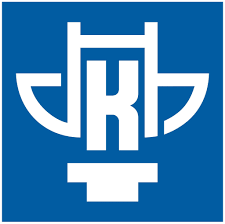
**TRƯỜNG ĐẠI HỌC KIẾN TRÚC HÀ NỘI**

**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO TIỂU LUẬN**

**MÔN HỌC: CÔNG NGHỆ JAVA**

***Đề tài***: **Phần mềm quản lý thông tin học viên trung tâm ngoại ngữ**

Nhóm 5

**Nhóm sinh viên thực hiện:**

1. Nguyễn Việt Anh 2155010016

**Lớp**: 21CN1

**Giảng viên hướng dẫn:** ThS. Lương Thị Bích Phượng

*Hà Nội, 17/01/2024*

**LỜI MỞ ĐẦU**

Trong thời đại công nghệ thông tin phát triển mạnh mẽ, việc quản lý thông tin học viên trong lớp học ngoại ngữ trở nên phức tạp và đòi hỏi sự chính xác và linh hoạt. Các lớp học ngoại ngữ thường có số lượng học viên đông đảo, với nhiều thông tin cá nhân và tiến độ học tập riêng biệt. Để đáp ứng nhu cầu quản lý hiệu quả, các giáo viên và nhà quản lý cần một hệ thống giúp họ tổ chức và lưu trữ thông tin một cách dễ dàng và có thể tra cứu nhanh chóng.

Mục tiêu của đề tài là áp dụng công nghệ Java để xây dựng một hệ thống quản lý học viên linh hoạt, tiện lợi và dễ sử dụng. Hệ thống này sẽ cung cấp một cơ sở dữ liệu để lưu trữ thông tin học viên, bao gồm tên, địa chỉ, thông tin liên lạc và thông tin học tập. Ngoài ra, hệ thống cũng sẽ cho xác định tiến độ học tập của từng học viên.

Trong quá trình nghiên cứu và xây dựng hệ thống, nhóm đã sử dụng Java và các công nghệ liên quan như lập trình hướng đối tượng (OOP), cơ sở dữ liệu, giao diện người dùng (GUI), và các thư viện hỗ trợ Java. Sử dụng lập trình hướng đối tượng, nhóm đã xây dựng các lớp và đối tượng để biểu diễn các khái niệm như học viên, lớp học và báo cáo. Cơ sở dữ liệu được sử dụng để lưu trữ và truy xuất thông tin một cách hiệu quả. Giao diện người dùng được thiết kế để đảm bảo tính thân thiện và dễ sử dụng cho các giáo viên và nhà quản lý.

Nhóm cũng đã áp dụng các nguyên tắc thiết kế và phát triển phần mềm để đảm bảo tính ổn định, mở rộng và bảo mật của hệ thống, chia nhỏ hệ thống thành các module riêng biệt, từ đó tăng khả năng mở rộng và bảo trì. Bên cạnh đó còn áp dụng các biện pháp bảo mật như xác thực người dùng và phân quyền truy cập để đảm bảo rằng thông tin học viên được bảo vệ và chỉ có người dùng có quyền truy cập mới có thể xem và chỉnh sửa.

**MỤC LỤC**

[**PHẦN I: ĐẶT VẤN ĐỀ** 4](#_Toc156343664)

[**PHẦN II: TÌM HIỂU ĐỀ TÀI** 6](#_Toc156343665)

[**I.** **Phân tích bài toán** 6](#_Toc156343666)

[**II.** **Cơ sở lý thuyết** 7](#_Toc156343667)

[**2.1.** **Ngôn ngữ lập trình Java** 7](#_Toc156343668)

[**2.2.** **Kỹ thuật JDBC kết nối cơ sở dữ liệu** 9](#_Toc156343669)

[**2.3.** **Lập trình ứng dụng GUI với java.** 13](#_Toc156343670)

[**III.** **Phân tích hệ thống** 16](#_Toc156343671)

[**3.1.** **Phân tích đề tài** 16](#_Toc156343672)

[**3.2.** **Sơ đồ Use case** 28](#_Toc156343673)

[**3.3.** **Biểu đồ phân cấp chức năng** 29](#_Toc156343674)

[**3.4.** **Biểu đồ luồng dữ liệu mức ngữ cảnh** 29](#_Toc156343675)

[**3.5.** **Biểu đồ luồng dữ liệu mức đỉnh** 30](#_Toc156343676)

[**3.6.** **Biểu đồ tuần tự** 30](#_Toc156343677)

[**IV.** **Bảng cơ sở dữ liệu** 34](#_Toc156343678)

[**4.1.** **Bảng qlhv** 35](#_Toc156343679)

[**4.2.** **Bảng lichhoc** 36](#_Toc156343680)

[**4.3.** **Bảng user** 37](#_Toc156343681)

[**4.4.** **Bảng bangdiem** 38](#_Toc156343682)

[**V.** **Thiết kế chương trình** 39](#_Toc156343683)

[**5.1.** **Giao diện đăng nhập** 39](#_Toc156343684)

[**5.2.** **Giao diện đăng ký** 39](#_Toc156343685)

[**5.3.** **Giao diện thay đổi mật khẩu** 40](#_Toc156343686)

[**5.4.** **Giao diện menu Lựa Chọn** 40](#_Toc156343687)

[**5.5.** **Giao diện quản lý lịch học** 41](#_Toc156343688)

[**5.6.** **Giao diện quản lý điểm** 41](#_Toc156343689)

[**5.7.** **Giao diện quản lý thông tin** 42](#_Toc156343690)

[**5.8.** **Giao diện khi chọn thoát chương trình** 42](#_Toc156343691)

[**VI.** **Kiểm thử và bảo trì** 43](#_Toc156343692)

[**6.1. Test case form Đăng Nhập, Đăng kí và Đổi Mật Khẩu** 43](#_Toc156343693)

[**6.2. Test case form Quản Lý Học Viên** 44](#_Toc156343694)

[**6.3. Test case form Quản Lý Điểm** 44](#_Toc156343695)

[**6.4. Test case form Quản Lý Lịch Học** 45](#_Toc156343696)

[**6.5. Bảo trì** 46](#_Toc156343697)

[**PHẦN 3: KẾT LUẬN** 47](#_Toc156343698)

[**TÀI LIỆU THAM KHẢO** 48](#_Toc156343699)

# **PHẦN I: ĐẶT VẤN ĐỀ**

1. **Lí do lựa chọn đề tài**

Trong lớp học ngoại ngữ, số lượng học viên thường đông đảo và đa dạng, điều này làm cho việc quản lý thông tin cá nhân, tiến độ học tập và kết quả đánh giá trở nên phức tạp. Do đó, một hệ thống quản lý học viên hiệu quả sẽ giúp tổ chức lớp học một cách chặt chẽ, tăng cường sự tổ chức và thuận tiện trong việc theo dõi và đánh giá tiến trình học tập.

Một hệ thống quản lý học viên chất lượng không chỉ giúp tăng cường hiệu quả quản lý mà còn tối ưu hóa quá trình giảng dạy. Bằng cách cung cấp thông tin chi tiết về tiến độ học tập và kết quả đánh giá của từng học viên, giáo viên có thể dễ dàng theo dõi sự tiến bộ và khả năng của học viên. Thông qua việc này, giáo viên có thể tạo ra những phương pháp giảng dạy phù hợp và cá nhân hóa, từ đó nâng cao chất lượng giáo dục và đảm bảo sự thành công của học viên.

Một hệ thống quản lý học viên cũng giúp quản lý dữ liệu hiệu quả. Thông qua việc tổ chức và lưu trữ thông tin học viên một cách cụ thể và hiệu quả, dữ liệu về học viên, kết quả đánh giá và các thông tin khác có thể được quản lý, tra cứu và bổ sung dễ dàng. Điều này giúp giảm thiểu sai sót trong việc lưu trữ và tìm kiếm thông tin, cung cấp một cơ sở dữ liệu chính xác và tin cậy để hỗ trợ quyết định và phân tích sâu hơn về quá trình học tập và tiến trình giảng dạy.

Cuối cùng, lựa chọn đề tài "Hệ thống quản lý học viên lớp học ngoại ngữ" còn phản ánh xu hướng phát triển công nghệ hiện đại trong lĩnh vực giáo dục. Áp dụng công nghệ vào quản lý học viên giúp nâng cao hiệu suất và chất lượng quản lý lớp học. Việc xây dựng và triển khai một hệ thống quản lý học viên lớp học ngoại ngữ mang lại nhiều lợi ích cho các bên liên quan.

1. **Bảng phân công công việc**

|  |  |  |  |
| --- | --- | --- | --- |
| **Tên thành viên** | **Mã sinh viên** | **Công việc được giao** | **Tiến độ hoàn thành** |
| Nguyễn Việt Anh (Nhóm trưởng) | 2155010016 | Tìm hiểu, lập trình và kết nối cở sở dữ liệu về Frame Quản lý Thông tin và Frame Đăng Nhập, Đăng Ký | 100% |
| Nguyễn Chu Kiều Trang | 2155010261 | Tìm hiểu, lập trình và kết nối cở sở dữ liệu về Frame Quản lý Lịch Học | 100% |
| Nguyễn Thị Tâm Nhi | 2155010191 | Tìm hiểu, lập trình và kết nối cở sở dữ liệu về Frame Quản lý Điểm | 100% |

# **PHẦN II: TÌM HIỂU ĐỀ TÀI**

1. **Phân tích bài toán**

Hệ thống quản lý học viên lớp học ngoại ngữ được thiết kế để giải quyết một số bài toán quan trọng. Một trong những bài toán đó là quản lý thông tin cá nhân của học viên. Hệ thống lưu trữ và cập nhật thông tin như mã học viên, họ tên, ngày sinh, địa chỉ, email, số điện thoại và giới tính. Quan trọng nhất là đảm bảo tính bảo mật và quyền riêng tư của thông tin này, đồng thời cho phép cập nhật và chỉnh sửa thông tin một cách dễ dàng và tiện lợi. Điều này giúp đảm bảo rằng thông tin học viên luôn được cập nhật và chính xác.

