

**BỘ GIÁO DỤC VÀ ĐÀO TẠO
TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT**

**BÁO CÁO TỔNG KẾT
NHIỆM VỤ KHOA HỌC VÀ CÔNG NGHỆ CẤP CƠ SỞ**

**XÂY DỰNG PHẦN MỀM SINH PHÁCH VÀ GHÉP ĐIỂM
CHO CÁC HỌC PHẦN KHẢO THÍ CỦA
TRƯỜNG ĐẠI HỌC MỎ - ĐỊA CHẤT**

Mã số: NV19-46

**Xác nhận của
Trường Đại học Mỏ - Địa chất**

Chủ nhiệm đề tài

ThS. Nông Thị Oanh

Hà Nội, 11/2020

**DANH SÁCH NHỮNG CÁN BỘ THAM GIA THỰC HIỆN VÀ
ĐƠN VỊ PHỐI HỢP CHÍNH**

Danh sách những cán bộ tham gia thực hiện:

1. Nông Thị Oanh, Khoa CNTT, Trường Đại học Mở-Địa chất
2. Lê Hồng Anh, Khoa CNTT, Trường Đại học Mở-Địa chất
3. Phạm Quang Hiên, Khoa CNTT, Trường Đại học Mở-Địa chất
4. Đặng Văn Nam, Khoa CNTT, Trường Đại học Mở-Địa chất
5. Nguyễn Thị Lan Hương, Phòng ĐBCLGD, Trường Đại học Mở-Địa chất

MỤC LỤC

DANH MỤC HÌNH VẼ.....	4
DANH MỤC BẢNG BIỂU	5
DANH MỤC CÁC CHỮ VIẾT TẮT.....	6
THÔNG TIN KẾT QUẢ NGHIÊN CỨU.....	7
1. Tình hình nghiên cứu và tính cấp thiết của đề tài	9
2. Mục tiêu của đề tài.....	9
3. Cách tiếp cận, đối tượng, phạm vi và phương pháp nghiên cứu.....	9
4. Đối tượng sử dụng và hiệu quả mang lại của đề tài	10
5. Nội dung thực hiện của đề tài	10
CHƯƠNG I. KHẢO SÁT HIỆN TRẠNG.....	11
1.1 Giới thiệu về Phòng Đảm bảo chất lượng giáo dục	11
1.2 Quy trình nghiệp vụ quản lý chấm thi và ghép điểm	13
1.3 Yêu cầu đối với hệ thống mới	14
1.3.1 Yêu cầu về dữ liệu	14
1.3.2 Yêu cầu về chức năng	14
1.3.3 Yêu cầu phi chức năng	15
CHƯƠNG 2. TỔNG QUAN VỀ MÔ HÌNH HÓA PHẦN MỀM	16
2.1 Khái niệm mô hình và mô hình hóa	16
2.1.1 Định nghĩa và ý nghĩa của mô hình.....	16
2.1.2. Khái niệm về mô hình hóa.....	17
2.1.3. Mục đích của mô hình hóa.....	17
2.2 Phương pháp mô hình hóa.....	18
2.3 Ngôn ngữ mô hình hóa	20
2.4 Kiến trúc hệ thống	22
2.5 Các pha chính trong quy trình phát triển phần mềm	23
CHƯƠNG 3. PHÂN TÍCH HỆ THỐNG.....	25
3.1 Mô hình hóa nghiệp vụ.....	25
3.1.1 Đặc tả các yêu cầu của hệ thống.....	25
3.1.2 Xây dựng biểu đồ ca sử dụng	28
3.2 Mô hình hóa sự tương tác	28
3.3 Mô hình hóa cấu trúc.....	31

3.3.1 Xác định các lớp đối tượng:	31
3.3.2 Xác định mối quan hệ giữa các lớp đối tượng.....	31
3.3.3 Các thuộc tính của các lớp đối tượng	32
3.3.4 Các phương thức của lớp đối tượng	32
CHƯƠNG 4: THIẾT KẾ HỆ THỐNG VÀ CÀI ĐẶT.....	33
4.1 Công cụ phát triển	33
4.1.1 Lựa chọn môi trường phát triển hệ thống	33
4.1.2 Giới thiệu về mô hình MVC.....	33
4.1.3 Giới thiệu về ASP.NET.....	36
4.1.4 Hệ quản trị cơ sở dữ liệu	40
4.2. Thiết kế cơ sở dữ liệu.....	41
4.3. Cài đặt hệ thống phần mềm.....	45
4.3.1 Dữ liệu thực nghiệm	45
4.3.2 Kết quả thực nghiệm	46
TÀI LIỆU THAM KHẢO.....	53

DANH MỤC HÌNH VẼ

Hình 2.1 Sơ đồ nguyên tắc hoạt động của phương pháp mô hình hóa

Hình 2.2 Các hướng mô hình hóa

Hình 2.3 Các thành phần cơ sở của UML

Hình 2.4 Mô hình hóa kiến trúc hệ thống

Hình 2.5 Quy trình xây dựng các mô hình cơ bản trong phân tích và thiết kế hệ thống hướng đối tượng bằng UML

Hình 3. 1. Mô hình Ca sử dụng tổng thể của hệ thống

Hình 3.2 Biểu đồ tuần tự ca sử dụng Sinh phách

Hình 3.3 Biểu đồ tuần tự ca sử dụng Ghép điểm

Hình 3.4 Biểu đồ lớp của hệ thống phần mềm

Hình 4.1 Công cụ, ngôn ngữ, hệ quản trị CSDL và mô hình hình phát triển hệ thống.

Hình 4.2 Mô hình MVC

Hình 4.3: Xây dựng CSDL của hệ thống

Hình 4.4. Giao diện chức năng Quản lý kỳ thi.

Hình 4.5. Giao diện chức năng Tạo mới kỳ thi.

Hình 4.6. Giao diện chức năng Upload file danh sách thí sinh dự thi

Hình 4.7. Thông báo khi sinh phách thành công

Hình 4.8. File phách sau khi in ra

Hình 4.9. Giao diện Upload điểm lên hệ thống

Hình 4.10. Giao diện một số thông báo khi Ghép điểm

Hình 4.11. Giao diện chức năng mở lại chức năng nhập danh sách vắng thi và ghép điểm thi

Hình 4.12. Giao diện chức năng Thống kê.

Hình 4.13. Giao diện chức năng nhập danh sách vắng thi và ghép điểm

Hình 4.14. Giao diện chức năng xử lý kỳ thi với vai trò chuyên viên/giảng viên

Hình 4.15. Giao diện chức năng nhập thí sinh vắng thi và ghép điểm từ file excel với vai trò chuyên viên/giảng viên

Hình 4.16. Giao diện chức năng nhập thí sinh vắng thi cho một túi bài thi với vai trò chuyên viên/giảng viên

DANH MỤC BẢNG BIỂU

Bảng 2.1 Danh sách các tác nhân và ca sử dụng

Bảng 4.1 Bảng QuyenHan

Bảng 4.2 Bảng TaiKhoan

Bảng 4.3 Bảng LichSuHoatDong

Bảng 4.4 Bảng KetQuaThi

Bảng 4.5 Bảng KyThi

Bảng 4.6 Bảng KyThiMonThi

Bảng 4.7 Bảng DanhSachThi

Bảng 4.8 Bảng DanhSachPhach

DANH MỤC CÁC CHỮ VIẾT TẮT

Từ/Cụm từ viết tắt	Ý nghĩa
UML	Unified Modeling Language
OMT	Object Modeling Technique
UC	UseCase
MVC	Model - View – Controller
GUI	Graphical User Interface
OOP	Object Oriented Programming
ASP	Active Server Pages
ADO	ActiveX Data Object
FSO	File System Object
CSDL	Cơ sở dữ liệu

THÔNG TIN KẾT QUẢ NGHIÊN CỨU**1. Thông tin chung:**

- Tên đề tài: ***Xây dựng phần mềm sinh phách và ghép điểm cho các học phần khảo thí của Trường Đại học Mở - Địa chất.***

- Mã số: **NV19-46**

- Chủ nhiệm: **ThS Nông Thị Oanh**

- Cơ quan chủ trì: **Trường Đại học Mở - địa chất**

- Thời gian thực hiện: **12 tháng (12/2019 – 11/2020)**

2. Mục tiêu:

Xây dựng được phần mềm quản lý sinh phách và ghép điểm đáp ứng yêu cầu đặt ra của phòng Đảm bảo chất lượng giáo dục.

3. Tính mới và sáng tạo:**4. Kết quả nghiên cứu:**

Đề tài đã hoàn thành theo đúng tiến độ về thời gian và hoàn thiện được những nội dung theo đúng thuyết minh đề tài được phê duyệt; Cụ thể đã đạt được các kết quả sau:

- Xây dựng được phần mềm quản lý sinh phách và ghép điểm trên nền tảng Web; Hệ thống đã được chạy thử nghiệm với các bộ dữ liệu thực tế. Các chức năng của phần mềm đã xây dựng phù hợp với mô hình nghiệp vụ của bài toán; Các chức năng đều hoạt động tốt, tốc độ nhanh, chính xác, đảm bảo độ an toàn và bảo mật; Giao diện phần mềm trực quan, dễ dàng sử dụng.
- Báo cáo tổng kết đề tài đầy đủ nội dung và đúng theo mẫu quy định.

5. Sản phẩm:

STT	Sản phẩm theo thuyết minh đề tài	Số lượng	Sản phẩm đã đạt được
1	Hệ thống phần mềm quản lý sinh phách và ghép điểm cho các học phần khảo thí của	01	Đáp ứng các yêu cầu của phòng Đảm bảo chất lượng giáo dục, trường Đại học Mở - Địa chất

	Trường Đại học Mỏ - Địa chất		
2	Tài liệu hướng dẫn sử dụng hệ thống phần mềm quản lý sinh phách và ghép điểm cho các học phần khảo thí của Trường Đại học Mỏ - Địa chất	01	Đáp ứng các yêu cầu của phòng Đảm bảo chất lượng giáo dục, trường Đại học Mỏ - Địa chất

6. Hiệu quả, phương thức chuyển giao kết quả nghiên cứu và khả năng áp dụng:

Hiệu quả: Hệ thống phần mềm sinh phách và ghép điểm được xây dựng, cài đặt và đang được sử dụng tại Phòng Đảm bảo chất lượng giáo dục, Trường Đại học Mỏ - Địa chất. Phần mềm cho phép sinh các phách ngẫu nhiên 5 số kèm theo mã vạch, không trùng lặp; cho phép sinh phách và quản lý tối đa 99999 phách ngẫu nhiên trong một kỳ thi. Phần mềm cho phép ghép phách-điểm của bài thi với dữ liệu thí sinh; cho phép cảnh báo trong trường hợp phách đang ghép thuộc về thí sinh vắng thi hoặc không tồn tại trong túi bài đang ghép phách-điểm.

Ngoài ra, phần mềm có một số hỗ trợ người dùng là chuyên viên, giảng viên thống kê tình trạng nhập điểm, ghép phách-điểm của các túi bài thi; hỗ trợ admin xuất dữ liệu có cấu trúc, đẩy điểm trực tiếp vào hệ thống quản lý đào tạo Edusoft của Nhà trường.

Phương thức chuyển giao: Cài đặt, triển khai và bàn giao phần mềm quản lý sinh phách và ghép điểm.

Khả năng và địa chỉ áp dụng: Phòng Đảm bảo chất lượng giáo dục, Trường Đại học Mỏ - Địa chất.

Cơ quan chủ trì
(ký, họ và tên, đóng dấu)

Ngày 29 tháng 10 năm 2020
Chủ nhiệm đề tài
(ký, họ và tên)

MỞ ĐẦU

1. Tình hình nghiên cứu và tính cấp thiết của đề tài

Ứng dụng Công nghệ thông tin trong quản lý đào tạo là một trong những vấn đề trọng tâm, được đặt lên hàng đầu và nhận được nhiều sự quan tâm của các cấp quản lý giáo dục nói chung và của Trường Đại học Mở - Địa chất nói riêng. Có rất nhiều phần mềm được xây dựng nhằm hỗ trợ các cán bộ xử lý công việc, nâng cao hiệu quả công việc trong công tác quản lý đào tạo. Đặc điểm chung của các phần mềm này là thương mại hóa nên giá thành khá cao và được xây dựng tổng thể cho hệ thống quản lý đào tạo nhiều trường đại học, cao đẳng.

Hiện tại, công việc chấm thi của Phòng Đảm bảo chất lượng giáo dục, Trường Đại học Mở - Địa chất gặp nhiều bất cập, việc sinh phách và ghép phách điểm cho các học phần khảo thí của Nhà trường đang tồn tại một số vấn đề như sau: số phách bị hạn chế và bị trùng lặp giữa các học phần khác nhau, mất rất nhiều thời gian để sinh phách, hệ thống chưa kiểm soát được các sai sót trong quá trình nhập điểm, gây ra nhiều khó khăn, nhầm lẫn trong việc ghép điểm theo phách và không thể nắm được tiến độ các công việc khảo thí.

Chính vì vậy, việc xây dựng một phần mềm sinh phách và ghép điểm cho Phòng Đảm bảo chất lượng giáo dục là một yêu cầu cấp thiết. Phần mềm này sẽ khắc phục được các nhược điểm của hệ thống hiện tại đang gặp phải, giúp cho việc quản lý sinh phách và ghép điểm được thực hiện một cách chuyên nghiệp, thuận tiện và chính xác, giúp nâng cao hiệu quả cho công tác quản lý đào tạo của Phòng Đảm bảo chất lượng giáo dục và Trường Đại học Mở - Địa chất.

2. Mục tiêu của đề tài

Xây dựng được phần mềm quản lý sinh phách và ghép điểm đáp ứng yêu cầu đặt ra của phòng Đảm bảo chất lượng giáo dục.

3. Cách tiếp cận, đối tượng, phạm vi và phương pháp nghiên cứu

Cách tiếp cận: Tiếp cận từ tổng thể đến chi tiết quy trình nghiệp vụ quản lý sinh phách và ghép điểm để xây dựng kiến trúc phần mềm.

Đối tượng nghiên cứu: Quy trình quản lý sinh phách và ghép điểm; Các yêu cầu mới đối với hệ thống sinh phách và ghép điểm.

Phạm vi nghiên cứu: Quản lý sinh phách và ghép điểm cho các học phần khảo thí của trường Đại học Mở - Địa chất.

Phương pháp nghiên cứu:

Phương pháp phỏng vấn người nghiệp vụ và người quản lý.

Phương pháp thực nghiệm xây dựng phần mềm sinh phách và ghép điểm.

4. Đối tượng sử dụng và hiệu quả mang lại của đề tài

Đối tượng sử dụng: Phòng Đảm bảo chất lượng giáo dục, Trường Đại học Mở - Địa chất.

Hiệu quả: Hệ thống phần mềm sinh phách và ghép điểm được xây dựng, cài đặt và đang được sử dụng tại Phòng Đảm bảo chất lượng giáo dục, Trường Đại học Mở - Địa chất. Phần mềm cho phép sinh các phách ngẫu nhiên 5 số kèm theo mã vạch, không trùng lặp; cho phép sinh phách và quản lý tối đa 99999 phách ngẫu nhiên trong một kỳ thi. Phần mềm cho phép ghép phách-điểm của bài thi với dữ liệu thí sinh; cho phép cảnh báo trong trường hợp phách đang ghép thuộc về thí sinh vắng thi hoặc không tồn tại trong túi bài đang ghép phách-điểm.

Ngoài ra, phần mềm có một số hỗ trợ người dùng là chuyên viên, giảng viên thống kê tình trạng nhập điểm, ghép phách-điểm của các túi bài thi; hỗ trợ admin xuất dữ liệu có cấu trúc, đẩy điểm trực tiếp vào hệ thống quản lý đào tạo Edusoft của Nhà trường.

5. Nội dung thực hiện của đề tài

- Khảo sát và đánh giá hiện trạng hệ thống quản lý sinh phách và ghép điểm các học phần khảo thí. Thu thập và phân loại các tài liệu, dữ liệu liên quan đến nghiệp vụ quản lý sinh phách và ghép điểm.
- Phân tích và thiết kế hệ thống quản lý sinh phách và ghép điểm.
- Lập trình mô-đun quản lý sinh phách. Lập trình mô-đun quản lý ghép điểm và mô-đun nhập điểm.
- Cài đặt và kiểm thử phần mềm quản lý sinh phách và ghép điểm.
- Thực nghiệm, đánh giá các kết quả của phần mềm và viết báo cáo tổng kết.

