**Mô tả hệ thống bằng ngôn ngữ tự nhiên**

**Các bước thực hiện**

* Bước 1: Giới thiệu mục đích hệ thống
* Bước 2: Phạm vi hệ thống: ai được dùng phần mềm/hệ thống? Mỗi người vào vào hệ thống được phép thực hiện các chức năng nào?
* Bước 3: Với mỗi chức năng mà người dùng được phép thực hiện ở bước 2, mô tả chi tiết hoạt động nghiệp vụ của chức năng đấy diễn ra như thế nào?
* Bước 4: Các đối tượng nào được quản lí, xử lí trong hệ thống? Mỗi đối tượng cần dùng/quản lí các thuộc tính nào?
* Bước 5: Quan hệ (số lượng) giữa các đối tượng đã nêu ở bước 4?

**Áp dụng**

***Bước 1: Mục đích của hệ thống***: Hệ thống trang web phục vụ công tác quản lí đăng kí tín chỉ của sinh viên, đăng kí giảng dạy của giảng viên, quản lí điểm các môn học của một trường đại học.

***Bước 2: Phạm vi hệ thống***: Những người được vào hệ thống và chức năng mỗi người được thực hiện khi vào hệ thống này được quy định như sau:

* Thành viên hệ thống:
  + Đăng nhập
  + Đăng xuất
  + Đổi mật khẩu cá nhân
* Sinh viên:
  + Được thực hiện các chức năng như thành viên
  + Đăng kí học, sửa thông tin đăng kí của mình
  + Xem lịch học của mình
  + Xem điểm của mình
* Giảng viên:
  + Đăng kí dạy, sửa thông tin đăng kí dạy
  + Nhập, sửa điểm các lớp mình dạy
  + Xem lịch dạy của mình
  + Xem thống kê liên quan đến các lớp mình dạy
* Nhân viên giáo vụ:
  + Quản lí thông tin sinh viên: thêm, xóa, sửa theo yêu cầu từ sinh viên
  + Quản lí thông tin giảng viên theo yêu cầu từ giảng viên
  + Quản lí thông tin môn học
  + Quản lí thông tin lớp học phần
* Nhân viên quản lí:
  + Xem các loại thống kê
* Nhân viên khảo thí:
  + Xuất bảng điểm theo yêu cầu của sinh viên

Những chức năng không đề cập đến thì mặc định là không thuộc phạm vi của hệ thống.

***Bước 3: Hoạt động nghiệp vụ của các chức năng***: Theo nguyên tắc, mỗi chức năng liệt kê trong bước 2 đều phải mô tả chi tiết. Tuy nhiên, trong phạm vi tài liệu này, chỉ có ba chức năng được mô tả chi tiết vì đây là các chức năng được dùng để minh họa cho các bước phân tích, thiết kế từ đầu đến cuối. Các chức năng còn lại coi như bài tập cho người học.

* *Sinh viên đăng kí học*: Sinh viên đăng nhập vào hệ thống -> chọn chức năng đăng kí tín chỉ (đang trong thời gian mở đăng kí mới được chọn) -> chọn kì đăng kí + ngành học (có thể có sinh viên học đồng thời hai chuyên ngành) -> hệ thống hiện danh sách các môn học có thể đăng kí (mã, tên môn học, số tín chỉ, mô tả), các lớp học phần đã đăng kí rồi, nếu có -> Sinh viên chọn môn học -> hệ thống hiện danh sách các lớp học phần của môn học đấy (mã, tên, sĩ số tối đa, sĩ số hiện tại, phòng học, giảng viên, lịch học hàng tuần vào các ngày nào trong tuần, kíp nào trong ngày): chỉ active các nhóm mà không bị trùng lịch học với các môn đã chon trước, các nhóm bị trùng lịch thì chỉ xem, không chọn được -> Sinh viên chọn lớp học phần mình thích -> hệ thống quay lại trang bắt đầu đăng kí với lớp học phần vừa chọn được bổ sung vào danh sách các lớp học phần đã chọn. Sinh viên lặp lại các bước trên cho đến khi chọn đủ số tín chỉ trong ngưỡng cho phép -> nút lưu được active -> Sinh viên click lưu thì thông tin đăng kí mới chính thức được lưu vào hệ thống, hệ thống quay về giao diện chính của sinh viên.
* *Giảng viên nhập điểm*: Giảng viên đăng nhập vào hệ thống -> chọn chức năng nhập điểm -> Chọn học kì đang active -> hệ thống hiện danh sách các môn học do giảng viên dạy của kì đã chọn (mã, tên, số tín chỉ, mô tả) -> Giảng viên click chọn môn học muốn nhập điểm -> Hệ thống hiện danh sách các lớp học phần do giảng viên dạy (mã, tên, sĩ số thực, phòng học, ngày học, kíp học) -> Giảng viên chọn 1 lớp học phần muốn nhập -> Hệ thống hiện danh sách các sinh viên đăng kí lớp học phần được chọn với điểm thành phần, nếu có: thứ tự, mã sinh viên, họ tên, các đầu điểm thành phần, điểm thi, cột trung bình môn và điểm chữ được tự tính sau khi nhập -> Giảng viên nhập đầu điểm muốn nhập cho tất cả sinh viên trong danh sách và click lưu -> hệ thống lưu điểm vào và quay về giao diện chính của giảng viên.
* *Quản lí xem thống kê theo loại học lực*: Nhân viên quản lí đăng nhập vào hệ thống -> chọn chức năng xem thống kê -> hệ thống hiện giao diện chọn thông tin thống kê -> chọn thống kê loại học lực -> Hệ thống hiện giao diện thống kê loại học lực -> Quản lí chọn học kì muốn thống kê -> Kết quả thống kê hiện lên, mỗi loại học lực trên một dòng, xếp theo thứ tự cao nhất đến thấp nhất của loại học lực trong bảng đánh giá (Ưu tú, xuất sắc, giỏi, khá, trung bình, yếu kém): thứ tự, loại học lực, tổng số sinh viên đạt loại đó, điểm trung bình sinh viên trong nhóm đạt loại đó trong học kì đã chọn -> Quản lí click vào một loại học lực -> Hệ thống hiện danh sách các sinh viên đạt loại học lực đấy lên, xếp theo thứ tự các ngành học, đến thứ tự abc của tên sinh viên: thứ tự, mã sinh viên, họ và tên, ngành học, khóa học, tổng số tín chỉ của học kì, điểm trung bình của học kì -> Quản lí click vào một sinh viên danh sách -> Hệ thống hiện lên danh sách các môn và kết quả của sinh viên đã học trong học kì đó, xếp theo thứ tự abc của tên môn học: thứ tự, tên môn học, số tín chỉ, điểm trung bình môn đó của sinh viên. Dòng cuối là tổng số tín chỉ, điểm trung bình cả học kì của sinh viên -> Quản lí click vào một môn học trong danh sách -> Hệ thống hiện lên điểm chi tiết của môn học của sinh viên: mã môn, tên môn, tổng tín chỉ, các đầu điểm thành phần dạng bảng: tên đầu điểm thành phần, tỉ lệ % tính của đầu điểm thành phần, điểm của sinh viên. Dòng cuối là điểm trung bình môn của sinh viên trong môn học đó.

