**TRƯỜNG ĐẠI HỌC PHENIKAA**

**TRƯỜNG CÔNG NGHỆ THÔNG TIN PHENIKAA**

A logo for a university

AI-generated content may be incorrect.

**BÁO CÁO BÀI TẬP LỚN**

HỌC PHẦN: PHÂN TÍCH THIẾT KẾ PHẦN MỀM

**ĐỀ TÀI: XÂY DỰNG ỨNG DỤNG WEBSITE TÍNH TIỀN LƯƠNG GIÁO VIÊN**

**Nhóm: 2**

|  |  |  |
| --- | --- | --- |
| **Lê Hải Đăng** | **22010260** | **22010260@st.phenikaa-uni.edu.vn** |
| **Nguyễn Trọng Hùng** | **23010083** | **23010083@st.phenikaa-uni.edu.vn** |
| **Nguyễn Quốc Thành** | **23010038** | **23010038@st.phenikaa-uni.edu.vn** |
| **Lê Nam Anh** | **21010628** | **21010628@st.phenikaa-uni.edu.vn** |

**Hà Nội, 9/2025**

**Họ Và Tên : Lê Hải Đăng**

**Mã sinh viên : 22010260**

**Công việc đã làm : Thiết kế database ; Code BE ,FE ; Kiểm thử ; vẽ sơ đồ ; thiết kế UI**

**Bảng phân chia công việc**

|  |  |
| --- | --- |
| **Lê Hải Đăng** | **25%** |
| **Nguyễn Trọng Hùng** | **25%** |
| **Nguyễn Quốc Thành** | **25%** |
| **Lê Nam Anh** | **25%** |

Mục lục

[1. Xác định yêu cầu](#_Toc201688515)

[1.1. Đặt vấn đề](#_Toc201688516)

[1.2. Yêu cầu chung](#_Toc201688517)

[1.3. Yêu cầu cụ thể](#_Toc201688518)

[2. Các yêu cầu chức năng](#_Toc201688519)

[2.1. Tác nhân hệ thống](#_Toc201688520)

[2.2. Các chức năng hệ thống](#_Toc201688521)

[2.3. Sơ đồ Usecase tổng quát](#_Toc201688522)

[2.4. Đặc tả Usecase](#_Toc201688523)

3.Các biểu đồ....................................................................................................

3.1 : Biểu đồ hoạt động................................................................................

3.1.1: Đăng nhập..............................................................................

3.1.2: Thêm giảng viên.....................................................................

3.1.3: Thêm môn học........................................................................

3.1.4: Thêm học kỳ...........................................................................

3.1.5: Lập lịch dạy............................................................................

3.1.6: Tính lương..............................................................................

3.2: Biểu đồ gói............................................................................................

3.3: Biểu đồ hoạt động.................................................................................

[4. Các yêu cầu phi chức năng](#_Toc201688524)

[4.1. Giao diện người dùng](#_Toc201688525)

[4.2. Hiệu suất](#_Toc201688526)

5. Thiết kế cơ sở dữ liệu và cài đặt...................................................................

[6. Giao diện hệ thống.............................................................................](#_Toc201688527)........

7. Kế hoạch kiểm thử........................................................................................

8. Xây dựng Testcase........................................................................................

**LỜI GIỚI THIỆU**

Kiểm thử và đánh giá chất lượng phần mềm là những giai đoạn quan trọng, mang tính quyết định trong vòng đời phát triển phần mềm. Đặc biệt, trong bối cảnh các hệ thống ngày càng trở nên phức tạp và đóng vai trò thiết yếu trong hoạt động của doanh nghiệp, tổ chức và xã hội, quá trình này không chỉ dừng lại ở việc kiểm tra chức năng mà còn đảm bảo phần mềm đáp ứng đầy đủ các yêu cầu về độ tin cậy, hiệu suất và trải nghiệm người dùng. Hơn nữa, việc phát hiện và khắc phục sớm các lỗi tiềm ẩn giúp nâng cao chất lượng tổng thể của sản phẩm, giảm thiểu rủi ro trong quá trình vận hành.

Trong kỷ nguyên chuyển đổi số, phạm vi kiểm thử phần mềm đã được mở rộng, không chỉ tập trung vào tính đúng đắn của chức năng mà còn bao gồm nhiều tiêu chí khác như: Tính khả dụng (khả năng dễ sử dụng, giao diện thân thiện), hiệu suất (khả năng xử lý ổn định dưới tải cao); khả năng mở rộng (dễ dàng nâng cấp khi cần thiết).Điều này đòi hỏi một quy trình kiểm thử bài bản, kết hợp đa dạng phương pháp và công cụ để đảm bảo phần mềm hoạt động tối ưu trong mọi điều kiện.

Xuất phát từ nhu cầu thực tiễn, nhóm nghiên cứu lựa chọn đề tài "*Đánh giá chất lượng phần mềm tính lương giảng viên*", nhằm giải quyết bài toán quản lý thu nhập trong môi trường giáo dục – nơi thường xuyên phải xử lý các quy định thuế phức tạp và tính toán lương đa dạng. Nghiên cứu này không chỉ mang tính ứng dụng cao mà còn hướng đến xây dựng một giải pháp tin cậy, chính xác và minh bạch, đem lại lợi ích thiết thực cho cả nhà quản lý và giảng viên.

Mục tiêu của nghiên cứu là áp dụng các phương pháp kiểm thử hiện đại để đánh giá toàn diện chất lượng phần mềm, từ đó đề xuất cải tiến nhằm nâng cao hiệu quả hệ thống. Kết quả mong đợi sẽ góp phần hoàn thiện quy trình phát triển phần mềm, đảm bảo sản phẩm cuối cùng đáp ứng tốt nhất nhu cầu người dùng.

1. **Xác định yêu cầu**
   1. Đặt vấn đề

Trong quản lý giáo dục đại học, việc tính toán và thanh toán lương cho giảng viên là một nhiệm vụ quan trọng nhưng phức tạp, đòi hỏi độ chính xác cao do liên quan đến nhiều yếu tố như: số giờ giảng dạy, loại môn học, bằng cấp, phụ cấp chức vụ, và các quy định về thuế thu nhập cá nhân. Hiện nay, nhiều trường đại học vẫn gặp khó khăn trong quá trình này do:

* Khối lượng dữ liệu lớn: Việc quản lý thông tin giảng dạy, điểm danh và tính lương cho hàng trăm giảng viên đòi hỏi hệ thống xử lý nhanh và chính xác.
* Nguy cơ sai sót: Tính toán thủ công hoặc sử dụng phần mềm không tối ưu dễ dẫn đến sai lệch trong thanh toán lương, ảnh hưởng đến quyền lợi giảng viên.
* Tốn thời gian và nhân lực: Quy trình tổng hợp dữ liệu từ lịch dạy, điểm danh đến tính lương thường kéo dài, gây chậm trễ trong thanh toán.
* Thiếu minh bạch: Giảng viên khó theo dõi chi tiết các khoản thu nhập, dẫn đến thiếu tin tưởng vào hệ thống quản lý.

Trước những thách thức trên, việc ứng dụng phần mềm tính lương giảng viên với đầy đủ các chức năng quản lý thông tin giảng viên, lịch dạy và điểm danh, tính lương và thanh toán, báo cáo thống kê là giải pháp cần thiết. Tuy nhiên, chất lượng của phần mềm hiện có vẫn cần được đánh giá kỹ lưỡng để đảm bảo tính chính xác, hiệu suất và khả năng đáp ứng nhu cầu thực tế.

Đề tài "Đánh giá chất lượng phần mềm tính lương giảng viên" được thực hiện nhằm phân tích hiệu quả của hệ thống hiện tại, phát hiện các điểm còn hạn chế và đề xuất cải tiến. Kết quả nghiên cứu sẽ góp phần nâng cao chất lượng phần mềm, hỗ trợ các trường đại học quản lý lương giảng viên một cách chính xác, minh bạch và hiệu quả hơn.