CHƯƠNG I. KHẢO SÁT HIỆN TRẠNG

1.1 Giới thiệu về Phòng Đảm bảo chất lượng giáo dục

Chức năng: Xây dựng và vận hành các quy trình khảo thí và đảm bảo chất lượng của Trường Đại học Mở - Địa chất.

Nhiệm vụ:

a. Công tác khảo thí:

- Chủ trì và phối hợp các đơn vị liên quan xây dựng quy trình kiểm tra và thi kết thúc học phần, thi tốt nghiệp, đánh giá đồ án tốt nghiệp thuộc các hệ đào tạo của Trường... theo đúng các quy chế và quy định của Bộ Giáo dục và Đào tạo.

- Tổ chức thực hiện đúng thủ tục, quy trình đối với thi học phần, bảo vệ tốt nghiệp thuộc các hệ đào tạo của Trường; quản lý, kiểm tra và giám sát việc tổ chức ra đề thi, sao in đề thi; lập lịch thi và tổ chức thi đối với các học phần được quy định; chủ trì thực hiện công tác chấm thi, làm phách bài thi, báo điểm và lưu trữ, phân tích, xử lý kết quả thi; chủ trì thẩm định và bảo đảm tính chính xác của việc chấm thi.

- Tổ chức xây dựng và quản lý hệ thống ngân hàng đề thi, đáp án thi; cải tiến và phát triển các hình thức thi phù hợp với yêu cầu đào tạo và đánh giá chất lượng học tập của các ngành, các hệ, các bậc đào tạo.

- Tham gia hội đồng xử lý kết quả học tập cuối mỗi học kỳ và hội đồng xét tốt nghiệp.

b. Công tác bảo đảm chất lượng giáo dục:

- Chủ trì xây dựng các mục tiêu, chính sách, quy định, quy trình và các kế hoạch cụ thể của hoạt động tự đánh giá, đánh giá ngoài, kiểm định và đảm bảo chất lượng giáo dục trong Trường.

- Chủ trì, chủ động phối hợp với các đơn vị liên quan để triển khai công tác đánh giá Trường, đánh giá chương trình đào tạo theo đúng các yêu cầu và tiêu chuẩn của hệ thống kiểm định; xây dựng các kế hoạch, hướng dẫn một cách có hệ thống và đồng bộ công tác tự đánh giá và đánh giá ngoài; đăng ký và triển khai công tác kiểm định chất lượng với các tổ chức kiểm định chất lượng giáo dục trong nước và quốc tế.

- Chủ trì, phối hợp với các đơn vị có liên quan cập nhật báo cáo tự đánh giá, đảm bảo chất lượng công tác đánh giá hàng năm.

- Chủ trì, phối hợp với các đơn vị liên quan để nghiên cứu và xây dựng hệ thống công cụ thông tin phản hồi kết quả giáo dục đối với các trình độ, các hệ đào tạo gồm

chương trình đào tạo, học phần, giảng viên, người học, việc làm sau khi tốt nghiệp của người học...

- Là đầu mối tổ chức tập huấn cho cán bộ, giảng viên trong Trường về công tác khảo thí và kiểm định chất lượng; làm dịch vụ khảo thí, tư vấn công tác đảm bảo chất lượng cho các tổ chức, cá nhân có nhu cầu.

c. Nhiệm vụ khác:

- Tham mưu, giúp việc cho lãnh đạo Trường triển khai các hoạt động chuyên môn và quản lý liên quan đến chức năng, nhiệm vụ được quy định cụ thể cho mỗi đơn vị trong Quy định này, nhằm hoàn thành tốt nhiệm vụ chính trị của Trường.

- Thường xuyên rà soát và cập nhật các văn bản pháp luật để triển khai thực hiện các nhiệm vụ của đơn vị theo đúng quy định; chủ động, chủ trì, phối hợp với các đơn vị liên quan xây dựng, sửa đổi, bổ sung các quy định nội bộ trình Hiệu trưởng ban hành nhằm chuẩn hóa và nâng cao hiệu quả công tác quản lý của Trường liên quan đến chức năng nhiệm vụ của đơn vị.

- Chủ động ra các quyết định quản lý, điều hành nếu được ủy quyền.

- Đề xuất với Hiệu trưởng kế hoạch năm, quý, tháng, tuần; phương án thực hiện các kế hoạch; và giải quyết các công việc đối nội, đối ngoại theo chức năng, nhiệm vụ được giao.

- Quản lý, theo dõi, tổng hợp và giải quyết công tác chuyên môn nghiệp vụ theo chức năng, nhiệm vụ và quyền được giao.

- Thực hiện công tác lưu trữ theo quy định tất cả các văn bản do đơn vị làm đầu mối thực hiện và các văn bản khác (bản cứng và bản mềm) liên quan đến chức năng nhiệm vụ của đơn vị.

- Hướng dẫn, giúp đỡ, kiểm tra, đôn đốc việc thực hiện các công tác nghiệp vụ đối với các đơn vị thuộc Trường trong phạm vi chức năng, nhiệm vụ được giao.

- Phối hợp chặt chẽ với các đơn vị khác để thực hiện tốt công việc của Trường. Quản lý trực tiếp và toàn diện cán bộ, viên chức và người lao động của đơn vị.

- Xây dựng quy hoạch, kế hoạch bồi dưỡng chuyên môn, nghiệp vụ, ngoại ngữ, tin học cho cán bộ, viên chức trong đơn vị nhằm nâng cao hiệu quả phục vụ và đáp ứng các yêu cầu phát triển của Trường.

- Quản lý, sử dụng tài sản được giao một cách an toàn, hiệu quả và tiết kiệm.

- Xây dựng và văn bản hóa quy trình giải quyết các công việc được giao theo chức năng nhiệm vụ của đơn vị.

- Chủ trì và làm đầu mối phối hợp với Phòng Quan hệ Công chúng và Doanh nghiệp tổ chức các sự kiện của Trường liên quan đến chức năng nhiệm vụ của đơn vị.

- Ngoài nhiệm vụ chung được quy định tại Điều này, các đơn vị chức năng còn có trách nhiệm thực hiện các nhiệm vụ cụ thể được quy định riêng cho mỗi đơn vị trong Quy định này và thực hiện các nhiệm vụ khác do Hội đồng trường (HĐT) và Ban Giám hiệu (BGH) chỉ đạo.

1.2 Quy trình nghiệp vụ quản lý chấm thi và ghép điểm

Tính đến tháng 4 năm 2020, toàn Trường có 795 học phần đăng ký khảo thí. Trong năm học 2019-2020, Nhà trường đã tổ chức thi đối với 1403 môn học (khoảng 110223 lượt thi). Số học phần khảo thí chiếm 51-61%, trong khi đó số bài thi của các học phần khảo thí chiếm tới 63-68%. So với năm học trước thì tỷ lệ học phần được tổ chức khảo thí tăng khoảng 3% và sẽ tiếp tục tăng trong những năm học tới.

Đối với các học phần khảo thí, sau khi cho các nhóm học phần thi xong, Phòng ĐBCLGD sẽ tiến hành làm phách, dán phách lên túi bài thi và bài thi; Các giảng viên sẽ chấm điểm và nhập điểm; Sau đó Phòng ĐBCLGD ghép phách và điểm cho các túi bài thi của các học phần khảo thí và đẩy điểm lên hệ thống EduSoft.

Công việc sinh phách và ghép điểm mặc dù đã có phần mềm hỗ trợ nhưng còn tồn tại những vấn đề sau:

- Số phách sinh ra bị hạn chế và bị trùng lặp giữa các học phần khác nhau.
- Thời gian sinh phách lâu, với những học kỳ có số lượng lượt thi lớn có thể phải mất 4-5 tiếng mới xong.
- Hệ thống chưa kiểm soát được các sai sót trong quá trình nhập điểm (ví dụ đầu quét barcode đọc phách sai), gây ra nhiều khó khăn, nhầm lẫn trong việc ghép điểm theo phách.
- Không thể nắm được tiến độ các công việc khảo thí như tiến độ nhập điểm, tiến độ ghép phách – điểm của các túi bài thi,...

Với những nhược điểm nêu trên của hệ thống, các cán bộ phòng ĐBCLGD thường xuyên gặp phải khó khăn, tốn nhiều thời gian khi thực hiện các quy trình nghiệp vụ khảo thí. Vì vậy, việc xây dựng một phần mềm sinh phách và ghép điểm cho Phòng Đảm bảo chất lượng giáo dục là một yêu cầu cấp thiết. Phần mềm này sẽ khắc phục được các nhược điểm của hệ thống hiện tại đang gặp phải, giúp nâng cao hiệu quả cho công tác quản lý đào tạo của Phòng ĐBCLGD.

1.3 Yêu cầu đối với hệ thống mới

1.3.1 Yêu cầu về dữ liệu

- Dữ liệu đầu vào:
 - ✓ Danh sách thí sinh tham dự kỳ thi (theo mẫu): để tiến hành đưa dữ liệu vào hệ thống xử lý và sinh phách.
 - ✓ File điểm đầu ra của phần mềm đọc điểm từ phách bài thi: sử dụng để làm dữ liệu đầu vào trên hệ thống để tiến hành ghép điểm và quản lý điểm.
 - ✓ Danh sách thí sinh vắng thi của các môn thi.
 - ✓ Danh sách thí sinh phúc khảo bài thi.
- Dữ liệu đầu ra:
 - ✓ File phách: với dữ liệu phách được sinh tự động ứng với dữ liệu sinh viên tham dự kỳ thi.
 - ✓ Tổng hợp và xuất dữ liệu theo yêu cầu của quản trị: Tiến độ ghép điểm, lịch sử làm việc...
 - ✓ File điểm: để làm dữ liệu đầu vào cho hệ thống Edusoft.
 - ✓ File điểm phúc khảo: để ghép điểm sau khi phúc khảo bài thi của sinh viên.

1.3.2 Yêu cầu về chức năng

- Phần mềm quản lý sinh phách và ghép điểm bao gồm các chức năng chính sau:
 - ✓ **Quản lý kỳ thi:**
 - Cho phép khởi tạo thông tin kỳ thi, quản lý thông tin kỳ thi.
 - Quản lý danh sách sinh viên dự thi của kỳ thi.
 - Sinh phách cho kỳ thi.
 - Quản lý tình trạng môn thi, túi bài thi: quản lý tình trạng ghép điểm của từng túi bài thi.
 - ✓ **Quản lý người dùng:** quản lý và phân quyền tài khoản
 - ✓ **Quản lý lịch sử hoạt động:** Thống kê lịch sử hoạt động của hệ thống khi có tương tác với chức năng hệ thống.
 - ✓ **Thống kê:** Tổng hợp và tìm kiếm dữ liệu về tiến độ ghép điểm của từng môn thi, túi bài thi
 - ✓ **Ghép điểm:** hệ thống sẽ phát hiện ra các phách không hợp lệ của môn thi, kỳ thi và báo cho người dùng biết để kiểm tra lại. Hệ thống sẽ thực hiện tự động ghép điểm dựa trên dữ liệu đầu vào được quét khi nhập điểm khi và chỉ khi các phách hợp lệ đối với kỳ thi, môn thi
 - ✓ **Phúc khảo:** quản lý danh sách sinh viên muốn phúc khảo bài thi và ghép điểm sau khi phúc khảo.

- ✓ **Xuất bảng điểm:** Xuất dữ liệu bảng điểm theo yêu cầu theo định dạng để làm file dữ liệu đầu vào cho hệ thống edusoft.
- ✓ **Tìm kiếm:** Theo các tiêu chí: mã môn học, túi bài thi, mã sinh viên, mã giảng viên.
- Các yêu cầu khác:
 - ✓ Hệ thống tự động backup dữ liệu theo lịch định kỳ.
 - ✓ Về giao diện: hình thức và màu sắc đơn giản, sử dụng ngôn ngữ là tiếng Việt, có thể dùng các biểu tượng thông dụng; có hướng dẫn sử dụng rõ ràng và dễ sử dụng; sử dụng được trên nhiều máy tính.
 - ✓ Về thao tác:
 - Các bước khởi tạo, upload dữ liệu và thực hiện lệnh không quá ba thao tác.
 - Các bước thực hiện phải đưa ra thông báo để người dùng lựa chọn các bước tiếp theo.

1.3.3 Yêu cầu phi chức năng

- **Hiệu suất:**
 - ✓ Yêu cầu về thời gian: Thời gian cho phép để hệ thống phản hồi lại thông tin đã tiếp nhận yêu cầu xử lý từ phía người sử dụng là 3 giây; thời gian cho phép để hiển thị đầy đủ thông tin là 5 giây; thời gian cho phép để gửi kết quả tìm kiếm thông tin là 10 giây;
 - ✓ Tài nguyên sử dụng: Tài nguyên lưu trữ chiếm dụng của hệ thống trong trạng thái hoạt động bình thường không được phép lớn hơn 80% tài nguyên lưu trữ được phép sử dụng;
 - ✓ Công suất tối đa: Đảm bảo khả năng 50 người sử dụng dịch vụ truy cập đồng thời; đảm bảo khả năng thực hiện 10 giao dịch/giây;
- **An toàn thông tin:**
 - ✓ Bảo mật: Hệ thống hỗ trợ khả năng cấu hình ứng dụng đảm bảo khả năng bảo mật nhiều mức (trình diễn, nghiệp vụ, truy cập dữ liệu); sử dụng kênh kết nối an toàn trong việc truy cập máy chủ ứng dụng và công cụ quản lý;
 - ✓ Toàn vẹn: Không cho phép tên tài khoản và mật khẩu đăng nhập được truyền trên môi trường mạng mà không được mã hóa;
 - ✓ Xác thực: Hệ thống hỗ trợ việc xác thực bằng tên tài khoản, mật khẩu đăng nhập.

CHƯƠNG 2. TỔNG QUAN VỀ MÔ HÌNH HÓA PHẦN MỀM

2.1 Khái niệm mô hình và mô hình hóa

2.1.1 Định nghĩa và ý nghĩa của mô hình

Mô hình (model) là một dạng trừu tượng hóa của một hệ thống thực. Mô hình chính là một hình ảnh (một biểu diễn) của một hệ thống thực, được diễn tả ở một mức độ trừu tượng nào đó, theo một quan điểm nào đó, theo một hình thức (hiểu được) nào đó như phương trình, bảng, đồ thị,... Mô hình có xu hướng dạng sơ đồ (diagrams) tức là đồ thị gồm các nút và cung.

Mô hình mô tả bản chất của một vấn đề hoặc một cấu trúc phức tạp bằng cách loại bỏ những chi tiết không quan trọng, làm cho bài toán trở nên dễ hiểu và dễ nắm bắt hơn. Để xây dựng một hệ thống phức tạp, những người phát triển phải trừu tượng hóa những khía cạnh khác nhau của hệ thống, xây dựng các mô hình bằng cách sử dụng các ký hiệu một cách rõ ràng, cẩn thận, kiểm tra xem các mô hình đã thỏa mãn các yêu cầu của hệ thống chưa và dần dần thêm vào các chi tiết để có thể chuyển đổi từ mô hình sang một cài đặt cụ thể.

Trong phân tích thiết kế hệ thống, các mô hình được tạo ra để trừu tượng hóa các đặc điểm quan trọng của các hệ thống thế giới thực. Trong lĩnh vực phần mềm, mô hình là kế hoạch chi tiết của hệ thống, là bức tranh hay mô tả của vấn đề đang được cố gắng giải quyết hay biểu diễn. Mô hình còn có thể là mô tả chính giải pháp, có thể dùng biểu tượng thay cho đối tượng thực. Tiến trình phát triển phần mềm là làm giảm một số đặc trưng của đối tượng để hình thành mô hình, làm giảm độ phức tạp bằng mô hình trừu tượng.

Mọi mô hình đều phản ánh hệ thống theo một khung nhìn trừu tượng hóa nào đó. Có 2 khung nhìn chính:

Khung nhìn logic: tập trung mô tả bản chất của hệ thống và mục đích hoạt động của hệ thống, bỏ qua các yếu tố về tổ chức thực hiện, về biện pháp cài đặt. Nói cách khác, mô hình logic trả lời các câu hỏi “là gì?” (What?)- như là chức năng gì, thông tin gì, ứng xử gì, bỏ qua các câu hỏi “như thế nào?” (How?). Ở tầng này, người ta tiến hành trên 3 phương diện xử lý, dữ liệu và động thái hệ thống.