***Bước 4: Thông tin các đối tượng cần xử lí, quản lí***:

Nhóm các thông tin liên quan đến con người:

* Thành viên: tên đăng nhập, mật khẩu, họ tên, địa chỉ, ngày sinh, email, số điện thoại
* Sinh viên: giống thành viên, có thêm: mã sinh viên. Theo mỗi ngành học còn có khóa học, ngành học
* Nhân viên: giống thành viên, có thêm: vị trí công việc.
* Nhân viên quản lí: giống nhân viên
* Nhân viên giáo vụ: giống nhân viên
* Nhân viên khảo thí: giống nhân viên
* Giảng viên: giống nhân viên

Nhóm các thông tin liên quan đến cơ sở vật chất:

* Tòa nhà: tên, mô tả
* Phòng học: tên, sức chứa tối đa, mô tả

Nhóm các thông tin liên quan đến đơn vị, tổ chức:

* Trường: tên, địa chỉ, mô tả
* Khoa: tên, mô tả
* Ngành học: tên, mô tả
* Bộ môn: tên, mô tả

Nhóm các thông tin liên quan đến chuyên môn, vận hành:

* Năm học: tên, mô tả
* Kì học: tên, mô tả
* Tuần học: tên, mô tả
* Ngày trong tuần: tên, mô tả
* Kíp học trong ngày: tên, mô tả
* Môn học: tên, số tín chỉ, mô tả
* Lớp học phần: tên, mô tả, sĩ số tối đa, sĩ số hiện tại, giảng viên dạy, phòng học, tuần nào học ngày nào, kíp nào.

Nhóm thông tin liên quan đến thống kê:

* Thống kê theo loại học lực
* Thống kê sinh viên theo kết quả học
* Thống kê các môn học theo kết quả học
* Thống kê giảng viên theo: số giờ dạy, kết quả học
* Thống kê học kì theo số sinh viên

***Bước 5: Quan hệ giữa các đối tượng, thông tin***:

* Một trường có nhiều khoa
* Một khoa có nhiều bộ môn
* Một khoa có nhiều ngành học
* Một bộ môn quản lí chuyên môn nhiều môn học
* Một bộ môn có nhiều giảng viên
* Một năm học có nhiều học kì
* Một học kì liên quan đến nhiều năm học. Một năm học + một học kì tạo ra một kì học (kì học # học kì).
* Một kì học có nhiều môn học
* Một môn học, vào một kì học, có nhiều lớp học phần
* Một lớp học phần có thể học vào nhiều buổi, mỗi buổi có thể liên quan đến 1 tuần khác nhau, 1 ngày khác nhau, 1 kíp khác nhau, 1 phòng học khác nhau, 1 giảng viên khác nhau.
* Một giảng viên có thể dạy nhiều môn học trong mỗi kì học
* Một môn học, trong một kì học, giảng viên có thể dạy nhiều lớp học phần khác nhau, miễn sao không trùng lịch buổi nào.
* Một lớp học phần, có thể có nhiều giảng viên dạy. Nhưng mỗi buổi học chỉ có một giảng viên dạy.
* Một môn học có nhiều đầu điểm thành phần.
* Mỗi đầu điểm thành phần, đối với mỗi môn học, có tỉ lệ % tính điểm nhất định.
* Một tuần có thể có nhiều buổi dạy/học
* Một ngày có thể có nhiều buổi học/dạy
* Một kíp có thể có nhiều buổi học/dạy của nhiều lớp học phần khác nhau
* Một phòng học có thể có nhiều lớp học phần vào học ở những buổi khác nhau.
* Một sinh viên có thể đăng kí học nhiều ngành khác nhau (tối đa 2 ngành đồng thời).
* Với mỗi ngành, sinh viên phải học một số môn nhất định, và điểm tính theo từng ngành. Các môn trùng nhau giữa các ngành thì sinh viên chỉ phải học 1 lần, qua là được.
* Mỗi sinh viên, mỗi môn học, có một diểm trung bình môn.

**Mô tả hệ thống bằng UML: use case**

**Các bước thực hiện**

Tóm tắt các bước thực hiện để vẽ biểu đồ use case tổng quan:

* Bước 1: Đề xuất các actor. Với mỗi người dùng, đề xuất thành một actor tương ứng. Nếu các actor có đặc điểm gì chung, có thể đề xuất actor trừu tượng thành actor cha của các actor tương ứng. Ngoài ra, cần xem xét cần có các actor gián tiếp tác động vào để thực hiện các chức năng hay không.
* Bước 2: Đề xuất use case. Với mỗi chức năng, đề xuất thành một use case tương ứng
* Bước 3: Mịn hóa các use case. Nếu có ít nhất 2 use case trùng nhau, cần xem xét gộp lại thành 1. Nếu gộp lại gây hiểu nhầm về số các actor tác động vào, thì có thể dùng use case trừu tượng cho các use case giống nhau, mỗi use case con liên quan đến nhóm các actor tương ứng mà thôi.

Tóm tắt các bước thực hiện để vẽ biểu đồ use case chi tiết:

* Bước 1: Trích phần use case của chức năng tương ứng từ biểu đồ use case tổng quan.
* Bước 2: Phân rã use case chính thành các use case con: mỗi giao diện (hoặc một số giao diện) tương tác với người dùng có thể đề xuất thành một use case con.
* Bước 3: Xác định quan hệ của use case con với use case chính: generalization, include, hay extend.
* Bước 4: Gộp các use case con tương tự nhau bằng cách dùng các use case trừu tượng tổng quát hơn.

Lưu ý tránh sai sót trong biểu đồ use case:

* Tên use case phải là động từ chỉ hành động của actor. Không nên là động từ chỉ hành động của hệ thống. Cũng không nên là danh từ, tính từ…
* Mỗi use case phải có tương tác với ít nhất một actor, có thể là trực tiếp hoặc gián tiếp: phải tồn tại ít nhất một đường đi từ một actor nào đó đến use case theo hướng: quan hệ include thì theo chiều mũi tên, quan hệ extend thì ngược chiều mũi tên, quan hệ kế thừa thì gộp lại.

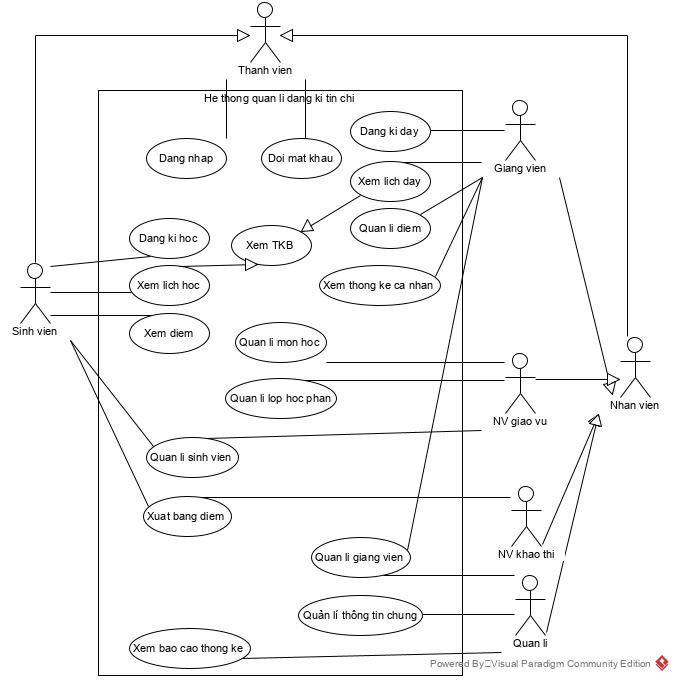
**Áp dụng**

***a. Biểu đồ use case tổng quan***

Ta có thể đề xuất được các actor của hệ thống: sinh viên, giảng viên, quản lí, giáo vụ, và khảo thí. Tất cả đều có chức năng giống thành viên nên kế thừa từ thành viên. Riêng giảng viên, quản lí, giáo vụ, khảo thí còn kế thừa từ actor nhân viên của trường. Nhân viên kế thừa trực tiếp từ thành viên.