* 1. Yêu cầu chung

Đề tài "Đánh giá chất lượng phần mềm tính lương giảng viên" hướng đến việc kiểm tra, phân tích và nâng cao hiệu quả của hệ thống phần mềm hiện có, đảm bảo đáp ứng các yêu cầu cơ bản sau:

* Tính chính xác: Đảm bảo phần mềm tính toán lương, thuế và các khoản phụ cấp đúng theo quy định của nhà trường và pháp luật.
* Hiệu suất: Hệ thống phải xử lý nhanh, ổn định ngay cả khi làm việc với lượng dữ liệu lớn (hàng trăm giảng viên).
* Dễ sử dụng: Giao diện trực quan, giúp người dùng (kế toán, quản lý, giảng viên) thao tác dễ dàng.
* Minh bạch: Cung cấp đầy đủ báo cáo chi tiết, cho phép giảng viên tự kiểm tra thông tin lương.
* Khả năng mở rộng: Dễ dàng tích hợp với các hệ thống khác (như phần mềm kế toán, quản lý đào tạo) và cập nhật khi có thay đổi chính sách.
  1. Yêu cầu cụ thể
     1. Yêu cầu chức năng
* Quản lý thông tin: Cập nhật và lưu trữ các thông tin như thông tin giảng viên, môn học, buổi dạy, bằng cấp và khoa.
* Lịch dạy và điểm danh: Theo dõi lịch giảng dạy thực tế và ghi nhận điểm danh, lập lịch dạy và quản lý học kì.
* Tính lương và thanh toán: Tính toán lương dựa trên thời gian giảng dạy thực tế, theo hệ số lương( bằng cấp, chức vụ), hỗ trợ thanh toán tự động.
* Báo cáo và thống kê: xuất báo cáo lương theo tháng/ quý/ năm.
  + 1. Yêu cầu phi chức năng
* Hiệu suất: Thời gian phản hồi dưới < 3 giây cho các thao tác thông thường. Xử lý được dữ liệu của 300 giảng viên mà không bị treo và chậm.
* Bảo mật : Phân quyền truy cập rõ ràng, mã hóa dữ liệu nhạy cảm.
* Khả năng tương thích: Chạy ổn định trên các hệ điều hàng Windows/ Linux. Hỗ trợ xuất báo cáo dưới dạng Excel.
  + 1. Yêu cầu kiểm thử
* Kiểm thử các chức năng: Đảm bảo các chức năng chính hoạt động đúng. Kiểm tra các trường hợp biên.
* Kiểm thử hiệu năng: Đánh giá tốc độ xử lý khi nhiều giảng viên cùng truy cập. Kiểm tra khả năng chịu tải.
* Kiểm thử giao diện: đánh giá tính thân thiện với người dùng. Kiểm tra khả năng tương thích trên nhiều trình duyệt và thiết bị.

1. **Các yêu cầu chức năng** 
   1. Tác nhân hệ thống

Tác nhân hệ thống bao gồm:

* Admin: Người thiết lập hệ thống cốt lõi, bao gồm cấu hình quy tắc tính lương và quản lý danh mục chung để đảm bảo tính nhất quán đối với giảng viên toàn trường.
* Giảng viên: Người trực tiếp tham gia giảng dạy, có nhiệm vụ khai báo thông tin cá nhân, cập nhật lịch dạy, điểm danh, cũng như xác nhận giờ giảng và các buổi làm việc.
* Kế toán : Người phụ trách quản lý và theo dõi các thông tin liên quan đến tài chính của giảng viên và nhà trường, có nhiệm vụ tổng hợp và xác nhận giờ giảng đã được phê duyệt, tính toán thù lao, lập bảng lương, và thực hiện các báo cáo tài chính liên quan đến hoạt động giảng dạy.
  1. Các chức năng hệ thống
* Nhóm chức năng 1: Nhóm chức năng quản lý giáo viên.
* Mục đích: Quản lý thông tin cơ bản của giáo viên và cơ cấu tổ chức.
* Các chức năng con:
* Quản lý danh mục bằng cấp: Xác định loại bằng cấp ( Thạc sĩ, Tiến sĩ, GS,...) để làm cơ sở tính hệ số lương.
* Quản lý khoa: Xác định rõ cơ cấu tổ chức ( khoa CNTT, khoa kinh tế) để phân công giảng dạy và thống kê báo cáo.
* Quản lý giáo viên: Lưu trữ hồ sơ giảng viên( họ tên, mã số, chức danh, khoa) để phân công và tính lương.
* Thống kê giáo viên: Tổng hợp số lượng giảng viên theo khoa/ chức danh, hỗ trợ quy hoạch nhân sự.
* Nhóm chức năng số 2: Nhóm chức năng quản lý lớp học
* Mục đích: Theo dõi hoạt động giảng dạy theo từng kỳ/ năm học.
* Các chức năng con:
* Quản lý học phần: Định nghĩa các môn học (tên, mã số, số tín chỉ) để làm cơ sở mở lớp và tính số tiết chuẩn.
* Quản lý kỳ học/năm học: Phân chia thời gian thành các kỳ (ví dụ: Kỳ 1/2023) để tổ chức lịch giảng dạy và tính lương theo chu kỳ.
* Quản lý TKB/lớp học: Ghi nhận thông tin lớp học (phòng, thời gian, sĩ số) để phân công giảng viên và tính hệ số lớp.
* Thống kê số lớp mở: Đánh giá khối lượng giảng dạy của từng học phần, hỗ trợ kế hoạch mở lớp năm sau.
* Nhóm chức năng số 3: Nhóm tính tiền dạy
* Mục đích: Thiết lập quy tắc tính lương và tự động hóa tính lương.
* Các chức năng con:
* Tính tiền dạy: Tự động tính lương dựa trên công thức.
* Công thức tính lương:

Lương = Hệ số lương x Mức lương cơ sở

Phụ cấp ưu đãi = lương x 30%

Tiền đóng bảo hiểm xã hội = Lương x 10.5%

Tổng lương thực nhận = Lương + Tiền phụ cấp ưu đãi – Tiền đóng bảo hiểm xã hội

* Nhóm chức năng số 4: Báo cáo
* Mục đích: Tổng quát lại dữ liệu lương do kế toán phụ trách
* Các chức năng con:
* Báo cáo theo giảng viên: Giúp giảng viên theo dõi thu nhập cá nhân trong năm, đối chiếu số liệu.
* Báo cáo theo khoa: Hỗ trợ trưởng khoa đánh giá ngân sách giảng dạy và phân bổ nhân lực.
* Báo cáo toàn trường: Cung cấp cho ban giám hiệu cái nhìn tổng thể về chi phí lương toàn trường.
  1. Sơ đồ Usecase tổng quát
     1. Sơ đồ Usecase chung

A screenshot of a phone

AI-generated content may be incorrect.

*Hình 1: Sơ đồ Usecase chung của hệ thống.*

* + 1. Mô tả Usecase

Biểu đồ Use Case thể hiện rõ vai trò và các chức năng chính mà người dùng có thể thực hiện trong hệ thống hỗ trợ quản lý giảng dạy tại cơ sở đào tạo. Toàn bộ hệ thống được thiết kế với mục tiêu phục vụ công tác tổ chức, vận hành và thống kê liên quan đến giảng viên, học phần, thời khóa biểu, cũng như tính toán và báo cáo tiền lương.

Hệ thống được chia thành 4 nhóm chức năng chính, mỗi nhóm bao gồm nhiều trường hợp sử dụng (use case) cụ thể, như sau:

* Nhóm 1: Quản lý giáo viên: Nhóm này bao gồm 4 chức năng nhằm mục tiêu giúp Admin tổ chức, theo dõi và cập nhật cơ sở dữ liệu liên quan đến giảng viên và các yếu tố cấu thành như khoa, bằng cấp:
* UC 1.1 – Quản lý danh mục bằng cấp: Cho phép Admin thêm, sửa, xóa các loại bằng cấp để làm dữ liệu tham chiếu cho giảng viên. Danh mục này giúp chuẩn hóa thông tin học thuật trong hồ sơ giảng viên.
* UC 1.2 – Quản lý khoa: Admin có thể tạo mới, cập nhật hoặc ngừng hoạt động các khoa trong trường. Đây là yếu tố phân loại quan trọng giúp giảng viên được sắp xếp hợp lý theo đơn vị chuyên môn.
* UC 1.3 – Quản lý giảng viên: cho phép Admin thực hiện các thao tác như thêm sửa xía hồ sơ giảng viên như họ tên, học vị, học hàm, khoa quản lý, trạng thái công tác,...
* UC 1.4 – Thống kê giảng viên: Cung cấp số liệu thống kê tổng quan về đội ngũ giảng viên như số lượng theo từng khoa, bằng cấp, tình trạng công tác,... phục vụ mục đích quản trị và lập kế hoạch.
* Nhóm 2: Quản lý lớp học . Nhóm chức năng này hỗ trợ Admin trong việc lập kế hoạch giảng dạy, theo dõi học phần và bố trí giảng viên phù hợp với lịch giảng:
* UC 2.1 – Quản lý học phần: Cho phép Admin thêm mới hoặc chỉnh sửa các học phần được giảng dạy, bao gồm thông tin như mã học phần, tên học phần, số tín chỉ, số tiết, hệ số giờ dạy,...
* UC 2.2 – Quản lý kỳ học: Quản lý các kỳ học theo năm học, học kỳ. Mỗi học phần, lớp học hay giảng viên đều phải gắn với một kỳ học cụ thể để đảm bảo tính nhất quán trong kế hoạch giảng dạy.
* UC 2.3 – Quản lý thời khóa biểu / lớp học: Hỗ trợ Giảng viên tạo mới và sắp xếp lớp học theo từng học phần, gắn liền với thời gian và giảng viên giảng dạy cụ thể.
* UC 2.4 – Thống kê số lớp mở cho các học phần: Cho phép thống kê và theo dõi số lượng lớp học được mở cho từng học phần theo từng kỳ học. Giúp kiểm soát và điều chỉnh việc phân bổ giảng viên và lớp học kịp thời.
* Nhóm 3: Tính lương Đây là chức năng đóng vai trò rất quan trọng, cho phép Kế toán thực hiện tính toán tiền lương giảng viên dựa trên số giờ giảng, hệ số giờ, hệ số học phần, thời điểm giảng dạy,...
* Nhóm 4: Báo cáo thống kê tiền giảng dạy cung cấp các báo cáo chi tiết nhằm hỗ trợ lãnh đạo nhà trường hoặc các bộ phận liên quan dễ dàng theo dõi, đánh giá tình hình giảng dạy cũng như quản lý chi phí nhân sự:
* UC 4.1 – Báo cáo tiền dạy của giảng viên trong một năm: Tổng hợp và thống kê toàn bộ số giờ giảng và tiền công tương ứng của từng giảng viên theo từng năm học.
* UC 4.2 – Báo cáo tiền dạy của giảng viên một khoa: Cho phép lọc và thống kê riêng biệt theo từng khoa để dễ dàng theo dõi hoạt động giảng dạy.
* UC 4.3 – Báo cáo tiền dạy của giảng viên toàn trường: Tổng hợp toàn bộ dữ liệu tiền giảng dạy của tất cả các giảng viên trong toàn trường trong khoảng thời gian nhất định, phục vụ mục tiêu tài chính và lập kế hoạch ngân sách.
  1. Đặc tả Usecase