Khung nhìn vật lý: Trả lời câu hỏi “như thế nào”, “ai làm”, “làm ở đâu”, “khi nào làm”, quan tâm đến các mặt như: phương pháp, biện pháp, công cụ, tác nhân, địa điểm, thời gian, hiệu năng,... Ở tầng này yêu cầu cần làm rõ kiến trúc vật lý của hệ thống.

Việc xây dựng mô hình có một ý nghĩa rất lớn trong quá trình phát triển phần mềm:

Mô hình giúp ta hiểu và thực hiện được sự trừu tượng, tổng quát hóa các khái niệm cơ sở để giảm thiểu độ phức tạp của hệ thống.

Qua mô hình chúng ta biết được hệ thống gồm những gì? và chúng hoạt động như thế nào.

Mô hình giúp chúng ta quan sát được hệ thống như nó vốn có trong thực tế hoặc nó phải có như ta mong muốn. Muốn hiểu và phát triển được hệ thống phần mềm theo yêu cầu thực tế thì ta phải quan sát nó theo nhiều góc nhìn khác nhau: theo chức năng sử dụng, theo các thành phần logic.

Mô hình thể hiện rõ các đặc tả cấu trúc và hành vi của hệ thống.

Đồng thời, mô hình là cơ sở để trao đổi, ghi lại những quyết định đã thực hiện trong nhóm tham gia dự án phát triển phần mềm. Mọi quan sát, mọi kết quả phân tích đều được ghi lại chi tiết để phục vụ cho cả quá trình phát triển phần mềm.

2.1.2. Khái niệm về mô hình hóa

Việc dùng mô hình để nhận thức và diễn tả một hệ thống được gọi là mô hình hóa. Trong khoa học máy tính, mô hình hóa bắt đầu từ việc mô tả vấn đề, sau đó là mô tả giải pháp vấn đề. Các hoạt động này còn được gọi là *phân tích và thiết kế*. Khi khảo sát hệ thống, người ta sưu tập yêu cầu cho hệ thống, ánh xạ chúng thành yêu cầu phần mềm, từ đó phát sinh mã trình, đảm bảo yêu cầu phù hợp với mã trình và khả năng chuyển đổi mã trình ngược lại thành yêu cầu. Tiến trình đó chính là thực hiện mô hình hóa.

Người ta có thể lấy thông tin từ mô hình và hiển thị đồ họa bằng tập các phần tử đồ họa chuẩn. Tiến trình đó được gọi là mô hình hóa trực quan. Tiêu chuẩn là cốt lõi để thực hiện một trong các lợi thế của mô hình trực quan, đó là vấn đề giao tiếp. Giao tiếp giữa tất cả các người tham gia dự án là mục tiêu quan trọng nhất của mô hình hóa trực quan. Tương tác này có thể được thực hiện bằng văn bản nhưng tốt hơn là bằng đồ họa. Các nhà tin học đã rất cố gắng để hình thành ký pháp mô hình hóa trực quan. Một số ký pháp quen thuộc là của Booch, OMT và UML. Phần mềm công cụ Rational Rose 2000 đã trợ giúp tốt vấn đề này.

2.1.3. Mục đích của mô hình hóa

Mục đích của mô hình hóa là để hiểu, để làm phương tiện trao đổi, để hoàn chỉnh hệ thống, hay nói rõ hơn:

Mô hình hóa giúp ta hiểu và thực hiện được sự trừu tượng, tổng quát hóa các khái niệm cơ sở để giảm thiểu độ phức tạp của hệ thống. Qua mô hình chúng ta biết được hệ thống gồm những gì? và chúng hoạt động như thế nào?.

Michael B., William P. [17] từng nói: “*Hiểu tức là mô hình hóa*”. Do vậy, quá trình phát triển phần mềm chẳng qua là quá trình nhận thức và mô tả lại hệ thống đó. Đó cũng là quá trình thiết lập, sử dụng và biến đổi các mô hình. Vậy, có một mô hình đúng sẽ giúp ta làm sáng tỏ những vấn đề phức tạp và cho ta cái nhìn thấu đáo về vấn đề cần giải quyết.

Mô hình hóa giúp chúng ta quan sát được hệ thống như nó vốn có trong thực tế hoặc nó phải có như ta mong muốn. Muốn hiểu và phát triển được hệ thống phần mềm theo yêu cầu thực tế thì ta phải quan sát nó theo nhiều góc nhìn khác nhau: theo chức năng sử dụng, theo các thành phần logic, theo phương diện triển khai,...

Mô hình hóa cho phép chúng ta đặc tả được cấu trúc và hành vi của hệ thống:

Đảm bảo hệ thống đạt được mục đích đã xác định trước. Mọi mô hình đều đơn giản hóa thế giới thực, nhưng phải đảm bảo sự đơn giản đó không loại bỏ đi những yếu tố quan trọng.

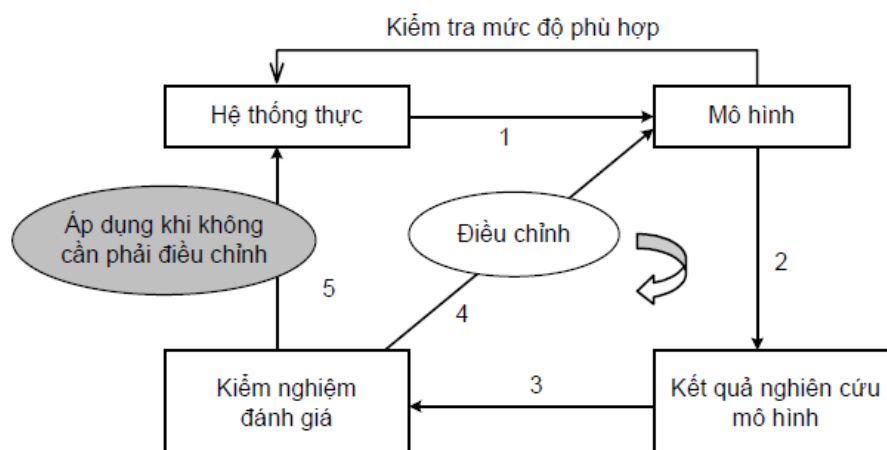
Kiểm tra được các quy định về cú pháp, ngữ nghĩa về tính chặt chẽ và đầy đủ của mô hình, khẳng định được tính đúng đắn của thiết kế, phù hợp với yêu cầu của khách hàng. Nghĩa là, mô hình hóa là quá trình hoàn thiện và tiến hóa liên tục.

Mô hình hóa là nhằm tạo ra khuôn mẫu (template) và hướng dẫn cách xây dựng hệ thống; cho phép thử nghiệm, mô phỏng và thực hiện, hoàn thiện theo mô hình.

2.2 Phương pháp mô hình hóa

Phương pháp là cách trực tiếp cấu trúc hóa sự suy nghĩ và hành động của con người. Phương pháp cho người sử dụng biết phải làm gì? làm như thế nào? khi nào? và tại sao? (mục đích của hành động). Phương pháp chứa các *mô hình (model)*, các mô hình được dùng để mô tả những gì sử dụng cho việc truyền đạt kết quả trong quá trình sử dụng phương pháp [2].

Phương pháp mô hình hóa là một trong những phương pháp quan trọng nhất để nghiên cứu hệ thống. Ý tưởng của phương pháp mô hình hóa là không nghiên cứu trực tiếp đối tượng mà thông qua việc nghiên cứu một đối tượng khác “trưng tự” hay là “hình ảnh” của nó mà có thể sử dụng được các công cụ khoa học. Kết quả nghiên cứu trên mô hình được áp dụng vào cho đối tượng thực tế.



Hình 1.1 Sơ đồ nguyên tắc hoạt động của phương pháp mô hình hóa

Việc mô hình hóa thể hiện một tiến độ triển khai, bao gồm các bước đi lần lượt, các hoạt động cần làm. Mô hình hóa giữ một vai trò đặc biệt quan trọng khi nó trở thành một công cụ trợ giúp. Đó là cơ sở tạo phần mềm giúp cho việc triển khai hệ thống thực hiện đúng và nhanh.

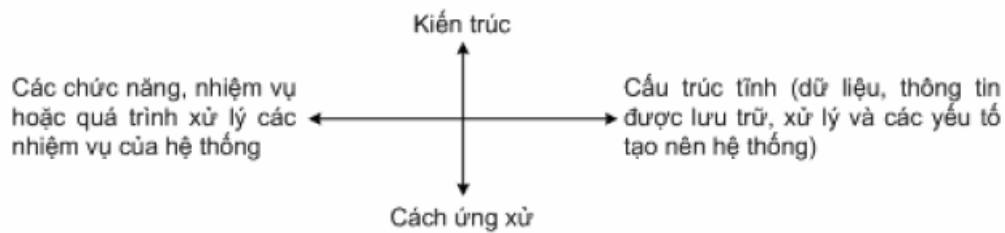
Như vậy, mô hình hóa là biểu diễn hệ thống dưới các dạng hình thức dễ hiểu như sơ đồ, đồ thị, công thức... Mô hình hóa giúp hiểu rõ bài toán, trao đổi thông tin giữa những người liên quan như khách hàng, chuyên gia, người phân tích, người thiết kế. Mô hình giúp cho việc xác định các yêu cầu tốt hơn, thiết kế rõ ràng hơn và khả năng bảo trì hệ thống cao hơn.

Mô hình hóa là phần trung tâm trong các công việc, các hoạt động để dẫn tới một phần mềm tốt. Chúng ta xây dựng mô hình để trao đổi, bàn bạc về cấu trúc và hành vi mong muốn của hệ thống. Đồng thời xây dựng mô hình để trực quan hóa và kiểm soát kiến trúc của hệ thống.

Mô hình hóa có thể mô tả các cấu trúc, nhấn mạnh về mặt tổ chức của hệ thống hoặc nó có thể mô tả các hành vi, tập trung vào mặt động của hệ thống.

Tóm lại, phương pháp mô hình hóa là phương pháp tiên tiến. Nó giúp chúng ta hiểu rõ hơn về hệ thống mà chúng ta đang xây dựng, tạo ra cơ hội để có thể đơn giản hóa và tái sử dụng. Ngoài ra phương pháp mô hình hóa còn giúp chúng ta dễ dàng kiểm soát rủi ro.

Theo Booch G., Rumbaugh J. and Jacobson I., mô hình hóa một hệ thống phải thực hiện theo cả bốn hướng [2]:



Hình 1.2 Các hướng mô hình hóa

Hướng của điểm xuất phát sẽ kéo theo phương pháp cần lựa chọn để phát triển phần mềm. Nếu ta bắt đầu từ bên trái, nghĩa là tập trung vào chức năng để phân tích thì chúng ta thực hiện, phát triển phần mềm theo cách tiếp cận hướng chức năng. Ngược lại, nếu bắt đầu từ bên phải, nghĩa là dựa vào dữ liệu là chính thì chúng ta sử dụng phương pháp hướng đối tượng.

2.3 Ngôn ngữ mô hình hóa

Mô hình được biểu diễn theo một ngôn ngữ mô hình hóa. Ngôn ngữ mô hình hóa bao gồm các ký hiệu (những biểu tượng được dùng trong mô hình) và một tập các quy tắc chỉ cách sử dụng chúng. Các quy tắc này bao gồm:

Cú pháp (Syntactic): cho biết hình dạng các biểu tượng và cách kết hợp chúng trong ngôn ngữ.

Ngữ nghĩa (Semantic): cho biết ý nghĩa của mỗi biểu tượng, chúng được hiểu thế nào khi nằm trong hoặc không nằm trong ngữ cảnh của các biểu tượng khác.

Mục đích (Pragmatic): định nghĩa ý nghĩa của biểu tượng để sao cho mục đích của mô hình được thể hiện và mọi người có thể hiểu được.

UML (Unified Modeling Language) là ngôn ngữ trực quan được dùng trong quy trình phát triển các hệ thống phần mềm. Nó là một *ngôn ngữ đặc tả hình thức (formal specification language)*. UML có một tập các phần tử và một tập các quy tắc riêng. Hầu hết các phần tử của UML là các đối tượng đồ họa như đường thẳng, hình chữ nhật, hình oval,... và thường có nhãn kèm theo để tăng thông tin. Các quy tắc trong UML được mô tả trong đặc tả UML như cú pháp trừu tượng (được biểu diễn như các sơ đồ và ngôn ngữ tự nhiên), quy tắc hình thức (nằm trong ngôn ngữ ràng buộc đối tượng) và ngữ nghĩa. Quy tắc xác định cách kết hợp giữa các phần tử.

Do UML là ngôn ngữ mô hình hóa chuẩn, ngôn ngữ mô hình đồ họa, trực quan, vừa đặc tả vừa có cấu trúc, đồng thời lại là ngôn ngữ làm tài liệu nên đối với việc phát triển phần mềm hướng đối tượng, UML đặc biệt có khả năng sau:

Cho phép mô tả toàn bộ các sản phẩm phân tích và thiết kế;

Trợ giúp việc tự động hóa quá trình thiết kế trên máy tính;

Trợ giúp việc dịch xuôi và dịch ngược các thiết kế sang mã nguồn của ngôn ngữ lập trình và CSDL.

UML được hợp nhất từ nhiều thành tựu và kinh nghiệm nghiên cứu và triển khai nhờ:

Cách tiếp cận của Grady Booch (Booch Approach),

Kỹ thuật mô hình hóa đối tượng (OMT: Object Modeling Technique) của James Rumbaugh,

Công nghệ phần mềm hướng đối tượng (OOSE: Object-Oriented Software Engineering) của Ivar Jacobson,

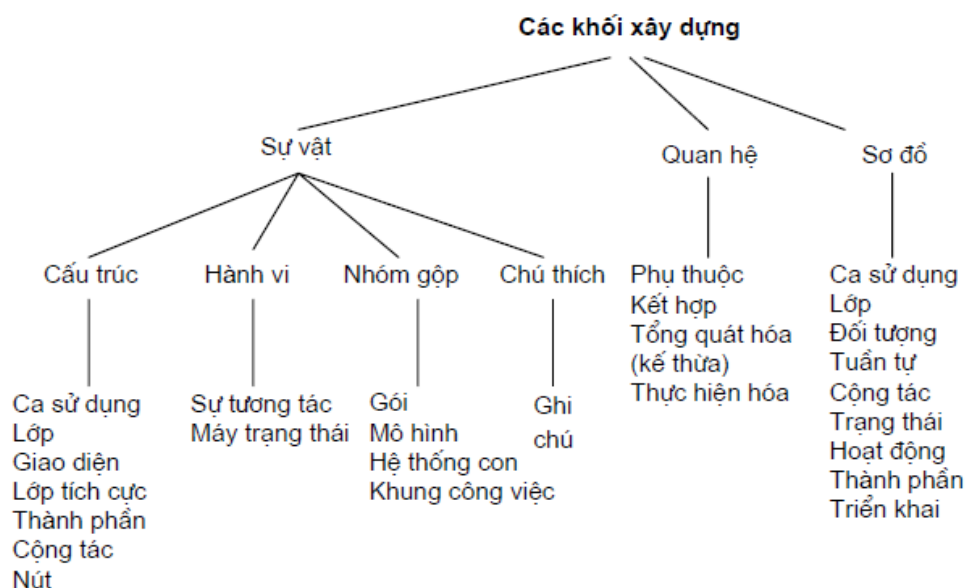
Thống nhất được nhiều ký pháp, khái niệm của nhiều phương pháp khác nhau. Quá trình hình thành UML bắt đầu từ ngôn ngữ Ada (Booch) trước năm 1990.

Mục đích chính của UML nhằm vào các hoạt động sau:

Mô hình hóa được các hệ thống và sử dụng được tất cả các khái niệm hướng đối tượng một cách thống nhất.

Cho phép đặc tả, hỗ trợ để đặc tả tường minh mối quan hệ giữa các khái niệm cơ bản trong hệ thống, đồng thời mô tả được mọi trạng thái hoạt động của hệ thống đối tượng. Nghĩa là cho phép mô tả được cả mô hình tĩnh lẫn mô hình động một cách đầy đủ và trực quan.

Tận dụng được những khả năng sử dụng lại và kế thừa ở phạm vi rộng để xây dựng được những hệ thống phức tạp và nhạy cảm như: các hệ thống động, hệ thống thời gian thực, hệ thống nhúng thời gian thực...



