan hệ sử dụng câu lệnh SQL để trao đổi dữ liệu giữa máy cài SQL Server và máy Client. Một Relational Database Management System – RDBMS gồm có: databases, datase engine và các chương trình ứng dụng dùng để quản lý các bộ phận trong RDBMS và những dữ liệu khác.

1. **Các thành phần cơ bản trong SQL Server.**

Các thành cơ bản trong SQL Server gồm có: Reporting Services, Database Engine, Integration Services, Notification Services, Full Text Search Service,… Tất cả kết hợp với nhau tạo thành một giải pháp hoàn chỉnh giúp cho việc phân tích và lưu trữ dữ liệu trở nên dễ dàng hơn.

**+ Database Engine:** Đây là một engine có khả năng chứa dữ liệu ở các quy mô dưới dạng support và table. Ngoài ra, nó còn có khả năng tự điều chỉnh ví dụ: trả lại tài nguyên cho ệ điều hành khi một user log off và sử dụng thêm các tài nguyên của máy khi cần.

**+ Integration Services:** là tập hợp các đối tượng lập trình và các công cụ đồ họa cho việc sao chép, di chuyển và chuyển đổi dữ liệu. Khi bạn làm việc trong một công ty lớn thì dữ liệu được lưu trữ ở nhiều nơi khác nhau như được chứa trong: Oracle, SQL Server, DB2, Microsoft Access,… và bạn chắc chắn sẽ có nhu cầu di chuyển dữ liệu giữa các server này. Ngoài ra, bạn còn muốn định dạng dữ liệu trước khi lưu vào database. Chắc chắn Integration Services sẽ giúp bạn giải quyết được công việc này dễ dàng.

+ **Analysis Services:** Đây là một dịch vụ phân tích dữ liệu rất hay của Microsoft. Dữ liệu khi được lưu trữ vào trong database mà bạn không thể lấy được những thông tin bổ ích thì coi như không có ý nghĩa gì. Chính vì thế, công cụ này ra đời giúp bạn trong việc phân tích dữ liệu một cách hiệu quả và dễ dàng bằng cách dùng kỹ thuật khai thác dữ liệu – datamining và khái niệm hình khối nhiều chiều – multi dimendion cubes.

**+ Notification Services:** Dịch vụ thông báo này là nền tảng cho sự phát triển và triển khai các ứng dụng soạn và gửi thông báo. Ngoài ra, dịch vụ này còn có chức năng gửi thông báo theo dịch thời đến hàng ngàn người dăng ký sử dụng trên nhiều loại thiết bị khác nhau.

+ **Reporting Services:** là một công cụ tạo, quản lý và triển khai báo cáo bao gồm: server và client. Ngoài ra, nó còn là nền tảng cho việc phát triển và xây dựng các ứng dụng báo cáo.

**+ Full Text Search Service:** là một thành phần đặc biệt trong việc truy vấn và đánh chỉ mục dữ liệu văn bản không cấu trúc được lưu trữ trong các cơ sở dữ liệu SQL Server.

**+ Service Broker:** là một môi trường lập trình cho việc tạo ra các ứng dụng trong việc nhảy qua các Instance.

1. **Ngôn ngữ lập trình Java.**
2. **Giới thiệu.**

Ngôn ngữ [lập trình Java](http://hoclaptrinhweb.org/hoc-lap-trinh/hoc-lap-trinh-java.html" \t "_blank) ban đầu được phát triển bởi Sun Microsystems do James Gosling khởi xướng và phát hành vào năm 1995 (Java 1.0 [J2SE]). Với ưu thế về đa nền tảng (multi platform) Java càng lúc càng được ứng dụng rộng rãi trên nhiều thiết bị từ máy tính đến mobile và nhiều thiết bị phần cứng khác...

1. **Khái niệm.**

Java là một trong những ngôn ngữ lập trình hướng đối tượng, được thiết kế độc lập với hệ điều hành, cho phép người lập trình viết chương trình một lần và có thể sử dụng tại bất kỳ đâu. Nó được sử dụng trong phát triển phần mềm, trang web, game hay ứng dụng trên các thiết bị di động.

1. **Các đặc điểm của ngôn ngữ lập trình java**.

* **Hướng đối tượng** – Trong Java, mọi thứ đều là một Object. Java có thể dễ dàng mở rộng và bảo trì vì nó được xây dựng dựa trên mô hình Object.
* ***Tính trừu tượng (Abstraction)*:** là tiến trình xác định và nhóm các thuộc tính, các hành động liên quan đến một thực thể đặc thù, xét trong mối tương quan với ứng dụng đang phát triển.
* ***Tính đa hình (Polymorphism):*** cho phép một phương thức có các tác động khác nhau trên nhiều loại đối tượng khác nhau. Với tính đa hình, nếu cùng một phương thức ứng dụng cho các đối tượng thuộc các lớp khác nhau thì

nó đưa đến những kết quả khác nhau. Bản chất của sự việc chính là phương thức này bao gồm cùng một số lượng các tham số.

* ***Tính kế thừa (Inheritance)*:**Điều này cho phép các đối tượng chia sẻ hay mở rộng các đặc tính sẵn có mà không phải tiến hành định nghĩa lại.
* ***Tính đóng gói (Encapsulation)*:**  là tiến trình che giấu việc thực thi những chi tiết của một đối tượng đối với người sử dụng đối tượng ấy.
* **Nền tảng độc lập** – Không giống nhiều ngôn ngữ lập trình khác bao gồm cả C và C ++, khi Java được biên dịch, nó không được biên dịch thành ngôn ngữ máy nền tảng cụ thể, thay vào mã byte – nền tảng độc lập. Mã byte này được thông dịch bởi máy ảo (JVM) trên nền tảng nào đó mà nó đang chạy.
* **Đơn giản** – Java được thiết kế để dễ học. Nếu bạn hiểu khái niệm cơ bản về OOP Java, sẽ rất dễ để trở thành master về java.
* **Bảo mật** – Với tính năng an toàn của Java, nó cho phép phát triển các hệ thống không có virut, giả mạo. Các kỹ thuật xác thực dựa trên mã hoá khóa công khai.
* **Kiến trúc – trung lập** – Trình biên dịch Java tạo ra định dạng tệp đối tượng kiến trúc trung lập, làm cho mã biên dịch được thực thi trên nhiều bộ vi xử lý, với sự hiện diện của hệ điều hành Java.
* **Portable** – Là kiến trúc tập trung và không có khía cạnh thực hiện phụ thuộc của đặc tả này làm cho Java khả chuyển. Trình biên dịch trong Java được viết bằng ANSI C, đó là một tập con POSIX.
* **Mạnh mẽ** – Java làm nỗ lực để loại trừ các tình huống dễ bị lỗi bằng cách kiểm tra lỗi tại thời gian biên dịch và kiểm tra lỗi tại runtime.
* **Đa luồng** – Với tính năng đa luồng của Java có thể viết các chương trình có thể thực hiện nhiều tác vụ đồng thời. Tính năng thiết kế này cho phép các nhà phát triển xây dựng các ứng dụng tương tác có thể chạy trơn tru hơn.
* **Thông dịch** – Mã byte Java được dịch trực tiếp tới các máy tính gốc và không được lưu trữ ở bất cứ đâu.
* **Hiệu năng cao** – Với việc sử dụng trình biên dịch Just-In-Time, Java cho phép thực hiện hiệu năng cao.
* **Phân tán** – Java được thiết kế cho môi trường phân tán của Internet.
* **Năng động** – Java là năng động hơn C hoặc C++ vì nó được thiết kế để thích nghi với môi trường đang phát triển. Các chương trình Java có thể mang một lượng lớn thông tin tại runtime mà có thể được sử dụng để xác minh và giải quyết các truy cập vào các đối tượng tại runtime.

1. **NETBEANS.**
2. **Giới thiệu.**

Bạn sẽ ngạc nhiên về những gì mà cộng đồng mã nguồn mở đem lại cho bạn. Nếu như Linux phải cần hơn 10 năm để có thể tiến gần đến mục tiêu là đối trọng của Windows cho dòng máy để bàn thì các ứng dụng IDE (Integrated Development Environment- môi trường phát triển tích hợp) chỉ mất một nữa thời gian để có thể là đối thủ tương xứng với các dòng sản phẩm thương mại danh tíếng như Borland Jbuilder, Code Warrior, Oracle JDeveloper ...Một trong những nguyên nhân chính giúp các sản phẩm mã nguồn mở trong dòng sản phẩm này phát triển như vũ bão như hiện nay là sự hỗ trợ từ các đại gia trong giới công nghệ, ví dụ như trong sản phẩm NetBeans là sự xuất hiện của Sun Microsystems và trong sản phẩm Eclipse là “ông trùm” IBM.

1. **Khái niệm.**

NetBeans là một môi trường phát triển tích hợp (IDE) cho Java. NetBeans cho phép các ứng dụng được phát triển từ một tập hợp các thành phần phần mềm được gọi là modules. NetBeans chạy trên Windows, macOS, Linux và Solaris. Ngoài việc phát triển Java, nó còn có các phần mở rộng cho các ngôn ngữ khác như PHP, C, C++, HTML5, và JavaScript. Các ứng dụng dựa trên NetBeans, bao gồm NetBeans IDE, có thể được mở rộng bởi các nhà phát triển bên thứ ba.

1. **Các chức năng chính của NetBeans**

Là công cụ tiện dụng và mạnh mẽ với chức năng chỉnh sửa nhanh và thông minh cung cấp các trình soạn thảo, trình hướng dẫn và các mẫu để giúp bạn tạo các ứng dụng bằng Java và nhiều ngôn ngữ khác.