*UC 1.1: Quản lý danh mục bằng cấp*

|  |  |  |
| --- | --- | --- |
| **UC1.1** | | **QUẢN LÝ DANH MỤC BẰNG CẤP** |
| **Mô tả ngắn gọn** | | Chức năng này cho phép quản trị viên thêm mới, chỉnh sửa, xóa và tìm kiếm các bằng cấp trong hệ thống. Bằng cấp sẽ được dùng để phân loại và thống kê giáo viên. Việc quản lý chính xác danh mục bằng cấp là cơ sở để tổ chức và lọc dữ liệu giảng viên. |
| **Tác nhân thực hiện** | | Quản trị viên |
| **Tiền điều kiện** | | Đã đăng nhập với quyền quản trị |
| **Hậu điều kiện** | **Thành công** | Danh sách bằng cấp được cập nhật đúng yêu cầu |
| **Lỗi** | Hệ thống hiển thị lỗi và không thay đổi dữ liệu |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| * **Luồng chính:**  1. Ở danh mục “Quản lý” người dùng chọn “Bằng cấp” 2. Hệ thống hiển thị danh sách các bằng cấp hiện có. 3. Quản trị viên chọn thao tác:  * **Thêm mới**: Nhập Tên bằng cấp, hệ số lương, hệ số giảng dạy→ Nhấn Lưu. * **Chỉnh sửa**: Chọn “Chỉnh sửa” → Cập nhật thông tin cần chỉnh sửa → Nhấn “Lưu thay đổi ”. * **Xóa**: Chọn bằng cấp cần xóa → Nhấn Xóa → Xác nhận.  1. Hệ thống phản hồi thành công và cập nhật danh sách. | | |
| **Luồng phát sinh** | | |
| * Nếu Bằng cấp đã có → Hệ thống báo lỗi và yêu cầu sửa. * Nếu bằng cấp đang được gán cho ít nhất một giáo viên → Hệ thống không cho xóa và hiển thị thông báo lỗi : “không được xóa vì đang có giáo viên sử dụng”. | | |

*UC 1.2: Quản lý khoa*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **QUẢN LÝ KHOA** |
| **Mô tả ngắn gọn** | | Chức năng cho phép quản trị viên thêm mới, chỉnh sửa, xóa và tìm kiếm thông tin khoa. Mỗi khoa có tên đầy đủ, tên viết tắt và mô tả nhiệm vụ riêng. Dữ liệu khoa là bắt buộc khi tạo thông tin giáo viên. |
| **Tác nhân thực hiện** | | Quản trị viên |
| **Tiền điều kiện** | | Đã đăng nhập với quyền quản trị |
| **Hậu điều kiện** | **Thành công** | Thêm mới, chỉnh sửa hoặc xóa khoa thành công |
| **Lỗi** | Hệ thống báo lỗi, không thực hiện thao tác |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| * **Luồng chính:**  1. Ở danh mục “ Quản lý” , người dùng chọn “Khoa” 2. Hệ thống hiển thị danh sách khoa gồm: Mã khoa, tên khoa 3. Quản trị viên chọn thao tác:  * Thêm mới: Nhập đầy đủ thông tin khoa → Nhấn Lưu. * Chỉnh sửa: Chọn khoa cần sửa → Cập nhật thông tin → Nhấn Lưu. * Xóa: Chọn khoa → Nhấn Xóa → Xác nhận xóa. * Tìm kiếm: Nhập từ khóa → Hệ thống lọc danh sách.  1. Hệ thống phản hồi kết quả: thành công hoặc báo lỗi. | | |
| **Luồng phát sinh** | | |
| * Trùng tên với một khoa đã có → Hiển thị lỗi, không cho lưu. * Khoa đang có giáo viên → Không thể xóa, hệ thống báo lỗi kèm cảnh báo liên kết dữ liệu. | | |

*UC 1.3: Quản lý giáo viên*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **QUẢN LÝ GIÁO VIÊN** |
| **Mô tả ngắn gọn** | | Chức năng cho phép quản trị viên thêm mới, chỉnh sửa, xóa và tìm kiếm thông tin giáo viên. Mỗi giáo viên có mã số nhân viên và tên đầy đủ. Dữ liệu giáo viên là bắt buộc khi tính lương. |
| **Tác nhân thực hiện** | | Quản trị viên |
| **Tiền điều kiện** | | Đã đăng nhập với quyền quản trị |
| **Hậu điều kiện** | **Thành công** | Thêm mới, xem, chỉnh sửa hoặc xóa thông tin giáo viên thành công |
| **Lỗi** | Hệ thống báo lỗi, không thực hiện thao tác |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| 1. Ở danh mục “Quản lý thông tin”, admin chọn “Giáo viên” 2. Hệ thống trả về danh sách giáo viên đã có. 3. Người dùng thao tác  * Thêm: Chọn “Thêm” → Cập nhật thông tin →Chọn “Thêm giáo viên” * Xem chi tiết: Chọn “Xem chi tiết” * Sửa: Chọn “Sửa” → Nhập thông tin thay đổi → Chọn “Lưu”. * Xóa: Chọn “Xóa” → Chọn đồng ý xóa. | | |
| **Luồng phát sinh** | | |
| * Khi nhập thiếu các trường bắt buộc, hệ thống báo lỗi và yêu cầu nhập đủ các trường bắt buộc. | | |

*UC 1.4: Thống kê giáo viên*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **THỐNG KÊ GIÁO VIÊN** |
| **Mô tả ngắn gọn** | | Tổng hợp và phân tích dữ liệu giáo viên theo khoa, chức danh, bằng cấp. |
| **Tác nhân thực hiện** | | Quản trị viên |
| **Tiền điều kiện** | | Đã đăng nhập với quyền quản trị  Có dữ liệu giáo viên trong hệ thống |
| **Hậu điều kiện** | **Thành công** | Thống kê số lượng giáo viên theo khoa, chức danh, bằng cấp |
| **Lỗi** | Hiển thị thông báo lỗi khi không có dữ liệu |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| 1. Người dùng chọn “ Thống kê giáo viên” trên danh mục “ Quản lý”. 2. Hệ thống hiển thị trang Thống kê 3. Người dùng chọn tiêu chí lựa chọn và chọn “ Thống kê” 4. Hệ thống hiển thị kết quả. | | |
| **Luồng phát sinh** | | |
| * Khi không có dữ liệu thỏa mãn điều kiện: Hệ thống hiển thị thông báo “ Không có dữ liệu phù hợp”. | | |