Hình 1.3 Các thành phần cơ sở của UML

2.4 Kiến trúc hệ thống

Khi mô hình hóa hệ thống, chúng ta cần quan tâm tới một loạt các khía cạnh khác nhau như về mặt chức năng (cấu trúc tĩnh của nó cũng như các tương tác động), về mặt phi chức năng (yêu cầu về thời gian, về độ đáng tin cậy, về quá trình thực thi) cũng như về khía cạnh tổ chức (tổ chức làm việc, ánh xạ nó vào các module..). Vì vậy một hệ thống khi mô hình hóa thường được xem xét trên một loạt các phương diện khác nhau, mỗi phương diện sẽ thể hiện một bức ảnh ánh xạ của toàn bộ hệ thống và chỉ ra một khía cạnh riêng của hệ thống.

Trong UML, kiến trúc của hệ thống phần mềm chuyên sâu (cho ta một cách nhìn khái quát nhất về hệ thống phần mềm ở các góc độ khác nhau) được mô tả theo 5 loại *khung nhìn* (views) khác nhau. Mỗi khung nhìn phản ánh về một khía cạnh của tổ chức và cấu trúc của hệ thống, tập trung vào từng mặt cụ thể giúp cho ta hiểu và sử dụng hệ thống tốt nhất. Mỗi khung nhìn thường được thể hiện trong một sơ đồ nhất định.



Hình 1.4 Mô hình hóa kiến trúc hệ thống

Khung nhìn ca sử dụng (Use case view): chứa các tác nhân, ca sử dụng, sơ đồ ca sử dụng (khía cạnh tĩnh của khung nhìn) trong hệ thống. Chúng cũng có thể bao gồm vài sơ đồ trình tự, sơ đồ cộng tác (khía cạnh động của khung nhìn) và gói. Khung nhìn ca sử dụng tập trung vào mức cao của cái hệ thống sẽ làm, không quan tâm đến hệ thống làm như thế nào.

Khung nhìn thiết kế logic (design view) tập trung vào hệ thống cài đặt hành vi trong ca sử dụng như thế nào. Nó bao gồm các lớp, sơ đồ lớp, sơ đồ đối tượng (khía cạnh tĩnh của khung nhìn), sơ đồ tương tác, sơ đồ hoạt động, sơ đồ biến đổi trạng thái (khía cạnh động của khung nhìn) và các gói. Trong khung nhìn, người ta quan tâm đến hai lớp quan trọng là lớp phân tích (lớp biên, lớp điều khiển, lớp dữ liệu) và lớp thiết kế (lớp phụ thuộc vào ngôn ngữ). Khung nhìn này tập trung vào cấu trúc logic của hệ thống nên còn được gọi là *khung nhìn logic (logical view)*. Khung nhìn này tập trung vào cấu trúc của hệ thống.

Nhờ nó, người ta nhận ra các bộ phận cơ bản cấu thành hệ thống thể hiện mọi quá trình trao đổi, xử lý thông tin cơ bản trong hệ thống.

Khung nhìn tiến trình/tương tranh (process view) biểu diễn sự phân chia các luồng thực hiện công việc, các lớp đối tượng cho các *tiến trình* và sự đồng bộ giữa các *luồng (thread)* trong hệ thống. Khung nhìn này tập trung vào các nhiệm vụ tương tranh, tương tác với nhau trong hệ thống đa nhiệm. Nó chủ yếu diễn đạt hiệu năng, quy mô và năng lực thông qua của hệ thống.

Khung nhìn cài đặt/thành phần (component/implementation view) bao gồm các thành phần (là mô-đun vật lý hay các file) để lắp ráp thành hệ thống vật lý. Khung nhìn này hướng đến việc quản lý cấu hình của hệ thống. Khung nhìn này bao gồm thành phần, sơ đồ thành phần và gói.

Khung nhìn triển khai/bố trí (Deployment view) bao gồm các nút tạo nên kết cấu phần cứng mà trên đó hệ thống vận hành. Khung nhìn này chủ yếu hướng đến sự phân tán và cài đặt cụ thể của hệ thống, tức là liên quan đến triển khai vật lý của hệ thống. Khung nhìn triển khai chỉ ra các tiến trình và thiết bị trên mạng và các kết nối vật lý giữa chúng. Sơ đồ triển khai cũng hiển thị tiến trình và chỉ ra tiến trình nào chạy trên máy nào. Khung nhìn này được thể hiện trong các sơ đồ triển khai/bố trí các nút của hệ thống.

2.5 Các pha chính trong quy trình phát triển phần mềm

Có năm pha chính cần thực hiện trong quy trình phát triển phần mềm:

Xác định các yêu cầu

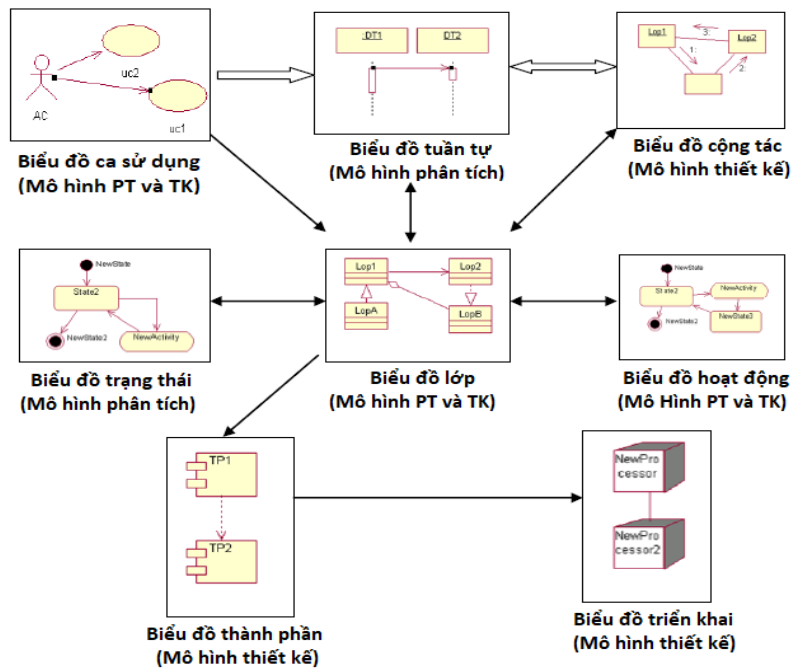
Phân tích hệ thống

Thiết kế hệ thống

Lập trình và kiểm tra hệ thống

Vận hành và bảo trì hệ thống.

Trong quy trình phát triển phần mềm kể trên, hai pha cốt lõi là phân tích và thiết kế hướng đối tượng bằng UML nhằm xây dựng các mô hình đặc tả các yêu cầu, các mô hình khái niệm, logic và kiến trúc của hệ thống.



Hình 1.5 Quy trình xây dựng các mô hình cơ bản trong phân tích và thiết kế hệ thống hướng đối tượng bằng UML

CHƯƠNG 3. PHÂN TÍCH HỆ THỐNG

3.1 Mô hình hóa nghiệp vụ

3.1.1 Đặc tả các yêu cầu của hệ thống

3.1.1.1 Danh sách các tác nhân của hệ thống

Admin: là người có toàn quyền thao tác toàn bộ các chức năng của hệ thống.

Giảng viên: là những người cần sử dụng chức năng Nhập điểm để thực hiện việc nhập điểm các học phần của mình phụ trách.

Chuyên viên: là những người cần sử dụng chức năng Thống kê để kiểm tra tiến độ công việc chấm điểm và lên điểm các học phần khảo thí và nhập danh sách thí sinh vắng thi của các môn thi.

3.1.1.2 Danh sách các ca sử dụng của hệ thống

- ✓ **Quản lý kỳ thi:** Thực hiện việc khởi tạo kì thi, chỉnh sửa thông tin của kỳ thi khi có sai sót. Upload dữ liệu thí sinh dự thi cho các học phần khảo thí, cho phép upload bổ sung, chỉnh sửa thông tin nếu có sai sót. Sinh phách cho dữ liệu thí sinh dự thi, sinh phách bổ sung khi upload bổ sung danh sách thí sinh dự thi.
- ✓ **Ghép điểm:** Chuyên viên tiến hành nhập danh sách sinh viên vắng thi vào hệ thống (các sinh viên vắng thi sẽ có điểm thi là V), sau khi nhập xong, chức năng này sẽ bị khoá lại. Giảng viên tiến hành nhập điểm trên phần mềm nhập điểm có sẵn, đầu ra là file excel có chứa dữ liệu về số phách và điểm thi. Giảng viên thực hiện việc upload file điểm có chứa dữ liệu về số phách và điểm thi của của mỗi học.
- ✓ **Thống kê:** Cho phép thống kê tiến độ ghép điểm của các học phần, túi bài thi (biết được môn thi, túi bài thi nào đã nhập danh sách thí sinh vắng thi hoặc ghép điểm).
- ✓ **Phức khảo:** Khi sinh viên đăng ký phúc khảo bài thi, Admin hoặc Chuyên viên sẽ tiến hành nhập danh sách thí sinh đăng ký phúc khảo. Khi có kết quả phúc khảo, quét điểm và nhập điểm, Giảng viên sẽ tiến hành ghép điểm.
- ✓ **Tìm kiếm:** thực hiện việc tìm kiếm dữ liệu theo các tiêu chí của người dùng.
- ✓ **Quản lý người dùng:** Cho phép thêm mới người dùng, xoá người dùng, phân quyền người dùng với hai vai trò Quản trị và Chuyên viên.
- ✓ **Quản lý lịch sử hoạt động:** Cho phép thống kê lại tất cả các lịch sử hoạt động của người dùng đối với hệ thống.

Bảng 2.1 Danh sách các tác nhân và ca sử dụng

Tác nhân	Ca sử dụng
Admin	Quản lý kỳ thi, Quản lý người dùng, Quản lý lịch sử hoạt động, Thống kê, Phúc khảo, Xuất bảng điểm, Tìm kiếm
Giảng viên	Ghép điểm, Phúc khảo, Tìm kiếm
Chuyên viên	Ghép điểm, Phúc khảo, Thống kê, Tìm kiếm

3.1.1.3 Đặc tả một số ca sử dụng quan trọng

1. Ca sử dụng: Quản lý kỳ thi

- **Tác nhân:** Admin

- **Mục đích:** khởi tạo kì thi, upload dữ liệu thí sinh dự thi, sinh phách cho các học phần khảo thí, xuất ra file theo định dạng cho trước và in phách theo định dạng yêu cầu.

- **Dữ liệu vào:** kì thi, dữ liệu các phòng thi bao gồm mã môn học, mã sinh viên, họ và tên, ngày sinh, số báo danh.

- **Dữ liệu ra:** bao gồm hai file:

- ✓ Một file excel (cho mỗi học phần) dùng để ghép điểm, gồm các thông tin: số thứ tự, mã môn học, mã sinh viên, họ và tên, ngày sinh, số báo danh. (định dạng và đặt tên file theo format của phòng ĐBCLGD).
- ✓ Một file word (cho mỗi học phần) dùng để in phách dán lên bài thi, gồm các thông tin: số thứ tự, mã học phần, mã sinh viên, phách và mã vạch. (định dạng và đặt tên file theo format của phòng ĐBCLGD).

- **Mô tả:**

- ✓ Bước 1. Admin khởi tạo kỳ thi gồm: mã kỳ thi và tên kỳ thi.
- ✓ Bước 2. Admin upload lên hệ thống phần mềm file (excel) dữ liệu các phòng thi gồm: mã môn học, tên môn học, mã sinh viên, họ và tên, ngày sinh, số báo danh, ngày thi, phòng thi, tiết bắt đầu, số tiết.
- ✓ Bước 3. Hệ thống tự động sinh số phách 5 ký tự duy nhất cho các phòng thi (gồm mã phách, mã học phần, tên phòng thi); số phách 5 ký tự duy nhất cho các học phần; xuất ra một file excel và một file word.

2. Ca sử dụng: Ghép điểm

- **Tác nhân:** Admin, Giảng viên

- **Mục đích:** nhập điểm cho các học phần khảo thí khi giảng viên đã chấm xong. Ghép phách và lên điểm cho mỗi học phần khảo thí.

- **Dữ liệu vào:** thông tin giảng viên, file excel (mỗi học phần) của module Sinh phách.

- **Dữ liệu ra:** file điểm (excel) do đã được giảng viên chấm quét mã vạch và hai file:

- ✓ Một file điểm cho mỗi học phần với đầy đủ thông tin: số thứ tự, mã học phần, mã sinh viên, họ và tên, ngày sinh, số báo danh, điểm thi, số phách (định dạng và đặt tên file theo format của phòng ĐBCLGD).
- ✓ Một file tổng hợp điểm với đầy đủ thông tin: điểm thi, mã sinh viên, mã học phần (định dạng và đặt tên file theo format của phòng ĐBCLGD).

- **Mô tả:**

- Chuyên viên tiến hành nhập danh sách sinh viên vắng thi vào hệ thống (các sinh viên vắng thi sẽ có điểm thi là V), sau khi nhập xong, chức năng này sẽ bị khoá lại.
- Giảng viên tiến hành nhập điểm trên phần mềm nhập điểm có sẵn, đầu ra là file excel có chứa dữ liệu về số phách và điểm thi.
- Giảng viên thực hiện việc upload file điểm có chứa dữ liệu về số phách và điểm thi của của mỗi học.
- Khi upload file để ghép điểm, hệ thống sẽ kiểm tra dữ liệu nếu có sai sót:
 - ✓ Phách thi không tồn tại (do máy đọc sai, chọn nhầm môn thi...)
 - ✓ Thí sinh vắng thi nhưng vẫn có điểm thi (do chuyên viên dán nhầm phách hoặc nhập nhầm danh sách sinh viên vắng thi)
 - ✓ Hệ thống sẽ có thông báo cụ thể dữ liệu bị sai phần nào, ở đâu để giảng viên kiểm tra và xử lý lại
 - ✓ Trong trường hợp dữ liệu sinh viên vắng thi bị nhập sai, Quản trị sẽ mở lại chức năng nhập sinh viên vắng thi cho chuyên viên xử lý.
- Trong trường hợp dữ liệu đúng, hệ thống sẽ tiến hành ghép điểm tự động, và báo lại cho người dùng biết ghép điểm thành công.

3. Ca sử dụng: Tìm kiếm

- **Tác nhân:** Admin.

- **Mục đích:** tìm kiếm thông tin về điểm của các học phần.

- **Dữ liệu vào:** một trong bốn tiêu chí tìm kiếm: mã học phần, tên học phần, mã giảng viên, tên giảng viên.

- **Dữ liệu ra:** hiển thị thông tin theo tiêu chí yêu cầu của người dùng.

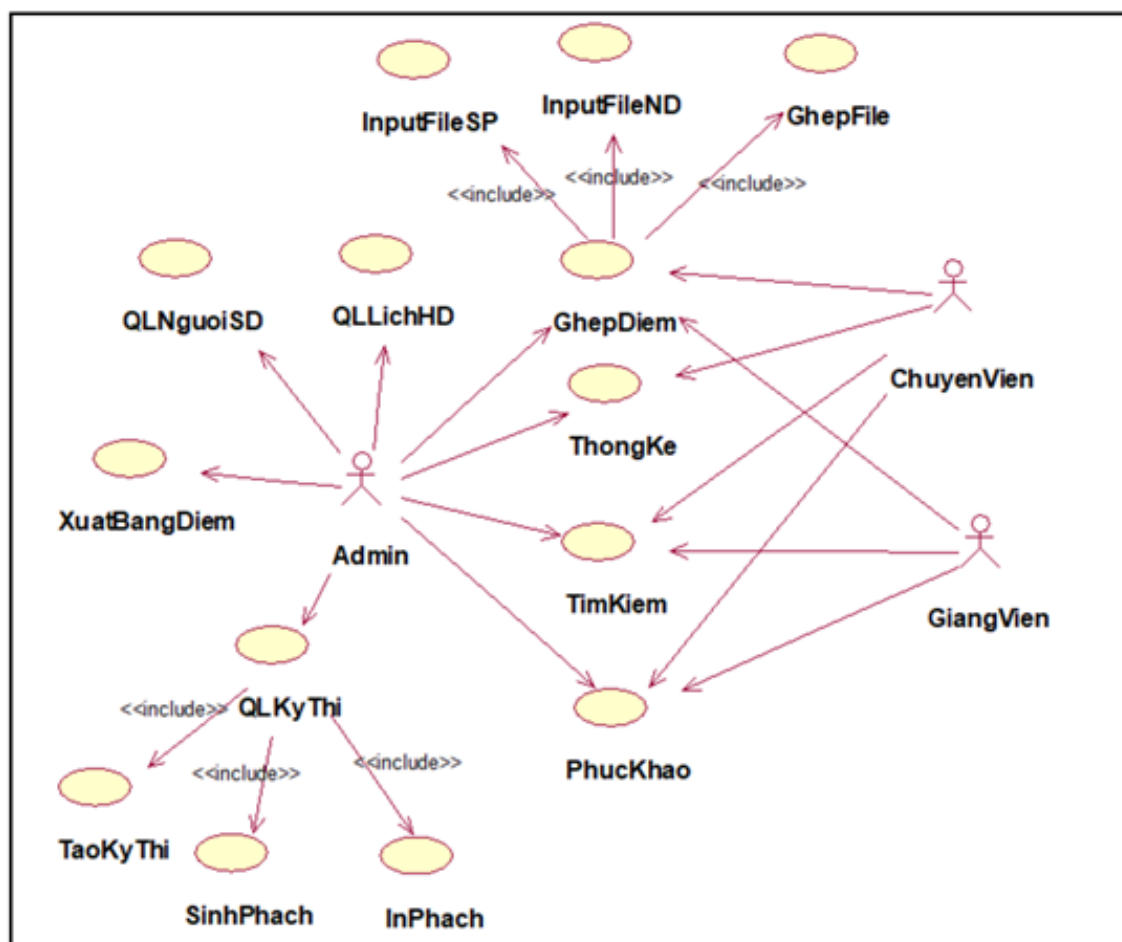
- **Mô tả:** khi Admin có nhu cầu tìm kiếm thông tin về điểm của các học phần thì có thể tìm kiếm theo một trong bốn tiêu chí: mã học phần, tên học phần, mã giảng viên, tên giảng viên. Hệ thống sẽ hiển thị thông tin theo yêu cầu

của người tìm kiếm.