Các chức năng tương ứng với từng actor:

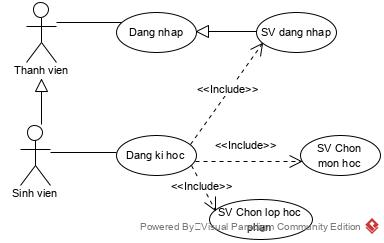
* Thành viên: đăng nhập, đổi mật khẩu
* Sinh viên: đăng kí học, xem lịch học, xem điểm. Ngoài ra có thể tham gia gián tiếp vào các chức năng: quản lí thông tin sinh viên, xuất bảng điểm cho sinh viên.
* Giảng viên: đăng kí dạy, xem lịch dạy, nhập điểm, xem thống kê cá nhân. Ngoài ra có thể tham gia gián tiếp vào chức năng quản lí thông tin giảng viên.
* Giáo vụ: quản lí thông tin sinh viên theo yêu cầu sinh viên, quản lí môn học, lớp học phần.
* Khảo thí: xuất bảng điểm theo yêu cầu sinh viên.
* Quản lí: quản lí thông tin chung, quản lí thông tin giảng viên theo yêu cầu giảng viên, xem các loại báo cáo thống kê..



Trong số các chức năng này, việc xem lịch học của sinh viên là tương tự chức năng xem lịch dạy của giảng viên. Nên hai use case này được cho kế thừa từ use case xem TKB. Như vậy, biểu đồ use case tổng quan của hệ thống được trình bày như Hình 3.1, với mô tả các use case như sau:

* Đăng kí học: UC này cho phép sinh viên vào hệ thống đăng kí các môn học theo nguyện vọng cá nhân.
* Xem lịch học: UC này cho phép sinh viên vào hệ thống xem lịch học cá nhân
* Xem điểm: UC này cho phép sinh viên vào hệ thống xem kết quả các môn học của mình.
* Đăng kí dạy: UC này cho phép giảng viên vào hệ thống để đăng kí lịch dạy của mình vào đầu mỗi kì học.
* Xem lịch dạy: UC này cho phép giảng viên xem lịch dạy cá nhân
* Nhập điểm: UC này cho phép giảng viên nhập điểm các lớp học phần do mình dạy
* Xem thống kê cá nhân: UC này cho phép giảng viên xem các thống kê cá nhân
* Quản lí thông tin giảng viên: UC này cho phép giáo vụ quản lí thông tin giảng viên theo yêu cầu của giảng viên tương ứng.
* Quản lí thông tin sinh viên: UC này cho phép giáo vụ quản lí thông tin sinh viên theo yêu cầu từ sinh viên tương ứng.
* Quản lí thông tin môn học: UC này cho phép giáo vụ quản lí thông tin các môn học
* Quản lí thông tin lớp học phần: UC này cho phép giáo vụ quản lí thông tin các lớp học phần.
* Xuất bảng điểm: UC này cho phép khảo thí xuất bảng điểm cho sinh viên theo yêu cầu từ sinh viên.
* Xem thống kê: UC này cho phép nhân viên quản lí xem các loại báo cáo thống kê.

***b. Use chi tiết của đăng kí học***



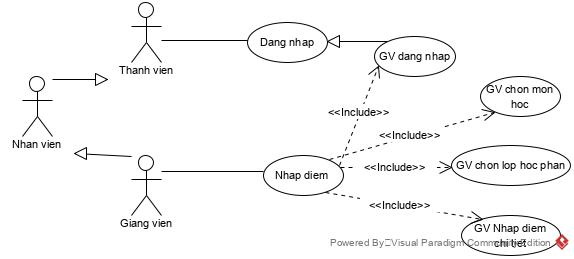
Chức năng đăng kí học có các giao diện tương tác với sinh viên:

* Đăng nhập -> đề xuất UC đăng nhập
* Đăng kí -> đề xuất UC đăng kí
* Chọn môn học -> đề xuất UC chọn môn học
* Chọn lớp học phần -> đề xuất UC chọn lớp học phần
* Đăng nhập, chọn môn học, chọn lớp học phần là bắt buộc mới hoàn thành được việc đăng kí, do đó UC đăng kí include các UC này.

Như vậy, biểu đồ UC chi tiết cho modul đăng kí học được trình bày trong Hình trên. Các UC được mô tả như sau:

* Chọn môn học: UC này cho phép sinh viên chọn môn học để đăng kí học
* Chọn lớp học phần: UC này cho phép sinh viên chọn lớp học phần để đăng kí học.

***c. Use case chi tiết modul nhập điểm***



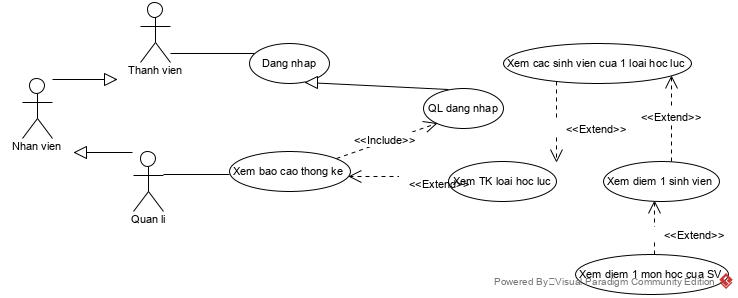
Trong chức năng nhập điểm, giảng viên phải tương tác với các giao diện:

* Đăng nhập -> thống nhất với UC đăng nhập
* Chọn kì học + môn học -> đề xuất UC chọn môn học
* Chọn lớp học phần -> đề xuất UC chọn lớp học phần
* Nhập điểm chi tiết -> đề xuất UC nhập điểm chi tiết
* Các UC trên đều bắt buộc thực hiện mới hoàn thành việc nhập điểm -> chúng đều bị chứa trong UC nhập điểm

Như vậy, biểu đồ UC chi tiết cho modul nhập điểm được trình bày trong hình trên. Trong đó các UC được mô tả như sau:

* Chọn môn học: UC này cho phép giảng viên chọn môn học để nhập điểm
* Chọn lớp học phần: UC này cho phép giảng viên chọn lớp học phần để nhập điểm
* Nhập điểm chi tiết: UC cho phép giảng viên nhập/sửa điểm chi tiết từng đầu điểm thành phần của từng sinh viên trong một lớp học phần do mình dạy

***d. Use case chi tiết cho modul xem thống kê loại học lực***



Trong chức năng này, nhân viên quản lí có thể phải tương tác với các giao diện:

* Đăng nhập -> thống nhất với UC đăng nhập
* Xem thống kê loại học lực -> đề xuất UC xem TK loại học lực
* Xem thống kê các sinh viên của 1 loại học lực -> đề xuất UC xem TK sinh viên của loại học lực.
* Xem kết quả của một sinh viên -> đề xuất UC xem điểm các môn học của sinh viên
* Xêm chi tiết một môn học của sinh viên -> đề xuất UC xem điểm chi tiết 1 môn học của sinh viên.
* Các giao diện thống kê lần lượt theo dạng: giao diện sau là tùy chọn từ giao diện trước. Do đó, chúng có quan hệ mở rộng lần lượt cái sau từ cái trước.

