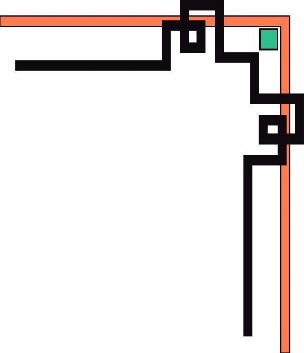
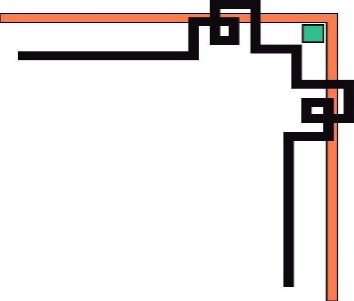
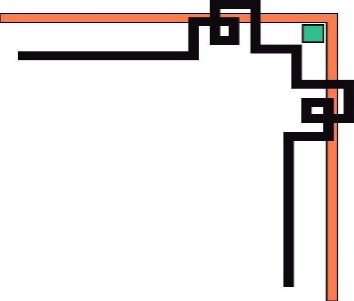
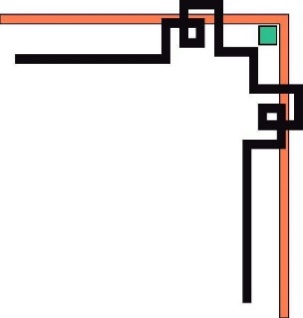
**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ GIAO THÔNG VẬN TẢI**



**KHOA CÔNG NGHỆ THÔNG TIN**

A logo of the university of transport technology

Description automatically generated

|  |
| --- |
| **BÁO CÁO HỌC PHẦN: JAVA NÂNG CAO**  **ĐỀ TÀI: XÂY DỰNG PHẦN MỀM QUẢN LÝ SÂN CẦU LÔNG**  **LỚP: 72DCTT23**  **NHÓM: 6** |

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:**  **Sinh viên thực hiện:** | **Bùi Thị Như**  **Nguyễn Thị Như Quỳnh**  **Bùi Hồng Khanh**  **Đỗ Ngọc Khánh**  **Lê Tuấn Hiệp** |

**Hà Nội - 2024**

# NHIỆM VỤ VÀ ĐÁNH GIÁ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Thành viên | Mã Sinh viên | Công việc | Đánh giá chuyên cần | Nhận xét giáo viên |
| Nguyễn Thị Như Quỳnh (nhóm trưởng) | 72DCTT20133 | Quản lý dịch vụ  Quản lý đặt hàng | Hoàn thành nhiệm vụ đúng hạn, theo dõi tiến trình công việc của nhóm |  |
| Bùi Hồng Khanh | 72DCTT20101 | Quản lý khách hàng  Quản lý loại sân | Hoàn thành đúng hạn |  |
| Lê Tuấn Hiệp | 72DCTT20122 | Quản lý hoá đơn  Báo cáo thống kê | Hoàn thành đúng hạn |  |
| Đỗ Ngọc Khánh | 72DCTT20150 | Đăng nhập  Menu chức năng  Quản lý đặt sân,thêm sân | Hoàn thành đúng hạn |  |

**MỤC LỤC**

[NHIỆM VỤ VÀ ĐÁNH GIÁ 2](#_Toc163403907)

[LỜI MỞ ĐẦU 4](#_Toc163403908)

[I. GIỚI THIỆU CHUNG 5](#_Toc163403909)

[1. Giới thiệu chung về học phần 5](#_Toc163403910)

[1.1.JavaSwing 5](#_Toc163403911)

[1.2.Mô hình MVC 6](#_Toc163403912)

[2. Giới thiệu chung về dự án 9](#_Toc163403913)

[II. THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG 12](#_Toc163403914)

[1.Cơ sở dữ liệu 12](#_Toc163403915)

[2.Mô hình thực hiện 12](#_Toc163403916)

[III. GIAO DIỆN CÁC CHỨC NĂNG CỦA PHẦM MỀM 13](#_Toc163403917)

[1. Chức năng đăng nhập 13](#_Toc163403918)

[2. Chức năng quản lý đặt sân 14](#_Toc163403919)

[3. Chức năng thêm và tính tiền dịch vụ cho từng sân 15](#_Toc163403920)