4. Ca sử dụng: Thống kê

- **Tác nhân:** Admin, người quản lý.
- **Mục đích:** thống kê số lượt phách, số lượt học phần đã được ghép phách; vẽ biểu đồ thể hiện tỷ lệ % công việc đã hoàn thành theo mã học phần hoặc theo lượt phách.
- **Dữ liệu vào:** một trong bốn tiêu chí thống kê: số lượt phách, các học phần đã được ghép phách, vẽ biểu đồ công việc (theo mã học phần hoặc theo phách)
- **Dữ liệu ra:** hiển thị thông tin theo tiêu chí thống kê yêu cầu của người dùng.
- **Mô tả:** Admin và người quản lý có thể thống kê theo các tiêu chí: số lượt phách, các học phần đã được ghép phách, tiến độ công việc đã hoàn thành.

3.1.2 Xây dựng biểu đồ ca sử dụng



Hình 2. 1. Mô hình Ca sử dụng tổng thể của hệ thống

3.2 Mô hình hóa sự tương tác

Mục đích của bước mô hình hóa tương tác là dùng các biểu đồ tương tác (chủ yếu là biểu đồ trình tự và biểu đồ giao tiếp) để diễn tả sự tương tác giữa các đối tượng

nhằm tạo ra các kịch bản của mỗi ca sử dụng của hệ thống.

Hình thức tương tác duy nhất có thể có giữa các đối tượng là chuyển giao thông điệp và khi thành lập các biểu đồ tương tác ta có thể áp dụng các nguyên tắc sau đây:

- Các đối tác chỉ có thể tương tác (gửi thông điệp) tới các đối tượng biên.
- Các đối tượng biên chỉ có thể tương tác tới các đối tượng điều khiển hay các đối tượng biên khác.
- Các đối tượng điều khiển có thể tương tác tới các đối tượng biên, các đối tượng thực thể hay các đối tượng điều khiển.
- Các đối tượng thực thể chỉ có thể tương tác với các đối tượng thực thể mà thôi.

Theo yêu cầu của giai đoạn phân tích, chúng ta chỉ cần định nghĩa hệ thống như một hộp đen, trong đó hành vi của hệ thống thể hiện được những gì (What?) nó cần thực hiện và không cần thể hiện những cái đó thực hiện như thế nào (How?). Vì vậy, nhiệm vụ chính của chúng ta trong giai đoạn này là xác định và mô tả được các hoạt động của hệ thống theo yêu cầu của các tác nhân. Nghĩa là phải tìm được các sự kiện, các thao tác (sự tương tác) của các đối tượng trong từng ca sử dụng. Biểu đồ tuần tự giúp chúng ta thực hiện được những nhiệm vụ đó.

Biểu đồ tuần tự (sequence diagram) là sơ đồ tương tác theo trình tự thời gian của các giao tiếp bằng thông điệp giữa các đối tượng đang hoạt động trong hệ thống. Mỗi ca sử dụng có nhiều luồng dữ liệu. Mỗi Biểu đồ tuần tự biểu diễn một luồng dữ liệu.

Biểu đồ tuần tự bao gồm các phần tử biểu diễn đối tượng, thông điệp và thời gian. Biểu đồ tuần tự được thể hiện theo hai trục:

Trục dọc trên xuống chỉ thời gian xảy ra các sự kiện, hay sự truyền thông điệp, được biểu diễn bằng các đường gạch - gạch thẳng đứng bắt đầu từ đỉnh đến đáy của sơ đồ.

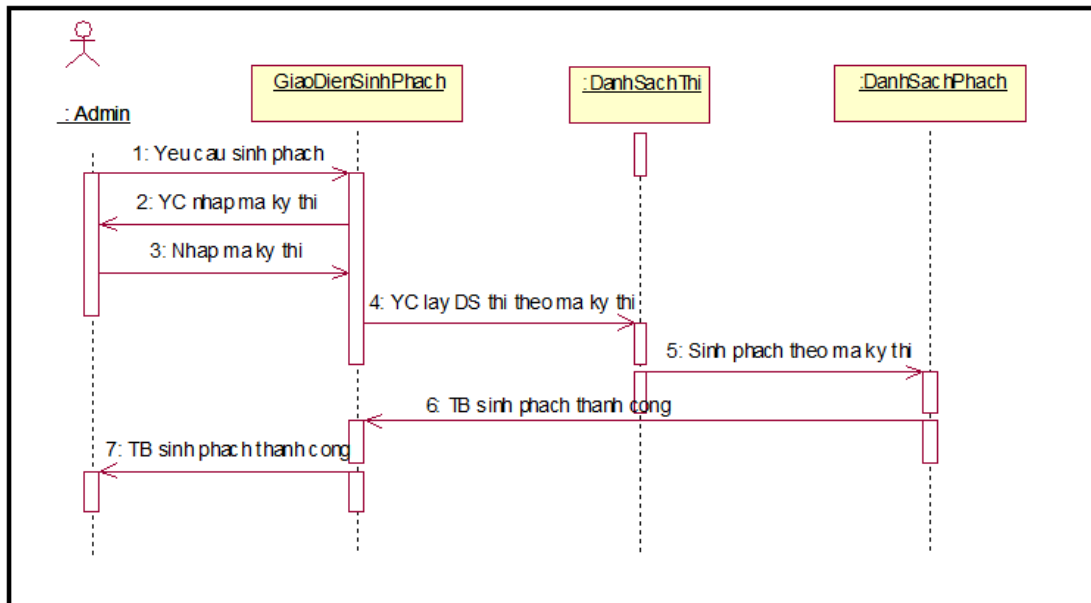
Trục ngang từ trái qua phải là dãy các đối tượng tham gia vào việc trao đổi các thông điệp với nhau theo chiều ngang, có thể có cả các tác nhân.

Đối tượng được biểu diễn bằng hình chữ nhật trong đó có tên đối tượng cụ thể và/hoặc tên lớp cùng được gạch dưới (hoặc tên lớp được gạch dưới biểu diễn cho một đối tượng bất kỳ của lớp đó).

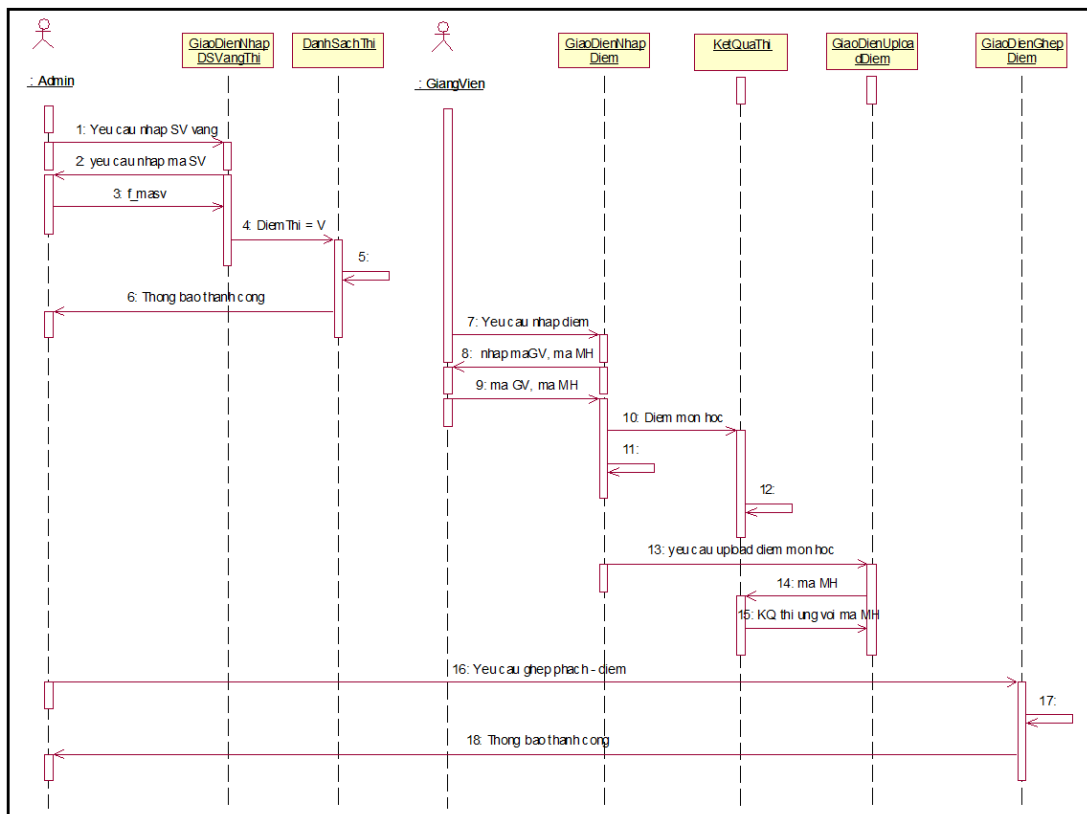
Biểu đồ tuần tự được đọc từ trên xuống dưới, từ trái sang phải. Thứ tự các đối tượng trong sơ đồ phải được sắp xếp sao cho đơn giản nhất có thể để dễ quan sát. Thời gian thực hiện một thông điệp của một đối tượng, hay còn gọi là hoạt động của đối tượng được biểu diễn bằng hình chữ nhật hẹp dọc theo trục thẳng đứng của đối tượng đó.

Mỗi thông điệp đều có tên gọi thể hiện được ý nghĩa của thông tin cần gửi và các tham số về dữ liệu liên quan. Thông thường đó là các lời gọi hàm. Khi định nghĩa

các lớp sau này thì mỗi thông điệp nhận được sẽ trở thành một phương thức. Một đối tượng có thể gửi thông điệp tới chính nó. Những thông điệp này gọi là phản thân, nó chỉ ra rằng đối tượng gọi chính các thao tác của mình để thực hiện.



Hình 2.2 Biểu đồ tuần tự ca sử dụng Sinh phách



Hình 2.3 Biểu đồ tuần tự ca sử dụng Ghép điểm

3.3 Mô hình hóa cấu trúc

Biểu đồ lớp mô tả quan sát tĩnh của hệ thống thông qua các lớp và các mối quan hệ của chúng. Nó được sử dụng để hiển thị các lớp và gói các lớp cùng các mối quan hệ của chúng. Biểu đồ lớp giúp người phát triển phần mềm quan sát và lập kế hoạch cấu trúc hệ thống trước khi lập trình. Nó đảm bảo rằng hệ thống được thiết kế tốt ngay từ đầu. Thông thường mỗi hệ thống có vài biểu đồ lớp. Có thể tạo rất nhiều biểu đồ lớp để mô tả toàn bộ bức tranh hệ thống.

Biểu đồ các lớp với các thuộc tính mô tả cấu trúc tĩnh của hệ thống. Nó mô tả mối liên kết có cấu trúc giữa các mục dữ liệu được thao tác, xử lý trong hệ thống. Nó cũng mô tả cách các thông tin được phân chia thành từng phần cho các đối tượng, chỉ ra cách các đối tượng được chia thành các lớp và thể hiện mối quan hệ giữa các đối tượng là gì.

3.3.1 Xác định các lớp đối tượng:

Có sáu cách chính để tìm

- Dựa vào văn bản, kịch bản mô tả bài toán: các danh từ có thể là các đại biểu của lớp.
- Dựa vào danh sách phân loại các phạm trù khái niệm.
- Dựa vào mục đích của các UC.
- Dựa vào kinh nghiệm và kiến thức của người phân tích.
- Dựa vào hồ sơ tài liệu những hệ thống có liên quan.
- Dựa vào ý kiến tham khảo với các chuyên gia hệ thống.

3.3.2 Xác định mối quan hệ giữa các lớp đối tượng

Hệ thống bao giờ cũng là một thể thống nhất, nghĩa là các phần tử của hệ thống phải có quan hệ tương tác với nhau. Do vậy, mô hình khái niệm phải bao gồm những khái niệm (các lớp đối tượng) khác nhau nhưng phải có sự tương tác, hợp tác với nhau.

Trong pha phân tích, chủ yếu tập trung phát hiện các mối quan hệ kết hợp, kết nhập của các lớp. Trong đó quan hệ kết hợp là quan trọng nhất, nó thể hiện các mối liên hệ giữa các lớp trong hệ thống.

Các đối tượng có thể có nhiều loại quan hệ và vì thế, các lớp (khái niệm) cũng phải có tất cả các loại quan hệ đó trong lĩnh vực ứng dụng.

- ❖ Quan hệ kết hợp giữa hai lớp là *sự kết nối vật lý hay khái niệm* giữa các đối tượng của hai lớp đó.
- ❖ Chỉ những đối tượng có mối quan hệ kết hợp với nhau mới có thể cộng tác với nhau theo các đường liên kết được thiết lập giữa các lớp.

3.3.3 Các thuộc tính của các lớp đối tượng

Thuộc tính của lớp mô tả các đặc tính của các đối tượng trong lớp đó. Ở mỗi thời điểm thuộc tính của một đối tượng của một lớp là giá trị dữ liệu logic biểu diễn cho tính chất tương ứng của đối tượng, được gọi là giá trị thuộc tính của đối tượng tại thời điểm đó.

Mỗi thuộc tính có:

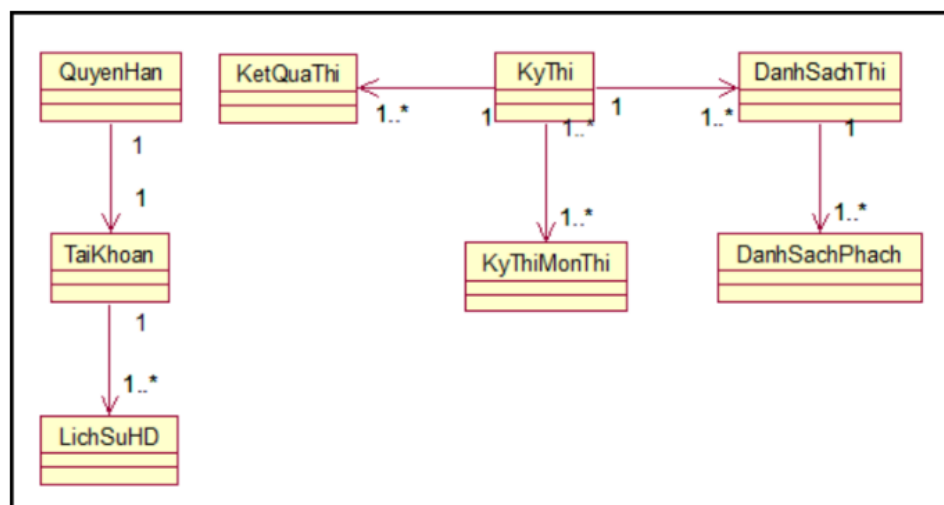
- Tên của thuộc tính,
- Kiểu xác định các loại giá trị mà thuộc tính mô tả,
- Giá trị mặc định (khởi đầu) cho mỗi thuộc tính. Việc gán giá trị khởi đầu cho thuộc tính là tùy chọn, không bắt buộc.