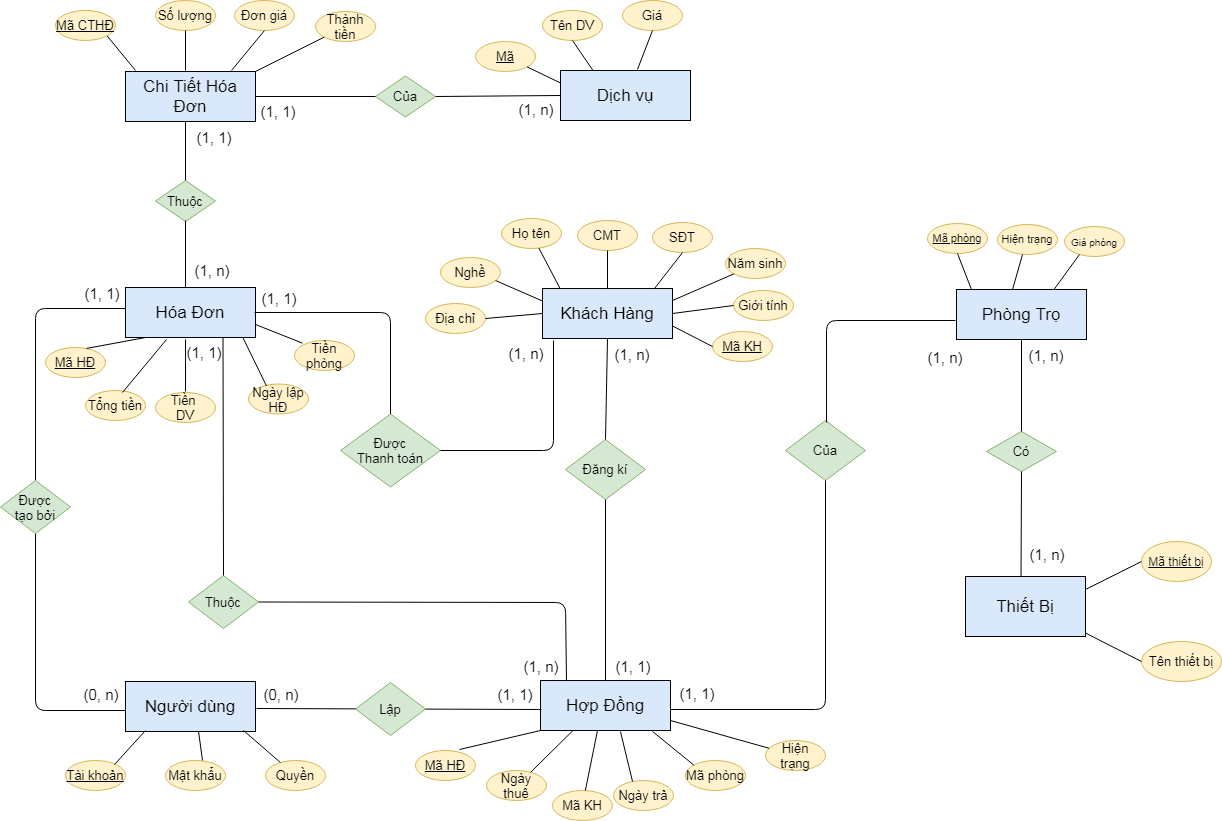
# **CHƯƠNG 2: PHÂN TÍCH VÀ THIẾT KẾ**

1. **Đặc tả bài toán.**

Một người cần quản lí thông tin về các phòng trọ của mình. Mỗi **phòng** gồm các thuộc tính: mã phòng, hiện trạng phòng, giá phòng, dãy phòng. Với mỗi khách hàng thuê phòng sẽ có một hợp đồng thuê phòng, các thuộc tính của **khách hàng**: mã khách hàng, số chứng minh nhân dân, họ tên, giới tính, năm sinh, nghề nghiệp, số điện thoại, địa chỉ. Các thuộc tính của **hợp đồng thuê phòng** gồm: mã hợp đồng, ngày thuê, ngày trả, mã phòng thuê, mã khách hàng, hiện trạng hợp đồng, người tạo hợp đồng. Mỗi khách hàng khi thuê phòng có thể sử dụng thêm các dịch vụ như: điện, nước, internet,... thuộc tính của **dịch vụ** gồm: mã dịch vụ, tên dịch vụ, giá dịch vụ. Mỗi phòng trọ còn có những thiết bị cần thiết, thuộc tính **thiết bị** gồm: mã thiết bị, tên thiết bị. Hàng tháng, khách hàng sẽ nhận được 1 hóa đơn đóng tiền trọ, thuộc tính của **hóa đơn** gồm: mã hóa đơn, ngày lập hóa đơn, tổng tiền phải trả, tiền phòng, tiền dịch vụ, tên khách hàng, người tạo hóa đơn, mã hợp đồng, mã phòng. Mỗi hóa đơn có một chi tiết hóa đơn, thuộc tính của **chi tiết hóa đơn** gồm: mã chi tiết hóa đơn, số lượng, đơn giá, thành tiền. Mỗi người dùng được cấp cho một tài khoản để sử dụng hệ thống, thông tin **tài khoản người dùng** bao gồm: tài khoản, mật khẩu, quyền.

Một khách hàng có thể thuê 1 hoặc nhiều phòng, tương ứng với nhiều hợp đồng. Mỗi hợp đồng chỉ do 1 khách hàng đăng kí thuê. Một phòng có thể có 1 hoặc nhiều thiết bị, với mỗi 1 thiết bị có thể được sử dụng ở 1 hay nhiều phòng. Mỗi chi tiết hóa đơn sẽ chỉ có 1 dịch vụ và một dịch vụ có thể có nhiều chi tiết hóa đơn. Mỗi chi tiết hóa đơn thuộc một hóa đơn và mỗi hóa đơn có thể có nhiều chi tiết hóa đơn hoặc. Hàng tháng, mỗi khách hàng sẽ nhận được 1 hóa đơn thanh toán, mỗi khách hàng có thể có 1 hoặc nhiều hóa đơn trong suốt thời gian thuê phòng. Ngược lại, mỗi hóa đơn chỉ thuộc về một khách hàng thuê phòng. Mỗi hóa đơn chỉ thuộc về một hợp đồng thuê phòng, nhưng 1 hợp đồng có thể có nhiều hóa đơn thanh toán.

1. **Mô hình ERD.**

****

1. **Lược đồ quan hệ.**

+ **Thiet\_bi** (MaTB, TenThietBi)

Mỗi thiết bị cần có *MaTB* để phân biệt các thiết bị với nhau.

+ **Phong\_co\_thiet\_bi** (MaTB, MaPhong, SoLuong)

Mỗi phòng có thiết bị cần có *MaTB* và *MaPhong* để xác định xem thiết bị nào đang được sử dụng ở phòng nào, và số lượng là bao nhiêu.

+ **Phong\_tro** (MaPhong, GiaPhong, LoaiPhong, HienTrang, DayPhong)

Mỗi phòng trọ cần có *MaPhong* để phân biệt *GiaPhong*, *LoaiPhong*, *HienTrang, DayPhong* của mỗi phòng trọ.

+ **Hop\_dong** (MaHopDong, NgayThue, NgayTra, HienTrang, MaPhong, MaKhachHang, NguoiTao)

Mỗi hợp đồng cần có *MaHopDong* để phân biệt *NgayThue, NgayTra, MaPhong, MaKhachHang, NguoiTao, HienTrang* của mỗi khách hàng khi đăng kí thuê trọ, có mã phòng cụ thể.

+ **Chi\_tiet\_hoa\_don** (MaCTHD, MaDichVu,SoLuong, DonGia, ThanhTien, MaHopDong)

Mỗi chi tiết hóa đơn cần có *MaCTHD* để phân biệt *MaDichVu, SoLuong, DonGia, ThanhTien, MaHopDong* của mỗi khách hàng khi sử dụng dịch vụ.

+ **Dich\_vu** (MaDichVu, TenDichVu, GiaDichVu)

Mỗi dịch vụ cần có *MaDichVu* để phân biệt *TenDichVu, GiaDichVu*

+ **Hoa\_don** (MaHoaDon, MaHopDong, NgayLapHD, TenKhachHang, MaPhong, TienDV, TienPhong ,TongTien, NguoiTao)

Mỗi hóa đơn có *MaHoaDon* để phân biệt *MaHopDong, TenKhachHang, TongTien, NgayLapHD, NguoiTao*

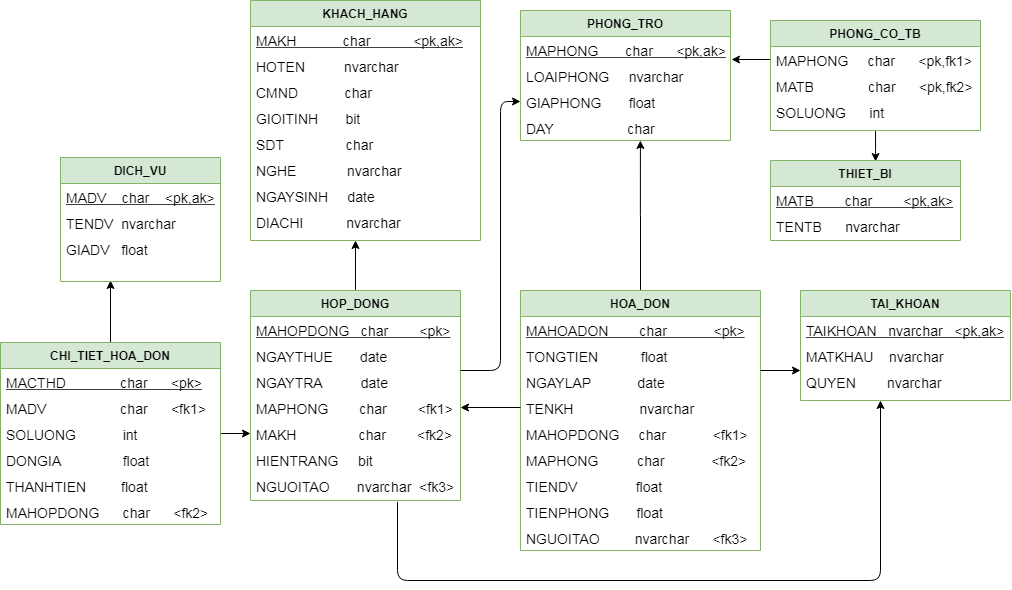
+ **Tai\_khoan** (TaiKhoan, MatKhau, Quyen)

Mỗi tài khoản cần có TaiKhoan để phân biệt các người dùng khác nhau.

+ **Khach\_hang** (MaKH, NgaySinh, SDT, CMT, HoTen, Nghe, DiaChi, GioiTinh)

Mỗi khách hàng cần có *MaKH* để phân biệt tên khách hàng, năm sinh, số chứng minh thư, số điện thoại, địa chỉ của mỗi khách hàng khi đăng kí.