Một bài toán quan trọng khác trong hệ thống là liên kết thông tin học viên với mã khóa học. Mã khóa học giúp phân loại và nhận dạng học viên theo khóa học mà họ đang tham gia. Điều này tạo ra sự rõ ràng và có tổ chức trong quá trình quản lý, cho phép dễ dàng tìm kiếm và phân nhóm học viên theo nhu cầu.

Hệ thống cũng ghi nhận và lưu trữ điểm thi của học viên trong từng lần thi. Điều này cho phép giáo viên và học viên theo dõi tiến trình học tập và đánh giá kết quả. Học viên có thể xem điểm thi của mình và theo dõi sự tiến bộ trong quá trình học tập. Đồng thời, giáo viên cũng có thể sử dụng thông tin này để đánh giá hiệu quả giảng dạy và đưa ra phương pháp hỗ trợ phù hợp cho học viên.

Một chức năng quan trọng khác của hệ thống là khả năng chỉnh sửa thông tin học viên. Người quản lý hoặc học viên có thể cập nhật và sửa đổi thông tin cá nhân như địa chỉ, email, số điện thoại và thông tin khác. Điều này đảm bảo rằng thông tin được cập nhật chính xác và nhanh chóng, và giúp học viên và người quản lý duy trì thông tin học viên một cách hiệu quả.

1. **Cơ sở lý thuyết**
   1. **Ngôn ngữ lập trình Java**
      1. **Lịch sử phát triển**

Cuối năm 1990, James Gosling được công ty Sun Microsystems giao nhiệm vụ xây dựng phần mềm cho các thiết bị điện tử dân dụng như lò nướng, PDA (Personal Digital Assistant),… Lúc đầu, Gosling và các cộng sự định chọn ngôn ngữ C++ nhưng thấy rằng C++ có những hạn chế. Chương trình viết bằng C++ khi chuyển sang chạy trên một hệ thống có bộ vi xử lý khác thì đòi hỏi phải biên dịch lại.

Gosling quyết định xây dựng một ngôn ngữ mới dựa trên ngôn ngữ C/C++ và đặt tên là Oak (cây sồi). Năm 1995, Oak đổi tên thành Java – một loại cà phê đến từ Indonesia. Năm 2009, Oracle mua lại Sun Microsystems và quyền phát triển Java thuộc về Oracle.

* + 1. **Đặc điểm của Java**

1. *Độc lập phần cứng và hệ điều hành*

Một chương trình viết bằng ngôn ngữ Java có thể chạy tốt ở nhiều môi trường khác nhau. Gọi là khả năng “cross-platform”. Khả năng độc lập phần cứng và hệ điều hành được thể hiện ở 2 cấp độ là cấp độ mã nguồn và cấp độ nhị phân.

Ở cấp độ mã nguồn: Kiểu dữ liệu trong Java nhất quán cho tất cả các hệ điều hành và phần cứng khác nhau. Java có riêng một bộ thư viện để hỗ trợ vấn đề này. Chương trình viết bằng ngôn ngữ Java có thể biên dịch trên nhiều loại máy khác nhau mà không gặp lỗi.

Ở cấp độ nhị phân: Một mã biên dịch có thể chạy trên nhiều nền tảng khác nhau mà không cần dịch lại mã nguồn. Tuy nhiên cần có Java Virtual Machine để thông dịch đoạn mã này.

1. *Ngôn ngữ thông dịch*

Ngôn ngữ lập trình thường được chia ra làm 2 loại (tùy theo các hiện thực hóa ngôn ngữ đó) là ngôn ngữ thông dịch và ngôn ngữ biên dịch.

Thông dịch (Interpreter) : Nó dịch từng lệnh rồi chạy từng lệnh, lần sau muốn chạy lại thì phải dịch lại.

Biên dịch (Compiler): Code sau khi được biên dịch sẽ tạo ra 1 file thường là .exe, và file .exe này có thể đem sử dụng lại không cần biên dịch nữa. Ngôn ngữ lập trình Java thuộc loại ngôn ngữ thông dịch. Chính xác hơn, Java là loại ngôn ngữ vừa biên dịch vừa thông dịch. Khi viết mã, hệ thống tạo ra một tệp .java. Khi biên dịch mã nguồn của chương trình sẽ được biên dịch ra mã byte code. Máy ảo Java (Java Virtual Machine) sẽ thông dịch mã byte code này thành machine code (hay native code) khi nhận được yêu cầu chạy chương trình.

1. *Cơ chế thu gom rác tự động*

Khi tạo ra các đối tượng trong Java, JRE sẽ tự động cấp phát không gian bộ nhớ cho các đối tượng ở trên heap.

Với ngôn ngữ như C C++, bạn sẽ phải yêu cầu hủy vùng nhớ mà bạn đã cấp phát, để tránh việc thất thoát vùng nhớ. Tuy nhiên vì một lý do nào đó, bạn không hủy một vài vùng nhớ, dẫn đến việc thất thoát và làm giảm hiệu năng chương trình.

Ngôn ngữ lập trình Java hỗ trợ cho bạn điều đó, nghĩa là bạn không phải tự gọi hủy các vùng nhớ. Bộ thu dọn rác của Java sẽ theo vết các tài nguyên đã được cấp. Khi không có tham chiếu nào đến vùng nhớ, bộ thu dọn rác sẽ tiến hành thu hồi vùng nhớ đã được cấp phát.

1. *Đa luồng*

Java hỗ trợ lập trình đa tiến trình (multithread) để thực thi các công việc đồng thời. Đồng thời cũng cung cấp giải pháp đồng bộ giữa các tiến trình (giải pháp sử dụng priority…).

1. *Tính an toàn và bảo mật*

*- Tính an toàn:*

Ngôn ngữ lập trình Java yêu cầu chặt chẽ về kiểu dữ liệu. Dữ liệu phải được khai báo tường minh. Không sử dụng con trỏ và các phép toán với con trỏ. Java kiểm soát chặt chẽ việc truy nhập đến mảng, chuỗi. Không cho phép sử dụng các kỹ thuật tràn. Do đó các truy nhập sẽ không vượt quá kích thước của mảng hoặc chuỗi.

Quá trình cấp phát và giải phóng bộ nhớ được thực hiện tự động. Cơ chế xử lý lỗi giúp việc xử lý và phục hồi lỗi dễ dàng hơn.

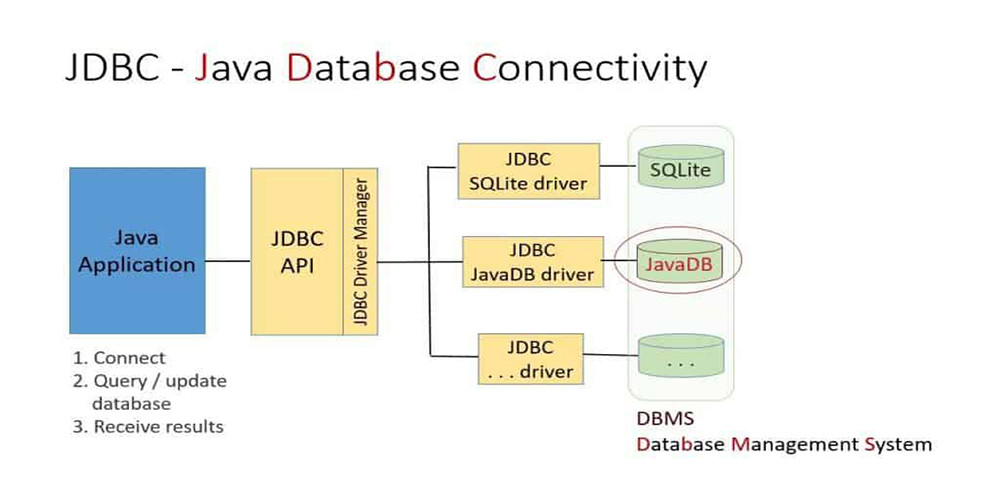
*- Tính bảo mật:*

Java cung cấp một môi trường quản lý chương trình với nhiều mức khác nhau.

* Mức 1 : Chỉ có thể truy xuất dữ liệu cũng như phương phức thông qua giao diện mà lớp cung cấp.
* Mức 2 : Trình biên dịch kiểm soát các đoạn mã sao cho tuân thủ các quy tắc của ngôn ngữ lập trình Java trước khi thông dịch.
* Mức 3 : Trình thông dịch sẽ kiểm tra mã byte code xem các đoạn mã này có đảm bảo được các quy định, quy tắc trước khi thực thi.
* Mức 4: Java kiểm soát việc nạp các lớp vào bộ nhớ để giám sát việc vi phạm giới hạn truy xuất trước khi nạp vào hệ thống.
  1. **Kỹ thuật JDBC kết nối cơ sở dữ liệu** 
     1. **Kỹ thuật JDBC**
        1. *JDBC là gì ?*

JDBC (Java Database Connectivity) là một API tiêu chuẩn dùng để tương tác với các loại cơ sở dữ liệu quan hệ. JDBC có một tập hợp các class và các Interface dùng cho ứng dụng Java có thể nói chuyện với các cơ sở dữ liệu.