3.3.4 Các phương thức của lớp đối tượng

Phương thức (thao tác, hoặc hàm thành phần) của lớp mô tả các hành vi và các mối quan hệ của các đối tượng trong hệ thống.

Mỗi phương thức được mô tả bởi phạm vi, tên gọi, danh sách các tham số và kiểu trả lại giá trị.

- Phạm vi: public, protected, private, hay mặc định
- Tên gọi: là tên gọi của phương thức, theo qui ước thường là các động từ (cụm động từ), viết hoa các chữ cái đầu tiên của các từ trừ từ đầu tiên.
- Danh sách các tham số.
- Các kiểu thuộc tính: là các kiểu của các thuộc tính đã khai báo trong lớp.
- Kiểu dữ liệu trả về sau khi phương thức kết thúc. Những phương thức không có kiểu trả lại (các thủ tục) thì sử dụng Void (void).



Hình 2.4 Biểu đồ lớp của hệ thống phần mềm

CHƯƠNG 4: THIẾT KẾ HỆ THỐNG VÀ CÀI ĐẶT

4.1 Công cụ phát triển

4.1.1 Lựa chọn môi trường phát triển hệ thống

Như đã xác định rõ ngay từ khi phát triển hệ thống, đó là hệ thống phần mềm sẽ chạy trên môi trường Web giúp cho nó có thể sử dụng ở bất kỳ nơi đâu, bất kỳ thời điểm nào, với đa dạng thiết bị phần cứng sử dụng. Để phát triển ứng dụng Web hiện nay có khá nhiều công cụ, mô hình phát triển và ngôn ngữ lập trình như ASP, PHP, Java... tuy nhiên, nhóm tác giả sử dụng nền tảng .NET; ngôn ngữ lập trình ASP; phát triển theo mô hình MVC (Model – View – Controller), sử dụng bộ công cụ Visual Studio và hệ quản trị cơ sở dữ liệu (CSDL) Microsoft SQL Server.



Hình 4.1. Công cụ, ngôn ngữ, hệ quản trị CSDL và mô hình hình phát triển hệ thống.

Bên cạnh đó, để tối ưu hóa hoạt động của hệ thống cho các thiết bị cuối sử dụng với các kích cỡ màn hình khác nhau, chúng tôi cũng sử dụng một số công nghệ và kỹ thuật mới trong phát triển ứng dụng Web hiện nay để xử lý vấn đề này như: Bootstrap, JQuery, LinQ, CSS3, HTML5...

4.1.2 Giới thiệu về mô hình MVC

MVC (Model - View – Controller) là một mô hình thiết kế hay kiến trúc mẫu (design pattern) được xây dựng nhằm giải quyết các vấn đề phát sinh cũng như các giải pháp cho chúng trong quá trình phát triển phần mềm.

Vào những năm 70 của thế kỷ 20, tại phòng thí nghiệm Xerox PARC ở Palo Alto. Sự ra đời của giao diện đồ họa (Graphical User Interface - GUI) và lập trình hướng đối tượng (Object Oriented Programming - OOP) cho phép lập trình viên làm việc với những thành phần đồ họa như những đối tượng đồ họa có thuộc tính và

phương thức riêng của nó. Không dừng lại ở đó, những nhà nghiên cứu ở Xerox PARC còn đi xa hơn nữa khi họ cho ra đời cái gọi là kiến trúc MVC (viết tắt của Model – View – Controller).

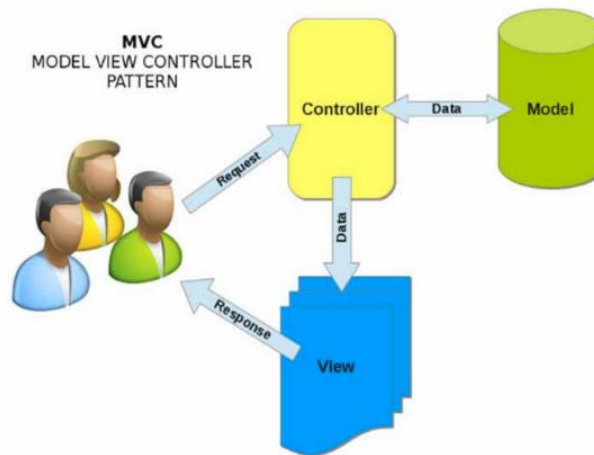
Trong kiến trúc MVC, một đối tượng đồ họa (GUI Component) bao gồm 3 thành phần cơ bản: Model, View và Controller. Mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác. Model có trách nhiệm đối với toàn bộ dữ liệu cũng như trạng thái của đối tượng đồ họa. View chính là thể hiện trực quan của Model, hay nói cách khác thì View chính là giao diện của đối tượng đồ họa. Controller điều khiển việc tương tác giữa đối tượng đồ họa với người sử dụng cũng như những đối tượng đồ họa khác.

Model là một đối tượng hoặc một tập hợp các đối tượng biểu diễn cho phần dữ liệu của chương trình. Nó chứa tất cả các nghiệp vụ logic, phương thức xử lý và truy xuất database, đối tượng mô tả dữ liệu, các hàm xử lý, trạng thái của đối tượng. Model thường là một lớp được ánh xạ từ một bảng trong CSDL. Tất cả các nghiệp vụ logic được thực thi ở Model. Dữ liệu vào từ người dùng sẽ thông qua View đến Controller và được kiểm tra ở Model trước khi lưu vào cơ sở dữ liệu. Việc truy xuất, xác nhận và lưu dữ liệu là một phần của Model. Ví dụ như chúng ta có lớp Products được sử dụng để mô tả dữ liệu từ bảng Products trong CSDL, bao gồm ProductID, OrderDate.

View là phần giao diện với người dùng, bao gồm việc hiện dữ liệu ra màn hình, cung cấp các menu, nút bấm, hộp đối thoại, chọn lựa,... để người dùng có thể thêm, sửa, xóa, tìm kiếm và làm các thao tác khác đối với dữ liệu trong hệ thống. Thông thường thì các thông tin cần hiển thị được lấy từ thành phần Models. Nói cách khác thì View là thành phần đảm nhiệm việc hiển thị thông tin, tương tác với người dùng, nơi chứa tất cả các đối tượng GUI như textbox, images... hay View là nơi tập hợp các form hoặc các file HTML. Ví dụ, đối tượng Product có một "Edit" View bao gồm các Textbox, Dropdown và các Checkbox để chỉnh sửa các thuộc tính sản phẩm, có một "Display" để hiển thị cột ProductID và OrderDate để xem thông tin về sản phẩm.

Controller là phần điều khiển của ứng dụng, điều hướng các nhiệm vụ (task) đến đúng phương thức (method) có chức năng xử lý nhiệm vụ. Nó chịu trách nhiệm xử lý các tác động về mặt giao diện, các thao tác đối với Model và cuối cùng là chọn một View thích hợp để hiển thị ra màn hình. Nói cách khác thì Controller giữ nhiệm vụ nhận điều hướng các yêu cầu từ người dùng và gọi đúng những phương thức xử

lý chúng... chẳng hạn thành phần này sẽ nhận request từ url và form để thao tác trực tiếp với Model.



Hình 4.2. Mô hình MVC

Luồng sự kiện được xử lý trong MVC: User tương tác với View, User gửi yêu cầu đi. Controller nhận và điều hướng chúng đến đúng phương thức xử lý ở Model. Model nhận thông tin và thực thi các yêu cầu. Khi Model hoàn tất việc xử lý, View sẽ nhận kết quả từ Model và hiển thị lại cho người dùng.

Khi người sử dụng hoặc những đối tượng khác cần thay đổi trạng thái của đối tượng đồ họa, nó sẽ tương tác thông qua Controller của đối tượng đồ họa. Controller sẽ thực hiện việc thay đổi trên Model. Khi có bất kỳ sự thay đổi nào xảy ra ở Model, nó sẽ phát thông điệp (broadcast message) thông báo cho View và Controller biết. Nhận được thông điệp đó từ Model, View sẽ cập nhật lại và đảm bảo rằng nó luôn là thể hiện trực quan chính xác của Model. Còn Controller, khi nhận được thông điệp từ Model, sẽ có những tương tác cần thiết phản hồi lại người sử dụng hoặc các đối tượng khác.

Kiến trúc MVC đã tách biệt (decoupling) sự phụ thuộc giữa các thành phần trong một đối tượng đồ họa, làm tăng tính linh động (flexibility) và tính tái sử dụng (reuseability) của đối tượng đồ họa đó. Một số đối tượng đồ họa có thể dễ dàng thay đổi giao diện bằng cách thay đổi thành phần View của nó trong khi cách thức lưu trữ (Model) cũng như xử lý (Controller) không hề thay đổi. Tương tự, ta có thể thay đổi cách thức lưu trữ (Model) hoặc xử lý (Controller) của đối tượng đồ họa mà những thành phần còn lại vẫn giữ nguyên.

Kiến trúc MVC đã được ứng dụng để xây dựng rất nhiều framework và thư viện đồ họa khác nhau. Tiêu biểu là bộ thư viện đồ họa của ngôn ngữ lập trình hướng đối tượng SmallTalk (cũng do Xerox PARC nghiên cứu và phát triển vào thập niên

70 của thế kỷ 20). Các Swing Components của Java cũng được xây dựng dựa trên kiến trúc MVC. Ví dụ đi cùng với JButton là ButtonUI (thành phần View) và ButtonModel (thành phần Model). Ta hoàn toàn có thể viết MyButtonUI hoặc YourButtonUI để thay đổi giao diện của JButton theo ý mình (tương tự cho ButtonModel). Một điểm khá thú vị đối với Swing Components là nó cho phép ta chỉ thay đổi giao diện một phần nào đó của component. Ví dụ ta có thể thay đổi thể hiện của list item trong JList thông qua ListCellRenderer.

- **Ưu điểm:**

- + Presentation Layer (PL) thể hiện giống như chức năng của View và Tính tin cậy (reliability): việc chia thành các thành phần riêng biệt giúp ta sửa đổi từng thành phần riêng biệt, có thể thay thế thành từng thành phần tương đương, có thể chia công việc theo nhóm, biên dịch độc lập, tăng cường khả năng tích hợp và khả năng đúng đắn cao.
- + Tính tương thích cao (compatibility): có khả năng sử dụng các công nghệ khác nhau mà không lệ thuộc vì chúng đã được tách biệt với nhau.
- + Tính tái sử dụng (reusable): chúng ta có thể sử dụng các thành phần của các ứng dụng khác để sử dụng cho ứng dụng hiện tại giúp tăng khả năng lập trình.
- + Khả năng triển khai và bảo trì nhanh chóng (quick deploy and easy maintenance): vì các thành phần độc lập với nhau nên việc bảo trì từng thành phần là khá dễ dàng mà ít ảnh hưởng tới các thành phần khác.

- **Nhược điểm:**

- + Đối với dự án nhỏ, việc áp dụng mô hình MVC gây cồng kềnh, tốn thời gian trong quá trình phát triển. Tốn thời gian trung chuyển dữ liệu của các tầng.
- + MVC là một nền tảng mới nên việc học khá khó khăn khi tài liệu rất hạn chế.

4.1.3 Giới thiệu về ASP.NET

Từ khoảng cuối thập niên 90, ASP (Active Server Pages) đã được nhiều lập trình viên lựa chọn để xây dựng và phát triển ứng dụng web động trên máy chủ sử dụng hệ điều hành Windows. ASP đã thể hiện được những ưu điểm của mình với mô hình lập trình thủ tục đơn giản, sử dụng hiệu quả các đối tượng COM: ADO (ActiveX Data Object) - xử lý dữ liệu, FSO (File System Object) - làm việc với hệ thống tập tin..., đồng thời, ASP cũng hỗ trợ nhiều ngôn ngữ: VBScript, JavaScript. Chính những ưu điểm đó, ASP đã được yêu thích trong một thời gian dài.

Tuy nhiên, ASP vẫn còn tồn đọng một số khó khăn như Code ASP và HTML lẫn lộn, điều này làm cho quá trình viết code khó khăn, thể hiện và trình bày code không trong sáng, hạn chế khả năng sử dụng lại code. Bên cạnh đó, khi triển khai cài đặt, do không được biên dịch trước nên dễ bị mất source code. Thêm vào đó, ASP không có hỗ trợ cache, không được biên dịch trước nên phần nào hạn chế về mặt tốc độ thực hiện. Quá trình xử lý postback khó khăn ...

Đầu năm 2002, Microsoft giới thiệu một kỹ thuật lập trình Web khá mới mẻ với tên gọi ban đầu là ASP+, tên chính thức sau này là ASP.Net. Với ASP.Net, không những không cần đòi hỏi bạn phải biết các tag HTML, thiết kế web, mà nó còn hỗ trợ mạnh lập trình hướng đối tượng trong quá trình xây dựng và phát triển ứng dụng Web.

ASP.Net là kỹ thuật lập trình và phát triển ứng dụng web ở phía Server (Server-side) dựa trên nền tảng của Microsoft .Net Framework. Hầu hết, những người mới đến với lập trình web đều bắt đầu tìm hiểu những kỹ thuật ở phía Client (Client-side) như: HTML, Java Script, CSS (Cascading Style Sheets). Khi Web browser yêu cầu một trang web (trang web sử dụng kỹ thuật client-side), Web server tìm trang web mà Client yêu cầu, sau đó gửi về cho Client. Client nhận kết quả trả về từ Server và hiển thị lên màn hình.

ASP.Net sử dụng kỹ thuật lập trình ở phía server thì hoàn toàn khác, mã lệnh ở phía server (ví dụ: mã lệnh trong trang ASP) sẽ được biên dịch và thi hành tại Web Server. Sau khi được Server đọc, biên dịch và thi hành, kết quả tự động được chuyển sang HTML/JavaScript/CSS và trả về cho Client. Tất cả các xử lý lệnh ASP.Net đều được thực hiện tại Server và do đó, gọi là kỹ thuật lập trình ở phía server.

- Ưu điểm của ASP.Net:

- + ASP.Net cho phép bạn lựa chọn một trong các ngôn ngữ lập trình mà bạn yêu thích: Visual Basic.Net, J#, C#...
- + Trang ASP.Net được biên dịch trước. Thay vì phải đọc và thông dịch mỗi khi trang web được yêu cầu, ASP.Net biên dịch những trang web động thành những tập tin DLL mà Server có thể thi hành nhanh chóng và hiệu quả. Yếu tố này là một bước nhảy vọt đáng kể so với kỹ thuật thông dịch của ASP.
- + ASP.Net hỗ trợ mạnh mẽ bộ thư viện phong phú và đa dạng của .Net Framework, làm việc với XML, Web Services, truy cập cơ sở dữ liệu qua ADO.Net,....

- + ASP.Net sử dụng phong cách lập trình mới đó là Code Behide. Tách code riêng, giao diện riêng. Việc này giúp người lập trình dễ đọc, dễ quản lý và bảo trì.
- + Hỗ trợ quản lý trạng thái của các control.
- + Tự động phát sinh mã HTML cho các Server control tương ứng với từng loại trình duyệt.
- + Hỗ trợ phát triển Web được truy cập trên các thiết bị di động.
- + Hỗ trợ thiết kế và xây dựng MasterPage lồng nhau.
- + Tăng cường tính bảo mật.
- + Triển khai cài đặt không cần lock, không cần đăng ký dll, cho phép nhiều hình thức cấu hình.
- + Hỗ trợ quản lý ứng dụng ở mức toàn cục.
- + Global.aspx có nhiều sự kiện hơn, quản lý session trên nhiều server.
- + ASP.NET là một mô hình thiết kế website thống nhất bao gồm cách dịch vụ cần thiết để xây dựng các ứng dụng website doanh nghiệp lớn. ASP.NET là một phần của NET Framework.

Ở chủ đề này sẽ mô tả các tính năng của ASP.NET và Visual Web Developer, môi trường phát triển tạo ra các ứng dụng ASP.NET.

- Visual Web Developer:

Visual Web Developer là một môi trường phát triển đầy đủ tính năng để tạo ra các ứng dụng website ASP.NET.

Các tính năng: Bố trí trang web phù hợp với các trang tổng thể và sự xuất hiện trang phù hợp với các chủ đề.

Chỉnh sửa mã: Trình biên tập cho phép bạn viết mã cho các website động trong Visual Basic hoặc C#.