[4. Chức năng quản lý dịch vụ 16](#_Toc163403921)

[5. Chức năng quản lý danh sách sân 16](#_Toc163403922)

[6. Chức năng quản lý khách hàng 17](#_Toc163403923)

[6. Chức năng quản lý hoá đơn và in hoá đơn 17](#_Toc163403924)

[7. Chức năng báo cáo thống kê 18](#_Toc163403925)

[IV. KIỂM THỬ 19](#_Toc163403926)

[1.Kết quả đạt được 19](#_Toc163403927)

[2.Hạn chế: 20](#_Toc163403928)

[KẾT LUẬN 21](#_Toc163403929)

# LỜI MỞ ĐẦU

Kính thưa quý thầy cô và các bạn,

Chúng em, nhóm 6, xin trân trọng gửi đến quý thầy cô và các bạn báo cáo về dự án Java nâng cao của chúng em với chủ đề "Quản lý Sân Cầu Lông". Trong suốt quá trình học tập và nghiên cứu, chúng em đã đặc biệt quan tâm đến việc áp dụng kiến thức và kỹ năng của môn Java vào việc phát triển một hệ thống quản lý hiệu quả cho các sân cầu lông.

Sân cầu lông là một trong những địa điểm phổ biến cho các hoạt động thể dục và giải trí. Tuy nhiên, việc quản lý các hoạt động và tài nguyên trên sân cầu lông có thể gặp phải nhiều thách thức, từ việc đặt lịch đến quản lý thành viên và tài liệu. Với mong muốn giải quyết những vấn đề này, chúng em đã xây dựng một ứng dụng phần mềm sử dụng ngôn ngữ lập trình Java.

Báo cáo này sẽ cung cấp một cái nhìn tổng quan về quá trình phát triển của chúng em, từ việc xác định yêu cầu đến triển khai và kiểm thử. Chúng em cũng sẽ trình bày các tính năng chính của ứng dụng và cách mà chúng được triển khai thông qua mã nguồn Javaswing.

Chúng em hy vọng rằng báo cáo này không chỉ là một cái nhìn sâu rộng về quá trình phát triển phần mềm sử dụng Java mà còn là một nguồn cảm hứng cho những ai quan tâm đến việc áp dụng kiến thức lập trình vào các vấn đề thực tế.

Xin chân thành cảm ơn sự quan tâm và hỗ trợ của quý thầy cô và các bạn.

Trân trọng,

Nhóm 6.

# GIỚI THIỆU CHUNG

## Giới thiệu chung về học phần

### 1.1.JavaSwing

Java Swing là một bộ công cụ giao diện người dùng (GUI) được xây dựng trên nền tảng Java, cho phép phát triển các ứng dụng có giao diện đồ họa đa năng và linh hoạt. Được giới thiệu từ phiên bản Java 1.2, Swing đã trở thành một phần không thể thiếu của hệ sinh thái Java cho đến ngày nay. Dưới đây là các điểm nổi bật và chi tiết về Java Swing:

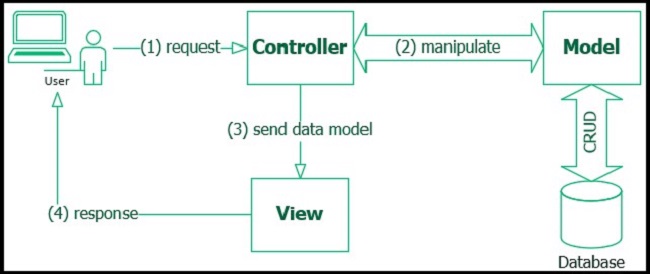
* Đa dạng và linh hoạt: Swing cung cấp một loạt các thành phần giao diện người dùng (UI components) đa dạng như JButton, JLabel, JTable, JList, JComboBox, JSlider, và nhiều hơn nữa. Các thành phần này có thể được kết hợp và tùy chỉnh để tạo ra giao diện người dùng phong phú và linh hoạt.
* Độc lập nền tảng: Ứng dụng sử dụng Java Swing có thể chạy trên nhiều nền tảng khác nhau mà không cần phải thay đổi mã nguồn, bao gồm Windows, macOS và Linux.
* Mô hình MVC (Model-View-Controller): Swing thúc đẩy việc sử dụng mô hình thiết kế Model-View-Controller (MVC), giúp tách biệt dữ liệu (Model) và giao diện người dùng (View) và logic điều khiển (Controller), tạo điều kiện cho việc phát triển và bảo trì ứng dụng một cách hiệu quả.
* Sự mở rộng: Swing là một bộ công cụ mở rộng cho AWT (Abstract Window Toolkit), mở ra khả năng tạo ra các giao diện người dùng phức tạp hơn và đẹp mắt hơn so với AWT.
* Hỗ trợ kỹ thuật kéo và thả (Drag and Drop): Swing cung cấp các công cụ để triển khai tính năng kéo và thả giữa các thành phần giao diện người dùng, giúp tăng trải nghiệm người dùng.
* Thư viện pluggable look-and-feel: Swing cho phép người dùng chọn lựa giao diện (look-and-feel) của ứng dụng, bao gồm các giao diện mặc định như Metal, Nimbus, và các giao diện tùy chỉnh.
* Hỗ trợ quốc tế hóa: Swing hỗ trợ việc phát triển ứng dụng đa ngôn ngữ và đa quốc gia thông qua việc sử dụng các gói ngôn ngữ và hệ thống ghi nhớ địa phương.
* Tích hợp dễ dàng: Swing tích hợp tốt với các công nghệ Java khác như JavaFX, JDBC, Servlets, và JSP.

### 1.2.Mô hình MVC

MVC được viết tắt từ Model View Controller. Đây là mô hình phần mềm dùng để tạo lập giao diện người dùng trên máy tính. Mô hình MVC bao gồm ba bộ phận chính là Model, View và Controller. Mỗi thành phần có chức năng, nhiệm vụ khác nhau nhưng giữa chúng có sự tương tác qua lại, hỗ trợ nhau. Trong đó:

* Model có chức năng quản lý và xử lý dữ liệu.
* View có nhiệm vụ hiển thị dữ liệu cho người dùng.
* Controller có chức năng điều khiển tương tác giữa Model và View.

Mô hình MVC trong Java giúp lập trình viên dễ dàng tách biệt giữa cách thức dữ liệu nội hàm với dữ liệu hiển thị. Sự tương tác qua lại giữa ba thành tố Model, View, Controller tạo nên hiệu quả tốt nhất cho việc lập trình. Chúng ta sẽ tìm hiểu chi tiết hơn về chức năng của từng thành phần trong MVC trong phần tiếp theo để từ đó hiểu và ứng dụng tốt nhất nhé!



Các thành phần trong mô hình MVC

a. Model

Một trong những thành phần quan trọng nhất của mô hình MVC trong Java. Đây là bộ phận làm nhiệm vụ quản lý dữ liệu. Model có chức năng vận chuyển thông tin từ nội hàm để hiển thị đến người dùng thông qua màn hình và xử lý các thông tin để người dùng dễ dàng tiếp cận nhất.

Model hoàn toàn độc lập với các thành phần còn lại trong MVC và nó chứa các tác vụ cần thiết nhất cho quá trình lập trình .

b. View

Thành phần tiếp theo chúng ta sẽ nhắc đến ở mô hình MVC trong Java, đó là View. Đối với người dùng thì View có vai trò thiết yếu. Nó thực hiện nhiệm vụ tạo tương tác với người dùng và hiển thị các kết quả từ tầng Controller. Đồng thời, View cũng thực hiện việc tiếp nhận các hoạt động, yêu cầu của người dùng để chuyển đến Controller xử lý.

Nói một cách dễ hiểu. Các thành phần trong MVC giống như một website. View chính là các trang giao diện hiển thị những gì mà người dùng nhìn thấy

c. Controller

Khi nhắc tới các thành phần ở mô hình MVC nhất định không thể bỏ qua Controller. Nếu không có thành phần này thì mọi hoạt động của Model hay View đều không còn giá trị.

Controller thực hiện chức năng kết nối tương tác giữa View và Model. Nó định nghĩa các lệnh và thực hiện xử lý các lệnh trong hệ thống. Controller đối chiếu hành động của người dùng từ View và tương tác với Model để chuyển tải thông tin cần thiết đến người dùng.