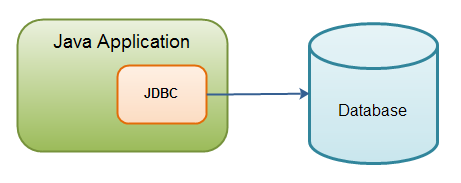
Các thành phần của JDBC Api về cơ bản bao gồm:

* *DriverManager:* Là một class, nó dùng để quản lý danh sách các Driver (database drivers).
* *Driver:* Là một Interface, nó dùng để liên kết các liên lạc với cơ sở dữ liệu, điều khiển các liên lạc với database. Một khi Driver được tải lên, lập trình viên không cần phải gọi nó một cách cụ thể.
* *Connection :* Là một Interface với tất cả các method cho việc liên lạc với database. Nó mô tả nội dung liên lạc. tất cả các thông tin liên lạc với cơ sở dữ liệu là thông qua chỉ có đối tượng Connection.
* *Statement :* Là một Interface, gói gọn một câu lệnh SQL gửi tới cơ sở dữ liệu được phân tích, tổng hợp, lập kế hoạch và thực hiện.
* *ResultSet:* ResultSet đại diện cho tập hợp các bản ghi lấy do thực hiện truy vấn.

*Hình 2.1. JDBC – Java Database Connectivity*

* + - 1. *Nguyên tắc kết nối*

Java sử dụng JDBC để làm việc với các cơ sở dữ liệu. Ví dụ bạn làm việc với cơ sở dữ liệu Oracle từ Java bạn cần phải có Driver (Đó là class điều khiển việc kết nối với loại cơ sở dữ liệu bạn muốn). Trong JDBC API chúng ta có java.sql.Driver, nó chỉ là một interface, và nó có sẵn trong JDK. Như vậy bạn phải download thư viện Driver ứng với loại Database mà bạn mong muốn.

Chẳng hạn với Oracle thì class thi hành Interface java.sql.Driver đó là: oracle.jdbc.driver.OracleDriver, java.sql.DriverManager là một class trong JDBC API. Nó làm nhiệm vụ quản lý các Driver.

*Hình 2.2. Nguyên tắc kết nối Java với Database*

Chúng ta có 2 cách để làm việc với một loại cơ sở dữ liệu cụ thể nào đó:

Cách 1: Bạn hãy cung cấp thư viện Driver điều khiển loại cơ sở dữ liệu đó, đây là cách trực tiếp. Nếu bạn dùng DB oracle (hoặc DB khác) bạn phải download thư viện dành cho loại DB này.

Cách 2: Khai báo một "ODBC DataSource", và sử dụng cầu nối JDBC-ODBC để kết nối với "ODBC DataSource" kia. Cầu nối JDBC-ODBC là thứ có sẵn trong JDBC API.

* + - 1. **Cách thức kết nối**

Bước 1: Tải driver JDBC MySQL từ trang chính thức của MySQL. Sau đó, thêm file JAR của driver vào classpath của dự án Java.

Bước 2: Trong mã nguồn Java, import các gói cần thiết như java.sql.Connection, java.sql.DriverManager, và java.sql.SQLException.

Bước 3: Sử dụng DriverManager.getConnection() để kết nối đến cơ sở dữ liệu MySQL. Cung cấp URL kết nối, tên người dùng và mật khẩu trong quá trình này.

Bước 4: Kiểm tra xem kết nối đã được thiết lập thành công hay không và in ra thông báo xác nhận nếu có.

Bước 5: Trong khối try-catch, thực hiện các thao tác cần thiết với cơ sở dữ liệu như thực hiện truy vấn SQL hoặc cập nhật dữ liệu. Đảm bảo rằng kết nối được đóng đúng cách trong khối finally bằng cách sử dụng connection.close().

* + 1. **Cơ sở dữ liệu MySQL**

1. *MySQL là gì ?*

MySQL là một hệ thống quản trị cơ sở dữ liệu mã nguồn mở (Relational Database Management System, viết tắt là RDBMS) hoạt động theo mô hình client-server. RDBMS là một phần mềm hay dịch vụ dùng để tạo và quản lý các cơ sở dữ liệu (Database) theo hình thức quản lý các mối liên hệ giữa chúng.

Database là tập hợp dữ liệu theo cùng một cấu trúc. Cơ sở dữ liệu, hay database, là nơi chứa và sắp đặt dữ liệu. Dữ liệu được đặt trong một bộ dữ liệu chung, dataset, được tổ chức sắp xếp giống như một bảng tính vậy. Mỗi “bảng” này có liên hệ với nhau theo cách nào đó.

1. *Khái niệm về XAMPP*

**XAMPP**là một phần mềm cho phép **giả lập** môi trường server hosting ngay trên máy tính của bạn, cho phép bạn **chạy demo website** mà **không**cần phải mua[hosting](https://www.thegioididong.com/hoi-dap/hosting-la-gi-co-may-loai-kinh-nghiem-chon-mua-hosting-1337771)hay[VPS](https://www.thegioididong.com/hoi-dap/vps-la-gi-dung-de-lam-gi-co-uu-nhuoc-diem-nhu-the-nao-1337214). Chính vì vậy, XAMPP hay được phục vụ cho hoạt động học tập giảng dạy thực hành và phát triển web.

**XAMPP** được viết tắt của **X + Apache + MySQL + PHP + Perl** vì nó được tích hợp sẵn [Apache](https://vi.wikipedia.org/wiki/Apache_(HTTP)), [MySQL](https://www.thegioididong.com/game-app/huong-dan-cach-tai-cai-dat-mysql-ban-moi-nhat-chi-tiet-tung-1299084), [PHP](https://vi.wikipedia.org/wiki/PHP), [FTP server](https://www.thegioididong.com/hoi-dap/ftp-la-gi-3-dieu-co-ban-ban-can-biet-ve-giao-thuc-ftp-1339006), Mail Server. Còn X thể hiện cho sự đa nền tảng của XAMPP vì nó có thể dùng được cho 4 hệ điều hành khác nhau: [Windows](https://www.thegioididong.com/phan-mem/windows-10-home-32-bit-64-bit-all-languages-kw9-0), [MacOS](http://www.thegioididong.com/hoi-dap/he-dieu-hanh-macos-la-gi-956642), [Linux](https://www.thegioididong.com/hoi-dap/he-dieu-hanh-linux-la-gi-uu-nhuoc-diem-cua-he-dieu-hanh-1312530) và [Solaris](https://vi.wikipedia.org/wiki/Solaris_(h%E1%BB%87_%C4%91i%E1%BB%81u_h%C3%A0nh)).

Việc sử dụng XAMPP đối với đề tài:

- Có thể chạy được linh hoạt được trên mọi hệ điều hành: Cross-platform, Windows, MacOS, Linux.

- Có cấu hình đơn giản nhưng mang đến nhiều chức năng hữu ích như: Lập Server giả định, lập Mail Server giả định và hỗ trợ SSL trên localhost.

- Tích hợp được nhiều tính năng với các thành phần quan trọng như: Apache, PHP, MySql. Vì thế, người dùng không cần cài đặt từng phần trên riêng lẻ mà chỉ cần cài XAMPP là có 1 web server hoàn chỉnh.

- Tạo mã nguồn mở: Giao diện quản lý dễ dàng và tiện lợi giúp người dùng luôn chủ động được trong chế độ khởi động lại hay bật/ tắt đối với máy chủ theo thời gian phù hợp nhất.

* 1. **Lập trình ứng dụng GUI với java.**

1. *GUI là gi?*

GUI là viết tắt của từ Graphic User Interface là giao diện người dùng đồ họa. Đối với Console là giao diện cửa sổ dòng lệnh. Một hệ thống GUI là sự kết hợp của các công nghệ, thiết bị để cung cấp cho người dùng một nền tảng cho phép người sử dụng có thể tương tác với nó.

Một chuỗi các thành phần của GUI tuân theo một ngôn ngữ trực quan (visual language) để biểu diễn thông tin được lưu trữ trong các máy tính. Thông dụng nhất khi kể đến sự kết hợp các thành phần như vậy là mô hình WIMP (Window, Icon, Menu, Pointing device) trong các máy tính cá nhân. Một số phần mềm GUI như là Eclipse, NetBeans, hay bộ Office của Microsoft đều là những phần mềm GUI. Java hỗ trợ các Framework để các lập trình viên có thể lập trình giao diện GUI cho ứng dụng một cách tiện lợi và dễ dàng nhất.

1. *Lập trình ứng dụng GUI với java*

Hiện tại có 3 Framework phổ biến hỗ trợ lập trình GUI trong Java:

* *AWT:* Xuất hiện từ những thế hệ đầu tiên, chính vì vậy Framework này tồn tại khá nhiều khiếm khuyết. AWT cung cấp một lớp trừu tượng cho các hệ điều hành khác nhau, khi viết chương trình thì nó sẽ có giao diện như chương trình gốc của hệ điều hành đó. AWT bao gồm:

Container: Container là một thành phần trong AWT có thể chứa các thành phần khác như button, textfield, label, v.v ... Các lớp mở rộng lớp Container được gọi là container như Frame, Dialog và Panel.

Window: Cửa sổ là container không có borders và menu bars. Bạn phải sử dụng khung, hộp thoại hoặc cửa sổ khác để tạo cửa sổ.

Panel: Panel là một container nhưng không chứa các thành phần như title bar và menu bars. Có có thể có các thành phần khác.

Frame: Frame là một container chứa các thành phần title bar và có thể chưa menu bars. Nó cũng có thể chứa các thành phần khác.

* *Swing:* Được coi là đứa em của AWT, Swing được ra đời với nhiều cải tiến và nâng cấp hơn so với người anh tiền nhiệm. Java Swing là một phần của Oracle’s Java Foundation Classes (JFC). Là một Framework cho lập trình viên lập trình giao diện trên Java. Swing được phát triển để cung cấp các thành phần GUI phức tạp hơn, nhiều hơn so với bộ công cụ AWT.

