**TRƯỜNG CAO ĐẲNG BÁCH KHOA TÂY NGUYÊN**

**KHOA CƠ BẢN**



**BÁO CÁO**

**KẾT THÚC HỌC PHẦN MÔN HỌC**

**MÔN:LẬP TRÌNH WEB VỚI PHP&SQL**

**Chuyên ngành: Công Nghệ Thông Tin**

**Khóa học: 2023-2024**

**Đắk Lắk, ngày 20 tháng 7 năm 2024**

**TRƯỜNG CAO ĐẲNG BÁCH KHOA TÂY NGUYÊN**

**KHOA CƠ BẢN**

******

**BÁO CÁO**

**KẾT THÚC HỌC PHẦN MÔN HỌC**

**MÔN: LẬP TRÌNH WEB VỚI PHP&SQL**

**Người hướng dẫn: Vàng Văn Tính**

**Đề tài: THIẾT KẾ TRANG ĐĂNG KÝ MÔN HỌC**

**Sinh viên thực hiện: TRẦN MINH HUY /MSSV:222013030**

**LÊ VĂN TIẾN /MSSV:222013022**

**Y ĐANIT ADRƠNG /MSSV:222013037**

**Đắk Lắk, ngày 20 tháng 07 năm 2024**

**Lời cảm ơn**

Em xin gửi lời cảm ơn chân thành nhất tới Thầy về sự hướng dẫn và hỗ trợ của Thầy trong quá trình làm báo cáo này.

Thầy đã là nguồn động viên quý báu, luôn sẵn sàng chia sẻ kiến thức và kinh nghiệm giúp em hiểu rõ hơn về chủ đề và cách triển khai bài báo cáo một cách hiệu quả.

Sự tận tâm và nhiệt huyết của Thầy không chỉ giúp em nắm bắt nội dung mà còn khơi gợi niềm đam mê và sự ham học hỏi. Thầy đã tạo ra một môi trường học tốt, khuyến khích sự sáng tạo và đồng thời giúp em phát triển kỹ năng tự học.

Em rất biết ơn về sự hỗ trợ chân thành và kiên nhẫn của Thầy trong suốt quá trình nghiên cứu và viết báo cáo. Những góp ý và phản hồi của Thầy đã giúp em hiểu rõ hơn về chất lượng công việc và cách để hoàn thiện nó.

Chân thành cảm ơn Thầy về mọi điều. Sự giúp đỡ của Thầy không chỉ là một phần quan trọng của bài báo cáo này mà còn là một phần quan trọng của hành trình học tập của em.

Em trân trọng.

Mục lục

[1. Giới thiệu 1](#_Toc172341237)

[**1.1. Mục đích của báo cáo** 1](#_Toc172341238)

[**1.2. Đối tượng sử dụng** 1](#_Toc172341239)

[**1.3. Phạm vi** 1](#_Toc172341240)

[2. Phân tích và thiết kế website 1](#_Toc172341241)

[**2.1. Phân tích yêu cầu** 1](#_Toc172341242)

[**2.1.1. Yêu cầu chức năng** 1](#_Toc172341243)

[**2.1.2. Yêu cầu phi chức năng** 1](#_Toc172341244)

[**2.2. Thiết kế hệ thống** 2](#_Toc172341245)

[**2.2.1. Kiến trúc hệ thống** 2](#_Toc172341246)

[**2.2.2. Thiết kế cơ sở dữ liệu** 2](#_Toc172341247)

[**2.2.3. Thiết kế giao diện người dùng** 2](#_Toc172341248)

[**2.2.4. Tổng thể Database** 3](#_Toc172341249)

[3. Phân tích và thiết kế 4](#_Toc172341250)

[**3.1. register.php** 4](#_Toc172341251)

[**3.2. admin.php** 6](#_Toc172341252)

[**3.4. view\_grades.php** 8](#_Toc172341253)

[4. Kết luận 9](#_Toc172341254)

[**4.1. Tính năng và Tính hiệu quả** 10](#_Toc172341255)

[**4.2. Đánh giá Hệ thống** 10](#_Toc172341256)

[**4.3. Đề xuất Cải tiến** 10](#_Toc172341257)

# 1. Giới thiệu

**1.1. Mục đích của báo cáo**

Báo cáo này nhằm mục đích cung cấp cái nhìn tổng quan về dự án phát triển một hệ thống quản lý học tập trực tuyến. Hệ thống này bao gồm các chức năng đăng ký học lại, xem bảng điểm, và quản lý người dùng.