-Ưu điểm mô hình MVC

* Mô hình MVC mang đến sự chuyên nghiệp trong lập trình và phân tích đối tượng.
* Phần mềm có băng thông nhẹ nên hoạt động của website sẽ ổn định hơn.
* Dễ dàng kiểm tra thông tin, phát hiện lỗi phần mềm.
* Các thành phần trong ứng dụng được phân tách rõ ràng và độc lập khiến cho việc sử dụng, thao tác các lệnh trở nên đơn giản, thuận tiện hơn. Việc nâng cấp, bảo trì từng thành tố cũng dễ dàng.
* Model – View – Controller dù độc lập nhưng lại có mối quan hệ tương tác với nhau. Nó tạo nên một thể nhất quán trong quá trình lập trình.
* MVC hỗ trợ test driven development, có thể tạo ra ứng dụng với unit test.

-Nhược điểm mô hình MVC

* MVC chưa thực sự thích hợp với các ứng dụng nhỏ.
* Sử dụng mô hình MVC tốn rất nhiều thời gian, dễ gây ra áp lực trong khi lập trình.
* Đòi hỏi người dùng phải có kiến thức để vận dụng.
* Mô hình này sử dụng rất nhiều ngôn ngữ lập trình khác nhau nên có thể tạo ra sự cồng kềnh trong quá trình trung chuyển dữ liệu.

## Giới thiệu chung về dự án

#### **2.1.Mục tiêu của dự án**

Dự án quản lý sân cầu lông là một dự án phần mềm quan trọng, được thiết kế để cung cấp một hệ thống tự động hóa và hiệu quả cho việc quản lý hoạt động và tài nguyên của các sân cầu lông. Dự án này không chỉ giúp tối ưu hóa quy trình quản lý mà còn mang lại trải nghiệm thuận lợi và tiện ích cho cả nhân viên quản lý sân và người sử dụng dịch vụ.

-Mục tiêu của dự án:

* Tối ưu hóa quản lý: Dự án nhằm mục đích tạo ra một hệ thống quản lý sân cầu lông linh hoạt và dễ dàng sử dụng, giúp quản lý sân cầu lông dễ dàng kiểm soát lịch trình, thông tin thành viên và tài nguyên cần thiết.
* Nâng cao trải nghiệm người dùng: Bằng cách cung cấp một giao diện người dùng thân thiện và dễ sử dụng, dự án giúp cải thiện trải nghiệm của người dùng khi sử dụng dịch vụ sân cầu lông, từ việc đặt lịch đến thanh toán.
* Tối ưu hóa hiệu suất: Sử dụng mô hình MVC (Model-View-Controller), dự án tập trung vào việc tách biệt dữ liệu, giao diện và logic điều khiển, tạo điều kiện cho việc phát triển và bảo trì ứng dụng một cách hiệu quả.

-Yêu cầu :

* Giao diện phần mềm được thiết kế dễ sử dụng và có tính thẩm mỹ , an toàn dữ liệu.
* Được viết trên nền tảng javaswing theo mô hình MVC
* Hiệu năng: tốc độ, khả năng và độ tin cậy của hệ thống
* Phần mềm có bảo mật ,yêu cầu đăng nhập để thực hiện các chức năng của hệ thống
* Công việc tính toán thực hiện chính xác, không chấp nhận sai sót.
* Đảm bảo an toàn dữ liệu khi chạy phần mềm
* Thông tin được cập nhật và sửa đổi thường xuyên.
* Dễ bảo trì , thay đổi hoặc nâng cấp

#### **2.2 Các tính năng của dự án:**

-Quản lý đặt lịch sân:

* Cho phép người quản lý thêm các lịch đặt sân theo yêu cầu khách đặt
* Hiển thị tiền sân theo giờ đặt và loại sân, đồng thời hiển thị tiền dịch vụ của sân(nếu có)
* Thêm, sửa đổi và xóa thông tin đặt sân một cách dễ dàng.

-Quản lý khách hàng:

* Ghi danh và quản lý thông tin của các khách hàng bao gồm thông tin liên hệ, lịch sử đặt sân.
* Thêm, sửa đổi và xóa,tìm kiếm,xuất excel thông tin khách một cách dễ dàng.