Swing cung cấp giao diện mô phỏng giao diện của một số nền tảng. Điểm khác so với AWT đó là các ứng dụng GUI sẽ không liên quan đến nền tảng bên dưới mà sẽ chạy trực tiếp bằng máy ảo của Java. Nó có các thành phần mạnh mẽ hơn AWT. Ngoài các thành phần cơ bản như button, textfield, label... Swing cũng cung cấp một số thành phần nâng cao như table panel, scroll panes, trees, table và lists. Không giống các thành phần của AWT, các thành phần của Swing không được triển khai theo mã dành riêng cho nền tảng.

* *JavaFX:* Đứa em út được ra đời sau cùng và được cho rằng sẽ kế thừa người anh của mình đó chính là JavaFX. JavaFX nó cung cấp giao diện đẹp hơn, trau chuốt hơn, do dựa trên CSS / FXML nhưng cũng phức tạp hơn so với Swing.

JavaFX là một tập hợp các packages và phương tiện cho phép các nhà phát triển thiết kế, kiểm tra, gỡ lỗi và triển khai các ứng dụng client một cách phong phú và hoạt động nhất quán trên các nền tảng khác nhau.

JavaFX được cải tiến hơn so với hai người anh trước đó vì những đặc điểm sau:

* JavaFX là một thư viện Java bao gồm các lớp và các giao diện được viết bằng mã Java gốc.
* FXML là một ngôn ngữ khai báo dựa trên XML để xây dựng một giao diện người dùng trong ứng dụng JavaFX.
* Lập trình viên có thể sử dụng JavaFX Scene Builder để thiết kế giao diện đồ họa GUI.
* JavaFX có thể được tùy biến giao diện thêm sinh động bằng cách sử dụng css.
* JavaFX còn hỗ trợ đồ họa 2D và 3D cũng như hỗ trợ cả âm thanh và video.
* JavaFX còn có WebView dựa trên trình duyệt Webkit, vì vậy bạn có thể nhúng các trang web hoặc các ứng dụng web vào bên trong JavaFX.

1. **Phân tích hệ thống**
   1. **Phân tích đề tài**
      1. **Đối tượng sử dụng**

Chương trình được tạo ra với mục đích là quản lý thông tin, quản lý lịch học và quản lý điểm của học học viên của trung tâm ngoại ngữ nên đối tượng sự dụng chương trình sẽ là nhân viên và giảng viên. Nhân viên là người quản lý thông tin học viên, thông tin lịch học của học viên và cả giáo viên cùng nhân viên sẽ đều có thể quản lý điểm của học viên trong phạm vi một trung tâm ngoại ngữ nên chương trình sẽ có đầy đủ chức năng để hỗ trợ cho nhân viên và giảng viên.

* + 1. **Chức Năng chính của hệ thống** 
       1. **Chức năng Thêm**

1. *Thêm danh sách học viên*
   * Mỗi sinh viên là 1 mã học viên, thông tin của học viên bao gồm: mã khóa học, mã học viên, họ tên, ngày sinh, địa chỉ, số điện thoại, email, giói tính, lớp, trình độ, điểm đánh giá đầu vào.
   * Hiển thị đầy đủ thông tin học viên vào danh sách khi ấn “Lưu”.
2. *Thêm lịch học cho học viên*

*-* Mỗi lịch học sẽ có 1 mã lịch học riêng, thông tin của lịch học bao gồm: ID, môn học, tên giảng viên, thời gian, địa điểm, lớp.

- Hiển thị đầy đủ thông tin lịch học vào danh sách khi ấn “Thêm”.

1. *Thêm điểm cho học viên*

- Mỗi học viên sẽ có 1 mã học viên riêng, điều đó tương đương với mã học viên trong việc quản lý điểm bao gồm: mã học viên, lớp, điểm kiểm tra lần 1, điểm kiểm tra lần 2, điểm kiểm tra lần 2, điểm trung bình, Đánh giá.

- Hiển thị đầy đủ thông tin điểm, tính điểm trung bình và tự động đưa ra đánh giá vào danh sách khi ấn “Thêm”.

* + - 1. **Chức năng Cập Nhật**

1. *Cập Nhật danh sách học viên*
   * Chỉnh sửa các thông tin của học viên, trừ mã học viên
   * Hiển thị đầy đủ lại thông tin học viên sau khi chỉnh sửa vào danh sách khi ấn “Lưu”.
2. *Cập Nhật lịch học cho học viên*

- Chỉnh sửa các thông tin của lịch học, trừ ID

- Hiển thị đầy đủ lại thông tin lịch học sau khi chỉnh sửa vào danh sách khi ấn “Lưu”.

1. *Cập Nhật điểm cho học viên*

- Chỉnh sửa các thông tin điểm của học viên, trừ mã học viên. Người dùng có thể chỉnh sửa các đầu điểm và kết quả sẽ được cập nhập tự động

- Hiển thị đầy đủ thông tin điểm, tính điểm trung bình và tự động đưa ra đánh giá vào danh sách khi ấn “Lưu”.

* + - 1. **Chức năng Xóa và Lưu danh sách**

1. *Xóa và Lưu danh sách học viên*
   * Xóa toàn bộ thông tin của học viên được chọn sau khi ấn nút “Xóa”.
   * Lưu toàn bộ thông tin trên bảng thành file NotePad sau khi ấn “Save”.
2. *Cập Nhật lịch học cho học viên*

* Xóa toàn bộ thông tin của lịch học được chọn sau khi ấn nút “Xóa”.
* Lưu toàn bộ thông tin trên bảng thành file NotePad sau khi ấn “Save”.

1. *Thêm điểm cho học viên*
   * Xóa toàn hộ thông tin điểm sau khi ấn nút “Xóa”.

* Lưu toàn bộ thông tin trên bảng thành file NotePad sau khi ấn “Save”.
  + - 1. **Chức năng Tìm Kiếm**

1. *Tìm kiếm thông tin học viên*
   * Tìm kiếm mã học viên và trình độ và hiển thị lên bảng đầy đủ thông tin về học viên sau khi ấn nút “Tìm Kiếm”
   * Trả lại danh sách học viên ban đầu khi người dùng ấn “Hủy Tìm Kiếm”.
2. *Tìm kiếm lịch học*

* Hiển thị các thông tin lịch học sau khi người dùng nhập tên môn học muốn tìm kiếm và ấn “Tìm Kiếm”.
  + Trả lại danh sách lịch học ban đầu khi người dùng ấn “Hủy Tìm Kiếm”.

1. *Tìm kiếm điểm* 
   * Hiển thị các thông tin điểm sau khi người dùng nhập mã học viên và ấn “Tìm Kiếm”.

* Trả lại danh sách điểm ban đầu sau khi người dùng ấn “Hủy Tìm Kiếm”.
  + 1. **Kịch bản phần mềm** 
       1. **Kịch bản Quản lý thông tin học viên**

1. *Thêm danh sách học viên*

|  |  |
| --- | --- |
| Kịch bản | Thêm học viên vào danh sách |
| Tác nhân | Nhân viên |
| Điều kiện cần | Nhân viên đăng nhập thành công |
| Mục đích cần đạt | Thêm thông tin thành công |
| Sự kiện chính | 1. Nhân viên chọn mục thông tin học viên từ menu chính 2. Nhân viên nhập thông tin học viên từ các ô nhập thông tin từ màn hình   Sau khi nhập đầy đủ thông tin nhân viên nhấn nút lưu   1. Thông tin được hiển thị vào Table. |
| Ngoại lệ | 1. Hệ thống báo khi    * Nhân viên nhập các thông tin chưa đầy đủ    * Nhân viên nhập sai thông tin, định dạng |

1. *Cập Nhật thông tin học viên*

|  |  |
| --- | --- |
| Kịch bản | Cập Nhật thông tin học viên |
| Tác nhân | Nhân viên |
| Điều kiện cần | Đăng nhập thành công |
| Mục đích cần đạt | Cập Nhật thành công thông tin học viên |
| Sự kiện chính | * Nhân viên chọn mục thông tin học viên trên menu chính * Nhân viên chọn một học viên cần sửa thông tin trên bảng Danh sách học viên. * Chỉnh sửa thông tin học viên chưa chính xác. * Nhân viên bấm lưu thông tin * Màn hình hiển thị thông báo sửa thành công |
| Ngoại lệ | * Thông tin cần sửa không hợp lệ |

1. *Xóa học viên*

|  |  |
| --- | --- |
| Kịch bản | Xoá học viên |
| Tác nhân | Nhân viên |
| Điều kiện cần | Nhân viên đăng nhập thành công |
| Mục đích đạt được | Xoá sinh viên khỏi danh sách |
| Sự kiện chính | * Nhân viên chọn mục thông tin học viên * Nhân viên chọn học viên cần xóa * Nhân viên có thể sử dụng nút tìm kiếm mã học viên nếu danh sanh sách học viên quá nhiều * Nhân viên chọn nút xoá * Học viên bị xóa khỏi danh sách |
| Ngoại lệ |  |

1. *Tìm kiếm học viên*

|  |  |
| --- | --- |
| Kịch bản | Tìm Kiếm học viên |
| Tác nhân | Nhân viên |
| Điều kiện cần | Nhân viên đăng nhập thành công |
| Mục đích đạt được | Tìm kiếm thành công học viên |
| Sự kiện chính | * Nhân viên chọn mục thông tin học viên * Nhân viên nhập thông tin vào khu vực tìm kiếm bao gồm: mã học viên và trình độ * Nhân viên ấn nút “Tìm kiếm” và học viên đúng với thông tin tìm kiếm sẽ hiển thị. * Nhân viên ấn nút “ Hủy Tìm kiếm” và danh sách học viên sẽ trở lại về trạng thái ban đầu |
| Ngoại lệ | - Thông tin tìm kiếm không chính xác |