**1.2. Đối tượng sử dụng**

Hệ thống được thiết kế cho sinh viên, giảng viên, và quản trị viên của trường đại học. Sinh viên có thể đăng ký học lại các môn học và xem bảng điểm của mình. Giảng viên và quản trị viên có thể quản lý việc đăng ký học lại và xem bảng điểm của tất cả sinh viên.

**1.3. Phạm vi**

Báo cáo này sẽ đi qua quá trình phân tích và thiết kế hệ thống, cũng như các bước xây dựng và triển khai hệ thống.

# 2. Phân tích và thiết kế website

**2.1. Phân tích yêu cầu**

**2.1.1. Yêu cầu chức năng**

**Đăng ký học lại**: Sinh viên có thể chọn môn học để đăng ký học lại nếu họ đã có điểm không đạt ở môn đó.

**Xem bảng điểm**: Sinh viên có thể xem điểm của mình theo các môn học.

**Quản lý đăng ký học lại**: Giảng viên hoặc quản trị viên có thể duyệt hoặc xóa các yêu cầu đăng ký học lại.

**Quản lý người dùng**: Quản trị viên có thể xem và quản lý thông tin của người dùng.

**2.1.2. Yêu cầu phi chức năng**

**Bảo mật**: Hệ thống cần đảm bảo tính bảo mật thông tin người dùng, đặc biệt là mật khẩu và dữ liệu cá nhân.

**Hiệu suất**: Hệ thống cần có khả năng xử lý nhiều yêu cầu đồng thời mà không làm giảm hiệu suất.

**Khả năng mở rộng**: Hệ thống nên có khả năng mở rộng để hỗ trợ nhiều người dùng và chức năng mới trong tương lai.

**2.2. Thiết kế hệ thống**

**2.2.1. Kiến trúc hệ thống**

Hệ thống được thiết kế theo mô hình 3 lớp gồm:

* **Lớp giao diện người dùng**: Các trang web mà người dùng tương tác, bao gồm HTML, CSS, và JavaScript.
* **Lớp logic ứng dụng** : Xử lý các yêu cầu từ người dùng, thực hiện các phép toán và xử lý dữ liệu, được xây dựng bằng PHP.
* **Lớp cơ sở dữ liệu (Database)**: Lưu trữ dữ liệu của hệ thống, sử dụng MySQL.

**2.2.2. Thiết kế cơ sở dữ liệu**

Cơ sở dữ liệu bao gồm các bảng chính:

* **users**: Lưu trữ thông tin người dùng.
* **students**: Lưu trữ thông tin sinh viên.
* **subjects**: Lưu trữ thông tin các môn học.
* **grades**: Lưu trữ điểm của sinh viên.
* **registrations**: Lưu trữ thông tin đăng ký học lại của sinh viên.