*UC 2.1: Quản lý học phần*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **QUẢN LÝ HỌC PHẦN** |
| **Mô tả ngắn gọn** | | Chức năng cho phép người dùng quản lý thông tin các học phần/ môm học trong chương trình đào tạo, làm cơ sở cho việc mở lớp và tính toán giờ dạy. |
| **Tác nhân thực hiện** | | Quản trị viên |
| **Tiền điều kiện** | | Đã đăng nhập với quyền quản trị  Phải có ít nhất 1 khoa để gán học phần |
| **Hậu điều kiện** | **Thành công** | Thêm, sửa, xóa học phần |
| **Lỗi** | Hệ thống báo lỗi, không thực hiện thao tác  Khi có lớp học đang mở, không thể xóa học phần |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| 1. Người dùng chọn “Học phần” trên danh mục “Quản lý môn học” 2. Hệ thống hiển thị danh sách học phần đã có. 3. Người dùng thao tác các chức năng:  * Thêm: Chọn “Thêm học phần ” → Cập nhật thông tin(tên môn học, số tiết, khoa, số tín chỉ,..)→Chọn “Thêm môn học”. * Sửa: Chọn “Sửa” → Nhập thông tin thay đổi → Chọn “Lưu”. * Xóa: Chọn “Xóa” → Hệ thống xác nhận xóa → Chọn “Xóa” | | |
| **Luồng phát sinh** | | |
|  | | |

*UC2.2: Quản lý kỳ học*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **QUẢN LÝ KỲ HỌC** |
| **Mô tả ngắn gọn** | | Chức năng cho phép quản trị viên kiểm soát thời gian năm học, là cơ sở để mở lớp. Người dùng có thể tạo mới, chỉnh sửa, xóa học kỳ, thiết kế thời gian bắt đầu và kết thúc học kỳ |
| **Tác nhân thực hiện** | | Quản trị viên |
| **Tiền điều kiện** | | Đã đăng nhập với quyền quản trị |
| **Hậu điều kiện** | **Thành công** | Kỳ học được lưu vào hệ thống |
| **Lỗi** | Hệ thống báo lỗi nếu thời gian trùng/ không hợp lệ. |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| 1. Ở danh mục “Quản lý môn học”, chọn “Học kỳ ” 2. Hệ thống trả về danh sách học kỳ đã có. 3. Người dùng thao tác thêm mới, chỉnh sửa, hoặc đóng kỳ 4. Hệ thống kiểm tra các thông tin được nhập ( không trùng với thời gian kỳ khác, ngày bắt đầu < ngày kết thúc) 5. Hệ thống hiển thị thông tin tạo kỳ thành công | | |
| **Luồng phát sinh** | | |
| * Khi thời gian học kỳ mới trùng với học kỳ đã có, hệ thống báo trùng lịch học. * Khi ngày kết thúc trước ngày bắt đầu, hệ thống báo lỗi. * Khi kỳ học mới bắt đầu khi chưa đóng kỳ học cũ, hệ thống báo lỗi. | | |

*UC 2.3: Quản lý TKB/ lớp học*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **QUẢN LÝ TKB/ LỚP HỌC** |
| **Mô tả ngắn gọn** | | Chức năng cho phép người dùng quản lý thông tin lớp học và thời khóa biểu của từng học kỳ bao gồm: Tạo và sắp xếp lớp, phân công giảng viên, kiểm tra lịch trùng,... |
| **Tác nhân thực hiện** | | Người dùng  Giảng viên |
| **Tiền điều kiện** | | Đã đăng nhập với quyền quản trị  Đăng nhập với quyền Giảng viên  Đã có danh sách học phần, danh sách giảng viên và kỳ học đang hoạt động |
| **Hậu điều kiện** | **Thành công** | Lớp học được tạo, TKB được hiển thị  Hệ thống tự động tính các hệ số để phục vụ việc tính lương. |
| **Lỗi** | Hệ thống báo lỗi nếu thời gian trùng. |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| 1. Người dùng chọn “Kỳ học” trên thanh “Quản lý môn học” 2. Hệ thống trả về Danh sách kỳ học. 3. Người dùng thao tác các chức năng có thể có  * Thêm: Chọn “Thêm” → Cập nhật thông tin lịch học →Chọn “Lập lịch” * Sửa: Chọn “Sửa” → Nhập thông tin thay đổi → Chọn “Lưu”. | | |
| **Luồng phát sinh** | | |
| * Khi đăng nhập với quyền giảng viên thì chỉ có thể sửa thông tin lớp học | | |

*UC2.4: Thống kê số lớp đã mở*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **THỐNG KÊ SỐ LỚP ĐÃ MỞ** |
| **Mô tả ngắn gọn** | | Tổng hợp và phân tích dữ liệu về các môn học( học phần) trong hệ thống |
| **Tác nhân thực hiện** | | Quản trị viên |
| **Tiền điều kiện** | | Đã đăng nhập với quyền quản trị  Có dữ liệu danh sách môn học trong hệ thống |
| **Hậu điều kiện** | **Thành công** | Hiển thị thống kê theo yêu cầu |
| **Lỗi** | Hiển thị thông báo lỗi khi không có dữ liệu |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| 1. Người dùng chọn “ Thống kê số lớp học phần” trên danh mục “ Quản lý môn học”. 2. Hệ thống hiển thị trang Thống kê 3. Người dùng chọn tiêu chí lựa chọn và chọn “ Thống kê” 4. Hệ thống hiển thị kết quả. | | |
| **Luồng phát sinh** | | |
| * Khi không có dữ liệu thỏa mãn điều kiện: Hệ thống hiển thị thông báo “ Không có dữ liệu phù hợp”. | | |

*UC 3.1: Tính lương*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **TÍNH LƯƠNG** |
| **Mô tả ngắn gọn** | | Người dùng chọn tên giáo viên và kỳ dạy. Hệ thống tự động tính toán tiền lương cho giảng viên dựa trên hệ số mà hệ thống tính toán. |
| **Tác nhân thực hiện** | | Kế toán |
| **Tiền điều kiện** | | Đã đăng nhập với kế toán  Có dữ liệu trong hệ thống (giáo viên, phân công giảng dạy) |
| **Hậu điều kiện** | **Thành công** | Hiển thị kết quả tính lương chi tiết theo từng môn học giảng dạy. |
| **Lỗi** | Hiển thị thông báo lỗi khi không có dữ liệu |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| 1. Người dùng chọn “Tính lương” trên danh mục “ Lương”. 2. Hệ thống hiển thị form điền thông tin giáo viên cần tính lương 3. Người dùng chọn giáo viên và kỳ học cần tính. 4. Hệ thống hiển thị kết quả. | | |
| **Luồng phát sinh** | | |
| * Khi không có dữ liệu thỏa mãn điều kiện: Hệ thống hiển thị thông báo “ Không có dữ liệu phù hợp”. | | |

*UC 4.1: Báo cáo*

|  |  |  |
| --- | --- | --- |
| **UC1.3** | | **BÁO CÁO** |
| **Mô tả ngắn gọn** | | Cung cấp báo cáo tổng hợp về lương của giáo viên theo thời gian, khoa. |
| **Tác nhân thực hiện** | | Kế toán |
| **Tiền điều kiện** | | Đã đăng nhập với quyền kế toán  Có dữ liệu giáo viên trong hệ thống |
| **Hậu điều kiện** | **Thành công** | Hiển thị báo cáo tiền lương giáo viên, |
| **Lỗi** | Hiển thị thông báo lỗi khi không có dữ liệu |
| **ĐẶC TẢ CHỨC NĂNG** | | |
| **Luồng sự kiện chính/Kịch bản chính** | | |
| 1. Người dùng chọn “ Báo cáo tiền lương” trên danh mục “ Báo cáo”. 2. Hệ thống hiển thị trang báo cáo tiền dạy 3. Người dùng chọn tiêu chí lựa chọn và chọn “ Xem báo cáo ” 4. Hệ thống hiển thị kết quả. | | |
| **Luồng phát sinh** | | |
| * Khi không có dữ liệu, hệ thống hiển thị thông báo “Không có dữ liệu lương trong khoảng thời gian này” | | |

1. **Biểu đồ hệ thống**

**3.1 Biểu đồ tuần tự**

**3.1.1 : Đăng nhập**

A diagram of a log-controller

AI-generated content may be incorrect.

**3.1.2:Thêm giảng viên**

A screenshot of a computer screen

AI-generated content may be incorrect.

**3.1.3: Thêm môn học**

A screenshot of a computer

AI-generated content may be incorrect.

**3.1.4:Thêm học kỳ**

A screenshot of a computer

AI-generated content may be incorrect.

**3.1.5: Lập lịch dạy** A screenshot of a computer program

AI-generated content may be incorrect.

**3.1.6 : Tính lương** A screenshot of a computer screen

AI-generated content may be incorrect.

**3.2 : Biểu đồ gói**

A diagram of a software company

AI-generated content may be incorrect.

**3.3: Biểu đồ hoạt động (active)**

A diagram of a computer

AI-generated content may be incorrect.