1. *Lưu danh sách học viên*

|  |  |
| --- | --- |
| Kịch bản | Lưu danh sách học viên |
| Tác nhân | Nhân viên |
| Điều kiện cần | Nhân viên đăng nhập thành công |
| Mục đích đạt được | Xuất thành công danh sách học viên |
| Sự kiện chính | * Nhân viên chọn mục thông tin học viên * Nhân viên chọn vào File và chọn nút Save * Màn hình hiển thị lên cửa sổ địa chỉ mà người dùng muốn lưu file * Danh sách được tải xuống. |
| Ngoại lệ |  |

* + - 1. **Kịch bản Quản lý lịch học**

1. *Thêm danh sách lịch học*

|  |  |
| --- | --- |
| Kịch bản | Thêm lịch học vào danh sách |
| Tác nhân | Nhân viên |
| Điều kiện cần | Nhân viên đăng nhập thành công |
| Mục đích cần đạt | Thêm thông tin lịch học thành công |
| Sự kiện chính | 1. Nhân viên chọn mục “Lịch học” từ menu chính 2. Nhân viên chọn “Thêm” 3. Nhân viên nhập thông tin học viên từ các ô nhập thông tin từ màn hình   Sau khi nhập đầy đủ thông tin nhân viên nhấn nút lưu   1. Thông tin lịch học được hiển thị vào Table. |
| Ngoại lệ | 1. Hệ thống báo khi    * Nhân viên nhập sai thông tin, định dạng    * Thêm lịch học không thành công |

1. *Cập Nhật thông tin lịch học*

|  |  |
| --- | --- |
| Kịch bản | Cập Nhật thông tin lịch học |
| Tác nhân | Nhân viên |
| Điều kiện cần | Đăng nhập thành công |
| Mục đích cần đạt | Cập Nhật thành công thông tin lịch học |
| Sự kiện chính | * Nhân viên chọn mục “Lịch học” trên menu chính * Nhân viên chọn một lịch học cần sửa thông tin trên bảng danh sách lịch học. * Chỉnh sửa thông tin lịch học. * Nhân viên bấm lưu thông tin * Màn hình hiển thị lịch học được cập nhật. |
| Ngoại lệ | * Thông tin cần sửa không hợp lệ |

1. *Xóa Lịch học*

|  |  |
| --- | --- |
| Kịch bản | Xoá lich học |
| Tác nhân | Nhân viên |
| Điều kiện cần | Nhân viên đăng nhập thành công |
| Mục đích đạt được | Xoá sinh viên khỏi danh sách |
| Sự kiện chính | * Nhân viên chọn mục Lịch Học * Nhân viên chọn lich học cần xóa * Nhân viên có thể sử dụng nút tìm kiếm tên môn học trong khu vực tìm kiếm * Nhân viên chọn nút xoá * Lịch học bị xóa khỏi danh sách |
| Ngoại lệ |  |

1. *Tìm kiếm Lịch Học*

|  |  |
| --- | --- |
| Kịch bản | Tìm kiếm lịch học |
| Tác nhân | Nhân viên |
| Điều kiện cần | Nhân viên đăng nhập thành công |
| Mục đích đạt được | Tìm kiếm thành công lịch học |
| Sự kiện chính | * Nhân viên chọn mục “Lịch Học” * Nhân viên nhập thông tin tên môn học vào khu vực tìm kiếm. * Nhân viên ấn nút “Tìm kiếm” và lịch học đúng với thông tin tìm kiếm sẽ hiển thị. * Nhân viên ấn nút “ Hủy Tìm kiếm” và danh sách lịch học sẽ trở lại về trạng thái ban đầu |
| Ngoại lệ | - Thông tin tìm kiếm không chính xác |

1. *Lưu danh sách học viên*

|  |  |
| --- | --- |
| Kịch bản | Lưu danh sách lịch học |
| Tác nhân | Nhân viên |
| Điều kiện cần | Nhân viên đăng nhập thành công |
| Mục đích đạt được | Xuất thành công danh sách lịch học |
| Sự kiện chính | * Nhân viên chọn mục “Lịch Học” * Nhân viên chọn nút Save * Màn hình hiển thị lên cửa sổ địa chỉ mà người dùng muốn lưu file * Danh sách được tải xuống. |
| Ngoại lệ |  |

* + - 1. **Kịch bản Quản lý điểm học viên**

1. *Thêm danh sách điểm*

|  |  |
| --- | --- |
| Kịch bản | Thêm lịch học vào danh sách |
| Tác nhân | Nhân viên hoặc giảng viên |
| Điều kiện cần | Nhân viên hoặc giảng viên đăng nhập thành công |
| Mục đích cần đạt | Thêm điểm thành công và đưa ra đánh giá |
| Sự kiện chính | * Nhân viên chọn mục “Quản Lý Điểm” từ menu chính * Nhân viên hoặc giảng viên nhập thông tin về các điểm từ các ô nhập thông tin từ màn hình * Sau khi nhập đầy đủ thông tin điểm nhấn nút lưu * Thông tin điểm được hiển thị và điểm trung bình sẽ được tính toán và tự động đưa ra đánh giá |
| Ngoại lệ | 1. Hệ thống báo khi    * Nhân viên hoặc giảng viên nhập sai thông tin, định dạng |

1. *Cập Nhật thông tin điểm*

|  |  |
| --- | --- |
| Kịch bản | Cập Nhật thông tin lịch học |
| Tác nhân | Nhân viên hoặc giảng viên |
| Điều kiện cần | Đăng nhập thành công |
| Mục đích cần đạt | Cập Nhật thành công thông tin điểm |
| Sự kiện chính | * Nhân viên hoặc giảng viên chọn mục “Quản Lý Điểm” trên menu chính * Nhân viên chọn một sinh viên cần cập nhật điểm trên bảng danh sách lịch học. * Chỉnh sửa thông tin điểm. * Nhân viên bấm lưu thông tin * Màn hình hiển thị thông tin điểm, điểm trung bình và đánh giá được được cập nhật. |
| Ngoại lệ | * Thông tin cần sửa không hợp lệ |

1. *Xóa Điểm*

|  |  |
| --- | --- |
| Kịch bản | Xoá Điểm |
| Tác nhân | Nhân viên hoặc giảng viên |
| Điều kiện cần | Nhân viên hoặc giảng viên đăng nhập thành công |
| Mục đích đạt được | Xoá điển của sinh viên khỏi danh sách |
| Sự kiện chính | * Nhân viên hoặc giảng viên chọn mục “Quản Lý Điểm” * Nhân viên hoặc giảng viên chọn mã học viên có điểm cần xóa * Nhân viên có thể sử dụng nút tìm kiếm mã học viên trong khu vực tìm kiếm * Nhân viên hoặc giảng viên chọn nút xoá * Điểm bị xóa khỏi danh sách |
| Ngoại lệ |  |

1. *Tìm kiếm Lịch Học*

|  |  |
| --- | --- |
| Kịch bản | Tìm kiếm điểm |
| Tác nhân | Nhân viên hoặc giảng viên |
| Điều kiện cần | Nhân viên hoặc giảng viên đăng nhập thành công |
| Mục đích đạt được | Tìm kiếm thành công điểm dựa vào mã học viên |
| Sự kiện chính | * Nhân viên chọn mục “Quản Lý Điểm” * Nhân viên nhập thông tin mã học viên vào khu vực tìm kiếm. * Nhân viên ấn nút “Tìm kiếm” và thông tin điểm có mã học viên đúng với thông tin tìm kiếm sẽ hiển thị. * Nhân viên ấn nút “ Hủy Tìm kiếm” và danh điểm sẽ trở lại về trạng thái ban đầu |
| Ngoại lệ | - Thông tin tìm kiếm không chính xác |

1. *Lưu danh sách học viên*

|  |  |
| --- | --- |
| Kịch bản | Lưu danh sách điểm |
| Tác nhân | Nhân viên hoặc giảng viên |
| Điều kiện cần | Nhân viên hoặc giảng viên đăng nhập thành công |
| Mục đích đạt được | Xuất thành công danh sách lịch học |
| Sự kiện chính | * Nhân viên hoặc giảng viên chọn mục “Quản Lý Điểm” * Nhân viên hoặc giảng viên chọn nút Save * Màn hình hiển thị lên cửa sổ địa chỉ mà người dùng muốn lưu file * Danh sách được tải xuống. |
| Ngoại lệ |  |

* + - 1. **Kịch bản Đăng nhập, Đăng ký và Đổi mật khẩu.**

1. *Đăng nhập*

|  |  |
| --- | --- |
| Kịch bản | Đăng nhập vào hệ thống |
| Tác nhân | Nhân viên hoặc giảng viên |
| Điều kiện cần | Nhân viên hoặc giảng viên có tài khoản đã được tạo. |
| Mục đích đạt được | Đăng nhập vào được phần mềm |
| Sự kiện chính | * Nhân viên hoặc giảng viên nhập tên tài khoản và mật khẩu đã tạo * Nhân viên hoặc giảng viên chọn Đăng Nhập * Màn hình Menu được hiển thị hiển thị |
| Ngoại lệ | Hệ thống thông báo không đăng nhập được khi:   * Nhân viên hoặc giảng viên nhập sai tài khoản * Nhân viên hoặc giảng viên không có tài khoản và cần tạo tài khoản. |

1. *Đăng Ký*

|  |  |
| --- | --- |
| Kịch bản | Đăng ký tài khoản |
| Tác nhân | Nhân viên hoặc giảng viên |
| Điều kiện cần |  |
| Mục đích đạt được | Đăng ký thành công tài khoản. |
| Sự kiện chính | * Nhân viên hoặc giảng viên chọn vào mục “Đăng ký” * Nhân viên hoặc giảng viên nhập tên tài khoản và mật khẩu mong muốn * Nhân viên hoặc giảng viên chọn Đăng Ký * Phần mềm thông báo tài khoản đã được tạo. |
| Ngoại lệ | Hệ thống thông báo không tạo được được khi:   * Nhân viên hoặc giảng viên không nhập tên tài khoản và mật khẩu khi Đăng ký. |