-Quản lý sân :

* Quản lý danh sách các sân ,giá tiền từng sân
* Thêm , sửa, xoá,tìm kiếm, xuất excel danh sách các sân

-Quản lý dịch vụ:

* Ghi danh sách các mặt hàng dịch vụ như đồ ăn, đồ uống, dụng cụ…
* Quản lý các thông tin như mã dịch vụ, tên,giá tiền,..
* Các nút thêm, sửa , xoá,xuất excel, tìm kiếm ,đếm các dịch vụ

-Phiếu đặt hàng(dịch vụ):

* Ngoài đặt sân thì khách hàng đến sân cầu chơi có thể order thêm các dịch vụ ăn uống khác theo từng sân
* Form phiếu đặt hàng cho phép người quản lý tìm kiếm, thêm, sửa, xoá các dịch vụ của một sân và tính tiền tổng dịch vụ mà khách đã mua, sau khi tính tiền , tổng tiền dịch vụ sẽ tự động nhảy ra form danh sách đặt sân, phục vụ cho phần quản lý hoá đơn.

-Quản lý hoá đơn

* Quản lý danh sách, tính tiền hoá đơn của các sân
* Xem chi tiết hoá đơn của các sân
* In hoá đơn của từng sân

-Báo cáo và thống kê:

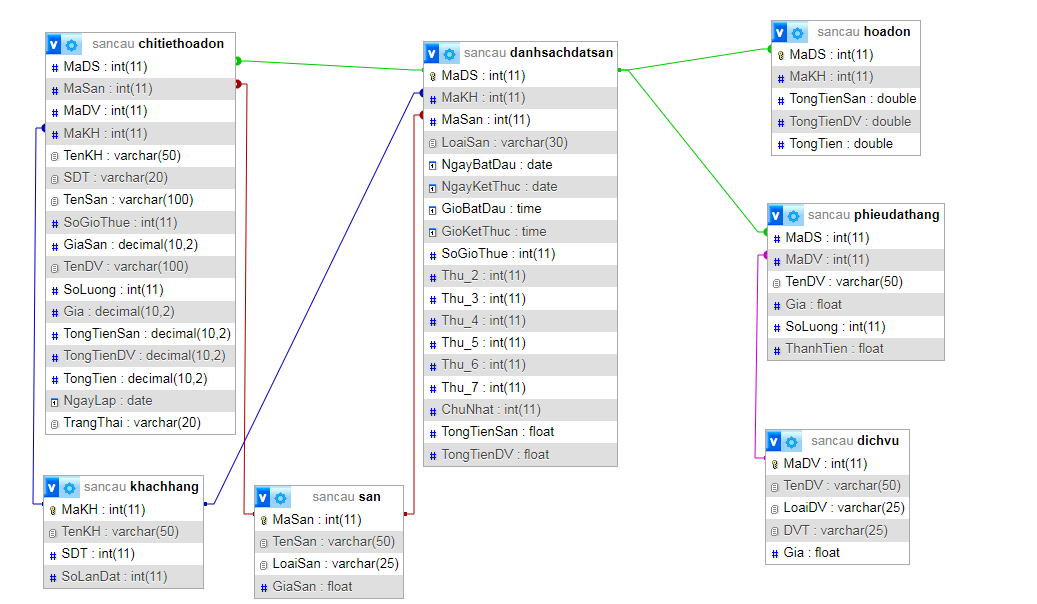
* Danh sách tất cả các giao dịch đặt sân, đặt dịch vụ của các sân
* Nút tìm kiếm trạng thái thanh toán

-Mô hình MVC (Model-View-Controller):

* Model: Bao gồm dữ liệu của ứng dụng, như thông tin khách hàng,đặt sân,dịch vụ, thông tin sân cầu lông, và các dữ liệu khác liên quan.
* View: Giao diện người dùng, hiển thị dữ liệu từ Model và cho phép người dùng tương tác.
* Controller: Logic điều khiển, xử lý sự kiện từ người dùng và cập nhật dữ liệu trong Model tương ứng.
* Sử dụng mô hình MVC giúp tách biệt các thành phần của ứng dụng, giúp dễ dàng bảo trì và mở rộng ứng dụng trong tương lai.