Như vậy, biểu đồ UC chi tiết cho chức năng thống kê loại học lực được trình bày như trong hình vẽ. Trong đó, các UC chi tiết được mô tả như sau:

* Xem TK loại học lực: UC này cho phép NVQL xem thống kê số lượng sinh viên trong mỗi loại học lực của một kì học
* Xem TK sinh viên của một loại học lực: UC này cho phép NVQL xem TK kết quả các sinh viên của một loại học lực
* Xem TK điểm của một sinh viên: UC này cho phép NVQL xem kết quả các môn học của một sinh viên trong một kì học
* Xem kết quả một một học của một sinh viên: UC này cho phép NVQL xem kết quả chi tiết một môn học của một sinh viên.

**Kịch bản**

Lưu ý tránh sai sót trong kịch bản:

* Mỗi use case chính trong biểu đồ use case tổng quan thường có một kịch bản chuẩn và có thể có nhiều kịch bản ngoại lệ.
* Use case có bao nhiêu actor thì kịch bản cũng có bấy nhiêu actor
* Kịch bản càng chi tiết càng tốt: thông tin hệ thống hiện lên, thông tin người dùng nhập vào…

**Chức năng đăng kí học**

|  |  |
| --- | --- |
| Use case | Đăng kí học |
| Actor | Sinh viên |
| Tiền điều kiện | Sinh viên đã đăng nhập thành công, đang trong thời gian sinh viên được phép đăng kí môn học đầu học kì |
| Hậu điều kiện | Sinh viên đăng kí xong môn học cho học kì |
| Kịch bản chính | 1. Sau khi đăng nhập, từ giao diện chính, sinh viên A chọn chức năng đăng kí học cho học kì tới. 2. Giao diện chọn kì học đăng kí, chọn ngành học hiện lên. Có danh sách các kì học đang mở đăng kí. Danh sách các ngành học mà sinh viên đang học. Nút vào đăng kí. 3. Sinh viên chọn ngành học, chọn kì học muốn đăng kí và click vào đăng kí. 4. Giao diện đăng kí học hiện lên, có ô chọn kì học muốn đăng kí; bảng danh sách các môn học/lớp học phần đã đăng kí đang rỗng; nút tiếp tục và nút lưu chưa được active. 5. Sinh viên chọn học kì tới trong danh sách và click vào nút tiếp tục. 6. Giao diện hiện lên danh sách các môn học mà sinh viên có thể đăng kí trong học kì: https://softwaredesignhome.files.wordpress.com/2020/09/kb-dangki-b6.jpg  7. Sinh viên click chọn môn Lập trình hướng đối tượng 8. Giao diện các lớp học phần hiện ra: https://softwaredesignhome.files.wordpress.com/2020/09/kb-dangki-b8.jpg  9. Sinh viên lick nhóm số 1 10. Hệ thống quay lại giao diện bước 2, bảng các môn đã chọn cập nhật lại: https://softwaredesignhome.files.wordpress.com/2020/09/kb-dangki-b10.jpg  (Lặp lại các bước 4-10 cho đến khi đạt số tín chỉ tối thiểu, nút lưu sẽ active) 11. Sinh viên click vào nút lưu. 12. Hệ thống báo thành công và quay về giao diện chính của sinh viên. |
| Ngoại lệ | 6. Không có môn nào để đăng kí 8. Các lớp học phần đều hết slot hoặc trùng giờ học với các môn đã chọn trước nên không chọn được. |

**Chức năng nhập điểm**

|  |  |
| --- | --- |
| Use case | Nhập điểm |
| Actor | Giảng viên |
| Tiền điều kiện | Giảng viên đã đăng nhập thành công, đã có sinh viên đăng kí vào các lớp do giảng viên dạy |
| Hậu điều kiện | Giảng viên nhập xong điểm cho lớp học phần |
| Kịch bản chính | 1. Giảng viên chọn chức năng nhập điểm từu giao diện chính sau khi đăng nhập 2. Giao diện chọn kì học đang học hiện lên, có danh sách các kì học đang học (active) để chọn. Nút tiếp tục. 3. Giảng viên chọn 1 kì học muốn nhập điểm và click tiếp tục. 4. Giao diện chọn môn học hiện lên: danh sách các môn học do giảng viên dạy kì tương ứng: https://softwaredesignhome.files.wordpress.com/2020/09/kb-nhapdiem-b04.jpg 5. Giảng viên click chọn môn Lập trình hướng đối tượng. 6. Giao diện các lớp học phần do giảng viên dạy hiện lên: https://softwaredesignhome.files.wordpress.com/2020/09/kb-nhapdiem-b06.jpg 7. Giảng viên click vào nhóm số 1 8. Giao diện nhập điểm chi tiết cho nhóm số 1 hiện lên: có thông tin nhóm số 1, môn học lập trình hướng đối tượng, và danh sách điểm sinh viên, dưới cùng là nút lưu: https://softwaredesignhome.files.wordpress.com/2020/09/kb-nhapdiem-b8.jpg 9. Giảng viên nhập đầu điểm thi: A: 5, C:6, D:7, S:8, V:9 và click lưu 10. Hệ thống báo thành công và quay về giao diện chính của giảng viên. |
| Ngoại lệ | 4. Không có môn học nào hiện lên 6. Chọn môn học xong, không có lớp học phần nào hiện lên 8. Chọn lớp xong, không có sinh viên nào hiện lên. |