Kiểm tra và gỡ rối: Máy chủ web thử nghiệm và gỡ rối giúp bạn tìm thấy lỗi trong chương trình.

Triển khai: Công cụ tự động hóa các nhiệm vụ cho việc triển khai một ứng dụng web đến một máy chủ lưu trữ hoặc nhà cung cấp lưu trữ.

Trang ASP.NET và điều khiển là một chương trình chạy trên máy chủ tự động đưa ra các trang ASP.NET. Các trang ASP.NET có thể được yêu cầu từ bất kỳ trình

duyet nào hoặc các thiết bị của khách hàng. Theo quy định bạn có thể sử dụng cùng một website cho nhiều trình duyệt. ASP.NET là công cụ thiết kế website hoàn toàn hướng đối tượng. Nó cũng cho phép đóng gói các chức năng giao diện người dùng thông thường để sử dụng, kiểm soát để có thể tái sử dụng.

Ngoài ra các chủ đề cũng giúp bạn xác định trang chủ để tạo ra và bố trí phù hợp cho các website trong ứng dụng của bạn. Khung trang ASP.NET cũng cho phép bạn xác định mô hình các URL sẽ được sử dụng trong website của bạn. Điều này giúp tối ưu hóa công cụ tìm kiếm và làm cho các URL thân thiện với người sử dụng.

- Biên dịch ASP.NET:

ASP.NET biên dịch, cho phép tối ưu hóa hiệu suất, tối ưu hóa ràng buộc và những lợi ích khác. Một khi mã được biên dịch, thời gian chạy sẽ được cải thiện.

- Bảo mật:

Ngoài các tính năng bảo mật cung cấp một sơ sở bảo mật tiên tiến để chứng thực và cho phép người dùng truy cập vào các nhiệm vụ liên quan đến an ninh. Bạn có thể xác thực bằng cách sử dụng window xác thực được cung cấp bởi IIS hoặc quản lý xác thực bằng cơ sở dữ liệu người dùng riêng.

- Cấu hình ASP.NET:

ASP.NET sử dụng một cấu hình hệ thống cho phép bạn xác định và thiết lập cấu hình cho máy chủ web của bạn. ASP.NET thiết lập các hình được lưu trữ trong các tập tin XML.

- Hỗ trợ gỡ lỗi:

ASP.NET có lợi thế gỡ lỗi trong thời gian chạy để cung cấp ngôn ngữ và hỗ trợ gỡ lỗi qua máy tính. Bạn có thể gỡ lỗi cả hai đối tượng quản lý và không được quản lý.

Web Service Framework ASP.NET hỗ trợ các dịch vụ web XML cho phép ứng dụng trao đổi thông tin qua bức tường lửa bằng cách sử dụng các tiêu chuẩn như HTTP và XML. Bất cứ ngôn ngữ lập trình web nào cũng có thể trao đổi thông tin và truy cập các dịch vụ web XML.

- ASP.NET sử dụng dữ liệu động:

ASP.NET sử dụng dữ liệu động cho phép bạn tạo ra các ứng dụng web hướng dữ liệu một cách dễ dàng. Bạn sẽ tìm thấy rằng bạn có thể nhận được các ứng dụng chạy dễ dàng hơn và với ít mã hơn so với trước đây. Đồng thời bạn có thể thêm các tính năng tùy chỉnh để các dữ liệu ASP.NET động để thích ứng với yêu cầu riêng của bạn.

4.1.4 Hệ quản trị cơ sở dữ liệu

Trong hệ thống này, chúng tôi sử dụng hệ quản trị cơ sở dữ liệu SQL Server 2008 để lưu trữ CSDL, Dưới đây chúng tôi sẽ giới thiệu sơ lược một số thông tin về Hệ quản trị CSDL này:

Một số ưu điểm của Microsoft SQL Server 2008:

- Có thể mở rộng.
- Có khả năng sẵn sàng cao.
- Có khả năng sao lưu, phục hồi dữ liệu.
- Được xây dựng trên mô hình dữ liệu quan hệ.
- Kiểm tra tính đúng đắn của dữ liệu và thẩm quyền truy vấn.
- Cho phép người sử dụng theo mô hình Client/Server.
- Có khả năng lưu trữ dữ liệu lâu dài.

Microsoft SQL server là một hệ quản trị cơ sở dữ liệu quan hệ (relational database management system – RDBMS) do Microsoft phát triển. SQL Server là một hệ quản trị cơ sở dữ liệu quan hệ mạng máy tính hoạt động theo mô hình khách chủ cho phép đồng thời cùng lúc có nhiều người dùng truy xuất đến dữ liệu, quản lý việc truy nhập hợp lệ và các quyền hạn của từng người dùng trên mạng. Ngôn ngữ truy vấn quan trọng của Microsoft SQL server là Transact-SQL. Transact-SQL là ngôn ngữ SQL mở rộng dựa trên SQL chuẩn của ISO (International Organization for Standardization) và ANSI (American National Standards Institute) được sử dụng trong SQL Server.

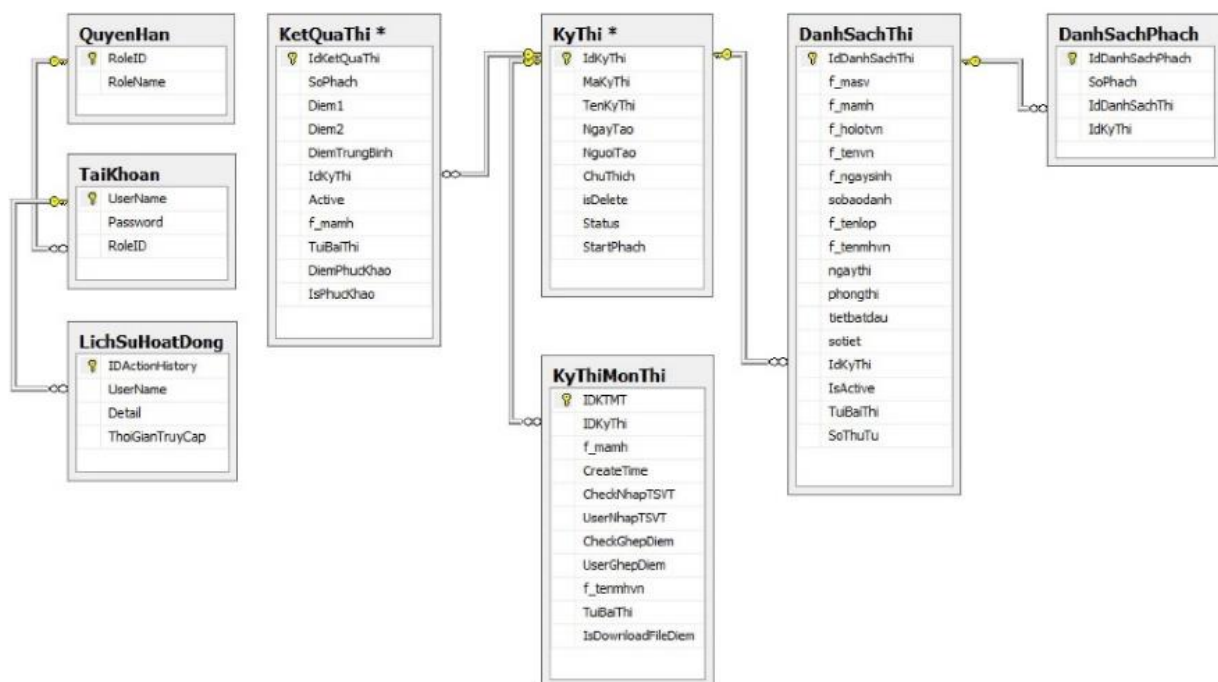
Quay lại lịch sử, lúc đầu Microsoft đã quay sang kết hợp với công ty Sybase để cùng hợp tác làm ra một sản phẩm thuộc loại hệ quản trị cơ sở dữ liệu (Database Management System). Với sự hợp tác tốt đẹp, hai công ty này đã thành công cho ra đời một sản phẩm thuộc loại cơ sở dữ liệu có tên là Ashton-Tate vào năm 1989, sản phẩm này hoạt động trên môi trường OS/2. Sau đó một thời gian, Sybase đã phát triển sản phẩm này trên môi trường UNIX và đổi tên riêng là Data Server mà ngày nay còn có tên khác là Sybase Adaptive Server. Microsoft quyết định không phát triển hệ điều hành OS/2 mà thay vào đó cho ra đời một hệ điều hành mạng máy tính có tên là Windows NT Server. Và thế là SQL Server chỉ hoạt động độc lập trên môi trường Windows NT Server mà thôi. Lần lượt các phiên bản của Microsoft SQL Server đã ra đời sau sự kiện này, từ 4.2 sau đó được nâng cấp thành 4.21, 6.0, 6.5, 7.0 và hiện

giờ là Microsoft SQL Server 2000, 2005, 2008 và mới nhất là Microsoft SQL Server 2012.

SQL Server 2005: SQL Server 2005, được phát hành vào tháng 11 năm 2005, là phiên bản tiếp theo của SQL Server 2000. SQL Server 2005 tập trung vào khả năng xử lý giao dịch trực tuyến trên diện rộng OLTP (large-scale online transactional processing), ứng dụng thương mại điện tử (e-commerce applications) và kho dữ liệu (data warehousing). Ngoài ra, những cải tiến quan trọng trong SQL Server 2005 là thêm các dịch vụ mới như: dịch vụ báo cáo Reporting Services, Service Broker và sự thay đổi đáng kể trong Database Engine.

SQL Server 2008: Có tên mã là “katmai”. Ngày 27/02/2008, Microsoft tổ chức một sự kiện có tên Heroes Happen Here nhằm giới thiệu sản phẩm mới SQL Server 2008 (cùng với những sản phẩm khác như Windows Server 2008; Visual Studio 2008). Bản SQL Server 2008 Release Candidate sẽ được trình làng trong quý II, trong khi đó, bản hoàn chỉnh sẽ mắt trong quý III (2008).

4.2. Thiết kế cơ sở dữ liệu



Hình 4.3 Xây dựng CSDL của hệ thống

Bảng 4.1 Bảng QuyenHan

Trường	Kiểu dữ liệu	Ý nghĩa
RoleId	Varchar (10)	Id vai trò
RoleName	Nvarchar (20)	Tên vai trò

Bảng 4.2 Bảng TaiKhoan

Trường	Kiểu dữ liệu	Ý nghĩa
UserName	Varchar (50)	Tên đăng nhập
Password	Varchar (max)	Mật khẩu, mã hoá MD5
RoleId	Nvarchar (20)	Tên vai trò

Bảng 4.3 Bảng LichSuHoatDong

Trường	Kiểu dữ liệu	Ý nghĩa
IdActionHistory	Double	Id hành động, tự động tăng
UserName	Varchar (50)	Tên đăng nhập
Detail	Nvarchar (max)	Mô tả chi tiết thao tác của người dùng
ThoiGianTruyCap	Datetime	Thời gian người dùng thực hiện thao tác

Bảng 4.4 Bảng KetQuaThi

Trường	Kiểu dữ liệu	Ý nghĩa
IdKetQuaThi	Double	Id kết quả thi
SoPhach	Int	Số phách
Diem1	Double	Điểm người chấm 1

Diem2	Double	Điểm người chấm 2
DiemTrungBinh	Double	Điểm trung bình
IDKyThi	Int	Id kỳ thi
Active	Bool	Tình trạng kỳ thi
f_manh	Varchar (15)	Mã môn học
TuiBaiThi	Int	Túi bài thi
DiemPhucKhao	Double	Điểm chấm phúc khảo
IsPhucKhao	Bool	Bài thi chấm phúc khảo hay không

Bảng 4.5 Bảng KyThi

Trường	Kiểu dữ liệu	Ý nghĩa
IdKyThi	Int	Id kỳ thi
MaKyThi	Varchar (15)	Mã kỳ thi
TenKyThi	Nvarchar (50)	Tên kỳ thi
NgayTao	Datetime	Ngày tạo
ChuThich	Varchar (max)	Chú thích
isDelete	Bool	Xoá kỳ thi hay không
Status	Bool	Kỳ thi đã hoàn thành hay chưa
StartPhach	Int	Phách bắt đầu

Bảng 4.6 Bảng KyThiMonThi

Trường	Kiểu dữ liệu	Ý nghĩa
--------	--------------	---------

IdKTMT	Double	Id kỳ thi – môn thi
IdKyThi	Int	Id kỳ thi
f_manh	Varchar (15)	Mã môn học
CreateTime	Datetime	Thời gian tạo
CheckNhapTSVT	Bool	Đã/chưa nhập thí sinh vắng thi
UserNhapTSVT	Varchar (50)	Tên người dùng nhập DS TSVT
CheckGhepDiem	Bool	Đã/chưa ghép điểm
UserGhepDiem	Varchar (50)	Tên ghép điểm
f_tenmhvn	Nvarchar (100)	Tên môn học
TuiBaiThi	Int	Túi bài thi
isDownloadFileDiem	Bool	Đã/chưa tải file điểm

Bảng 4.7 Bảng DanhSachThi

Trường	Kiểu dữ liệu	Ý nghĩa
IdDanhSachThi	Double	Id danh sách thi
f_masv	Varchar (15)	Mã sinh viên
f_manh	Varchar (15)	Mã môn học
f_holotvn	Nvarchar (50)	Họ lót sinh viên
f_tenvn	Nvarchar (50)	Tên sinh viên
f_ngaysinh	Datetime	Ngày sinh
sobaodanh	Int	Số báo danh
f_tenlop	Varchar (10)	Mã lớp

f_tenmhvn	Nvarchar (50)	Tên môn học
ngaythi	Datetime	Ngày thi
phongthi	Varchar (10)	Phòng thi
tietbatdau	SmallInt	Tiết bắt đầu
sotiet	SmallInt	Số tiết
IdKyThi	Int	Id kỳ thi
IsActive	Bool	Có/không tham dự thi
TuiBaiThi	Int	Túi bài thi
SoThuTu	Int	Số thứ tự

Bảng 4.8 Bảng DanhSachPhach

Trường	Kiểu dữ liệu	Ý nghĩa
IdDanhSachPhach	Double	Id danh sách phách
SoPhach	Int	Số phách
IdDanhSachThi	Double	Id danh sách thi
IdKyThi	Int	Id kỳ thi

4.3. Cài đặt hệ thống phần mềm

4.3.1 Dữ liệu thực nghiệm

Tất cả các file dữ liệu đầu vào và mẫu dữ liệu đầu ra được cung cấp bởi Phòng Đảm bảo chất lượng giáo dục, Trường Đại học Mở - Địa chất, cụ thể:

- Danh sách thi (bao gồm các thông tin về sinh viên, môn thi, ngày thi...)
- File phách (được sinh theo mẫu yêu cầu của phòng Đảm bảo chất lượng giáo dục) sử dụng để dán phách bài thi.
- File danh sách thí sinh vắng thi (để nhập thông tin thí sinh vắng thi vào hệ thống).

- File danh sách thí sinh phúc khảo bài thi.
- File điểm thi (file được xuất từ hệ thống, được sử dụng làm dữ liệu đầu vào cho hệ thống Edusoft).

4.3.2 Kết quả thực nghiệm

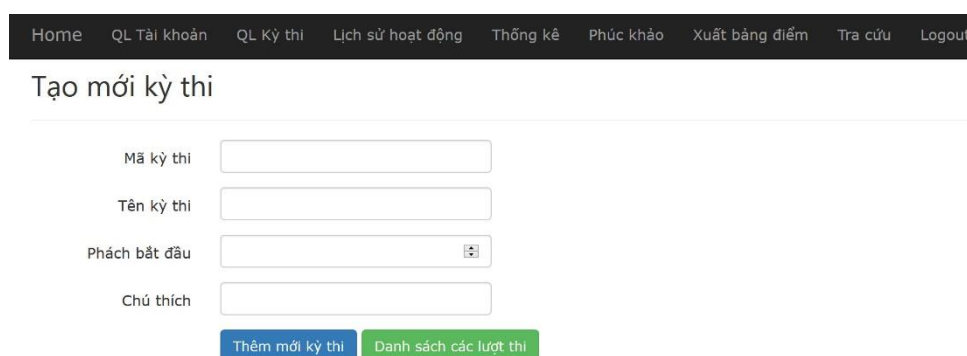
Một số giao diện chức năng chính của phần mềm:

Chức năng Quản lý kỳ thi: Trong chức năng quản lý kỳ thi → Danh sách thí sinh, quản trị viên có thể quản trị viên có thể thực hiện các chức năng sau:

- Tải file mẫu để ghi dữ liệu thí sinh dự thi sau đó upload lên hệ thống.
- Thêm thí sinh/thêm bổ sung thí sinh vào kỳ thi bằng cách upload file excel (theo định dạng có sẵn, người dùng có thể tải file mẫu về) chứa dữ liệu đầu vào.
- Sinh phách/sinh phách bổ sung cho kỳ thi dựa trên dữ liệu về các thí sinh có trong kỳ thi.
- Tải file phách của kỳ thi theo định dạng Word hoặc PDF



Hình 4.4. Giao diện chức năng Quản lý kỳ thi.