1. *Đổi mật khẩu*

|  |  |
| --- | --- |
| Kịch bản | Đổi mật khẩu |
| Tác nhân | Nhân viên hoặc giảng viên |
| Điều kiện cần |  |
| Mục đích đạt được | Thay đổi thành công mật khẩu |
| Sự kiện chính | * Nhân viên hoặc giảng viên nhập tên tài khoản vào màn hình Đăng Nhập * Nhân viên hoặc giảng viên chọn vào mục “Đổi mật khẩu”. * Nhân viên hoặc giảng viên nhập lại mật khẩu cũ * Phần mềm thông báo mật khẩu đã được tạo. |
| Ngoại lệ | Hệ thống thông báo không tạo được được khi:   * Nhân viên hoặc giảng viên không nhập lại mật khẩu mới. |

* 1. **Sơ đồ Use case**

**A diagram of a network

Description automatically generated***Hình 3.1. Sơ đồ Use case phần mềm*

Phần mềm quản lý giáo dục đơn giản và linh hoạt, giúp nhân viên và giáo viên thực hiện công việc một cách hiệu quả. Cả hai đối tượng đều bắt đầu bằng quá trình đăng nhập, sau đó nhân viên có thể quản lý thông tin, lịch học và điểm số, trong khi giáo viên chỉ cần quản lý điểm. Các tính năng đều được thiết kế để tiết kiệm thời gian và dễ sử dụng. Đặc biệt, việc đăng xuất sau khi hoàn thành công việc là bước quan trọng để đảm bảo an toàn thông tin và bảo vệ tính riêng tư của người dùng. Tóm lại, phần mềm này không chỉ đơn giản mà còn mạnh mẽ trong việc giúp quản lý thông tin giáo dục một cách thuận lợi.

* 1. **A diagram of a company

     Description automatically generatedBiểu đồ phân cấp chức năng**

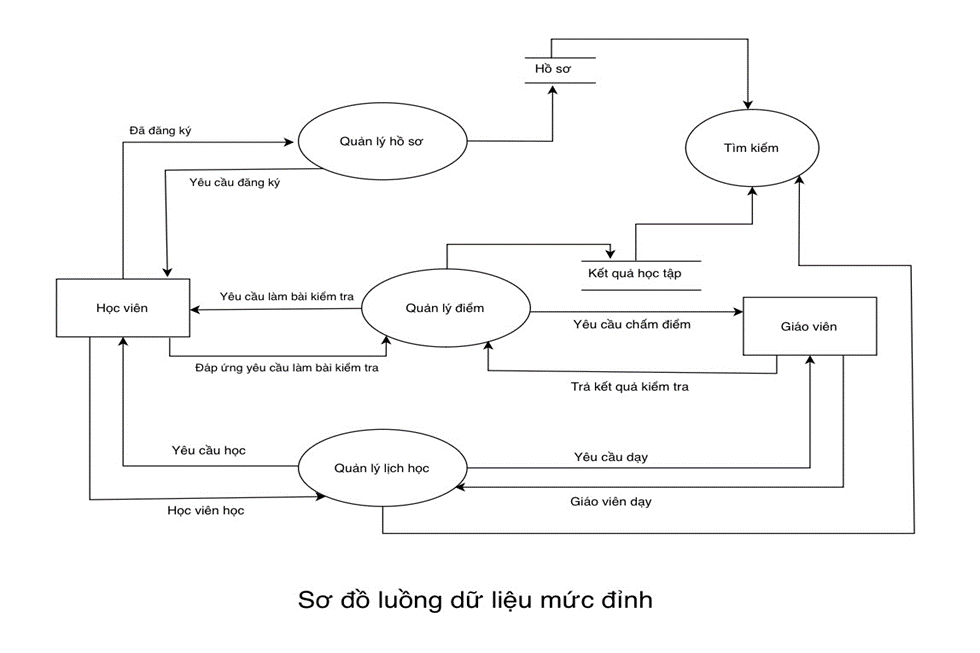
*Hình 3.2. Sơ đồ BPC phần mềm*

* 1. **Biểu đồ luồng dữ liệu mức ngữ cảnh**

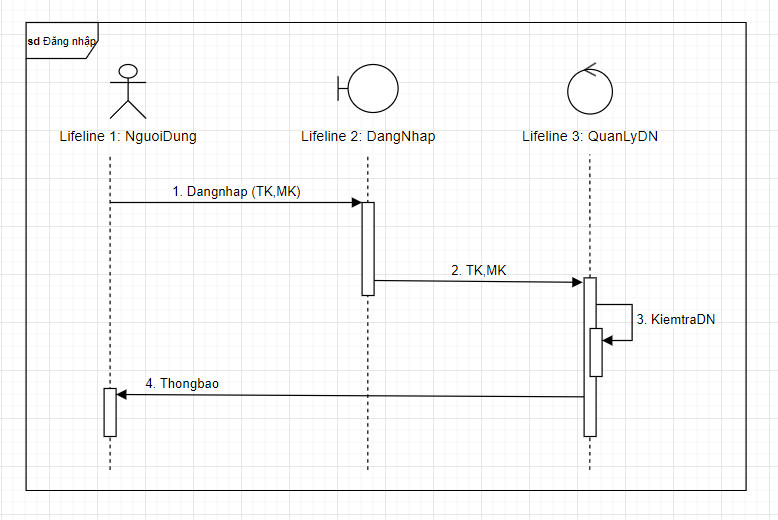
**A diagram of a diagram

Description automatically generated**

*Hình 3.3. Biểu đồ luồng dữ liệu mức ngữ cảnh*

* 1. **Biểu đồ luồng dữ liệu mức đỉnh**

*Hình 3.4. Biểu đồ luồng dữ liệu mức đỉnh*

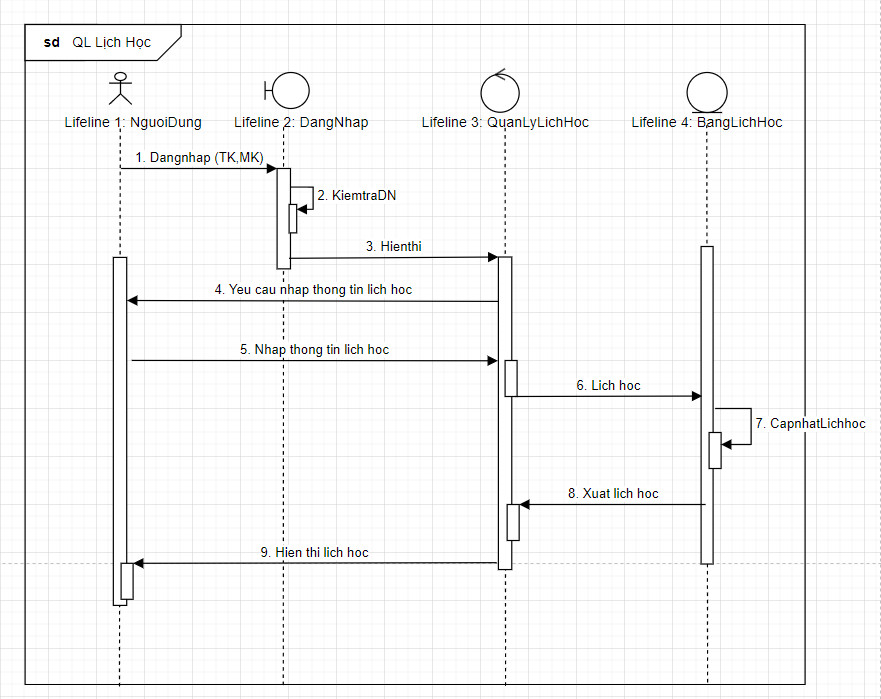
* 1. **Biểu đồ tuần tự**
     1. **Biểu đồ tuần tự đăng nhập**