**Chức năng xem thống kê loại học lực**

|  |  |
| --- | --- |
| Use case | Xem thống kê loại học lực |
| Actor | Nhân viên quản lí |
| Tiền điều kiện | Nhân viên quản lí đã đăng nhập thành công, kì học đã có điểm của tất cả các môn học |
| Hậu điều kiện |  |
| Kịch bản chính | 1. Quản lí A chọn chức năng xem thống kê từ giao diện chính sau khi đăng nhập 2. Giao diện chọn loại thống kê và kì học hiện lên. Có danh sách các loại thống kê: loại học lực, sinh viên, môn học… và anh sách các kì học đã kết thúc. 3. Quản lí chọn chức năng thống kê loại học lực, thống kê học kì 1 năm học 2019-2020 4. Giao diện thống kê loại học lực, danh sách thống kê hiện lên: https://softwaredesignhome.files.wordpress.com/2020/09/kb-thongke-04.jpg 5. Quản lí click vào dòng Xuất sắc 6. Giao diện thống kê các sinh viên xuất sắc trong kì kiện lên: https://softwaredesignhome.files.wordpress.com/2020/09/kb-thongke-b06.jpg 7. Quản lí click vào sinh viên Đoàn văn H 8. Giao diện kết quả các môn học của sinh viên Đoàn văn H hiện lên: https://softwaredesignhome.files.wordpress.com/2020/09/kb-thongke-b08.jpg 9. Quản lí click vào môn Lập trình hướng đối tượng 10. Giao diện kết quả môn Lập trình hướng đối tượng của Đoàn Văn H hiện lên: https://softwaredesignhome.files.wordpress.com/2020/09/kb-thongke-b10.jpg |
| Ngoại lệ | 4. Các loại học lực đều có 0 sinh viên |

**Lớp thực thể (phân tích)**

**Các bước thực hiện**

Tóm tắt các bước trong phương pháp trích danh từ để trích lớp thực thể cho hệ thống:

* Bước 1: Mô tả ngắn gọn nhưng phải đầy đủ hệ thống trong một đoạn văn. Có thể mô tả lần lượt các chức năng của hệ thống theo từng người dùng để tránh bị sót. Có thể thay thế bước này bằng cách tổng hợp tất cả các kịch bản (bao gồm kịch bản chuẩn và kịch bản ngoại lệ) của tất cả các use case của hệ thống.
* Bước 2: Trích các danh từ xuất hiện trong đoạn văn của bước 1. Mỗi danh từ xuất hiện chỉ cần tính một lần.
* Bước 3: Đánh giá các danh từ.
  + Một là đề xuất danh từ thành một lớp thực thể.
  + Hai là đề xuất danh từ thành thuộc tính của lớp nào đó.
  + Ba là loại bỏ danh từ đó vì quá chung chung, trừu tượng, hoặc ngoài pham vi quản lí của hệ thống.
* Bước 4: Xác định quan hệ số lượng giữa các thực thể.
  + Nếu quan hệ 1-1 thì có thể giữ nguyên hoặc gộp lại.
  + Nếu quan hệ 1-n thì giữ nguyên.
  + Nếu quan hệ là n-n thì phải đề xuất các lớp thực thể trung gian để tách thành ít nhất 2 quan hệ 1-n.
* Bước 5: Xác định quan hệ đối tượng giữa các thực thể: quan hệ kế thừa, hợp thành (aggregation), gắn chặt (composition), liên kết (association), phụ thuộc (dependence)…

Lưu ý khi vẽ biểu đồ lớp thực thể pha phân tích:

* Tên lớp, tên thuộc tính có thể dùng ngôn ngữ tự nhiên vẫn được, tuy nhiên, nên đặt tên theo chuẩn code convention để vào pha thiết kế đỡ phải thay đổi.
* Thuộc tính chưa cần kiểu dữ liệu
* Các lớp thực thể chưa cần thuộc tính id trong pha phân tích.

Áp dụng cho hệ thống quản lí đăng kí tín chỉ, các bước được thực hiện như sau:

**Áp dụng**

***Bước 1: Mô tả hệ thống trong một đoạn văn***

Hệ thống là một trang web hỗ trợ quản lí đăng kí tín chỉ và điểm của sinh viên trong một trường đại học. Trong đó, sinh viên được phép đăng kí các môn học, lớp học phần theo lịch học cá nhân vào đầu mỗi học kì; xem lịch học hàng ngày với đầy đủ thông tin về kíp học, ngày học, tuần học, phòng học, tòa nhà; xem kết quả học chi tiết với từng môn học, từng đầu điểm thành phần của môn học, cũng như điểm trung bình tích lũy các kì học của từng năm học hay từ đầu khóa học theo ngành học của sinh viên. Giảng viên được phép đăng kí dạy các môn học, các lớp học phần theo nguyện vọng cá nhân, sau đó sẽ được bộ môn quản lí duyệt và phân công chính thức; nhập hay sửa điểm của các sinh viên trong các lớp học phần do mình dạy; xem lịch dạy hàng tuần; và có thể xem các thống kê cá nhân về tỉ lệ sinh viên qua môn, điểm trung bình của sinh viên theo các môn học hay lớp học phần do mình dạy. Nhân viên giáo vụ có thể quản lí thông tin liên quan đến các môn học; lên lịch hay điều chỉnh lịch học cho các lớp học phần; quản lí thông tin sinh viên theo yêu cầu của sinh viên; quản lí thông tin giảng viên theo yêu cầu của giảng viên. Nhân viên khảo thí có thể xuất bảng điểm toàn khóa theo yêu cầu của sinh viên. Nhân viên quản lí có thể xem các loại thống kê: thống kê loại học lực, thống kê sinh viên theo kết quả học, theo ngành học, theo khoa, theo năm học, hay học kì; thống kê giảng viên theo khối lượng giảng dạy, theo kết quả sinh viên, theo khoa hay bộ môn.

***Bước 2: Trích các danh từ xuất hiện trong đoạn văn***

* Các danh từ liên quan đến người: sinh viên, nhân viên, giảng viên, nhân viên giáo vụ, nhân viên khảo thí, nhân viên quản lí
* Các danh từ liên quan đến vật: trường đại học, toà nhà, phòng học, bảng điểm
* Các danh từ liên quan đến thông tin: Khoa, ngành học, bộ môn, năm học, học kì, môn học, lớp học phần, điểm thành phần, điểm tích lũy, điểm trung bình môn, tín chỉ, học lực, khóa học, lịch học, thời khóa biểu, tuần học, ngày học, kíp học, báo cáo thống kê, thống kê học lực, thống kê sinh viên, thống kê giảng viên, thống kê môn học, thống kê học kì.