**2.2.3. Thiết kế giao diện người dùng**

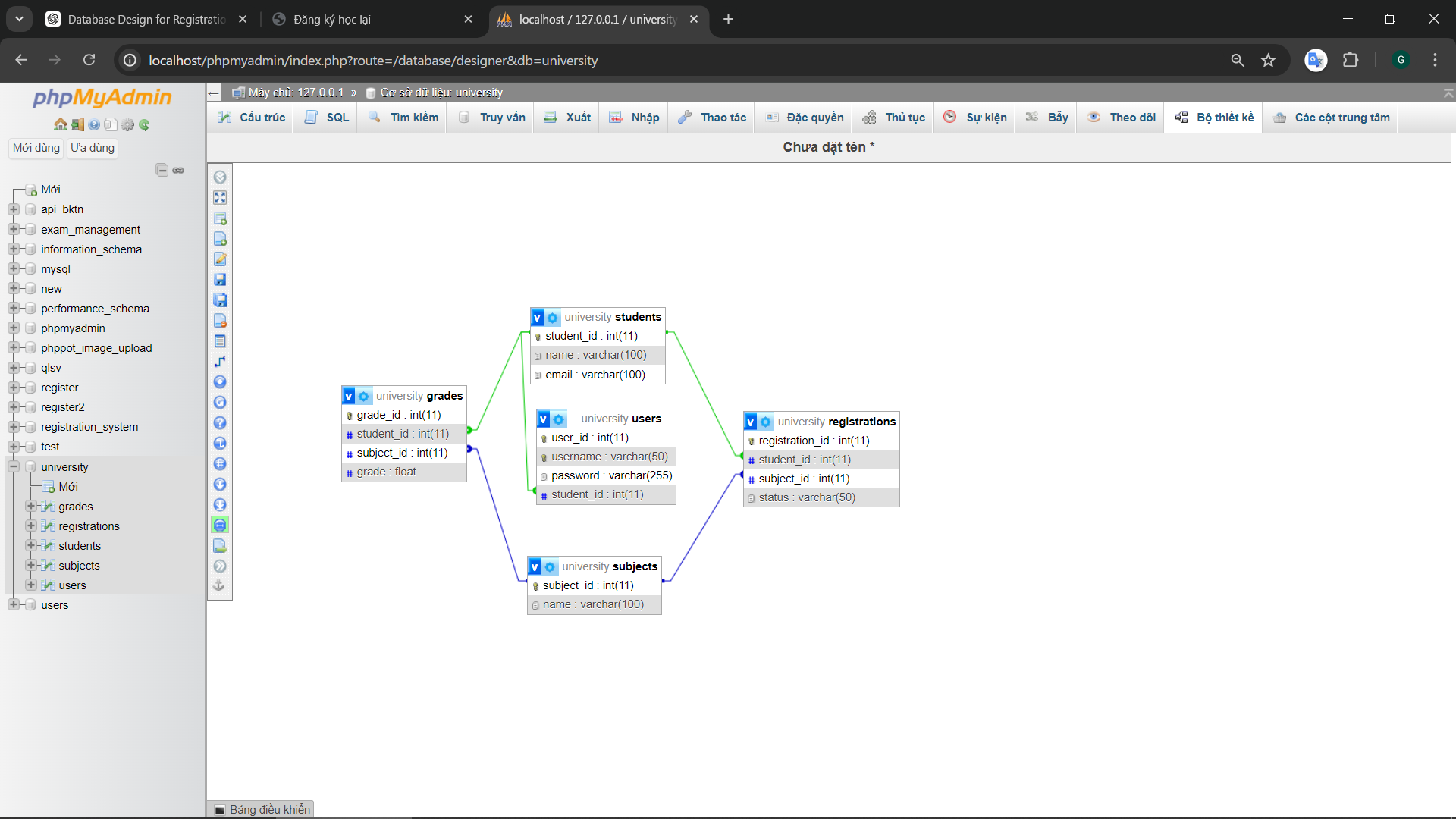
**Trang đăng nhập**: Cho phép người dùng đăng nhập vào hệ thống.

**Trang đăng ký học lại**: Cho phép sinh viên đăng ký học lại môn học.

**Trang bảng điểm**: Hiển thị điểm của sinh viên.

**Trang quản lý đăng ký học lại**: Cho phép quản trị viên duyệt và xóa các yêu cầu đăng ký học lại.

**2.2.4. Tổng thể Database**



Hình 1. 1

# 3. Phân tích và thiết kế

**3.1. register.php**

**Mục đích:** Cho phép sinh viên đăng ký học lại môn học nếu cần thiết.

**Chức năng:**

* **Xác thực sinh viên:** Kiểm tra xem sinh viên đã đăng nhập hay chưa. Nếu chưa, chuyển hướng đến trang đăng nhập.
* **Hiển thị form đăng ký:** Hiển thị form cho phép sinh viên chọn môn học để đăng ký học lại.
* **Xử lý đăng ký:** Khi form được gửi đi, kiểm tra điểm hiện tại của sinh viên. Nếu điểm của môn học cần đăng ký dưới 5, sinh viên có thể đăng ký lại. Nếu đã đăng ký rồi hoặc điểm không cần học lại, hiển thị thông báo phù hợp.