1. **Các yêu cầu phi chức năng** 
   1. Giao diện người dùng

Giao diện người dùng (UI) của hệ thống tính lương giáo viên cần phải

đáp ứng các tiêu chí sau:

• Thân thiện và dễ sử dụng: Giao diện phải trực quan, dễ hiểu và dễ sử dụng

ngay cả với những người không có nhiều kinh nghiệm về công nghệ hoặc

kiến thức.

• Thiết kế đáp ứng (Responsive Design): Giao diện phải tương thích với nhiều

loại thiết bị và kích thước màn hình khác nhau, bao gồm máy tính để bàn,

máy tính xách tay, máy tính bảng và điện thoại di động.

• Đồng nhất và chuyên nghiệp: Sử dụng phong cách thiết kế đồng nhất trong

toàn bộ hệ thống để tạo cảm giác chuyên nghiệp và tin cậy.

• Tương tác người dùng: Cung cấp các thông báo và hướng dẫn rõ ràng để hỗ

trợ người dùng trong quá trình nhập liệu và thực hiện các tác vụ

* 1. Hiệu suất

Hệ thống phải đảm bảo hiệu suất cao để đáp ứng nhu cầu của người dùng:

• Thời gian phản hồi: Hệ thống phải có thời gian phản hồi nhanh, đảm bảo

các trang web và chức năng tải nhanh chóng, tối đa trong vòng 2 giây.

• Khả năng xử lý đồng thời: Hệ thống phải có khả năng xử lý nhiều yêu cầu

từ người dùng cùng một lúc mà không bị chậm hoặc lỗi.

• Quản lý tải cao: Hệ thống phải hoạt động ổn định ngay cả khi có lượng truy

cập lớn, đặc biệt trong các kỳ khai báo và nộp thuế cao điểm.

**V . Thiết kế cơ sở dữ liệu và cài đặt**

* 1. **Thiết kế cơ sở dữ liệu**
  2. Sơ đồ cơ sở dữ liệu

**A computer screen shot of a computer

AI-generated content may be incorrect.**

*Hình 1: Thiết kế cơ sở dữ liệu hệ thống*

* 1. Danh sách chi tiết các bảng trong cơ sở dữ liệu
* Khoa

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_khoa | VARCHAR(10) | Khóa chính |
| ten\_khoa | VARCHAR(100) | Tên khoa |

* Bằng cấp

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_bangcap | VARCHAR(10) | Khóa chính |
| ten\_bangcap | VARCHAR(50) |  |
| he\_so | DECIMAL(3,1) | Hệ số quy đổi giảng |

* Học kỳ

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_hk | VARCHAR(10) | Khóa chính |
| ten\_hk | VARCHAR(50) |  |
| ngay\_bat\_dau | DATE |  |
| ngay\_ket\_thuc | DATE |  |
| nam\_hoc | VARCHAR(20) |  |
| Luong\_hocky | INT |  |
| trang\_thai | ENUM | 'Sắp diễn ra', 'Đang diễn ra', 'Đã kết thúc' |

* Môn học

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_mon | VARCHAR(10) | Khóa chính |
| ten\_mon | VARCHAR(100) |  |
| so\_tiet | INT |  |
| so\_tin\_chi | INT | Mặc định: 2 |
| mo\_ta | TEXT |  |
| ma\_khoa | VARCHAR(10) | FK → khoa |
| he\_so | DECIMAL(3,1) | Mặc định: 1.0 |

* Giáo viên

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_gv | VARCHAR(10) | Khóa chính |
| ho\_ten | VARCHAR(100) |  |
| gioi\_tinh | ENUM('Nam','Nữ') |  |
| ngay\_sinh | DATE |  |
| dia\_chi | TEXT |  |
| email | VARCHAR(100) | Duy nhất |
| so\_dien\_thoai | VARCHAR(15) |  |
| ma\_khoa | VARCHAR(10) | FK → khoa |
| ma\_bangcap | VARCHAR(10) | FK → bangcap |
| ngay\_vao\_lam | DATE |  |
| trang\_thai | TINYINT(1) | Mặc định: 1 |

* User

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| id | INT | AUTO\_INCREMENT, PK |
| username | VARCHAR(100) | UNIQUE |
| password | VARCHAR(255) | Có thể NULL |
| ma\_gv | VARCHAR(10) | FK → giaovien |
| role | ENUM | admin, teacher, accountant |
| active | TINYINT(1) | Mặc định: 1 |

* Lớp học

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_lop | VARCHAR(10) | PK |
| ten\_lop | VARCHAR(50) |  |
| so\_sinh\_vien | INT |  |
| ma\_mon | VARCHAR(10) | FK |
| ma\_hk | VARCHAR(10) | FK |

* Lịch dạy

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_lich | VARCHAR(15) | PK |
| ma\_gv | VARCHAR(10) | FK |
| ma\_mon | VARCHAR(10) | FK |
| ma\_hk | VARCHAR(10) | FK |
| ngay\_day | DATE |  |
| tiet\_bat\_dau | INT |  |
| so\_tiet | INT |  |
| phong\_hoc | VARCHAR(20) |  |
| ten\_lop | VARCHAR(50) |  |
| ten\_lop\_hoc | VARCHAR(50) |  |
| so\_sinh\_vien | INT | Mặc định 40 |
| thu\_trong\_tuan | INT |  |
| so\_buoi\_tuan | INT | Mặc định 1 |
| ma\_lich\_goc | VARCHAR(15) | FK → lich\_day |
| ghi\_chu | TEXT |  |
| he\_so\_lop | DECIMAL(3,1) | Mặc định 0 |

* Buổi dạy

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_buoi | VARCHAR(10) | PK |
| ma\_gv | VARCHAR(10) | FK |
| ma\_mon | VARCHAR(10) | FK |
| ngay\_day | DATE |  |
| tiet\_bat\_dau | INT |  |
| so\_tiet | INT |  |
| phong\_hoc | VARCHAR(20) |  |

* Lương

|  |  |  |
| --- | --- | --- |
| Cột | Kiểu dữ liệu | Ghi chú |
| ma\_luong | VARCHAR(10) | PK |
| ma\_gv | VARCHAR(10) | FK |
| thang | INT |  |
| nam | INT |  |
| so\_tiet | INT |  |
| he\_so\_luong | DECIMAL(10,2) |  |
| thuc\_lanh | DECIMAL(20,2) |  |

1. **Cài đặt phần mềm**
   1. Lựa chọn công nghệ

A blue elephant with a face and text

AI-generated content may be incorrect.

* + 1. Mô tả sơ đồ

Sơ đồ triển khai hệ thống mô tả cách thức hoạt động của trang web. Đầu

tiên, người dùng sử dụng thiết bị để truy cập trang web và tương tác với

giao diện được xây dựng bằng HTML, CSS và JavaScript. Khi người dùng

muốn xem thông tin, giao diện gửi yêu cầu "Nhận dữ liệu" đến máy chủ

sử dụng PHP. Máy chủ hỏi cơ sở dữ liệu MySQL qua “truy vẫn dữ liệu”

để lấy thông tin, sau đó mySQL trả về kết quả truy vấn. Máy chủ gửi lại

dữ liệu để hiển thị cho người dùng.

* + 1. Các công nghệ sử dụng
* HTML:

HTML là ngôn ngữ tiêu chuẩn cho việc tạo cấu trúc của trang web, cung

cấp một nền tảng vững chắc cho nội dung và các thành phần giao diện

người dùng. Nó sử dụng các thẻ để xác định các phần tử khác nhau trên

trang, cho phép tạo ra cấu trúc cho trang web với các phần như tiêu đề,

đoạn văn, bảng, danh sách, hình ảnh và liên kết. Bên cạnh đó, HTML còn

xác định các thành phần giao diện người dùng như form nhập liệu, nút bấm

và các phần tử tương tác khác. Với những lợi ích như dễ học và sử dụng,

phù hợp cho cả người mới bắt đầu và lập trình viên có kinh nghiệm, cùng

với tính tương thích với tất cả các trình duyệt web hiện đại, HTML giúp

đảm bảo tính khả dụng và trải nghiệm người dùng tốt.

* CSS

CSS là ngôn ngữ dùng để định dạng và tạo kiểu cho nội dung HTML, cho

phép tách biệt nội dung và hình thức, từ đó giúp dễ dàng quản lý và bảo trì

thiết kế. CSS tạo ra giao diện hấp dẫn, thân thiện với người dùng và dễ

dàng tùy chỉnh. Chức năng của CSS bao gồm định dạng văn bản với các

thuộc tính như màu sắc, kích thước và kiểu chữ; thiết lập bố cục thông qua các kỹ thuật như flexbox và grid; cũng như tạo hiệu ứng chuyển động và

hoạt ảnh. Những lợi ích của CSS không chỉ giúp tạo ra giao diện đẹp mắt

mà còn giảm thiểu sự lặp lại trong mã, giúp tiết kiệm thời gian và công sức

cho lập trình viên.