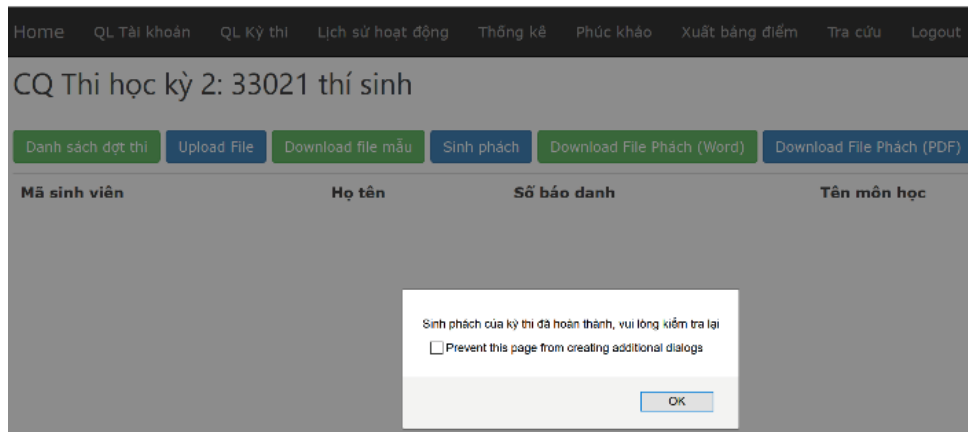


Hình 4.5. Giao diện chức năng Tạo mới kỳ thi.



Hình 4.6. Giao diện chức năng Upload file danh sách thí sinh dự thi

Chức năng Sinh phách: Sau khi đăng nhập vào hệ thống vào truy cập vào chức năng, Quản lý Kỳ thi → Danh sách thí sinh, Để tiến hành sinh phách cho kỳ thi Quản trị (Admin) nhấn vào nút lệnh Sinh phách. Sau khi sinh phách thành công, Admin sẽ lưu file phách về, phách được in và dán phách vào bài thi.



Hình 4.7. Thông báo khi sinh phách thành công

TRƯỜNG ĐẠI HỌC MÔ ĐỊA CHẤT - PHÒNG ĐAM BẢO CHẤT LƯỢNG GIÁO DỤC			
192HN - CQ Thi Học kỳ 2		Tài liệu mật 1/1,263	
Tối: số 1- 4010102 - Giải tích 1			
1-1621060568-27188 Đặng Công Chí	 27188	2-1621060274-23443 Nguyễn Minh Chiến	 23443
3-1521030010-20883 Đặng Hữu Chủ	 20883	4-1721060161-13401 Đỗ Ngọc Chung	 13401
5-1611060020-15367 Lê Thành Đạt	 15367	6-1721050135-28500 Lê Thành Đạt	 28500
7-1621060051-39764 Trần Minh Đại	 39764	8-1521050035-13165 Trần Tiến Đại	 13165
9-1621070005-36405 Nguyễn Văn Đình	 36405	10-1621040215-27542 Đặng Văn Dã	 27542
11-1624010073-42262 Nguyễn Đức Dũng	 42262	12-1621040042-37627 Trần Mạnh Dũng	 37627
13-1621050353-18407 Lại Đức Dương	 18407	14-1721040078-35330 Đào Thị Hà	 35330
15-1621060895-27320 Lê Thanh Hải	 27320	16-1721060335-39528 Nguyễn Hữu Hải	 39528
17-1621060567-17227 Nguyễn Quang Hiệp	 17227	18-1621010186-26771 Bùi Khắc Hưng	 26771
19-1621060481-12311 Nguyễn Hiền Hoàng	 12311	20-1621020218-41786 Phạm Văn Hòa	 41786
21-1621030238-12657 Trần Đức Huy	 12657	22-1621010018-21488 Nguyễn Trọng Huỳnh	 21488
23-1621050314-40690 Lê Duy Khánh	 40690	24-1621060158-20828 Nguyễn Thế Lợi	 20828
25-1621060184-25861 Nguyễn Công Minh	 25861	26-1721030129-33670 Tạ Tấn Minh	 33670
27-1621060979-31610 Nguyễn Anh Mười	 31610	28-1621010217-33793 Nguyễn Bá Nam	 33793
29-1621050731-26808 Trần Thị Ngọc	 26808	30-1621070084-12488 Nguyễn Thiện Nhân	 12488

Hình 4.8. File phách sau khi in ra

Chức năng nhập điểm: Sau khi đăng nhập vào hệ thống với quyền Chuyên viên, người dùng truy cập vào chức năng Kiểm tra phách, và tiến hành nhập danh sách thí sinh vắng thi vào hệ thống. Sau khi Nhập danh sách thí sinh vắng thi, người dùng (với quyền Giảng viên hoặc Chuyên viên) truy cập vào chức năng Kiểm tra phách, tiến hành ghép điểm theo môn thi, túi bài thi bằng cách click vào nút lệnh “Upload File Điểm”, sau đó sẽ hiển thị phần chọn file điểm để upload lên hệ thống. Trong trường hợp ghép điểm thành công hay không thành công, hệ thống đều có thông báo cho người dùng biết cùng với nguyên nhân cụ thể.

Trang chủ Đổi mật khẩu Log out

CQ Thi học kỳ 2

Browse... Le-192HN27...161652.xls
Upload

Search by subject Id... Nhập danh sách thí sinh vắng thi từ file

STT	Mã môn	Môn thi	Túi bài thi	Thao tác
1	4090107	An toàn điện	278	Hoàn thành
2	7060302	An toàn và Kỹ thuật môi trường trong CN Hóa học	167	Hoàn thành
3	4010403	Autocad + TH	10366	Nhập DSSV vắng thi Upload File Điểm
4	7050602	Biên tập bản đồ số	10368	Nhập DSSV vắng thi Upload File Điểm
5	4040403	Các mỏ khoáng không kim loại +TH	142	Nhập DSSV vắng thi Upload File Điểm

Hình 4.9. Giao diện Upload điểm lên hệ thống

Trang chủ Đổi mật khẩu Log out

CQ Thi học kỳ 2

Thông tin ghép phách môn An toàn điện, túi số 278: Không có phách nào hợp lệ, vui lòng kiểm tra lại, lý do có thể do: Bạn đã chọn nhầm file điểm, môn thi, túi bài thi hoặc kỳ thi

Trang chủ Đổi mật khẩu Log out

CQ Thi học kỳ 2

Thông tin ghép phách môn An toàn điện, túi số 278: Các phách không hợp lệ bao gồm: phách thứ 1, 5, 7

Trang chủ Đổi mật khẩu Log out

CQ Thi học kỳ 2

Thông tin ghép phách môn An toàn điện, túi số 278: Các phách sau không có điểm: 11378, 14622, 21055

Trang chủ Đổi mật khẩu Log out

CQ Thi học kỳ 2

Thông tin ghép phách môn An toàn điện, túi số 278: Ghép phách môn thi đã hoàn thành

Hình 4.10. Giao diện một số thông báo khi Ghép điểm

CQ Thi học kỳ 2

107						
STT	Mã môn	Môn thi	Túi bài thi	Nhập DSSV vắng thi	Ghép điểm thi	
1	4090107	An toàn điện	278	✓	✓	
116	7110107	Hóa học môi trường + TN (2+1)	335	✗	✗	
177	4070107	Luật kinh tế	206	✗	✗	
178	4070107	Luật kinh tế	207	✗	✗	
179	7070107	Luật Kinh tế	208	✗	✗	
180	7070107	Luật Kinh tế	209	✗	✗	
181	7070107	Luật Kinh tế	210	✗	✗	
182	7070107	Luật Kinh tế	211	✗	✗	

Hình 4.11. Giao diện chức năng mở lại chức năng nhập danh sách vắng thi và ghép điểm thi

Chức năng Thống kê: Thống kê tiến độ nhập dữ liệu thí sinh vắng thi và ghép điểm theo môn thi và túi bài thi, cho phép xuất dữ liệu thống kê ra file excel.

Home	QL Tài khoản	QL Kỳ thi	Lịch sử hoạt động	Thống kê	Phúc khảo	Xuất bảng điểm	Tra cứu	Logout
CQ Thi học kỳ 2								
Nhập mã môn học để tìm...		Nhập tên chuyên viên...		Xuất dữ liệu				
STT	Mã môn	Môn thi	Túi bài thi	Nhập DS Vắng thi	Ghép điểm			
1	4090107	An toàn điện	278	✓	✓			
2	7060302	An toàn và Kỹ thuật môi trường trong CN Hóa học	167	✓	✓			
3	4010403	Autocad + TH	10366	✓	✗			
4	7050602	Biên tập bản đồ số	10368	✗	✗			
5	4040403	Các mô khoáng khổng kim loại +TH	142	✗	✗			
6	4040402	Các mô khoáng kim loại + TH	141	✗	✗			
7	4040404	Các mô nhiên liệu	143	✗	✗			
8	4090215	Các phần tử tự động + TH	294	✗	✗			
9	7060304	Các phương pháp phân tích dụng cụ	169	✗	✗			
10	4060313	Các phương pháp phân tích dụng cụ	175	✗	✗			
11	4100101	Các phương pháp số	323	✗	✗			
12	7030103	Các quá trình sản xuất trên mô lý thiên	108	✗	✗			

Hình 4.12. Giao diện chức năng Thống kê.

Chức năng Nhập danh sách vắng thi và ghép điểm: Với vai trò Chuyên viên hoặc Giảng viên, người dùng có thể tiến hành nhập danh sách thí sinh vắng thi hoặc ghép điểm (upload file excel đầu vào là dữ liệu đầu ra của phần mềm nhập điểm) cho môn thi theo theo túi bài thi

Home	QL Tài khoản	QL Kỳ thi	Lịch sử hoạt động	Thống kê	Phúc khảo	Xuất bảng điểm	Tra cứu	Logout
------	--------------	-----------	-------------------	----------	-----------	----------------	---------	--------

CQ Thi học kỳ 2

STT	Mã môn	Môn thi	Túi bài thi	Nhập DS Vắng thi	Ghép điểm
1	4090107	An toàn điện	278	✓	✓
2	7060302	An toàn và Kỹ thuật môi trường trong CN Hóa học	167	✓	✓
3	4010403	Autocad + TH	10366	✓	✗
4	7050602	Biên tập bản đồ số	10368	✗	✗
5	4040403	Các mỏ khoáng không kim loại + TH	142	✗	✗
6	4040402	Các mỏ khoáng kim loại + TH	141	✗	✗
7	4040404	Các mỏ nhiên liệu	143	✗	✗
8	4090215	Các phần tử tự động + TH	294	✗	✗
9	7060304	Các phương pháp phân tích dụng cụ	169	✗	✗
10	4060313	Các phương pháp phân tích dụng cụ	175	✗	✗
11	4100101	Các phương pháp số	323	✗	✗
12	7030103	Các quá trình sản xuất trên mỏ lộ thiên	108	✗	✗

Hình 4.13. Giao diện chức năng nhập danh sách vắng thi và ghép điểm

Trang chủ	Đổi mật khẩu	Log out
-----------	--------------	---------

Danh sách kỳ thi

Mã kỳ thi	Tên kỳ thi	
HN_CQ_Ky2_20192020	Hà Nội Đại học Chính Quy Thi Học kỳ 2	Xử lý kỳ thi
192 HN	CQ Thi học kỳ 2	Xử lý kỳ thi

Hình 4.14. Giao diện chức năng xử lý kỳ thi với vai trò chuyên viên/giảng viên

Trang chủ Đổi mật khẩu Log out

CQ Thi học kỳ 2

Le-192HN27...161652.xls

Search by subject Id...

STT	Mã môn	Môn thi	Túi bài thi	Thao tác
1	4090107	An toàn điện	278	Hoàn thành
2	7060302	An toàn và Kỹ thuật môi trường trong CN Hóa học	167	Hoàn thành
3	4010403	Autocad + TH	10366	Nhập DSSV vắng thi Upload File Điểm
4	7050602	Biên tập bản đồ số	10368	Nhập DSSV vắng thi Upload File Điểm

Hình 4.15. Giao diện chức năng nhập thí sinh vắng thi và ghép điểm từ file excel với vai trò chuyên viên/giảng viên

Trang chủ Đổi mật khẩu Log out

CQ Thi học kỳ 2

Bỏ tích checkbox đối với sinh viên vắng thi bao gồm cả sinh viên bị cấm thi

Thông tin môn thi: Biên tập bản đồ số - 10368

Search by StudentID...

STT	Mã sinh viên	Họ tên	Có mặt
1	1821030164	Nguyễn Đình Tường An	☒
2	1821030017	Đình Hồ Bắc	☒
3	1821030247	Nguyễn Đức Chính	☒
4	1821030019	Đỗ Quang Danh	☒
5	1621070212	Nguyễn Thành Đô	☒
6	1821030159	Tạ Minh Đức	☒
7	1821030084	Nguyễn Văn Dương	☒
8	1621030138	Trần Hoàng Hà	☒

Hình 4.16. Giao diện chức năng nhập thí sinh vắng thi cho một túi bài thi với vai trò chuyên viên/giảng viên

KẾT LUẬN

Hệ thống phần mềm sinh phách và ghép điểm cho phòng Đảm bảo chất lượng giáo dục, trường Đại học Mở - Địa chất được phát triển trên nền tảng Web, sử dụng ngôn ngữ lập trình ASP.NET C# theo mô hình MVC và Hệ quản trị cơ sở dữ liệu SQL Server, cùng với những công nghệ và kỹ thuật mới trong phát triển ứng dụng Web hiện nay. Cho phép hệ thống chạy trên đa nền tảng phần cứng và đáp ứng được các yêu cầu về tốc độ xử lý, độ tin cậy cao, an toàn bảo mật...

Đề tài “*Xây dựng phần mềm sinh phách và ghép điểm cho các học phần khảo thí của Trường Đại học Mở - Địa chất*” Mã số NV19-46 đã hoàn thành theo đúng thời gian yêu cầu và đạt được các kết quả chính như sau:

- Xây dựng được phần mềm quản lý sinh phách và ghép điểm đáp ứng yêu cầu đặt ra của phòng Đảm bảo chất lượng giáo dục. Cụ thể:
 - + Phần mềm cho phép sinh các phách ngẫu nhiên 5 số kèm theo mã vạch, không trùng lặp; cho phép sinh phách và quản lý tối đa 99999 phách ngẫu nhiên trong một kỳ thi.
 - + Phần mềm cho phép ghép phách-điểm của bài thi với dữ liệu thí sinh; cho phép cảnh báo trong trường hợp phách đang ghép thuộc về thí sinh vắng thi hoặc không tồn tại trong túi bài đang ghép phách-điểm.
 - + Ngoài ra, phần mềm có một số hỗ trợ người dùng là chuyên viên, giảng viên thống kê tình trạng nhập điểm, ghép phách-điểm của các túi bài thi; hỗ trợ admin xuất dữ liệu có cấu trúc, đẩy điểm trực tiếp vào hệ thống quản lý đào tạo Edusoft của Nhà trường.
- Hoàn thiện báo cáo tổng kết đề tài đầy đủ nội dung và đúng theo mẫu quy định.

Hà Nội, ngày tháng 10 năm 2020

Nhóm tác giả

TÀI LIỆU THAM KHẢO

- [1]. Nguyễn Văn Ba, 2008. *Phát triển hệ thống hướng đối tượng với UML 2.0 và C++*, Nhà xuất bản Giáo dục.
- [2]. Microsoft Learning, *Introduction to Web Development with Visual Studio 2010*, Microsoft Official Course.
- [3]. Don Gosselin, *ASP.NET Programming with C# and SQL Server*, Cengage Learning.
- [4]. Trương Ninh Thuận, Đặng Đức Hạnh, 2013. *Giáo trình phân tích và thiết kế hướng đối tượng*. Nhà xuất bản Đại học Quốc gia Hà Nội.