1. **Mô hình quan hệ.**

****

1. **Từ điển dòng dữ liệu.**

**Thiet\_bi** (**MaTB**, TenThietBi)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Thiet\_bi | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2 | **MaTB**  TenThietBi | Mã thiết bị  Tên thiết bị | Text  Text | 10  50 | Khóa chính |

**Phong\_co\_thiet\_bi** (MaTB, MaPhong, SoLuong)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Phong\_co\_thiet\_bi | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2 | MaTB  MaPhong | Mã thiết bị  Mã phòng | Characters  Characters | 10  10 | Khóa ngoại  Khóa ngoại |

**Phong\_tro** (**MaPhong**, GiaPhong, LoaiPhong, HienTrang, DayPhong)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Phong\_tro | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2  3  4  5 | **MaPhong**  GiaPhong  LoaiPhong  HienTrang  DayPhong | Mã phòng  Mã phòng  Loại phòng  Hiện trạng  Dãy phòng | Characters  Float  Characters  Characters  Characters | 10  10  50  10  10 | Khóa chính |

**Hop\_dong** (**MaHopDong**, NgayThue, NgayTra, HienTrang, MaPhong, MaKhachHang, NguoiTao)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Hop\_dong | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2  3  4  5  6  7 | **MaHopDong**  NgayThue  NgayTra  HienTrang  MaPhong  MaKhachHang  NguoiTao | Mã phòng  Ngày thuê  Ngày trả  Hiện trạng  Mã phòng  Mã K.Hàng  Người tạo | Characters  Date  Date  Boolean  Characters  Characters  Characters | 10  1  10  10  50 | Khóa chính  Khóa ngoại  Khóa ngoại  Khóa ngoại |

**Chi\_tiet\_hoa\_don** (**MaCTHD**, MaDichVu,SoLuong, DonGia, ThanhTien, MaHopDong)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Chi\_tiet\_hoa\_don | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2  3  4  5  6 | **MaCTHD**  MaDichVu  SoLuong  DonGia  ThanhTien  MaHopDong | Mã chi tiết HĐ  Mã dịch vụ  Số lượng  Đơn giá  Thành tiền  Mã hợp đồng | Characters  Characters  Number  Float  Float  Characters | 10  10  8  20  20  10 | Khóa chính  Khóa ngoại  Khóa ngoại |

**Dich\_vu** (**MaDichVu**, TenDichVu, GiaDichVu)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Dich\_vu | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2  3 | **MaDichVu**  TenDichVu  GiaDichVu | Mã dịch vụ  Tên dịch vụ  Giá dịch vụ | Characters  Text  Float | 10  50 | Khóa chính |

**Hoa\_don** (**MaHoaDon**, MaHopDong, NgayLapHD, TenKhachHang, MaPhong, TienDV, TienPhong ,TongTien, NguoiTao)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Hoa\_don | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2  3  4  5  6  7  8  9 | **MaHoaDon**  MaHopDong  NgayLapHD  TenKhachHang  MaPhong  TienDV  TienPhong  TongTien  NguoiTao | Mã hóa đơn  Mã hợp đồng  Ngày lập  Tên khách hàng  Mã phòng  Tiền dịch vụ  Tiền phòng  Tổng tiền  Người tạo | Characters  Characters  Date  Text  Characters  Float  Float  Float  Characters | 10  10  50  20  10  10  10  50 | Khóa chính  Khóa ngoại  Khóa ngoại |

**Khach\_hang** (**MaKH**, NgaySinh, SDT, CMT, HoTen, Nghe, DiaChi, GioiTinh)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Khach\_hang | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2  3  4  5  6  7  8 | **MaKH**  NgaySinh  SDT  CMT  HoTen  Nghe  DiaChi  GioiTinh | Mã khách hàng  Ngày sinh  Số điện thoại  Chứng minh thư  Họ tên  Nghề  Địa chỉ  Giới tính | Characters  Date  Number  Characters  Text  Text  Text  Boolean | 10  10  20  50  50  50  1 | Khóa chính |

**Tai\_khoan** (**TaiKhoan**, MatKhau, Quyen)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Phần mềm quản lí nhà trọ | | Loại thực thể: Tai\_khoan | |  | |
| STT | Tên thuộc tính | Diễn giải | Kiểu | Kích thước | Ghi chú |
| 1  2  3 | **TaiKhoan**  MatKhau  Quyen | Tài khoản  Mật khẩu  Quyền | Text  Text  Text | 50  50  20 | Khóa chính |

1. **Ràng buộc toàn vẹn.**

**2.6.1 Ràng buộc về miền giá trị của thuộc tính**

**- R1: Trong quan hệ Phong\_tro, giá phòng phải là một số thực lớn hơn 0.**

R1: bảng Phong\_tro

Điều kiện: p1  Phong\_tro thì p1.Giaphong > 0

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R1 | Thêm | Xóa | Sửa |
| Phong\_tro | +(Giaphong) | - | -(\*) |

**- R2: Trong quan hệ Hoa\_don, tổng tiền phải là một số thực và lớn hơn 0.**

R2: bảng Hoa\_don

Điều kiện: hd1  Hoa\_don thì hd1 .Tongtien > 0

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R2 | Thêm | Xóa | Sửa |
| Hoa\_don | +(Tongtien) | - | -(\*) |

**- R3: Trong quan hệ Chi\_tiet\_hoa\_don, số lượng phải là một số nguyên lớn hơn 0.**

R3: bảng Chi\_tiet\_hoa\_don

Điều kiện: cthd1  Chi\_tiet\_hoa\_don thì cthd1.Soluong > 0

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R3 | Thêm | Xóa | Sửa |
| Chi\_tiet\_hoa\_don | +(Soluong) | - | -(\*) |

**- R4: Trong quan hệ Khach\_hang, giới tính khách hàng chỉ có thể chọn là nam hoặc là nữ.**

R4: bảng Khach\_hang

Điều kiện: kh1 Khach\_hang thì kh1.Gioitinh IN {‘Nam’, ‘Nữ’}

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R4 | Thêm | Xóa | Sửa |
| Khach\_hang | +(Gioitinh) | - | -(\*) |

**2.6.2 Ràng buộc toàn vẹn liên thuộc tính:**

**- R1: Ngaythue < Ngaytra**

R1: bảng Hop\_dong

Điều kiện: n1 Hop\_dong | n1.Ngaythue < n1.Ngaytra

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R1 | Thêm | Xóa | Sửa |
| Hop\_dong | +(Ngaythue) | - | -(\*) |

**- R2: Thanhtien = Soluong \* Dongia**

R2: bảng Chi\_tiet\_hoa\_don

Điều kiện: ct  Chi\_tiet\_hoa\_don | ct.Thanhtien = ct.Soluong \* ct.Dongia

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R2 | Thêm | Xóa | Sửa |
| Chi\_tiet\_hoa\_don | + | - | -(\*) |

**- R3: Tongtien = TienDV + Tienphong**

R3: bảng Hoa\_don

Điều kiện: hd  Hoa\_don | hd.Thanhtien = hd.TienDV + hd.Tienphong

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R3 | Thêm | Xóa | Sửa |
| Hoa\_don | + | - | -(\*) |

**2.6.3 Ràng buộc toàn vẹn liên bộ:**

**- R1: mỗi dịch vụ có một MADV để phân biệt**

R1: bảng Dich\_vu

Điều kiện: dv1,dv2  Dich\_vu | dv1.MADV  dv2.MADV

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R1 | Thêm | Xóa | Sửa |
| Dich\_vu | +(MADV) | - | +(MADV) |

**- R2: mỗi thiết bị có một MATB để phân biệt**

R2: bảng Thiet\_bi

Điều kiện: tb1,tb2  Dich\_vu | tb1.MATB  tb2.MATB

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R2 | Thêm | Xóa | Sửa |
| Thiet\_bi | +(MATB) | - | +(MATB) |

**- R3: mỗi phòng có một MAPHONG để phân biệt**

R3: bảng Phong\_tro

Điều kiện: p1,p2  Phong\_tro | p1.MAPHONG  p1.MAPHONG

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R3 | Thêm | Xóa | Sửa |
| Phong\_tro | +(MAPHONG) | - | +(MAPHONG) |

**- R4: mỗi khách hàng có một MAKH để phân biệt**

R4: bảng Khach\_hang

Điều kiện: kh1,kh2  Khach\_hang | kh1.MAKH  kh2.MAKH

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R4 | Thêm | Xóa | Sửa |
| Khach\_hang | +(MAKH) | - | +(MAKH) |

**- R5: mỗi hợp đồng có một MAHD để phân biệt**

R5: bảng Hop\_dong

Điều kiện: hd1,hd2  Hop\_dong | hd1.MAHD  hd1.MAHD

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R5 | Thêm | Xóa | Sửa |
| Hop\_dong | +(MAHD) | - | +(MAHD) |

**- R6: mỗi hóa đơn có một MAHDON để phân biệt**

R6: bảng Hoa\_don

Điều kiện: hd1,hd2  Hoa\_don | hd1.MAHDON hd1.MAHDON

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R6 | Thêm | Xóa | Sửa |
| Hoa\_don | +(MAHDON) | - | +(MAHDON) |

**- R7: mỗi chi tiết hóa đơn có một MACTHD để phân biệt**

R7: bảng Chi\_tiet\_hoa\_don

Điều kiện: ct1,ct2  Chi\_tiet\_hoa\_don | ct1.MACTHDct1.MACTHD

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R7 | Thêm | Xóa | Sửa |
| Chi\_tiet\_hoa\_don | +(MACTHD) | - | +(MACTHD) |

**- R8: mỗi người dùng có một TAIKHOAN để phân biệt**

R8: bảng Nguoi\_dung

Điều kiện: n1,n2  Nguoi\_dung | n1.TAIKHOAN n1.TAIKHOAN

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R8 | Thêm | Xóa | Sửa |
| Nguoi\_dung | +(TAIKHOAN) | - | +(TAIKHOAN) |

**2.6.4 Ràng buộc phụ thuộc tồn tại:**

**- R1: bảng Hop\_dong[Maphong]  Phong\_tro[Maphong]**

Bối cảnh: Ma\_phong, Hop\_dong

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R1 | Thêm | Xóa | Sửa |
| Phong\_tro | - | -(\*) | + |
| Hop\_dong | + | + | - |

**- R2: bảng Hoa\_don[MaKH]  Khach\_hang[MaKH]**

Bối cảnh: Hoa\_don, Khach\_hang

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R2 | Thêm | Xóa | Sửa |
| Khach\_hang | - | -(\*) | + |
| Hoa\_don | + | + | - |