***Bước 3: Đánh giá và lựa chọn các danh từ làm lớp thực thể hoặc thuộc tính***

* Các danh từ trừu tượng: hệ thống, trang web, báo cáo thống kê, thông tin -> loại.
* Các danh từ liên quan đến người:
  + Thành viên hệ thống -> Thanhvien (trừu tượng): tên, tên đăng nhập, mật khẩu, ngày sinh, địa chỉ, email, điện thoại
  + Sinh viên -> lớp Sinhien: kế thùa lớp Thành viên, mã sinh viên.
  + Nhân viên -> lớp Nhanvien (trừu tượng): kế thừa lớp thành viên, vị trí.
  + Giảng viên -> lớp Giangvien: kế thừa Thanhvien
  + Nhân viên giáo vụ -> lớp Giaovu: Kế thừa Thanhvien
  + Nhân viên khảo thí -> lớp NVKhaothi: kế thừa Thanhvien
  + Nhân viên quản lí -> lớp NVQuanli: Kế thừa Thanhvien
* Các danh từ liên quan đến vật:
  + Trường đại học -> lớp Truong: tên, địa chỉ, mô tả
  + Tòa nhà -> lớp Toanha: Tên, mô tả
  + Phòng học -> lớp Phonghoc: Tên, sức chứa, mô tả
  + Bảng điểm, là số nhiều của một điểm của một môn học nên không đề xuất lớp.
* Các danh từ thông tin:
  + Khoa + ngành -> lớp Khoa: Tên, mô tả
  + Bộ môn -> lớp Bomon: Tên, mô tả
  + Năm học -> lớp Namhoc: Tên, mô tả
  + Học kì -> lớp Hocki: Tên, mô tả
  + Môn học -> lớp Monhoc: Tên, số tín chỉ, mô tả
  + Lớp học phần -> lớp Lophocphan: Tên
  + Đầu điểm thành phần -> Daudiem: tên, mô tả
  + Điểm tích lũy, điểm trung mình môn là thông tin dẫn xuất nên đề xuất là thuộc tính của một lớp nào đó, quyết định sau.
  + Tín chỉ là thuộc tính của môn học
  + Học lực -> lớp Loaihocluc: tên, mô tả
  + Khóa học là thuộc tính của sinh viên học một ngành học.
  + Lịch học -> lớp Lichhoc: liên quan đến tuần học, ngày học, kíp học, phòng học, giảng viên.
  + Thời khóa biểu là số nhiều của lịch học.
  + Tuần học -> lớp Tuanhoc: tên, mô tả
  + Ngày học -> lớp Ngayhoc: tên thứ ngày trong tuần, mô tả
  + Kíp học -> lớp Kip: tên, mô tả
  + Thống kê loại học lực -> lớp TKLoaihocluc
  + Thống kê môn học -> lớp TKMonhoc
  + Thống kê giảng viên -> lớp TKGiangvien
  + Thống kê sinh viên -> TKSinhvien
  + Thống kê học kì -> TKHocki

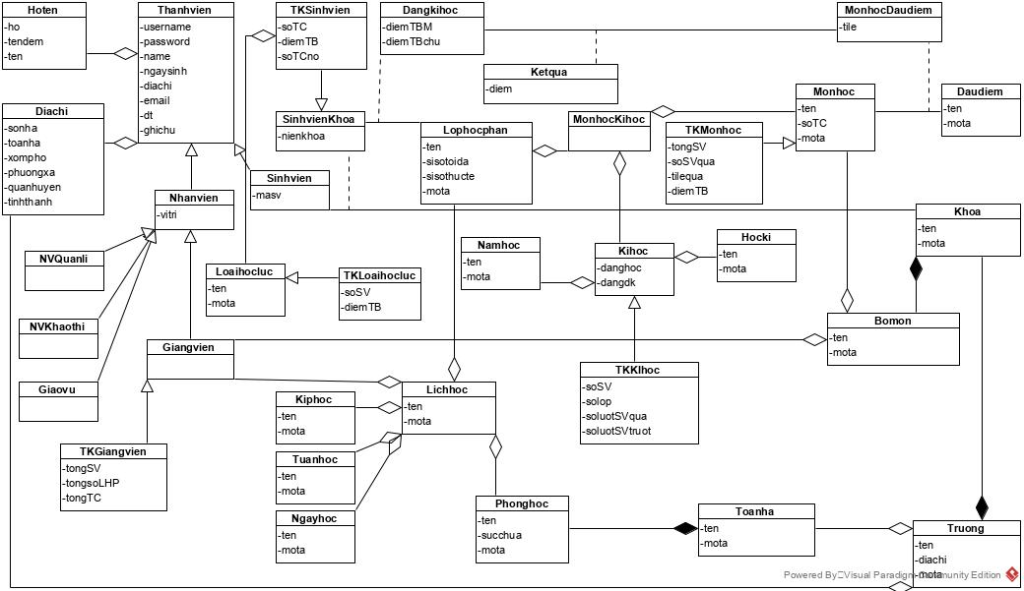
***Bước 4: Xác định quan hệ số lượng giữa các thực thể***

* Trường có nhiều khoa/ngành -> quan hệ giữa Truong và Khoa là 1-n
* Khoa có nhiều bộ môn -> Khoa – Bomon là 1-n
* Bộ môn có nhiều giảng viên -> Bomon – Giangvien là 1-n
* Trường có nhiều tòa nhà -> Truong – Toanha là 1-n
* Tòa nhà có nhiều phòng học -> Toanha – Phonghoc là 1-n
* Một năm học có nhiều học kì, một học kì có thể nằm trong nhiều năm học -> Namhoc – Hocki là n-n -> đề xuất lớp Kihoc kết nối giữa 1 năm học và 1 học kì (Kihoc # Hocki).
* Một kì học có nhiều môn học, một môn học có thể học trong nhiều kì học -> Kihoc – Monhoc là n-n -> đề xuất lớp MonhocKihoc xác định duy nhất một môn học dạy trong một kì học.
* Một môn học có một số môn học tiên quyết.
* Một môn học, trong một kì học, có nhiều lớp học phần -> MonhocKihoc – Lophocphan là 1-n.
* Một môn học có nhiều đầu điểm thành phần, một đầu điểm thành phần có thể xuất hiện trong nhiều môn học -> Monhoc – Daudiem là n-n -> đề xuất lớp MonhocDaudiem xác định duy nhất một đầu điểm trong một môn học.
* Một sinh viên có thể học nhiều ngành học/khoa, một ngành/khoa có thể có nhiều sinh viên -> Sinhvien – Khoa là n-n -> đề xuất lớp SinhvienKhoa xác định duy nhất một sinh viên học một ngành/khoa nhất định.
* Một sinh viên, đối với một ngành học, có thể học nhiều lớp học phần của nhiều môn học (mỗi môn học 1 lớp), mỗi lớp học phần có thể có nhiều sinh viên -> SingvienKhoa – Lophocphan là n-n -> đề xuất lớp Dangkihoc xác định duy nhất một sinh viên đăng kí vào một lớp học phần của một môn học nhất định.
* Một lớp học phần có nhiều buổi học (lịch học) khác nhau -> Lophocphan – Lichhoc là 1-n.
* Tương tự, các quan hệ Giangvien – Lichhoc, Phonghoc – Lichhoc – Tuanhoc – Lichhoc, Ngayhoc – Lichhoc, Kiphoc – Lichhoc đều là 1-n.
* Ngoài ra, thông tin Họ vè tên có thể tách thành một lớp thực thể riêng. thông tin địa chỉ cũng có thể tách thành một lớp thực thể riêng.

***Bước 5: Xác định quan hệ đối tượng giữa các thực thể***

* Khoa là thành phần của Trường
* Bộ môn là thành phần của Khoa.
* Giảng viên là thành phần của Bộ môn
* Môn học là thành phần của Bộ môn
* Tòa nhà là thành phần của Trường
* Phòng học là thành phần của tòa nhà
* Năm học và Học kì liên kết tạo ra Kì học duy nhất
* Sinh viên và ngành học liên kết tạo ra SinhvienKhoa duy nhất.
* Môn học và đầu điểm liên kết tạo ra MonhocDaudiem duy nhất
* Sinh viên và lớp học phần liên kết tạo ra Dangkihoc duy nhất
* Đăng kí học và MonhocDaudiem liên kết tạo ra điểm thành phần duy nhất cho sinh viên đối với môn học.
* Thông tin môn học nằm trong thông tin lớp học phần
* Thông tin lớp học phần, giảng viên, phòng học, tuần học, ngày học, kíp học nằm trong thông tin lịch học.
* Thông tin địa chỉ nằm trong thông tin về thành viên, trường
* Thông tin họ và tên nằm trong thông tin thành viên.
* Các lớp thống kê kế thừa từ các thực thể tương ứng.