*Hình 3.5. Biểu đồ tuần tự module đăng nhập*

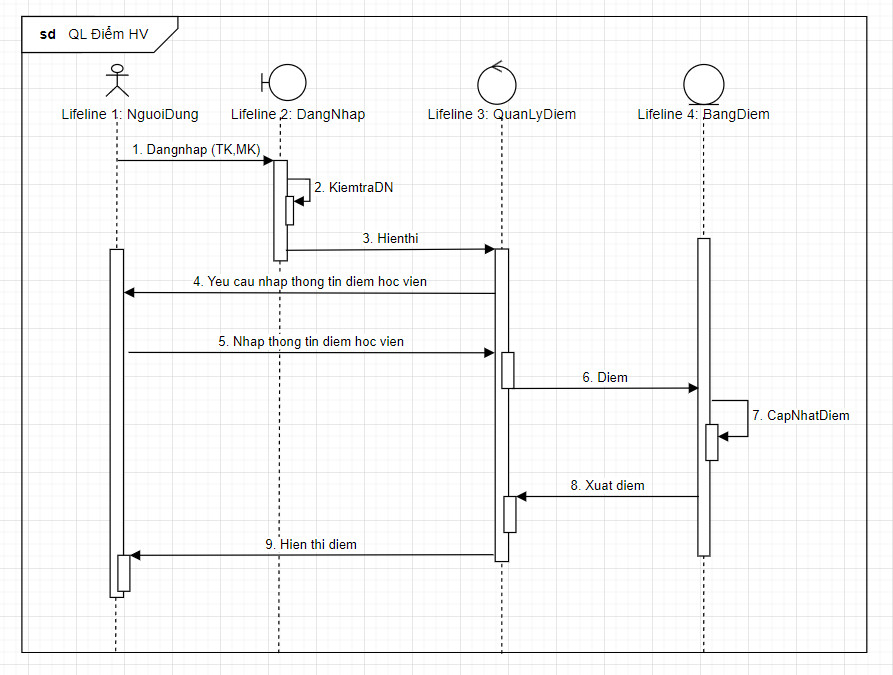
* + 1. **A diagram of a diagram

       Description automatically generatedBiểu đồ tuần tự đăng ký và đổi mật khẩu**

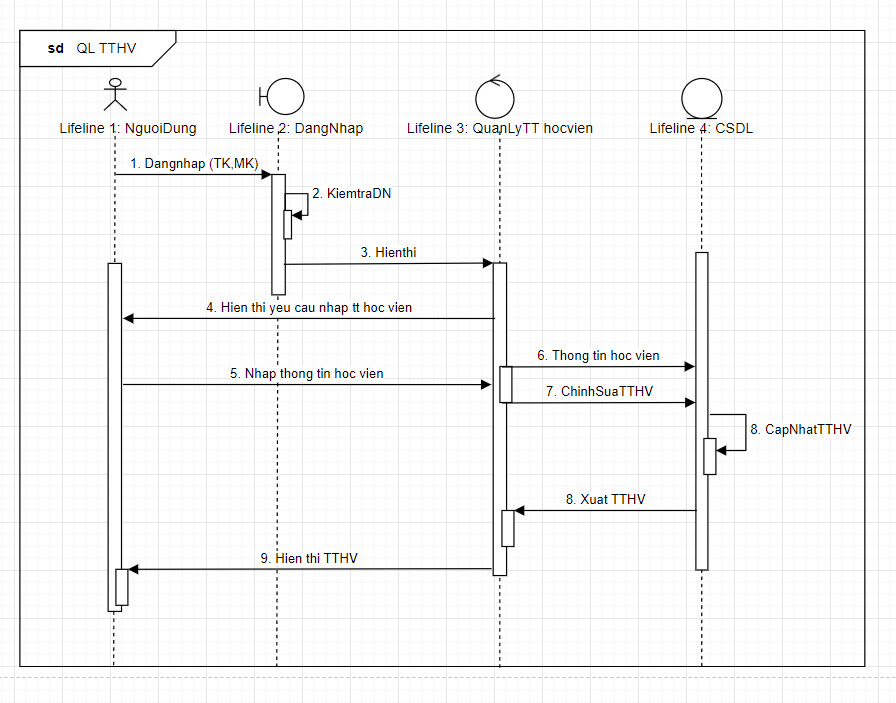
*Hình 3.4. Biểu đồ tuần tự module đăng ký và đổi mật khẩu*

* + 1. **Biểu đồ tuần tự quản lý lịch học**

*Hình 3.4. Biểu đồ tuần tự module quản lý lịch học*

* + 1. **Biểu đồ tuần tự quản lý điểm**

*Hình 3.5. Biểu đồ tuần tự module Quản lý điểm*

* + 1. **Biểu đồ tuần tự quản lý thông tin học viên**

*Hình 3.6. Biểu đồ tuần tự module Quản lý thông tin*

1. **Bảng cơ sở dữ liệu**

Cơ sở dữ liệu là một thành phần không thể thiếu và vô cùng quan trọng trong quá trình quản lý và lưu trữ dữ liệu trong các hệ thống phần mềm hiện đại. Trong khi việc nhập liệu trực tiếp trên phần mềm chỉ tạo ra những bản sao tạm thời và dễ mất mát khi phần mềm tắt đi, cơ sở dữ liệu mang lại tính bền vững cho dữ liệu bằng cách lưu trữ nó trên các thiết bị lưu trữ ổn định và có khả năng tồn tại dài hạn.

Một trong những ưu điểm quan trọng của cơ sở dữ liệu là khả năng quản lý và tổ chức dữ liệu một cách có hệ thống. Dữ liệu được tổ chức vào các bảng, cột và quan hệ, tạo ra một cấu trúc rõ ràng giúp dễ dàng trong việc truy xuất và quản lý thông tin. Điều này không chỉ giúp tối ưu hóa quá trình làm việc mà còn tạo ra một nền tảng linh hoạt cho phát triển và mở rộng hệ thống.

*A screenshot of a computer

Description automatically generatedHình 4.1. Cơ sở dữ liệu được sử dụng trong đề tài*

Cơ sở dữ liệu cũng đảm bảo tính an toàn cho dữ liệu thông qua các cơ chế bảo mật. Quyền truy cập có thể được kiểm soát chặt chẽ, giảm thiểu rủi ro mất mát dữ liệu hoặc truy cập trái phép. Điều này làm tăng tính riêng tư và độ tin cậy của hệ thống.

Ngoài ra, cơ sở dữ liệu còn mang lại các công cụ để tối ưu hóa hiệu suất. Sử dụng các chỉ mục, tối ưu hóa truy vấn và các biện pháp khác giúp đảm bảo rằng hệ thống có thể xử lý dữ liệu lớn một cách hiệu quả và nhanh chóng. Điều này là quan trọng đặc biệt khi xử lý ứng dụng và dự án có yêu cầu về khả năng mở rộng và độ phức tạp cao.

* 1. **Bảng qlhv**

*A screenshot of a computer

Description automatically generated*Khi người dùng chọn "Thông tin Học Viên", ứng dụng sẽ thực hiện truy vấn đến bảng "qlhv" trong cơ sở dữ liệu "testdb" để lấy thông tin chi tiết về học viên. Thông tin này sau đó có thể được hiển thị trên giao diện người dùng để cung cấp thông tin đầy đủ và chi tiết về học viên.

*Hình 4.2. Cơ sở dữ liệu cho bảng qlhv*

Quá trình này giúp tạo ra một trải nghiệm người dùng tốt, cho phép họ dễ dàng truy cập và quản lý thông tin cá nhân. Đồng thời, việc liên kết ứng dụng với cơ sở dữ liệu giúp đảm bảo tính chính xác và đồng bộ của dữ liệu.

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường dữ liệu** | **Kiểu dữ liệu** | **Ý nghĩa** |
| 1 | maKhoaHoc | Varchar | Mã Khóa Học |
| 2 | maHocVien | Int | Mã Học Viên |
| 3 | tenHocVien | Varchar | Tên Học Viên |
| 4 | trinhDo | Varchar | Trình Độ |
| 5 | diaChi | Varchar | Địa Chỉ |
| 6 | ngaySinh | Date | Ngày Sinh |
| 7 | gioiTinh | Varchar | Giới Tính |
| 8 | email | Varchar | Email |
| 9 | lop | Varchar | Lớp |
| 10 | phone | Varchar | Phone |
| 11 | diemKT1 | Float | Điểm Placement Test |

* 1. **Bảng lichhoc**

Khi người dùng chọn "Lịch Học", ứng dụng sẽ thực hiện truy vấn đến bảng "lichhoc" trong cơ sở dữ liệu "testdb" để lấy thông tin chi tiết về lịch học. Thông tin này sau đó có thể được hiển thị trên giao diện người dùng để cung cấp thông tin đầy đủ và chi tiết về lịch học.

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường dữ liệu** | **Kiểu dữ liệu** | **Ý nghĩa** |
| 1 | id | Varchar | ID |
| 2 | tenmonhoc | Varchar | Tên Môn Học |
| 3 | tengiangvien | Varchar | Tên giảng Viên |
| 4 | thoigian | Varchar | Thời gian |
| 5 | diadiem | Varchar | Địa Điểm |
| 6 | Lop | Varchar | Lớp |

A screenshot of a computer

Description automatically generated

*Hình 4.3. Cơ sở dữ liệu cho bảng lichhoc*

* 1. **Bảng user**

Khi người dùng đăng nhập vào phần mềm, ứng dụng sẽ thực hiện truy vấn đến bảng "user" trong cơ sở dữ liệu "testdb" để lấy thông tin chi tiết tài khoản đăng nhập. Thông tin này sau đó có thể được hiển thị trên giao diện người dùng để cung cấp thông tin đầy đủ và chi tiết về lịch học. Trước khi tiến hành đăng nhập, nếu người dùng chưa có tài khoảng thì có thể tiến hành đăng ký tài khoản. Trong trường hợp người dùng muốn đổi mật khẩu, phần mầm có thể hỗ trợ chức năng đó.

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường dữ liệu** | **Kiểu dữ liệu** | **Ý nghĩa** |
| 1 | username | Varchar | Tên tài khoản |
| 2 | password | Varchar | Mật khẩu |

A screenshot of a computer

Description automatically generated

*Hình 4.4. Cơ sở dữ liệu cho bảng user*

* 1. **Bảng bangdiem**

Là một chức năng của phần mềm, khi người dùng chọn "Quản Lý Điểm", ứng dụng sẽ thực hiện truy vấn đến bảng "bangdiem" trong cơ sở dữ liệu "testdb" để lấy thông tin chi tiết về bảng điểm. Thông tin này sau đó có thể được hiển thị trên giao diện người dùng để cung cấp thông tin đầy đủ và chi tiết về bảng điểm của học viên.