**Cấu trúc:**

Logic php: <?php

session\_start();

require\_once 'database.php';

if (!isset($\_SESSION['student\_id'])) {

    header('Location: login.php');

    exit();

}

$db = new Database();

$conn = $db->getConnection();

$message = ''; // Biến để lưu thông báo

$student\_id = $\_SESSION['student\_id'];

$student\_name = $\_SESSION['student\_name']; // Lấy tên sinh viên từ session

if ($\_SERVER['REQUEST\_METHOD'] === 'POST') {

    $subject\_id = $\_POST['subject\_id'];

    // Kiểm tra xem sinh viên đã có điểm không đạt ở môn học này chưa

    $grade\_result = $db->select("SELECT grade FROM grades WHERE student\_id = ? AND subject\_id = ?", [$student\_id, $subject\_id]);

    if ($grade\_result && $grade\_result[0]['grade'] < 5.0) {

        // Kiểm tra xem đã đăng ký môn này chưa

        $registration\_result = $db->select("SELECT \* FROM registrations WHERE student\_id = ? AND subject\_id = ?", [$student\_id, $subject\_id]);

        if (empty($registration\_result)) {

            // Thêm đăng ký học lại vào cơ sở dữ liệu

            $db->insert("INSERT INTO registrations (student\_id, subject\_id) VALUES (?, ?)", [$student\_id, $subject\_id]);

            $message = "Đăng ký học lại thành công.";

        } else {

            $message = "Bạn đã đăng ký môn học này rồi.";

        }

    } else {

        $message = "Sinh viên không cần đăng ký học lại môn này.";

    }

}

$db->closeConnection();

?>

Form:  
<body>

    <!-- Navbar -->

    <nav class="navbar navbar-expand-lg navbar-custom">

        <div class="container">

            <a class="navbar-brand text-white" href="#">Hệ thống đăng ký</a>

            <div class="collapse navbar-collapse">

                <ul class="navbar-nav ml-auto">

                    <li class="nav-item">

                        <a class="nav-link text-white" href="logout.php">Đăng xuất</a>

                    </li>

                </ul>

            </div>

        </div>

    </nav>

    <div class="container mt-4 form-container">

        <h2 class="text-center">Đăng ký học lại</h2>

        <div class="alert alert-primary text-center" role="alert">

            Xin chào, <?php echo htmlspecialchars($student\_name); ?>!

        </div>

        <?php if (!empty($message)): ?>

            <div class="alert alert-info" role="alert">

                <?php echo $message; ?>

            </div>

        <?php endif; ?>

        <form action="register.php" method="POST">

            <div class="form-group">

                <label for="subject\_id">Chọn môn học:</label>

                <select class="form-control" name="subject\_id" id="subject\_id">

                    <!-- Dữ liệu môn học được tải từ cơ sở dữ liệu -->

                    <?php

                    $db = new Database();

                    $conn = $db->getConnection();

                    $subjects = $db->select("SELECT \* FROM subjects");

                    foreach ($subjects as $subject) {

                        echo "<option value='{$subject['subject\_id']}'>{$subject['name']}</option>";

                    }

                    $db->closeConnection();

                    ?>

                </select>

            </div>

            <button type="submit" class="btn btn-primary btn-block">Đăng ký học lại</button>

        </form>

    </div>

## **3.2. admin.php**

**Mục đích:** Quản lý và duyệt các yêu cầu đăng ký học lại của sinh viên.

**Chức năng:**

**Hiển thị danh sách đăng ký:** Hiển thị tất cả các yêu cầu đăng ký học lại của sinh viên.

**Duyệt hoặc xóa đăng ký:** Cho phép admin duyệt hoặc xóa các yêu cầu đăng ký.

**Cấu trúc:**  <div class="container table-container">

        <h2 class="text-center">Quản lý đăng ký học lại</h2>

        <table class="table table-striped">

            <thead>

                <tr>

                    <th>Sinh viên</th>

                    <th>Môn học</th>

                    <th>Trạng thái</th>

                    <th>Hành động</th>

                </tr>

            </thead>

            <tbody>

                <?php

                require\_once 'database.php';