**- R3: bảng Hop\_dong[MaKH]  Khach\_hang[MaKH]**

Bối cảnh: Hop\_dong, Khach\_hang

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R3 | Thêm | Xóa | Sửa |
| Khach\_hang | - | -(\*) | + |
| Hop\_dong | + | + | - |

**- R4: bảng Hoa\_don[MaHopDong]  Hop\_dong[MaHopDong]**

Bối cảnh: Hop\_dong, Hoa\_don

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R4 | Thêm | Xóa | Sửa |
| Hop\_dong | - | -(\*) | + |
| Hoa\_don | + | + | - |

**- R5: bảng Chi\_tiet\_hoa\_don[MaDV]  Dich\_vu[MaDV]**

Bối cảnh: Chi\_tiet\_hoa\_don, Dich\_vu

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R5 | Thêm | Xóa | Sửa |
| Dich\_vu | - | -(\*) | + |
| Chi\_tiet\_hoa\_don | + | + | - |

**2.6.5 Ràng buộc liên bộ - liên quan hệ:**

**- R1: Tổng tất cả Thanhtien trong Chi\_tiet\_hoa\_don có cùng MaHopDong phải bằng TienDV của hóa đơn đó trong Hoa\_don**

Điều kiện: hd  Hoa\_don | hd.TienDV = SUM(cthd.Thanhtien) đối với các cthd  Chi\_tiet\_hoa\_don sao cho cthd.Mahopdong = hd.Mahopdong

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R1 | Thêm | Xóa | Sửa |
| Hoa\_don | - | +(TienDV) | + |
| Chi\_tiet\_hoa\_don | + | +(ThanhTien) | + |

**2.6.6 Ràng buộc liên thuộc tính - liên quan hệ:**

**- R1: Mỗi khách hàng sẽ có ngày thuê, ngày lập hóa đơn có thể giống hoặc khác nhau**

R1: h  Hoa\_don, hd  Hop\_dong | h.NgayThue <= hd.NgayLapHD

Tầm ảnh hưởng:

|  |  |  |  |
| --- | --- | --- | --- |
| R1 | Thêm | Xóa | Sửa |
| Hop\_dong | - | - | -(\*) |
| Hoa\_don | + | - | -(\*) |

1. **Phụ thuộc hàm**

* **Trong quan hệ PHONG\_CO\_TB có các phụ thuộc hàm:**

- f1: MAPHONG, MATB-> SOLUONG

* **Trong quan hệ HOA\_DON có các phụ thuộc hàm:**

- f1: MAHOADON -> MAHOPDONG

- f2: MAHOADON-> MAPHONG

- f3: MAPHONG -> TIENPHONG

- f4: TIENDV, TIENPHONG -> TONGTIEN

- f5: MAHOPDONG -> TENKH

- f6: MAHOADON -> NGAYLAP

- f7: MAHOADON -> NGUOITAO

* **Trong quan hệ THIET\_BI có các phụ thuộc hàm:**

- f1: MATB -> TENTB

* **Trong quan hệ NGUOI\_DUNG có các phụ thuộc hàm:**

- f1: TAIKHOAN -> QUYEN

- f2: TAIKHOAN -> MATKHAU

* **Trong quan hệ PHONG\_TRO có các phụ thuộc hàm:**

- f1: MAPHONG -> LOAIPHONG

- f2: MAPHONG -> GIAPHONG

- f3: MAPHONG -> DAYPHONG

* **Trong quan hệ KHACH\_HANG có các phụ thuộc hàm**

- f1: MAKH -> TENKH

- f2: MAKH -> NAMSINH

- f3: MAKH -> DIACHI

- f4: MAKH -> SODIENTHOAI

- f5: MAKH -> NGHENGHIEP

* **Trong quan hệ DICH\_VU có các phụ thuộc hàm**

- f1: MADV -> TENDV

- f2: MADV -> GIADV

* **Trong quan hệ CHI\_TIET\_HOA\_DON có các phụ thuộc hàm**

- f1: SOLUONG, DONGIA -> THANHTIEN

- f2: MACTHD -> THANHTIEN

- f3: MACTHD, MADV -> SOLUONG

- f4: MACTHD, MADV -> DONGIA

* **Trong quan hệ HOP\_DONG có các phụ thuộc hàm**

- f1: MAHDONG -> MAKH

- f2: MAHDONG -> NGAYTHUE

- f3: MAHDONG -> NGAYTRA

- f4: MAHDONG, MAPHONG -> HIENTRANG

- f5: MAHDONG, MAKH -> NGAYTHUE

1. **Truy vấn SQL và đại số quan hệ.**

* **Tìm những phòng mà khách đang thuê có họ 'Nguyễn'**

- Truy vấn SQL:

**SELECT** \*

**FROM** PHONG\_TRO

**WHERE** MAPHONG IN (

**SELECT** HD.MAPHONG

**FROM** HOP\_DONG HD, KHACH\_HANG KH

**WHERE** HD.MAKHACHHANG = KH.MAKHACHHANG AND

KH.HOTEN LIKE N'NGUYỄN%' AND HD.HIENTRANG = 1 )

- Đại số quan hệ:

Πmaphong((hoten = ‘Nguyễn%’)(KHACH\_HANG \* HOP\_DONG \* PHONG\_TRO))

* **Tìm thông tin khách hàng có hóa đơn cao nhất**

- Truy vấn SQL:

**SELECT** \*

**FROM** KHACH\_HANG

**WHERE** MAKHACHHANG IN (

**SELECT** HDG.MAKHACHHANG

**FROM** HOA\_DON HDN, HOP\_DONG HDG

**WHERE** HDN.MAHOPDONG = HDG.MAHOPDONG AND

TONGTIEN = (**SELECT** MAX(HOA\_DON.TONGTIEN) **FROM** HOA\_DON))

- Đại số quan hệ:

Πhoten((tongtien = MAX(TONGTIEN)(HOADON))(KHACH\_HANG \* HOA\_DON \* HOP\_DONG))

* **Cho biết dãy A có bao nhiêu phòng trọ**

- Truy vấn SQL:

**SELECT** COUNT(\*) AS ‘SOLUONG’

**FROM** PHONG\_TRO

**WHERE** DAY = ‘A’

**GROUP BY** MAPHONG

- Đại số quan hệ:

COUNT(DAY)((DAY = ‘A’)(PHONG\_TRO))

* **Tìm những khách hàng đã thuê trọ 2 lần trở lên**

- Truy vấn SQL:

**SELECT** \* **FROM** KHACH\_HANG

**WHERE** MAKHACHHANG IN (

**SELECT** MAKHACHHANG

**FROM** HOP\_DONG

**GROUP BY** MAKHACHHANG

**HAVING** COUNT(MAKHACHHANG) > 1)

* **Tìm doanh thu trung bình trong tháng 7 năm 2020**

- Truy vấn SQL:

**SELECT** AVG(TONGTIEN) AS 'DOANH THU'

**FROM** HOA\_DON

**WHERE** YEAR(NGAYLAPHOADON) = 2020 AND MONTH(NGAYLAPHOADON) = 7

- Đại số quan hệ:

AVG(TONGTIEN)((NGAYLAPHD = ‘2020-07-%’)(HOA\_DON))

* **Tìm những khách hàng có ngày tháng năm sinh là 01-01-2000**

- Truy vấn SQL:

**SELECT** \*

**FROM** KHACH\_HANG

**WHERE** NAMSINH LIKE ‘2000-01-01’

- Đại số quan hệ:

Π((NAMSINH = ‘2000-01-01’)(KHACH\_HANG))

* **Tìm những phòng đang trống**

- Truy vấn SQL:

**SELECT** \*

**FROM** PHONG\_TRO

**WHERE** HIENTRANG = 1

- Đại số quan hệ:

Π((TINHTRANG = 1)(PHONG\_TRO))

* **Tìm những phòng trả vào ngày 31-07-2020**

- Truy vấn SQL:

**SELECT** \*

**FROM** PHONG\_TRO

**WHERE** MAPHONG IN (

**SELECT** MAPHONG

**FROM** HOP\_DONG

**WHERE** NGAYTRA LIKE ‘31-07-2020’)

- Đại số quan hệ:

Πmaphong((NGAYTRA= ‘2020-07-31’)(HOPDONG\*PHONG\_TRO))

* **Tính tổng tiền thu được trong tháng 6-2020**

- Truy vấn SQL:

**SELECT** SUM (TONGTIEN)

**FROM** HOA\_DON

**WHERE** NGAYLAPHOADON LIKE ‘2020-06-%’

- Đại số quan hệ:

SUM(TONGTIEN)((NGAYLAPHOADON = ‘2020-06-%’)(HOA\_DON)

* **Đưa ra thông tin phòng có sử dụng dịch vụ WiFi trong tháng này**

- Truy vấn SQL:

**SELECT** \*

**FROM** PHONG\_TRO

**WHERE** MAPHONG IN (

**SELECT** DISTINCT HOP\_DONG.MAPHONG

**FROM** CHI\_TIET\_HOA\_DON, DICH\_VU, PHONG\_TRO, HOP\_DONG

**WHERE** CHI\_TIET\_HOA\_DON.MADV = DICH\_VU.MADICHVU AND HOP\_DONG.MAPHONG = PHONG\_TRO.MAPHONG AND CHI\_TIET\_HOA\_DON.MADV = 'DV0001'

AND CHI\_TIET\_HOA\_DON.MAHOPDONG = HOP\_DONG.MAHOPDONG)

- Đại số quan hệ:

Πmaphong((MADV= ‘DV001’)(CHI\_TIET\_HD\*PHONG\_TRO))

* **Tìm khách hàng có thời gian thuê trọ lâu nhất**

- Truy vấn SQL:

**SELECT** \*

**FROM** KHACH\_HANG

**WHERE** MAKHACHHANG IN (**SELECT** H1.MAKHACHHANG

**FROM** HOP\_DONG H1

**WHERE** (**SELECT** COUNT(\*) **FROM** HOP\_DONG H2

**WHERE** DATEDIFF(DAY,H1.NGAYTRA, H1.NGAYTHUE) > DATEDIFF(DAY,H2.NGAYTRA, H2.NGAYTHUE)) < 1)

* **Cho biết số lượng phòng trọ VIP đang được thuê**

- Truy vấn SQL:

**SELECT** COUNT(\*) AS ‘SOLUONG’

**FROM** HOP\_DONG, PHONG\_TRO

**WHERE** HOP\_DONG.MAPHONG = PHONG\_TRO.MAPHONG AND LOAIPHONG = ‘VIP’

**GROUP BY** MA\_PHONG

- Đại số quan hệ:

COUNT(MAPHONG)((LOAIPHONG= ‘VIP’)(HOPDONG\*PHONGTRO)

* **Cho biết số lượng khách hàng là sinh viên**

- Truy vấn SQL:

**SELECT** COUNT(\*) AS ‘SOLUONG’

**FROM** KHACH\_HANG

**WHERE** NGHE = N‘Sinh Viên’

**GROUP BY** MAKHACHHANG

- Đại số quan hệ:

COUNT(NGHE)((NGHE= ‘Sinh Viên’)(KHACH\_HANG)

* **Cho biết số lượng phòng đang thuê có giá từ 2 triệu trở lên**

- Truy vấn SQL:

**SELECT** COUNT(\*) AS ‘SOLUONG’

**FROM** HOP\_DONG, PHONG\_TRO

**WHERE** HOP\_DONG.MAPHONG = PHONG\_TRO.MAPHONG AND

GIAPHONG >= 2000000

**GROUP BY** MAPHONG

- Đại số quan hệ:

COUNT(MAPHONG)((GIAPHONG>=‘2000000’)(HOPDONG\*PHONGTRO)

1. **Chuẩn hóa quan hệ**

* **Chuẩn hóa 1NF:**

**- Vi phạm:**

+ **Chi\_tiet\_hoa\_don** (**MaCTHD**, MaDichVu,SoLuong, DonGia, ThanhTien, MaHopDong)

+ **Hoa\_don** (**MaHoaDon**, MaHopDong, NgayLapHD, TenKhachHang, MaPhong, TienDV, TienPhong ,TongTien, NguoiTao)

**- Chuẩn hóa:**

+ Loại bỏ thuộc tính ThanhTien của bảng Chi\_tiet\_hoa\_don vì ThanhTien có thể suy ra được từ SoLuong \* DonGia

Chi\_tiet\_hoa\_don (**MaCTHD**, MaDichVu,SoLuong, DonGia, MaHopDong)

+ Loại bỏ thuộc tính TongTien của bảng Hoa\_don

Hoa\_don (**MaHoaDon**, MaHopDong, NgayLapHD, TenKhachHang, MaPhong, TienDV, TienPhong, NguoiTao)

* **Chuẩn hóa 2NF:**

**- Vi phạm:**

**+ Chi\_tiet\_hoa\_don** (**MaCTHD**, MaDichVu,SoLuong, DonGia, MaHopDong)

**- Chuẩn hóa:**

+ Thuộc tính DonGia của bảng Chi\_tiet\_hoa\_don phụ thuộc một phần vào khóa MaDichVu chứ không cần phụ thuộc vào cả tập khóa (MaCTHD, MADV, MaHopDong)

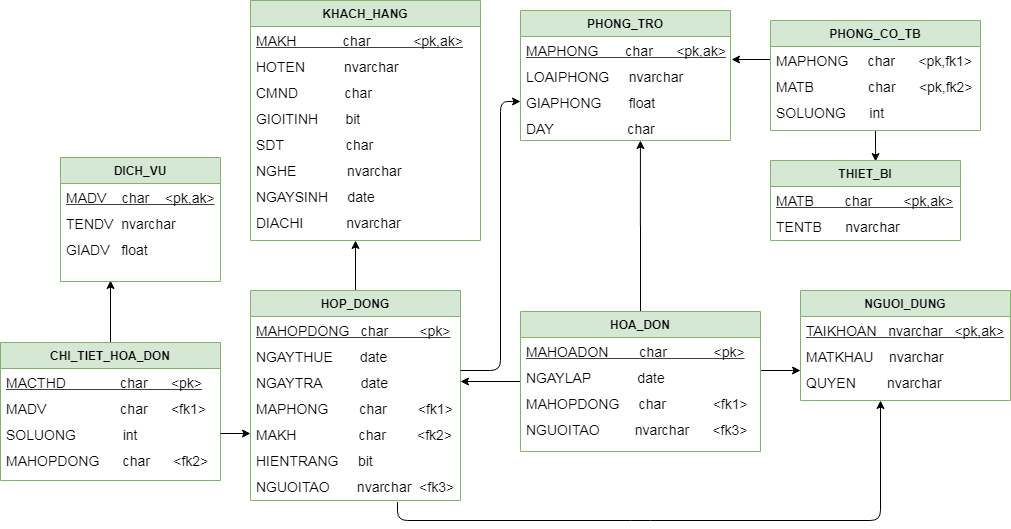
**Chi\_tiet\_hoa\_don** (**MaCTHD**, MaDichVu,SoLuong, MaHopDong)

+ Thuộc tính TienPhong của bảng Hoa\_don phụ thuộc một phần vào MaPhong chứ không cần phụ thuộc vào cả tập khóa. Tương tư, TienDV có thể suy ra từ MaHopDong thông qua bảng Chi\_tiet\_hoa\_don.

+ MaPhong có thể suy ra từ MaHopDong

**Hoa\_don** (**MaHoaDon**, MaHopDong, NgayLapHD, NguoiTao)

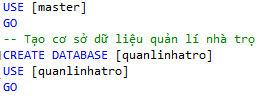
* **Đã chuẩn 3NF**
* **Mô hình quan hệ sau khi chuẩn hóa**