* JavaScript

JavaScript là ngôn ngữ lập trình cho phép tạo ra các tính năng tương tác

trên trang web, giúp xử lý sự kiện, thay đổi nội dung trang mà không cần

tải lại và tương tác với API. Ngôn ngữ này cung cấp khả năng tương tác

cao cho người dùng, hỗ trợ các chức năng như tìm kiếm, lọc thực đơn và

xử lý sự kiện. Các chức năng của JavaScript bao gồm việc tạo các hiệu ứng

động và tương tác cho người dùng, như mở menu và hiển thị thông báo;

gửi và nhận dữ liệu từ máy chủ thông qua AJAX; và xử lý các sự kiện như

click, hover, v.v. Những lợi ích của JavaScript không chỉ nâng cao khả

năng tương tác của website, mà còn cải thiện trải nghiệm người dùng.

Ngoài ra, nó còn hỗ trợ nhiều thư viện và framework như jQuery, React,

Angular, giúp phát triển các ứng dụng phức tạp hơn.

* PHP

PHP là ngôn ngữ lập trình phía máy chủ được sử dụng để phát triển các

ứng dụng web động. Nó cho phép xử lý dữ liệu từ cơ sở dữ liệu, quản lý

người dùng và thực hiện các chức năng phức tạp như gửi email và lưu trữ

thông tin đặt bàn. Các chức năng chính của PHP bao gồm xử lý dữ liệu từ

biểu mẫu và lưu trữ vào cơ sở dữ liệu; tạo nội dung động dựa trên dữ liệu

từ cơ sở dữ liệu; và quản lý phiên người dùng cũng như xác thực

(login/logout). Những lợi ích của PHP bao gồm khả năng dễ dàng tích hợp

với MySQL để quản lý dữ liệu, cùng với cộng đồng lớn và nhiều tài liệu

hỗ trợ, giúp người dùng dễ dàng tìm kiếm giải pháp cho các vấn đề gặp

phải.

* MySQL

MySQL là một hệ quản trị cơ sở dữ liệu mã nguồn mở, được sử dụng phổ

biến để lưu trữ và quản lý dữ liệu cho các ứng dụng web. Nó cho phép

người dùng thực hiện các truy vấn để lấy, thêm, sửa đổi và xóa dữ liệu một

cách dễ dàng. MySQL nổi bật với hiệu suất cao và khả năng tích hợp tốt

với ngôn ngữ lập trình PHP, giúp quản lý thông tin như thực đơn, đặt bàn

và thông tin khách hàng một cách hiệu quả. Với khả năng hỗ trợ các truy vấn phức tạp, MySQL còn giúp người dùng phân tích và báo cáo dữ liệu

một cách nhanh chóng và chính xác.

* + 1. Thiết lập môi trường cài đặt
* Yêu cầu hệ thống
* Hệ điều hành : Windows, macOS hoặc Linux
* Trình duyệt : Chrome, Firefox, Safari.
* Kết nối Internet: Để tải xuống các phần mềm cần thiết.
* Thiết lập môi trường
* Cài đặt XAMPP
* Bước 1. Tải xuống XAMPP

Truy cập (https://www.apachefriends.org/index.html).

Chọn phiên bản phù hợp với hệ điều hành của bạn và tải xuống

* Bước 2: Cài đặt XAMPP
* Mở tệp cài đặt đã tải xuống.
* Làm theo hướng dẫn cài đặt, chọn các thành phần cần thiết (Apache, MySQL, PHP).
* Chọn thư mục cài đặt (mặc định là `C:\xampp` trên Windows).
* Nhấn "Finish" để hoàn tất cài đặt.
* Khởi động XAMPP
* Mở XAMPP Control Panel.
* Nhấn nút "Start" bên cạnh Apache và MySQL để khởi động máy chủ web và cơ sở dữ liệu.
* Kiểm tra hoạt động của Apache bằng cách mở trình duyệt và truy cập `http://localhost`. Nếu thấy trang "XAMPP Welcome", nghĩa là Apache đã hoạt động.
* Cài đặt trình soạn thảo
* Cài đặt Visual Studio Code: https://code.visualstudio.com.
* Mở Visual Studio Code và cài đặt các tiện ích mở rộng cần thiết như PHP IntelliSense, Live Server và Prettier.
* Tạo cơ sở dữ liệu MySQL
* Mở phpMyAdmin bằng cách truy cập:
* http://localhost/phpmyadmin.
* Đăng nhập bằng tài khoản mặc định (thường không có mật khẩu).
* Nhấn "Databases" để tạo một cơ sở dữ liệu mới cho dự án.
* Nhập tên cơ sở dữ liệu và nhấn "Create".
  1. Cấu trúc mã nguồn

https://github.com/leedangg/PTTKPM25-26\_ClassN06\_Nhom02

1. Cấu trúc thư mục

A screen shot of a computer

AI-generated content may be incorrect.

Hệ thống được tổ chức theo hướng mô-đun để dễ bảo trì và mở rộng. Cấu trúc chính gồm:

config/: Chứa tệp cấu hình kết nối cơ sở dữ liệu (database.php).

modules/: Gồm các chức năng như quản lý giáo viên, bằng cấp, khoa, buổi dạy, lịch dạy, tính lương và báo cáo. Mỗi chức năng là một thư mục riêng, chứa các tệp index.php, them.php, sua.php, xoa.php.

assets/: Chứa tài nguyên tĩnh như CSS (style.css) và JavaScript (scripts.js).

index.php: Trang chính của hệ thống.

**VI. Giao diện hệ thống**

**Giao diện học phần**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Giao diện thêm môn học mới**

**A screenshot of a computer

AI-generated content may be incorrect.  
Giao diện thống kê giáo viên**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Giao diện quản lý khoa**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Giao diện thêm giảng viên**

**A screenshot of a computer

AI-generated content may be incorrect.**

**Giao diện màn hình chỉnh giảng viên**

**A screenshot of a computer

AI-generated content may be incorrect.**

**VII .Kế hoạch kiểm thử**

* 1. **Giới thiệu**

1.1 : Mục đích

Tài liệu kế hoạch kiểm thử này đưa ra các mục đích sau:

• Xác định thông tin cơ bản về dự án và các thành phần chức năng được kiểm thử và không được kiểm thử.

• Liệt kê những yêu cầu cho việc kiểm thử (Test Requirements).

• Những chiến lược kiểm thử nên được sử dụng.

• Ước lượng những yêu cầu về tài nguyên và chi phí cho việc kiểm thử.

• Những tài liệu được lập sau khi hoàn thành việc kiểm thử.

1.2 : Tổng quan

Báo cáo cung cấp chi tiết về phạm vi dự án, các yêu cầu, chức năng và các công việc kiểm thử. Dự án phát triển hệ thống tính lương giáo viên toàn trường với các chức năng như thêm giáo viên, lập lịch dạy, tính lương, và báo cáo,... Mỗi nhóm chức năng sẽ được kiểm thử để đảm bảo hệ thống hoạt động đúng yêu cầu và không gặp lỗi khi sử dụng.

1.3 : Phạm vi

Phạm vi của việc kiểm thử trong dự án Hệ thống tính lương giáo viên bao gồm việc đảm bảo các chức năng của hệ thống hoạt động đúng, dữ liệu nhất quán và hệ thống vận hành ổn định theo nghiệp vụ. Các dịch vụ và chức năng cần kiểm tra bao gồm:

* Quản lý giảng viên
* Quản lý môn học
* Tính lương
* Báo cáo

1.4 : Những người sử dụng tài liệu

Báo cáo này được sử dụng bởi các bên liên quan trong quá trình kiểm thử phần mềm, bao gồm:

* Test Manager: Quản lý toàn bộ quy trình kiểm thử, đảm bảo kế hoạch kiểm thử được thực hiện đúng tiến độ và yêu cầu.
* Test Designer: Thiết kế các trường hợp kiểm thử (test case), xác định chiến lược và phương pháp kiểm thử.
* Tester: Thực hiện kiểm thử theo các trường hợp kiểm thử đã được thiết kế, báo cáo lỗi và theo dõi quá trình sửa lỗi.