Kết quả thu được biểu đồ lớp thực thể pha phân tích toàn hệ thống như trong hình.



**Biểu đồ lớp modul (phân tích)**

**Các bước thực hiện**

Tóm tắt các bước thực hiện để vẽ biểu đồ lớp pha phân tích:

* Bước 1: Mỗi giao diện xuất hiện có thể đề xuất thành một lớp biên.
* Bước 2: Với mỗi giao diện, xem xét cần các thành phần nào để nhận dữ liệu vào, hiện dữ liệu lên, hay submit dữ liệu vào hệ thống. Mỗi thành phần như vậy đề xuất thành một thuộc tính của giao diện, đặt tên với các tiền tố tương ứng:
  + in: tiền tố của tên thuộc tính nhận dữ liệu vào (ô text, ô chọn….)
  + out: tiền tố của tên thuộc tính hiện dữ liệu lên (bảng biểu, thông báo…)
  + sub: tiền tố của tên thuộc tính để submit dữ liệu (nút nhấn, liên kết…)
* Bước 3: Với mỗi chức năng phải thực hiện tầng dưới tầng giao diện, thực hiện đề xuất phương thức:
  + Đề xuất tên phương thức (có thể đặt tên gợi nhớ, và tuân thủ code convention)
  + Xác định tham số đầu vào là gì, tham số đầu ra là gì.
  + Đề xuất gán hành động tương ứng với chức năng này cho lớp thực thể nào. Theo nguyên tắc:
    - Nếu tham số đầu ra liên quan lớp thực thể nào thì gán phương thức này cho lớp đó
    - Nếu không, xét tham số đầu vào. Nếu tham số đầu vào chỉ liên quan đến 1 thực thể thì gán phương thức này cho thực thể đó. Nếu tham số vào liên quan đến hai hay nhiều thực thể, thì tìm thực thể nào nhỏ nhất mà chứa được càng nhiều càng tốt các tham số đầu vào, gán phương thức này cho lớp thực thể đó.
* Vẽ biểu đồ lớp pha phân tích cho từng modul.

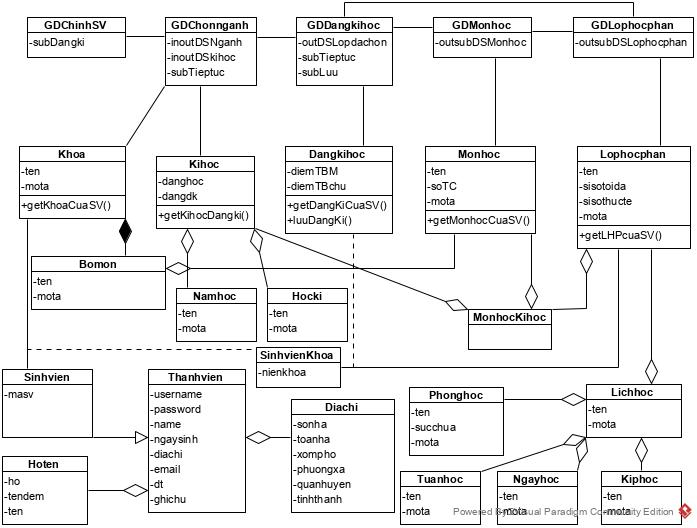
Lưu ý khi vẽ biểu đồ lớp pha phân tích:

* Tên lớp, tên thuộc tính có thể dùng ngôn ngữ tự nhiên vẫn được, tuy nhiên, nên đặt tên theo chuẩn code convention để vào pha thiết kế đỡ phải thay đổi.
* Thuộc tính chưa cần kiểu dữ liệu
* Quan hệ giữa các lớp thực thể phải thống nhất, đồng bộ với quan hệ giữa chúng trong biểu đồ lớp thực thể đã vẽ ở bước trước.

***a. Phân tích tĩnh modul đăng kí học***

* Ban đầu, giao diện chính của sinh viên -> đề xuất lớp GDChinhSV, cần các thành phần:
  + chọn đăng kí học: kiểu submit
* Bước 2, giao diện chọn kì học, ngành học -> đề xuất lớp GDChonnganh, cần có các thành phần:
  + Chọn ngành học để đăng kí: vừa input, vừa output
  + Chọn kì học: vừa input, vừa output
  + Nút chọn tiếp tục: kiểu submit
* Để có danh sách ngành học để chọn ở bước 2, cần xử lí dưới hệ thống:
  + Tìm các ngành học của sinh viên
  + input: sinh viên
  + output: danh sách Khoa
  + đề xuất phương thức getKhoacuaSV(), gán cho lớp Khoa.
* Để có danh sách kì học đnag mở đăng kí, cần xử lí bên dưới hệ thống:
  + Tìm các kì học đang mở đăng kí
  + input: không có
  + output: danh sách Kihoc
  + đề xuất phương thức getKihocDangki(), gán vào lớp Kihoc
* Bước 4, giao diện đăng kí -> đề xuất lớp GDDangki, cần có các thành phần:
  + Bảng danh sách các môn học: vừa output, vừa submit
  + Nút tiếp tục, nút lưu: kiểu submit
* Để có danh sách các đăng kí đã có, cần xử lí bên dưới hệ thống:
  + Tìm các môn học/lớp học phần đã đăng của sinh viên ở kì học đã chọn
  + input: học kì, sinh viên
  + output: danh sách Dangkihoc
  + đề xuất phương thức getDangkicuaSV(), gán vào lớp Dangkihoc
* Bước 6, giao diện chọn môn học -> đề xuất lớp GDMonhoc, cần các thành phần:
  + bảng danh sách các môn học có thể đăng kí: vừa output, vừa submit
* Để hiển thị được danh sách các môn học, cần xử lí trước dưới hệ thống:
  + Tìm các môn học có thể dăng kí của sinh viên
  + input: sinh viên, kì học
  + output: danh sách Monhoc
  + đề xuất phướng thức getMonhoccuaSV(), gán cho lớp Monhoc.
* Bước 8, giao diện lớp học phần -> đề xuất lớp GDLophocphan, cần có:
  + bảng danh sách lớp học phần: vừa output, vừa submit
* Để hiển thị danh sách lớp học phần, cần xử lí trước từ dưới hệ thống:
  + Tìm các lớp học phần của môn học
  + input: monhoc, kì học
  + output: danh sách Lophocphan
  + đề xuất phương thức getLHPcuaMH(), gán cho lớp Lophocphan.