                $db = new Database();

                $conn = $db->getConnection();

                $result = $db->select("SELECT r.registration\_id, s.name AS student\_name, subj.name AS subject\_name, r.status

                                       FROM registrations r

                                       JOIN students s ON r.student\_id = s.student\_id

                                       JOIN subjects subj ON r.subject\_id = subj.subject\_id");

                foreach ($result as $row) {

                    echo "<tr>";

                    echo "<td>" . htmlspecialchars($row['student\_name']) . "</td>";

                    echo "<td>" . htmlspecialchars($row['subject\_name']) . "</td>";

                    echo "<td>" . htmlspecialchars($row['status']) . "</td>";

                    if ($row['status'] == 'Pending') {

                        echo "<td>

                                <a href='action.php?id={$row['registration\_id']}&action=approve' class='btn btn-sm btn-success'>Duyệt</a>

                                <a href='action.php?id={$row['registration\_id']}&action=delete' class='btn btn-sm btn-danger'>Xóa</a>

                              </td>";

                    } else {

                        echo "<td>Không có hành động</td>";

                    }

                    echo "</tr>";

                }

                $db->closeConnection();

                ?>

            </tbody>

        </table>

    </div>

**3.3. action.php**

**Mục đích:** Xử lý các hành động liên quan đến các yêu cầu đăng ký học lại.

**Chức năng:**

**Duyệt hoặc xóa đăng ký:** Xử lý hành động duyệt hoặc xóa một yêu cầu đăng ký học lại dựa trên tham số action

**Cấu trúc:**

<?php

require\_once 'database.php';

if (isset($\_GET['id']) && isset($\_GET['action'])) {

    $registration\_id = $\_GET['id'];

    $action = $\_GET['action'];

    $db = new Database();

    $conn = $db->getConnection();

    if ($action == 'approve') {

        $db->update("UPDATE registrations SET status = 'Approved' WHERE registration\_id = ?", [$registration\_id]);

    } elseif ($action == 'delete') {

        $db->update("DELETE FROM registrations WHERE registration\_id = ?", [$registration\_id]);

    }

    $db->closeConnection();

}

header('Location: admin.php');

exit();

?>

**3.4. view\_grades.php**

**Mục đích:** Hiển thị bảng điểm của sinh viên.

**Chức năng:**

**Hiển thị bảng điểm:** Truy xuất và hiển thị điểm của các môn học theo sinh viên.

**Cấu trúc:** <div class="container table-container">

        <h2 class="text-center">Bảng điểm sinh viên</h2>

        <table class="table table-striped">

            <thead>

                <tr>

                    <th>Sinh viên</th>

                    <th>C++</th>

                    <th>C#</th>

                    <th>Cơ sở dữ liệu</th>

                    <th>PHP</th>

                    <th>Java</th>

                </tr>

            </thead>

            <tbody>

                <?php

                $db = new Database();

                $conn = $db->getConnection();

                $query = "

                    SELECT s.name AS student\_name,

                           MAX(CASE WHEN subj.name = 'C++' THEN g.grade END) AS cpp,

                           MAX(CASE WHEN subj.name = 'C#' THEN g.grade END) AS cs,

                           MAX(CASE WHEN subj.name = 'Cơ sở dữ liệu' THEN g.grade END) AS db\_course,

                           MAX(CASE WHEN subj.name = 'PHP' THEN g.grade END) AS php,

                           MAX(CASE WHEN subj.name = 'Java' THEN g.grade END) AS java

                    FROM grades g

                    JOIN students s ON g.student\_id = s.student\_id

                    JOIN subjects subj ON g.subject\_id = subj.subject\_id

                    GROUP BY s.student\_id

                    ORDER BY s.student\_id;

                ";

                $result = $db->select($query);

                foreach ($result as $row) {

                    echo "<tr>";

                    echo "<td>" . htmlspecialchars($row['student\_name']) . "</td>";

                    // Hàm inline kiểm tra điểm và gán class CSS tương ứng

                    $displayGrade = function ($grade) {

                        if ($grade === null) return "<td>-</td>";