1. **Những yêu cầu về tài nguyên**

2.1. Phần cứng

Để đảm bảo quá trình kiểm thử phần mềm tính lương giáo viên diễn ra hiệu quả, yêu cầu về phần cứng bao gồm:

|  |  |  |  |
| --- | --- | --- | --- |
| **CPU** | **RAM** | **HDD** | **Architecture** |
| Intel Core i3 3.5GHz | 4GB | 512GB | 64 bit |

2.2. Phần mềm

Các phần mềm cần thiết để hỗ trợ quá trình kiểm thử bao gồm:

|  |  |  |
| --- | --- | --- |
| Tên phần mềm | Phiên bản | Loại |
| Microsoft Edge | Version 131.0.2903.70 | Web site |
| Microsoft Windows 11 | Windows 11.5 | Operating system |

2.3. Công cụ kiểm thử

Các công cụ kiểm thử được sử dụng để quản lý, thực hiện và theo dõi quá trình kiểm thử:

|  |  |  |  |
| --- | --- | --- | --- |
| Hoạt động | Công cụ | Nhà cung cấp | Phiên bản |
| Quản lý Test Case | Microsoft Excel | Microsoft | 2024 |
| Defect Tracking | Microsoft Excel | Microsoft | 2024 |
| Quản lý tiến độ kiểm thử | Microsoft Excel | Microsoft | 2024 |

2.4. Môi trường kiểm thử

Môi trường kiểm thử bao gồm cá thiết bị và cấu hình cần thiết để thực hiện quá trình kiểm thử:

* Máy tính cá nhân có kết nối Internet để truy cập vào hệ thống tính thuế thu nhập cá nhân.
* Hệ điều hành: Microsoft Windows 10.
* Trình duyệt web: Microsoft Egde/ Cốc Cốc/ Google Chorme

1. **Phạm vi kiểm thử**

3.1. Những chức năng được kiểm thử

* Quản lý giáo viên
* Kiểm tra định dạng dữ liệu.
* Kiểm tra thông tin của giảng viên, có thể thêm sửa xóa giảng viên
* Quản lý bằng cấp và khoa
* Kiểm tra thêm mới / cập nhật tên khoa, mã khoa có bị trùng không
* Kiểm tra hệ số lương và hệ số môn học hợp lệ ( số dương, đúng định dạng )
* Quản lý kỳ học
* Kiểm tra Ngày bắt đầu và ngày kết thúc, ngày kết lúc phải > ngày bắt đầu.
* Quản lí học phần:
* Kiểm tra số tiết và số tín chỉ hợp lệ
* Kiểm tra liên kết với khoa
* Kiểm tra đúng hệ số môn học
* Chức năng tính lương:
* Kiểm tra đúng hệ số bằng cấp và hệ số môn học

1. **Chiến lược kiểm thử**

4.1. Các hình thức kiểm thử và các bước kiểm thử

* + 1. Kiểm thử hộp đen

Định nghĩa: Kiểm thử hộp đen là một phương pháp kiểm thử không cần quan tâm đến cấu trúc, mã nguồn, hay cách thức hoạt động bên trong của hệ thống. Thay vào đó, họ chỉ tập trung vào đầu vào và đầu ra của hệ thống để kiểm tra xem phần mềm có hoạt động đúng như mong đợi hay không. Điều này phù hợp vói chức năng như đăng nhập, khai báo thuế và tính thuế, yêu cầu hệ thống hoạt động đúng theo yêu cầu dưới góc độ người sử dụng.

Các bước kiểm thử:

• Bước 1: Lập kế hoạch test o Xây dụng tài liệu và những yêu cầu cụ thể. o Đưa ra các công cụ kiểm thử phù hợp.

* Tìm hiểu các chức năng cần kiểm tra.

• Bước 2: Thiết kế Test Case

* Đưa ra một bản test case đầy đủ chức các thông tin và dữ liệu đầu vào, mô tả chính xác thông tin đầu ra xem đúng kế hoạch đưa ra hay chưa.

• Bước 3: Thực thi kiểm thử

* Kiểm tra dựa vào các test case đã viết. o Ghi chép đầy đủ các trạng thái đã test thành công hay thất bại, ghi rõ các lỗi nếu có.

• Bước 4: Ghi nhận và báo cáo o Viết báo cáo kết quả đạt được.

* Đưa ra lỗi và tỷ lệ lỗi.
* Chuyển báo cáo cho nhóm phát triển để khắc phục.

• Bước 5: Kiểm thử lại o Sau khi các lỗi để được nhóm phát triển khắc phục.

* Đảm bảo các lỗi đã được sửa và báo cáo lại.

Khó khăn: Dữ liệu đầu vào khá nặng, các test case sẽ mất thời gian vì các mô tổ và thông số chưa rõ ràng. Độ chính xác của phạm vi kiểm thử khó có thể xác định.

* + 1. Kiểm thử hộp trắng

Định nghĩa: Kiểm thử hộp trắng là một cách kiểm thử khi biết cấu trúc bên trong và mã nguồn của hệ thống. Từ đó có thể kiểm tra code và thuật toán. Kiểm thử này phù hợp với các chức năng quan trọng như thuật toán tính thuế.

Các bước kiểm thử:

* Bước 1: Phân tích mã nguồn và cấu trúc
* Đọc và phân tích các thuật toán.
* Xây dựng tài liệu.
* Bước 2: Thiết kế Test Case
* Viết các test case dựa trên thuật toán trong mã nguồn.
* Tạo các test cho nhánh, vòng lặp và các điều kiện trong mã nguồn.
* Bước 3: Thực thi kiểm thử
* Kiểm tra dựa vào các test case đã viết.
* Ghi chép đầy đủ các trạng thái đã test thành công hay thất bại, ghi rõ các lỗi nếu có.
* Bước 4: Báo cáo và sửa lỗi.
* Viết báo cáo kết quả đạt được.
* Đưa ra lỗi và tỷ lệ lỗi. o Sửa các lỗi và kiểm thử lại.
  + 1. Kiểm thử thủ công

Định nghĩa: Kiểm thử thủ công là quá trình làm mọi công việc bằng tay, như thực hiện một số sự kiến của các nút bám, các thao tác nhập liệu. Phù hợp với các chức năng đăng nhập, khai báo thuế.

Các bước kiểm thử:

* Bước 1: Chuẩn bị môi trường
* Thiết lập môi trường bao gồm phần cứng, phần mềm và dữ liệu cần kiểm thử.
* Bước 2: Thực thi thủ công
* Thực thi các test thủ công.
* Quan sát và ghi nhận kết quả.
* Bước 3: Ghi nhận và báo cáo
* Viết báo cáo kết quả đạt được.
* Đưa ra lỗi và tỷ lệ lỗi.
* Chuyển báo cáo cho nhóm phát triển để khắc phục.
* Bước 4: Kiểm thử lại
* Sau khi các lỗi để được nhóm phát triển khắc phục.
* Đảm bảo các lỗi đã được sửa và báo cáo lại.

1. **Điều kiện chấp thuận**

Điều kiện chấp nhận là tập hợp các tiêu chí mà hệ thống phải đáp ứng để được coi là hoàn thiện và sẵn sàng đưa vào sử dụng. Đối với hệ thống quản lý giảng viên, điều kiện chấp nhận bao gồm các nội dung sau:

* Hoàn thành toàn bộ các test case
* Tất cả các trường hợp kiểm thử đã được xây dựng, bao gồm kiểm thử chức năng và phi chức năng, đều phải được thực thi đầy đủ và đạt kết quả "đạt".
* Các chức năng chính như: quản lý giảng viên, lịch dạy, tính lương,... đều phải được kiểm thử kỹ càng.
* Các lỗi nghiêm trọng (nếu có) phải được xử lý trước khi triển khai chính thức.
* Hoạt động ổn định trên nhiều nền tảng
* Hệ thống phải chạy mượt mà trên các trình duyệt phổ biến như Chrome, Firefox, Edge,... mà không xảy ra lỗi giao diện hoặc lỗi thao tác.
* Đảm bảo người dùng có thể truy cập và sử dụng hệ thống ổn định từ các thiết bị khác nhau như máy tính, laptop, hoặc máy tính bảng.
* Server phải đảm bảo vận hành ổn định trong thời gian dài, xử lý mượt mà các thao tác như tra cứu lịch dạy, tính lương...
* Yêu cầu về bảo mật:
* Tuân thủ các tiêu chuẩn cơ bản về bảo mật hệ thống và bảo vệ thông tin cá nhân.
* Khả năng bảo trì và mở rộng
* Mã nguồn được thiết kế theo cấu trúc rõ ràng, dễ bảo trì và dễ nâng cấp khi cần mở rộng hệ thống.
* Có khả năng cập nhật dữ liệu (như hệ số lương, học kỳ, môn học) một cách linh hoạt mà không ảnh hưởng đến dữ liệu hiện có.