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên trường dữ liệu** | **Kiểu dữ liệu** | **Ý nghĩa** |
| 1 | maHocVien | Varchar | Mã Học Viên |
| 2 | Lop | Varchar | Lớp |
| 3 | diemKT1 | Double | Điểm Kiểm Tra 1 |
| 4 | diemKT1 | Double | Điểm Kiểm Tra 2 |
| 5 | diemKT1 | Double | Điểm Kiểm Tra 3 |
| 6 | diemTB | Double | Điểm Trung Bình |
| 7 | Danhgia | Varchar | Đánh giá |

*A screenshot of a computer

Description automatically generatedHình 4.5. Cơ sở dữ liệu cho bảng bangdiem*

* + 1. **Thiết kế chương trình**

1. ***A screenshot of a login screen

   Description automatically generated*Giao diện đăng nhập**

*Hình 5.1. Giao diện đăng nhập*

1. **A screenshot of a computer

   Description automatically generatedGiao diện đăng ký**

*Hình 5.2. Giao diện màn hình đăng ký tài khoản*

1. **A screenshot of a computer

   Description automatically generatedGiao diện thay đổi mật khẩu**

*Hình 5.3. Giao diện màn hình thay đổi mật khẩu*

1. **A screenshot of a menu

   Description automatically generatedGiao diện menu Lựa Chọn**

*Hình 5.4. Giao diện màn hình menu Lựa Chọn*

1. **A screenshot of a computer

   Description automatically generatedGiao diện quản lý lịch học**

*Hình 5.5. Giao diện quản lý lịch học*

1. *A screenshot of a computer

   Description automatically generated***Giao diện quản lý điểm**

*Hình 5.6. Giao diện màn hình quản lý điểm*

1. **A screenshot of a computer

   Description automatically generatedGiao diện quản lý thông tin**

*Hình 5.7. Giao diện màn hình quản lý thông tin học viên*

1. **A screenshot of a computer

   Description automatically generatedGiao diện khi chọn thoát chương trình**

*Hình 5.8. Giao diện màn hình Thoát khỏi chương trình*

* + 1. **Kiểm thử và bảo trì**

**6.1. Test case form Đăng Nhập, Đăng kí và Đổi Mật Khẩu**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Testcase** | **Dữ liệu Test** | **Kết quả mong muốn** | **Kết quả thực tế** |
| **1** | Kiểm tra chức năng mở Login |  | Hiện form đăng nhập | Thành công |
| **2** | Kiểm tra trường hợp không nhập trường nào |  | Thông báo lỗi | Thành công |
| **3** | Kiểm tra trường hợp nhập trường Tên đăng nhập và không nhập trường Mật khẩu |  | Thông báo lỗi | Thành công |
| **4** | Kiểm tra trường hợp nhập trường Mật khẩu mà không nhập trường Tên đăng nhập |  | Thông báo lỗi | Thành công |
| **5** | Kiểm tra trường hợp nhập đúng user và đúng pass |  | Đăng Nhập được vào phần mềm | Thành công |
| **6** | Kiểm tra trường hợp nhập đúng user và sai pass |  | Thông báo lỗi | Thành công |
| **7** | Sai cả user và pass |  | Thông báo đăng nhập lại | Thành công |
| **8** | Kiểm tra trường hợp tạo tài khoản |  | Tài khoản được tạo | Thành công |
| **9** | Kiểm tra trường hợp người dùng thay đổi mật khẩu |  | Mật khẩu mới được cập nhật | Thành công |

**6.2. Test case form Quản Lý Học Viên**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Testcase** | **Dữ liệu Test** | **Kết quả mong muốn** | **Kết quả thực tế** |
| **1** | Kiểm tra chức năng Thêm học viên |  | Các dữ liệu trên màn hình tại jtextField sẽ bị xóa | Thành công |
| **2** | Kiểm tra chức năng Lưu |  | Các dữ liệu sẽ được lưu vào bảng và cơ sở dữ liệu | Thành công |
| **3** | Kiểm tra chức năng Cập Nhật |  | Thông tin học viên mới sẽ được cập nhật thành công | Thành công |
| **4** | Kiểm tra chức năng xóa |  | Thông tin học viên sẽ xóa thành công | Thành công |
| **5** | Kiểm tra chức năng tìm kiếm |  | Dữ liệu sinh viên sau khi tìm kiếm được hiển thị | Thành công |
| **6** | Kiểm tra chức năng Lưu danh sách |  | Danh sách được tải về | Thành công |

**6.3. Test case form Quản Lý Điểm**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Testcase** | **Dữ liệu Test** | **Kết quả mong muốn** | **Kết quả thực tế** |
| **1** | Kiểm tra chức năng Thêm điểm cho học viên |  | Các dữ liệu trên màn hình được lưu vào trong bảng và cơ sở dữ liệu | Thành công |
| **3** | Kiểm tra chức năng Cập Nhật |  | Thông tin điểm mới sẽ được cập nhật thành công | Thành công |
| **4** | Kiểm tra chức năng Xóa |  | Thông tin điểm sẽ xóa thành công | Thành công |
| **5** | Kiểm tra chức năng tìm kiếm |  | Dữ liệu điểm sau khi tìm kiếm theo mã học viên được hiển thị | Thành công |
| **6** | Kiểm tra chức năng Lưu danh sách |  | Danh sách được tải về | Thành công |

**6.4. Test case form Quản Lý Lịch Học**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **STT** | **Testcase** | **Dữ liệu Test** | **Kết quả mong muốn** | **Kết quả thực tế** |
| **1** | Kiểm tra chức năng Thêm học viên |  | Các dữ liệu trên màn hình được lưu vào trong bảng và cơ sở dữ liệu | Thành công |
| **3** | Kiểm tra chức năng Cập Nhật |  | Thông tin lịch học mới sẽ được cập nhật thành công | Thành công |
| **4** | Kiểm tra chức năng Xóa |  | Thông tin lịch học sẽ xóa thành công | Thành công |
| **5** | Kiểm tra chức năng tìm kiếm |  | Dữ liệu lịch học sau khi tìm kiếm được hiển thị | Thành công |
| **6** | Kiểm tra chức năng Lưu danh sách |  | Danh sách được tải về | Thành công |

**6.5. Bảo trì**

Bảo trì phần mềm quản lý học viên là một khía cạnh quan trọng để đảm bảo rằng hệ thống luôn hoạt động ổn định, hiệu quả và đáp ứng được đầy đủ nhu cầu của trung tâm Ngoại Ngữ. Quá trình này không chỉ giúp giữ cho mọi chức năng của phần mềm hoạt động mượt mà mà còn mang lại nhiều lợi ích khác trong quản lý thông tin học viên.

Một trong những nhiệm vụ quan trọng của bảo trì phần mềm là đảm bảo tính ổn định của hệ thống. Các kiểm tra định kỳ và sửa lỗi kỹ thuật giúp phát hiện và khắc phục các vấn đề một cách nhanh chóng, giữ cho phần mềm luôn hoạt động mà không gặp sự gián đoạn không mong muốn.

Ngoài ra, việc cập nhật phần mềm là một phần quan trọng của quá trình bảo trì. Các bản cập nhật thường xuyên không chỉ cung cấp sửa lỗi mà còn mang lại các tính năng mới, cải tiến hiệu suất và đảm bảo tính tương thích với các nền tảng và môi trường mới nhất. Điều này giúp người quản trị hệ thống không chỉ duy trì được ổn định mà còn nâng cao khả năng của phần mềm để đáp ứng những thách thức ngày càng phức tạp.

Bảo trì phần mềm quản lý học viên cũng liên quan chặt chẽ đến bảo mật thông tin. Việc duy trì các biện pháp bảo mật giúp bảo vệ dữ liệu cá nhân của học viên khỏi rủi ro mất mát hoặc truy cập trái phép. Các biện pháp như mã hóa dữ liệu, kiểm soát quyền truy cập và theo dõi hoạt động người dùng đều quan trọng để duy trì sự an toàn và tin cậy của hệ thống.

**PHẦN 3: KẾT LUẬN**

Đề tài đã mở ra một hành trình đầy thách thức và học hỏi cho cả nhóm về quá trình xây dựng và triển khai một ứng dụng phần mềm đơn giản. Bằng cách sử dụng JavaSwing để tạo giao diện người dùng và kết nối với cơ sở dữ liệu thông qua JDBC, nhóm đã đạt được những tiến bộ đáng kể trong việc tạo ra một ứng dụng có khả năng tương tác cao và khả năng lưu trữ dữ liệu một cách có tổ chức.

Tuy nhiên, những thành công này cũng đi kèm với những thách thức và cơ hội để cải thiện. Mặc dù JavaSwing là một công nghệ giao diện người dùng mạnh mẽ, nhưng có thể cần xem xét các lựa chọn hiện đại hơn để tạo ra một giao diện thân thiện và hiện đại hơn, ví dụ như sử dụng JavaFX hoặc các framework phát triển web.

Một điểm quan trọng cần xem xét là bảo mật. Đảm bảo rằng phần mềm được thiết kế với các biện pháp bảo mật mạnh mẽ để ngăn chặn các mối đe dọa tiềm ẩn và đảm bảo an toàn cho dữ liệu của người dùng. Cũng quan trọng là kiểm tra và tối ưu hóa hiệu suất để đảm bảo rằng ứng dụng có thể đáp ứng hiệu quả với nhiều điều kiện sử dụng khác nhau.

Vấn đề khác cần xem xét là khả năng bảo trì và mở rộng của phần mềm. Thiết kế hệ thống sao cho nó có thể dễ dàng bảo trì và mở rộng trong tương lai, đặc biệt là khi xuất hiện yêu cầu mới và thay đổi. Việc viết tài liệu đầy đủ và chi tiết cũng là quan trọng để hỗ trợ cả nhóm phát triển và người sử dụng.

Bằng cách này, đề tài không chỉ là một sản phẩm tạm thời mà còn là một nền tảng cho sự phát triển và mở rộng trong tương lai. Qua những thất bại và thành công, nhóm đã có cơ hội để đúc kết kinh nghiệm, nâng cao kỹ năng, và xây dựng cơ sở cho những dự án phần mềm phức tạp hơn trong tương lai.

**TÀI LIỆU THAM KHẢO**

1. Giáo trình lý thuyết và bài tập Java – chủ biên Trần Tiến Dũng
2. Java Core – [www.updatesofts.com](http://www.updatesofts.com)
3. ThinkJava – Allen B.Downey & Chris Mayfield
4. Lập trình Java – Phạm Quang Dũng
5. <https://topdev.vn/blog/lap-trinh-huong-doi-tuong-oops-trong-java/>
6. https://luanvan1080.com/huong-dan-chi-tiet-cach-trinh-bay-tieu-luan-chuan-form.html