****

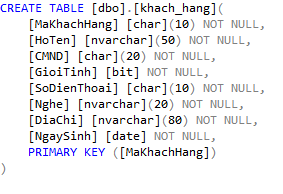
1. **Cài đặt**

**2.10.1 Tạo cơ sở dữ liệu và tạo bảng**

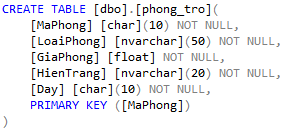
* Tạo cơ sở dữ liệu QUANLINHATRO



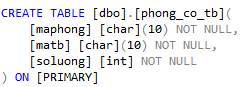
* **Tạo bảng KHACH\_HANG**



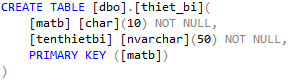
* **Tạo bảng PHONG\_TRO**



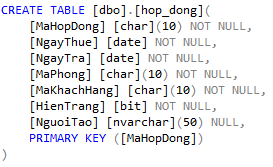
* **Tạo bảng PHONG\_CO\_TB**



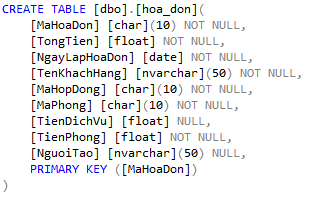
* **Tạo bảng THIET\_BI**



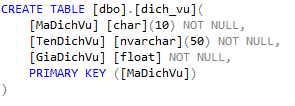
* **Tạo bảng HOP\_DONG**



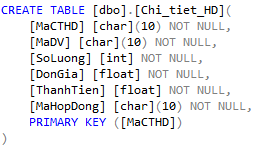
* **Tạo bảng HOA\_DON**



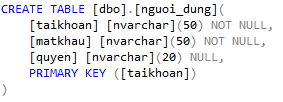
* **Tạo bảng DICH\_VU**



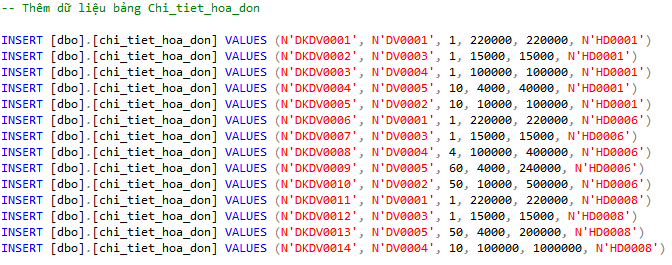
* **Tạo bảng CHI\_TIET\_HD**

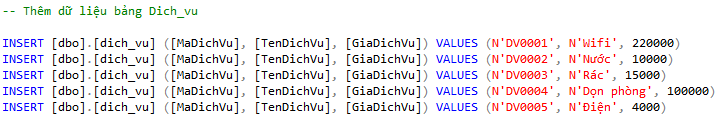


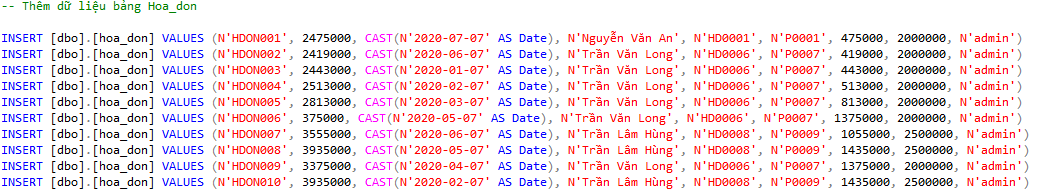
* **Tạo bảng NGUOI\_DUNG**

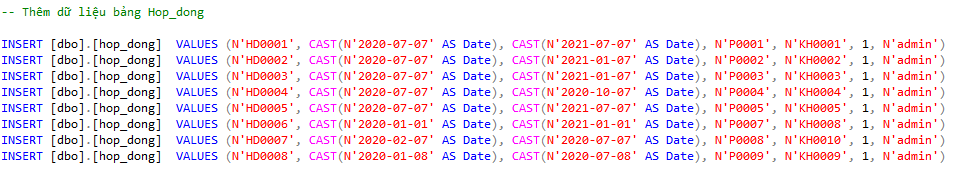


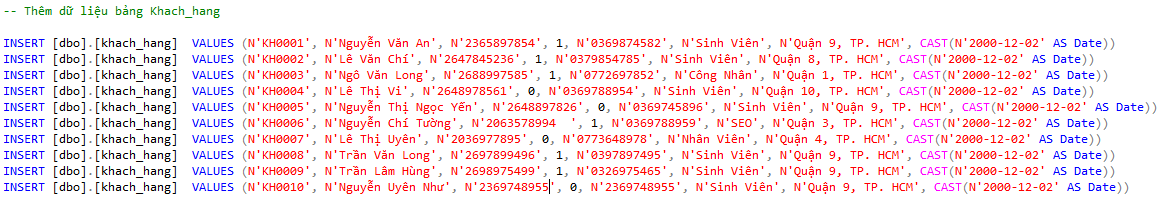
**2.10.2 Thêm dữ liệu vào các bảng**



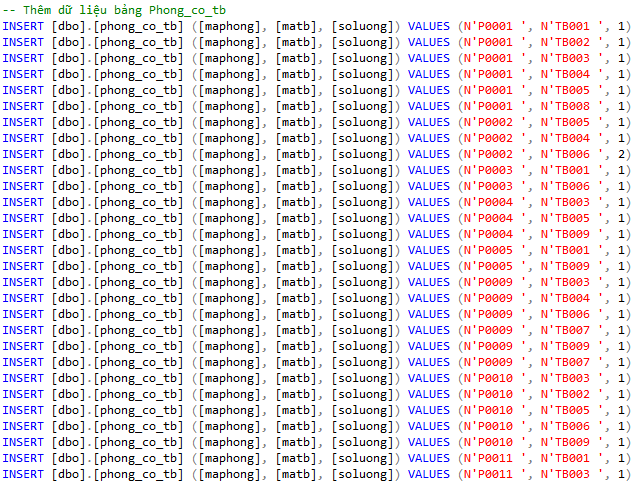


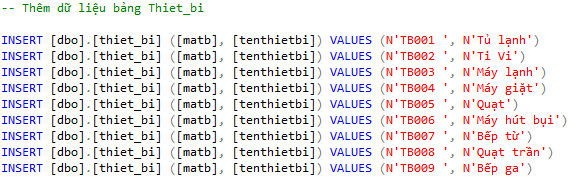


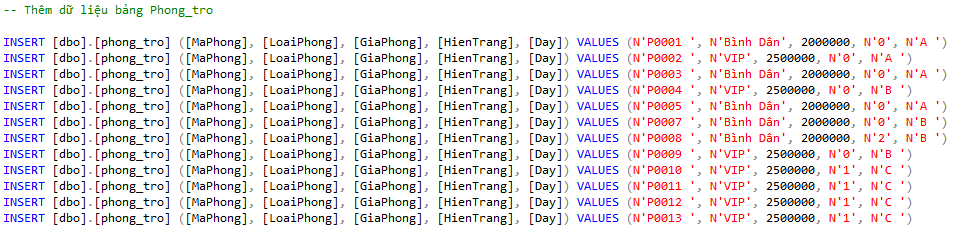




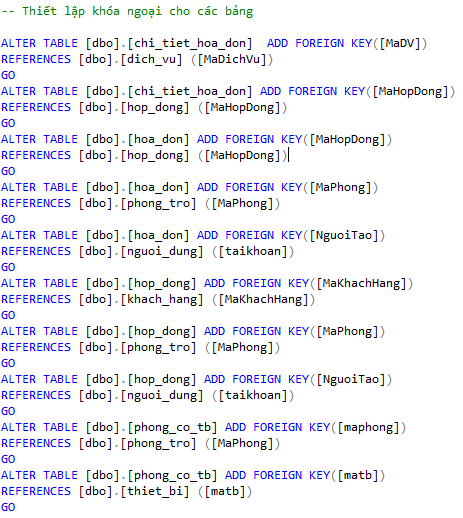




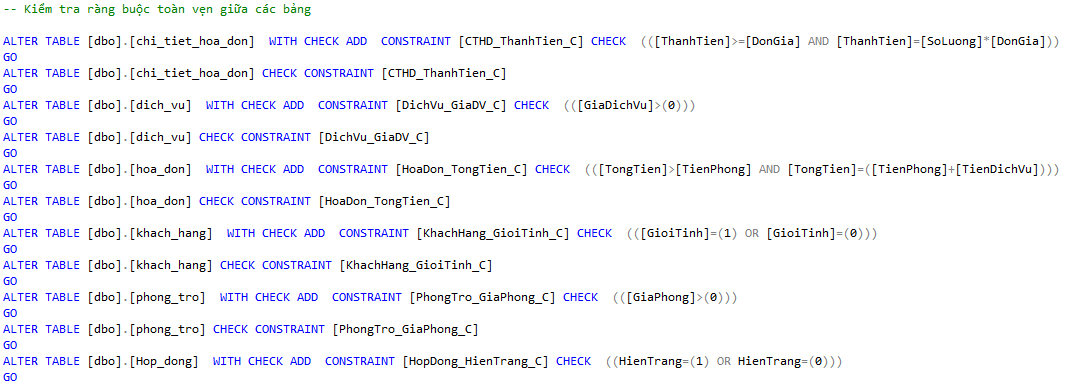




**2.10.3 Tạo khóa ngoại cho các bảng**



**2.10.4 Kiểm tra ràng buộc toàn vẹn cho các bảng**



**CHƯƠNG 3: THỬ NGHIỆM ỨNG DỤNG**

1. **Môi trường cài đặt**

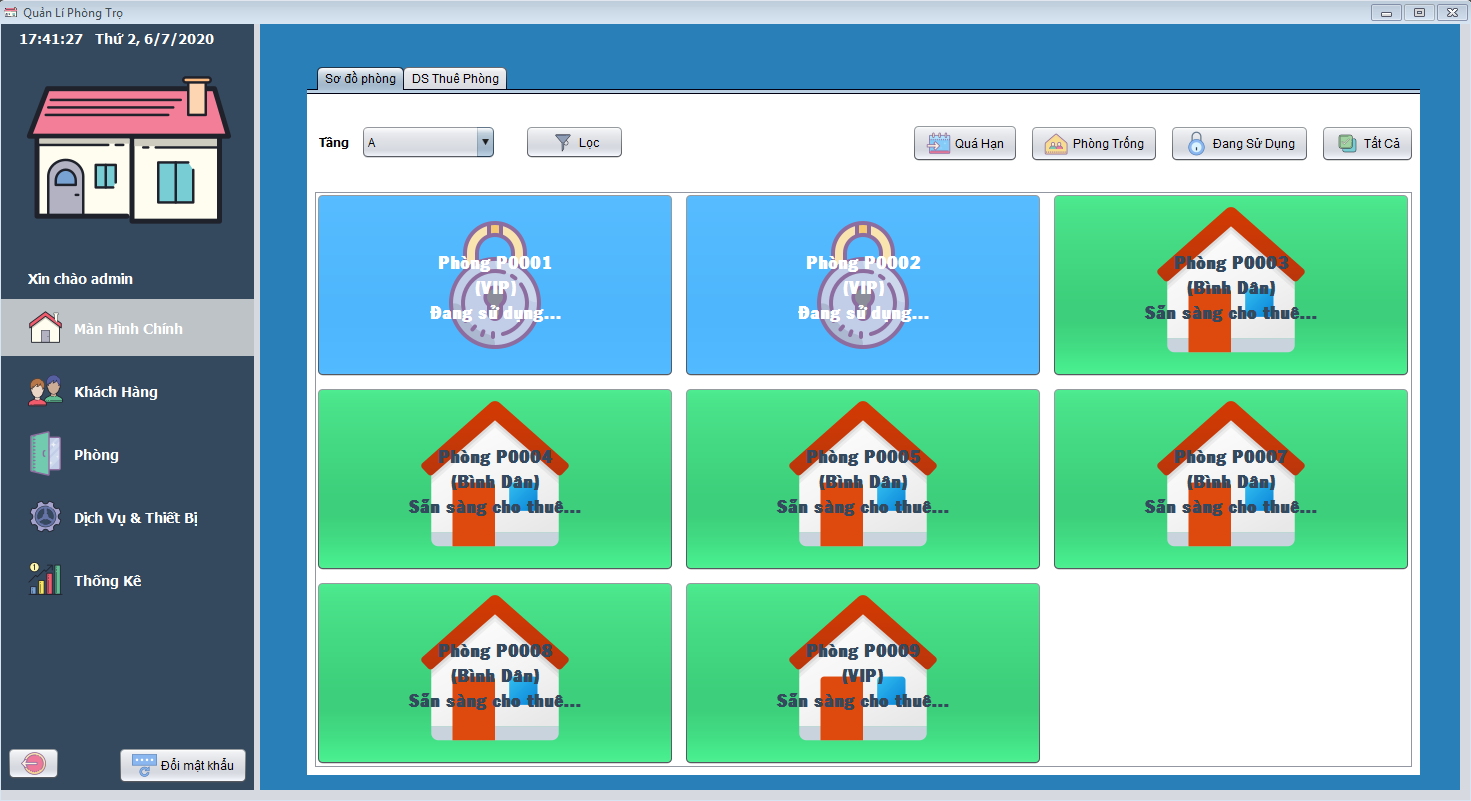
Từ những cơ sở lý thuyết, phương pháp phân tích và thiết kế được đề cập ở chương 2. Em đã xây dựng và phát triển thành công ứng dụng trên môi trường Netbean IDE 8.2 RC, viết bằng ngôn ngữ lập trình Java.

1. **Giao diện đăng nhập**



*Hình 3. 1 – Giao diện đăng nhập hệ thống.*

Cũng như bao ứng dụng desktop khác. Đầu tiên chúng ta tiến hành đăng nhập để có quyền truy cập vào hệ thống. Tài khoản và mật khẩu đã được lưu trong cơ sở dữ liệu trước đó, chỉ có quản trị viên mới biết và có quyền truy cập.



*Hình 3. 2 – Giao diện chính sau khi đăng nhập thành công.*

Đây chính để chúng ta tiến hành sử dụng. Có đầy đủ các chức năng cơ bản để quản lí hiệu quả phòng trọ.

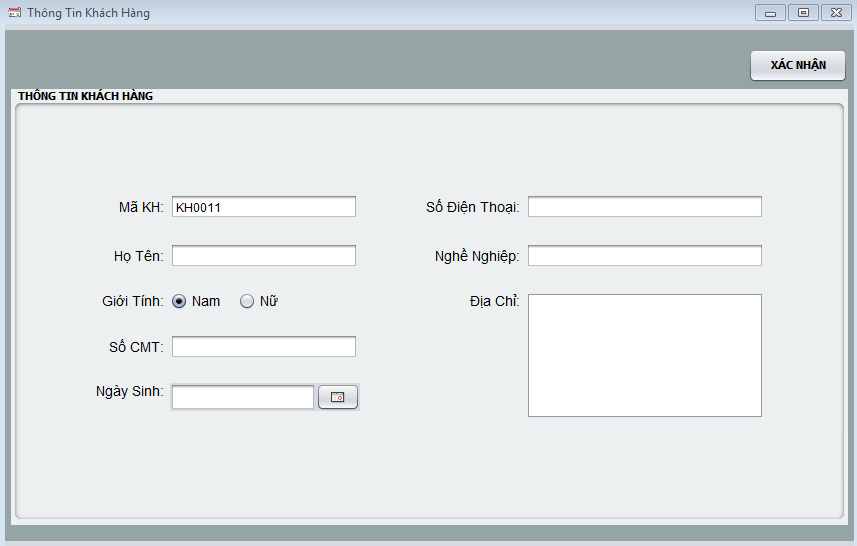
1. **Giao diện đăng ký hợp đồng**



*Hình 3. 3 – Giao diện đăng ký một hợp đồng mới..*

Đây là biểu mẫu dùng để nhập thông tin khách hàng, cùng với ngày thuê ngày trả để từ đó lưu vào cơ sở dữ liệu., có thể đặt thêm các dịch vụ tại đây.