1. **Defect Tracking**

6.1. Phân loại lỗi

|  |  |
| --- | --- |
| Mức độ nghiêm trọng | Đặc tả lỗi |
| Cao | Các chức năng chính của như tính thuế, đăng nhập không sử dụng được.  Các nút bấm không hoạt động.  Hệ thống bị treo.  Các ô nhập liệu không thể nhập. |
| Trung bình | Thông tin các nhân sai sót.  Không thể sửa thông tin cá nhân. |
| Thấp | Hiển thị kết quả không đúng form. Chức năng tính chạy chậm không ảnh hưởng lớn tới hệ thống.  Lỗi giao diện người dùng. |

6.2. Xử lý lỗi

Bước 1: Ghi nhận lỗi

* Mọi lỗi phát hiện trong quá trình kiểm thử phải ghi vào hệ thống theo dõi.
* Mô tả chi tiết lỗi, bao gồm các bước như cách thức tạo ra lỗi, bộ test tạo lỗi và kết quả của lỗi.

Bước 2: Phân loại lỗi

* Phân loại theo mức độ nghiệm trọng (Cao, trung bình, thấp) dựa trên sự ảnh hưởng của lỗi.

Bước 3: Phân công xử lý

* Lỗi được chuyển lại cho đội phát triển để chia nhiệm vụ cho từng thành viên trách nghiệm chức năng lỗi.

Bước 4: Sửa lỗi

* Đội phát triển tiến thành sửa các lỗi dựa trên báo cáo gửi vể.
* Sau khi sửa lỗi gửi lại cho nhóm Tester triển khai kiểm thử lại.

Bước 5: Kiểm thử lại

* Tiến hành kiểm tra đảm bảo không còn lỗi nào nào, và xem có phát sinh thêm lỗi nào khi sửa lỗi cũ không.

Bước 6: Báo cáo lỗi:

* Lỗi và trạng thái xử lý cập nhật liên tục cho các bên như quản lý dự án, tester, đội phát triển.
* Các báo cáo bao gồm số lượng lỗi, mức độ nghiêm trọng và trạng thái của lỗi đã sửa hay chưa.

Bước 7: Đánh giá và tổng kết

* Sau khi không còn lỗi, đội kiểm thử tiến hành đánh giá tổng thể về các lỗi và quá trình sửa lỗi.
* Đưa ra các bài học, cải thiện cho tương lai.

**VIII. Xây dựng các Testcase**

1. **Chức năng Quản lý giáo viên**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A close-up of a white sheet

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A close-up of a list

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Chức năng Quản lý bằng cấp**

**A screenshot of a computer

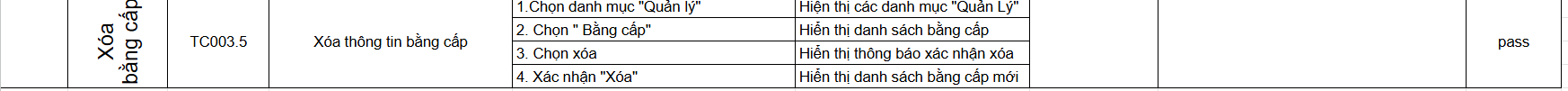
AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A close-up of a text

AI-generated content may be incorrect.**

****

1. **Chức năng Quản lý khoa**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Chức năng Thống kê giáo viên**

**A screenshot of a computer screen

AI-generated content may be incorrect.**

1. **Chức năng Quản lý học phần**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A white rectangular box with black text

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Chức năng Quản lý học kỳ**

**A screenshot of a computer screen

AI-generated content may be incorrect.**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**A screenshot of a computer program

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Chức năng Quản lý tkb / lớp học**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A white sheet with black lines and text

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

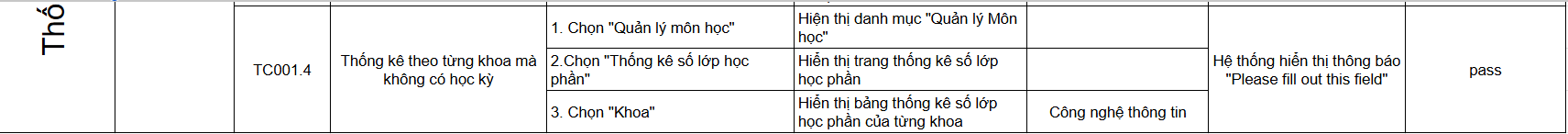
**A screenshot of a computer

AI-generated content may be incorrect.**

1. **Chức năng Thống kê số lớp học phần**

**A table of information with text

AI-generated content may be incorrect.**

****

1. **Chức năng Tính lương**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A white background with black and blue text

AI-generated content may be incorrect.**

1. **Chức năng Báo cáo**

**A screen shot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A close up of a list

AI-generated content may be incorrect.**

**A white sheet with black lines and text

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A white sheet with black text

AI-generated content may be incorrect.**

**A close up of a document

AI-generated content may be incorrect.**

**A screenshot of a computer

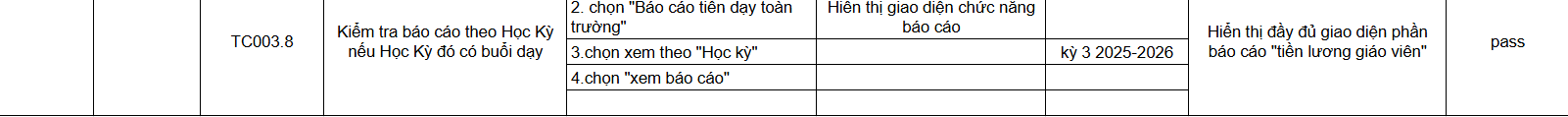
AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

**A screenshot of a computer

AI-generated content may be incorrect.**

****

**KẾT LUẬN**

Sau thời gian nghiên cứu và phát triển, đề án "Hệ thống quản lý giảng viên" đã hoàn thành với đầy đủ các chức năng cơ bản đáp ứng nhu cầu quản lý của nhà trường. Hệ thống đã xây dựng thành công cơ sở dữ liệu với 10 bảng chính, bao gồm quản lý thông tin giảng viên, lịch giảng dạy, bảng lương, và các thông tin liên quan đến khoa, môn học, học kỳ.

Về mặt kỹ thuật, hệ thống sử dụng MySQL làm hệ quản trị cơ sở dữ liệu với các ràng buộc toàn vẹn dữ liệu chặt chẽ, đảm bảo tính nhất quán và chính xác. Các trigger và function được tích hợp giúp tự động hóa nhiều quy trình như tạo tài khoản, tính toán hệ số lương, số tiết quy đổi, giảm thiểu sai sót từ thao tác thủ công.

Hệ thống đã đạt được các mục tiêu đề ra: số hóa quy trình quản lý giảng viên, giảm khối lượng công việc hành chính, tăng tính minh bạch trong việc tính toán lương và theo dõi giờ giảng. Giao diện phân quyền rõ ràng cho admin, giảng viên và kế toán giúp mỗi đối tượng dễ dàng thực hiện công việc của mình.

Tuy nhiên, hệ thống vẫn còn một số hạn chế cần khắc phục trong tương lai như: chưa có module báo cáo thống kê chi tiết, chưa tích hợp hệ thống thông báo tự động, chưa có tính năng xuất file Excel/PDF. Hướng phát triển tiếp theo sẽ tập trung vào việc bổ sung các tính năng này, tối ưu hiệu suất xử lý với lượng dữ liệu lớn, và cải thiện trải nghiệm người dùng.

Với những kết quả đạt được, đề án đã khẳng định tính khả thi của việc ứng dụng công nghệ thông tin vào quản lý giáo dục, góp phần hiện đại hóa công tác quản lý và nâng cao chất lượng giảng dạy tại nhà trường.

Sau quá trình kiểm thử, hệ thống đã được đánh giá là hoạt động ổn định và đáp ứng đúng các yêu cầu đề ra. Các chức năng chính như quản lý giảng viên, học kỳ, lớp học, lịch dạy, tính lương và báo cáo đều được kiểm tra kỹ lưỡng và cho kết quả chính xác.

Hệ thống có thể chạy tốt trên nhiều trình duyệt, xử lý dữ liệu đúng, bảo mật thông tin người dùng, và chịu được số lượng người truy cập cùng lúc. Một số lỗi nhỏ phát hiện trong quá trình kiểm thử đã được sửa ngay, không ảnh hưởng đến toàn bộ hoạt động của hệ thống.

Trong quá trình xây dựng và kiểm thử hệ thống quản lý giảng viên, nhóm phát triển đã có cơ hội tiếp cận và thực hành nhiều kỹ năng quan trọng như phân tích yêu cầu, thiết kế cơ sở dữ liệu, lập trình web, xử lý nghiệp vụ và viết kiểm thử. Dự án giúp nhóm hiểu rõ hơn về quy trình phát triển phần mềm thực tế và cách giải quyết các tình huống phát sinh trong quá trình triển khai.

Quan dự án này,nhóm không chỉ nâng cao kỹ năng lập trình và làm việc nhóm mà còn được rèn luyện tinh thần trách nhiệm, kiên trì và chủ động học hỏi.