                        $class = $grade < 5 ? 'grade-fail' : 'grade-pass';

                        return "<td class='$class'>" . htmlspecialchars($grade) . "</td>";

                    };

                    echo $displayGrade($row['cpp']);

                    echo $displayGrade($row['cs']);

                    echo $displayGrade($row['db\_course']);

                    echo $displayGrade($row['php']);

                    echo $displayGrade($row['java']);

                    echo "</tr>";

                }

                $db->closeConnection();

                ?>

            </tbody>

        </table>

    </div>

# 4. Kết luận

Hệ thống quản lý đăng ký học lại đã được thiết kế và triển khai nhằm đáp ứng nhu cầu quan trọng của sinh viên trong việc quản lý môn học và điểm số của mình. Hệ thống này bao gồm các chức năng chính như đăng ký học lại, duyệt các yêu cầu đăng ký, xem bảng điểm, và quản lý người dùng.

**4.1. Tính năng và Tính hiệu quả**

**Đăng ký học lại:** Sinh viên có thể dễ dàng đăng ký học lại các môn học có điểm dưới mức yêu cầu. Hệ thống kiểm tra điểm và cho phép hoặc từ chối đăng ký dựa trên tiêu chí cụ thể.

**Quản lý yêu cầu:** Các yêu cầu đăng ký học lại được quản lý bởi admin, người có quyền duyệt hoặc xóa yêu cầu dựa trên trạng thái của nó. Điều này đảm bảo rằng chỉ những yêu cầu hợp lệ và cần thiết được phê duyệt.

**Xem bảng điểm:** Sinh viên có thể tra cứu bảng điểm của mình để theo dõi tiến trình học tập. Điều này giúp sinh viên xác định các môn học cần cải thiện và lên kế hoạch cho các hành động tiếp theo.

**4.2. Đánh giá Hệ thống**

Hệ thống đã được phát triển với giao diện người dùng thân thiện, sử dụng công nghệ web hiện đại như HTML, CSS, và PHP. Việc sử dụng Bootstrap giúp tạo ra giao diện dễ nhìn và dễ sử dụng. Các chức năng quan trọng đều được kiểm tra kỹ lưỡng để đảm bảo tính chính xác và độ tin cậy cao.

**4.3. Đề xuất Cải tiến**

Mặc dù hệ thống hiện tại đáp ứng được các nhu cầu cơ bản, vẫn có thể cải tiến thêm như:

* **Bảo mật:** Cần nâng cao bảo mật, chẳng hạn như sử dụng các phương pháp mã hóa mạnh mẽ hơn thay vì MD5 cho mật khẩu.
* **Tính năng báo cáo:** Thêm các chức năng báo cáo chi tiết cho admin về số lượng yêu cầu đăng ký, trạng thái xử lý, và thống kê học sinh.
* **Giao diện người dùng:** Tiếp tục cải thiện giao diện để đáp ứng các yêu cầu trải nghiệm người dùng ngày càng cao, bao gồm các tính năng tương tác động và hiển thị phản hồi tức thì.

**Kết luận**

Tóm lại, hệ thống quản lý đăng ký học lại đã cung cấp một nền tảng hữu ích cho sinh viên và quản lý học tập. Hệ thống này không chỉ giúp sinh viên dễ dàng quản lý việc học lại mà còn cho phép các quản trị viên thực hiện quản lý và duyệt yêu cầu một cách hiệu quả. Những cải tiến và nâng cấp trong tương lai sẽ giúp nâng cao tính năng và hiệu suất của hệ thống, đáp ứng tốt hơn nhu cầu của người dùng.

**Ý kiến của người hướng dẫn:**

***Nhận xét***:

***Đối với khóa luận tốt nghiệp*** (Đánh dấu x và ký tên vào ý kiến chọn lựa sau):

|  |  |  |
| --- | --- | --- |
|  |  | Ký tên |
| Đồng ý thông qua báo cáo | ¨ |  |
| Không đồng ý thông qua báo cáo | ¨ |  |

|  |  |
| --- | --- |
|  | ....................., *ngày ... tháng .. năm.............*  **NGƯỜI HƯỚNG DẪN**  (*Ký và ghi rõ họ tên*) |