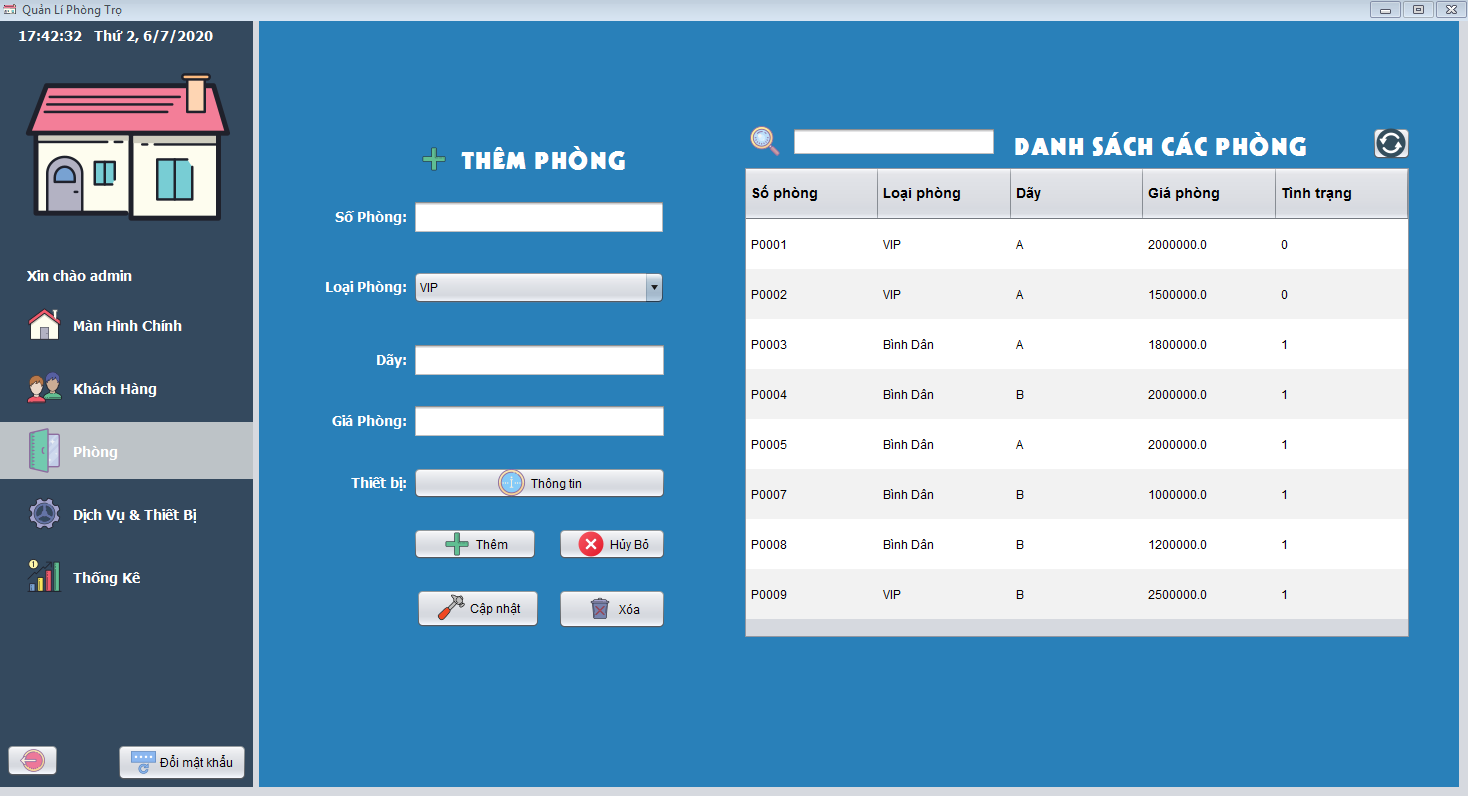
1. **Giao diện thêm khách hàng mới**



*Hình 3. 4 – Giao diện thêm mới một khách hàng*

Đây là giao diện cho phép thêm một khách hàng mới để đăng kí hợp đồng, mã khách hàng được tự động cập nhật sẵn, khách hàng khi đăng kí chỉ nhập các thông tin còn lại.

1. Giao diện thêm mới phòng và hiện thị danh sách phòng



*Hình 3. 6 – Giao diện thêm mới phòng và hiện thị danh sách phòng.*

Đây là giao diện khi cần thêm mới một phòng, với đầy đủ các chức năng thêm sửa xóa và tìm kiếm.

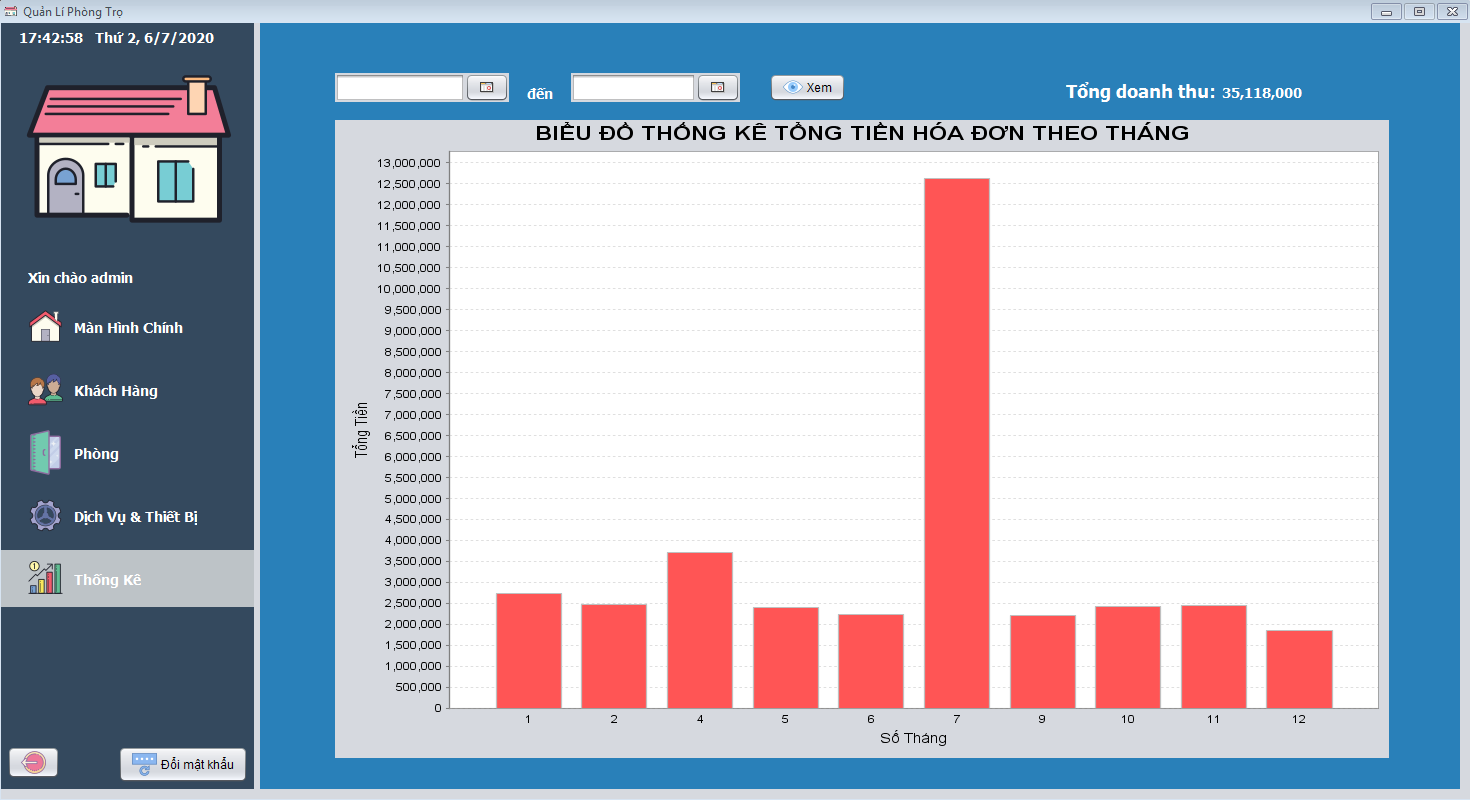
1. **Giao diện thêm dịch vụ và thiết bị**



*Hình 3. 7 – Giao diện thêm dịch vụ và thiết bị.*

Đây là giao diện thêm mới dịch vụ và thiết bị, với đầy đủ các chức năng cơ bản cần thiết như: thêm, sửa, xóa, tìm kiếm.

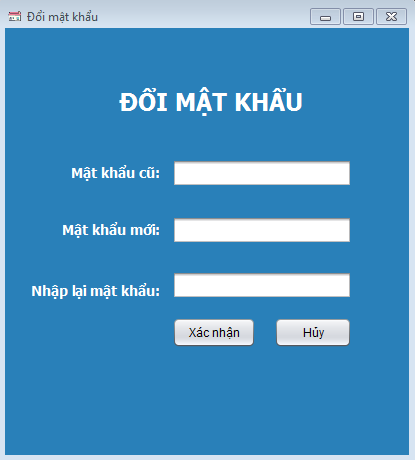
1. **Giao diện thống kê**



*Hình 3. 8 – Giao diện thống kê hóa đơn theo tháng*

Đây là tính năng cho phép người dùng xem được doanh thu hàng tháng của dãy trọ. Ngoài ra còn có thể lựa chọn xem thống kê hóa đơn theo từng ngày. Có hiển thị tổng doanh thu theo thời gian xem.