# THIẾT KẾ GIAO DIỆN NGƯỜI DÙNG

## 1.Cơ sở dữ liệu



## 2.Mô hình thực hiện

* Lớp model

A screenshot of a computer

Description automatically generatedA screenshot of a computer

Description automatically generated

* Lớp view

A screenshot of a computer

Description automatically generated

* Lớp controller

A screenshot of a computer

Description automatically generated

# GIAO DIỆN CÁC CHỨC NĂNG CỦA PHẦM MỀM

## Chức năng đăng nhập

A screenshot of a computer

Description automatically generated

## Chức năng quản lý đặt sân

-Danh sách đặt sân

A screenshot of a computer

Description automatically generated

-Thêm lịch đặt sân

A screenshot of a computer

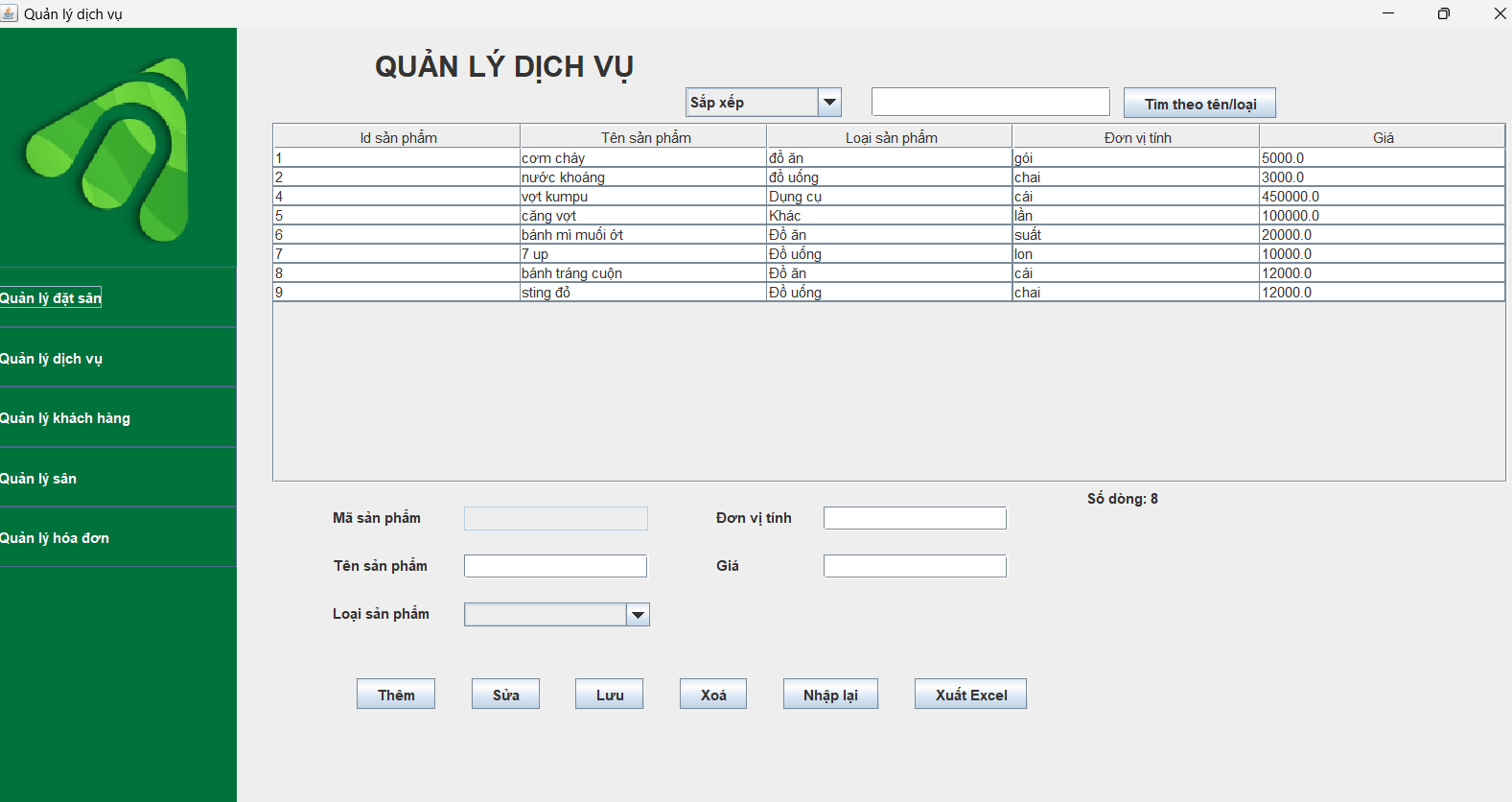
Description automatically generated

## Chức năng thêm và tính tiền dịch vụ cho từng sân

A screenshot of a computer

Description automatically generated

## Chức năng quản lý dịch vụ



## Chức năng quản lý danh sách sân

A screenshot of a computer

Description automatically generated

## 6. Chức năng quản lý khách hàng

A screenshot of a computer

Description automatically generated

## Chức năng quản lý hoá đơn và in hoá đơn

-Danh sách hoá đơn và chi tiết hoá đơn từng sân

A screenshot of a computer

Description automatically generated

-In hoá đơn

A white paper with black text

Description automatically generated

## Chức năng báo cáo thống kê

A screenshot of a computer

Description automatically generated

# KIỂM THỬ

## 1.Kết quả đạt được

* **Về yêu cầu:**
* Phầm mềm đã chạy được các chức năng đã đặt ra ban đầu
* Sử dụng ngôn ngữ đúng theo yêu cầu của giáo viên
* **Về công nghệ**
* Hiểu biết rõ hơn về quy trình phần mềm của một dự án thực tế: phân tích yêu cầu khác hàng, quá trình từ lúc bắt đầu và kết thúc của một dự án.
* Xây dựng thành công phần mềm quản lý sân cầu lông
* Thiết kế giao diện phần mềm đẹp và tiện lợi.