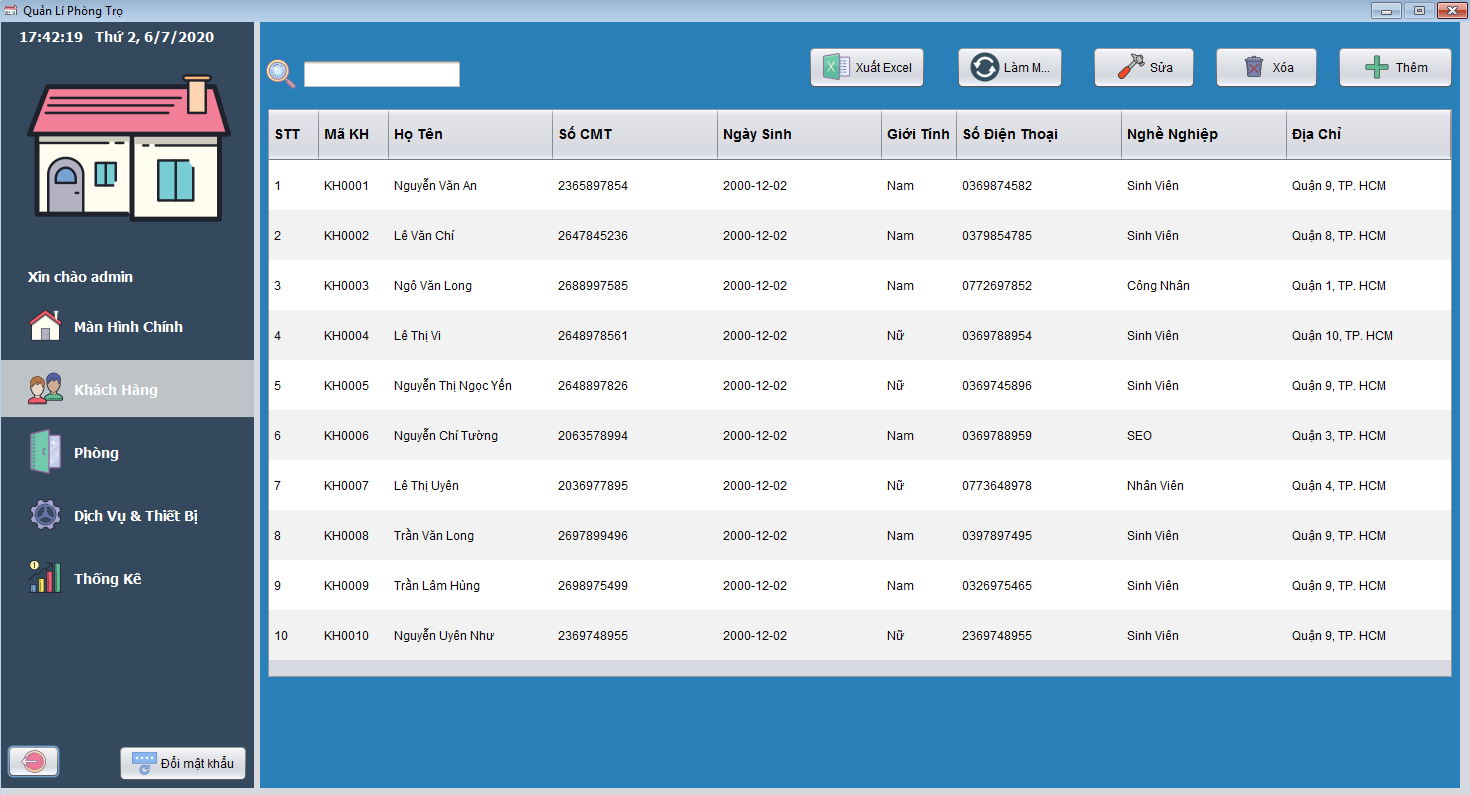
1. **Giao diện nếu người dùng muốn đổi mật khẩu**



*Hình 3. 9 – Giao diện đổi mật khẩu.*

Đây là giao diện giúp người dùng trong quá trình sử dụng có thể đổi mật khẩu nếu không muốn để mật khẩu mặc định từ quản trị viên của ứng dụng.

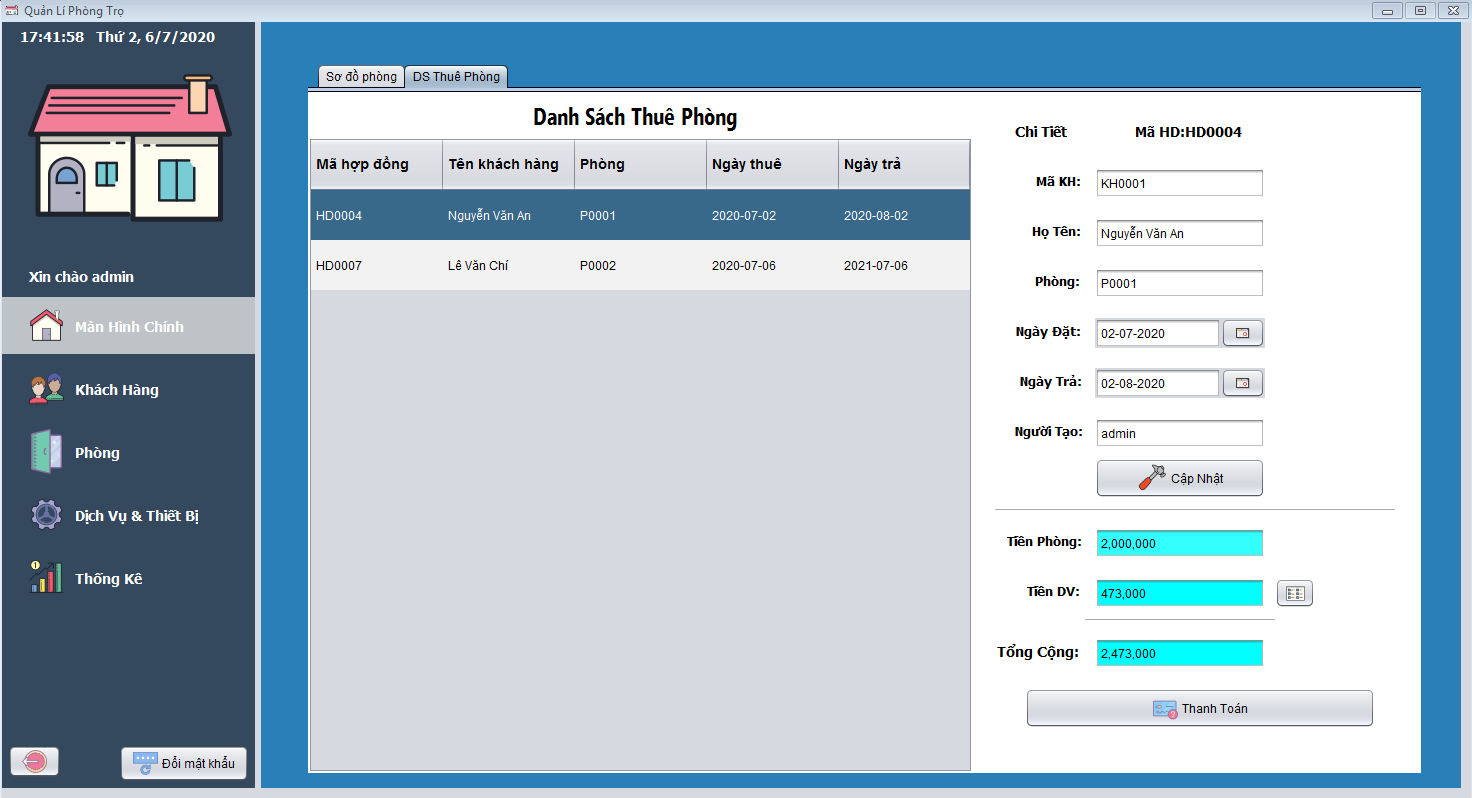
1. **Giao diện khách hàng**



*Hình 3. 10 – Giao diện quản lí thông tin khách hàng.*

Cung cấp đầy đủ các chức năng thêm xóa sửa tìm kiếm một khách hàng, ngoài ra còn có thể xuất danh sách khách hàng ra tệp tin Excel để dễ dàng lấy thông tin.

1. **Giao diện danh sách thuê phòng**

*****3.11. Giao diện danh sách phòng đang thuê*

Đây là giao diện hiển thị danh sách các phòng đang thuê, giúp cho người dùng sễ dàng xuất hóa đơn hàng tháng để tính tiền phòng cho khách hàng. Ngoài ra còn có thể thêm dịch vụ mà khách hàng nếu cần tại đây.

1. **Kết quả và đánh giá**

### Kết quả

Ứng dụng đã được phát triển thành công. Đảm bảo đầy đủ các tính năng cơ bản để quản lí nhà trọ một cách hiệu quả, thuận lợi.

Nhờ sử dụng hệ quản trị cơ sở dữ liệu SQL Server mà ứng dụng quản lí hệ thống hiệu quả, truy xuất thông tin nhanh chóng, chính xác.

Các dữ liệu được lưu trực tiếp xuống cơ sở dữ liệu, giúp cho người dùng không lo việc mất mác dữ liệu khách hàng.

Ứng dụng có tính năng đăng nhập phân quyền, giúp đảm bảo hệ thống bảo đảm và an toàn hơn. Phòng ngừa được nguy cơ mất dữ liệu do lỗi từ người dùng.

### Đánh giá

Ứng dụng hoạt động tốt trên máy tính, laptop. Hiệu suất nhanh chóng ổn định, giao diện đẹp đẽ, thân thiện, dễ sử dụng, dễ thao tác.

Với tính năng đăng nhập phân quyền và đặt phòng trọ chỉ vài cú nhấp chuột đã khiến thao thác người dùng trở nên đơn giản hơn, nhanh hơn, góp phần vào việc quản lí dữ liệu cho người dùng một cách nhanh chóng, chính xác.

**TÀI LIỆU THAM KHẢO**

[1]. <https://www.w3schools.com/sql/>

[2]. Chuẩn hóa dữ liệu:

<https://www.youtube.com/watch?v=cYsibRLo_hI&t=1058s>

[3]. <https://www.howkteam.vn/course/su-dung-sql-server-31>

[4]. <https://www.howkteam.vn/course/khoa-thuc-chien-sql-cung-kteam-du-an-quan-ly-sinh-vien-61>

[5]. <https://ttdungit.wordpress.com/bg-csdl/>