* Nắm vững về cách xây dựng một phầm mềm thông qua ngôn ngữ javaswing
* Sử dụng thành thạo các tool hỗ trợ lập trình: xampp, visual studio code, Netbean…
* **Về cài đặt chương trình**
* Giới thiệu về phần mềm quản lý sân cầu
* Cho phép người dùng đăng nhập , thực hiện các chức năng quản lý sân cầu lông, quản lý dịch vụ ăn uống kèm theo, đồng thời tính tiền , thông kê các số liệu
* Phần mềm cung cấp đầy đủ tính năng của một phần mềm quản lý : quản lý danh mục và duyệt các đăng ký
* **Tính năng khác**
* Giao diện thân thiện với người dùng
* Tính bảo mật khá cao, yêu cầu đăng nhập

## 2.Hạn chế:

- Một số chức năng chưa thực sự tối ưu nhất có thể

- Giao diện chưa thực sự bắt mắt

# KẾT LUẬN

Trong quá trình thực hiện dự án này, chúng em đã học hỏi rất nhiều từ việc áp dụng kiến thức Java vào việc phát triển một ứng dụng thực tế. Việc tạo ra một hệ thống quản lý sân cầu lông không chỉ đòi hỏi kiến thức kỹ thuật mà còn yêu cầu khả năng tư duy logic, phân tích và thiết kế hệ thống.

Chúng em muốn gửi lời cảm ơn sâu sắc tới quý thầy cô đã tận tình hướng dẫn và hỗ trợ trong suốt quá trình thực hiện dự án này. Sự chỉ dạy của quý thầy cô đã giúp chúng em vượt qua những thách thức và phát triển kỹ năng lập trình của mình.

Ngoài ra, chúng em cũng muốn bày tỏ lòng biết ơn đến các bạn trong nhóm vì sự cống hiến và cộng tác chặt chẽ của mọi người. Sự đoàn kết và hỗ trợ từ các thành viên đã giúp chúng em vượt qua mọi khó khăn và hoàn thành dự án một cách thành công.

Cuối cùng, chúng em hy vọng rằng ứng dụng quản lý sân cầu lông mà chúng em đã phát triển sẽ đóng góp vào việc cải thiện hiệu quả quản lý và tăng trải nghiệm cho người sử dụng. Chúng em sẽ tiếp tục nỗ lực học hỏi và phát triển kỹ năng của mình để có thể đóng góp tích cực hơn vào lĩnh vực công nghệ thông tin trong tương lai.

Xin chân thành cảm ơn!

Trân trọng,

Nhóm 6.