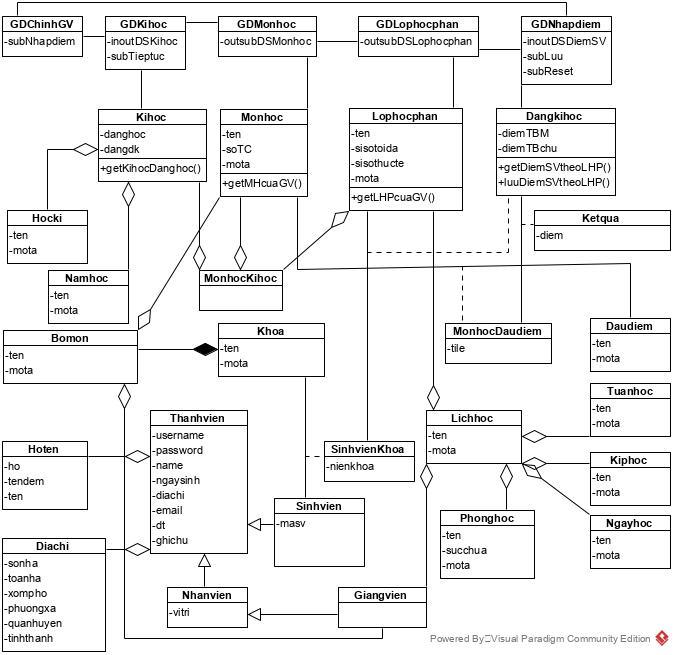
Kết quả thu được biểu đồ lớp cho modul đăng kí học như trong hình sau



***b. Phân tích tĩnh cho modul nhập điểm***

* Ban đầu, giao diện chính của giảng viên -> đề xuất lớp GDChinhGV, cần có:
  + chọn nhập/sửa điểm: kiểu submit
* Bước 2, giao diện chọn kì học -> đề xuất GDKihoc, cần có:
  + danh sách kì học: vừa input, vừa output
  + nút tiếp tục: kiểu submit
* Để có danh sách kì học đang học, cần xử lí dưới hệ thống:
  + Tìm các kì học đnag học
  + input: không có
  + output: danh sách các Kihoc
  + đề xuất phương thức getKihocDanghoc(), gán cho lớp Kihoc.
* Bước 4, giao diện chọn môn học -> đề xuất GDMonhoc, cần có:
  + danh sách các môn học: vừa output, vừa submit
* Sau khi chọn kì học, cần xử lí dưới hệ thống để lấy danh sách môn học:
  + Tìm các môn học của giảng viên đang dạy
  + input: giảng viên, kì học
  + output: danh sách các MonhocDaudiem
  + đề xuất phương thức getMHcuaGV(), gán cho lớp Monhoc.
* Bước 6, giao diện chọn lớp học phần -> đề xuất lớp GDLophocphan, cần có:
  + bảng danh sách lớp học phần: vừa output, vừa submit
* Để có danh sách lớp học phần, cần xử lí trước từ dưới hệ thống:
  + Lấy danh sách các lớp học phần của môn học do giảng viên dạy
  + input: môn học, giảng viên
  + output: danh sách Lophocphan
  + đề xuất phương thức getLHPcuaGV(), gán cho lớp Lophocphan
* Bước 8, giao diện nhập điểm -> đề xuất lớp GDNhapdiem, cần có:
  + Danh sách sinh viên + các đầu điểm: vừa output, vừa input
  + Nút lưu, nút reset: kiểu submit
* Để có danh sách sinh viên, cần xử lí trước từ dưới hệ thống:
  + lấy danh sách sinh viên + các đầu điểm đã có lên
  + input: lớp học phần
  + output: thông tin sinh viên + các đầu điểm thành phần của môn học (Dangkihoc)
  + đề xuất phương thức getDiemSVtheoLHP(), gán cho lớp Dangkihoc.
* Sau khi click lưu điểm, cần xử lí dưới hệ thống:
  + lưu danh sách sinh viên + các đầu điểm vào
  + input: thông tin sinh viên + các đầu điểm thành phần của môn học (Dangkihoc)
  + output: thành công hay không
  + đề xuất phương thức luuDiemSVtheoLHP(), gán cho lớp Dangkihoc.

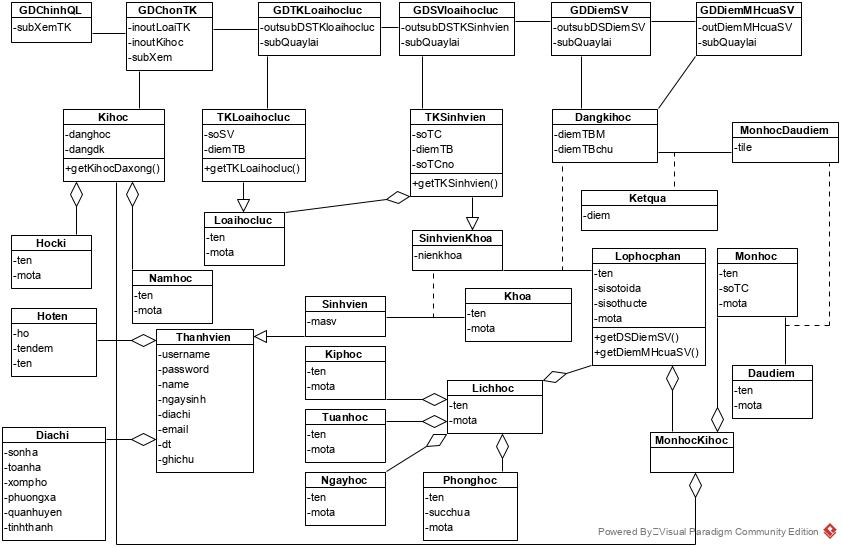
Kết quả thu được biểu đồ lớp phân tích modul nhập điểm như trong hình sau



***c. Phân tích tĩnh cho modul xem thống kê loại học lực***

* Ban đầu, giao diện chính của quản lí -> đề xuất lớp GDChinhQL, cần có
  + chọn xem thống kê: kiểu submit
* Bước 2, giao diện chọn loại thống kê và kì học -> đề xuất lớp GDChonTK, cần có:
  + danh sách loại thống kê: vừa input, vừa output
  + danh sách kì học đã kết thúc: vừa input, vừa output
  + nút xem: kiểu submit
* Đề có danh sách kì học đã kết thúc, cần xử lí trước từ dưới hệ thống:
  + lấy danh sách kì học đã kết thúc
  + input: không có
  + output: tất cả kì học đã kết thúc
  + đề xuất phương thức getKihocDaxong(), gán cho lớp Kihoc
* Bước 4+6, giao diện thống kê loại học lực -> đề xuất lớp GDTKLoaihocluc, cần có:
  + danh sách thống kê loại học lực: output, submit
* Để có dữ liệu thống kê, cần xử lí từ dưới hệ thống:
  + lấy dữ liệu thống kê loại học lực của học kì đã chọn
  + input: học kì
  + output: danh sách TKLoaihocluc
  + đề xuất phương thức getTKHocluc(), gán cho lớp TKLoaihocluc
* Bước 8, giao diện các sinh viên của loại học lực -> GDSVLoaihocluc, cần có:
  + Danh sách các sinh viên + điểm: kiểu output, submit
* Để có danh sách các sinh viên và điểm tổng kết, cần xử lí:
  + lấy danh sách sinh viên + kết quả theo học kì
  + input: loại học lực, học kì
  + output: danh sách TKSinhvien
  + đề xuất phương thức getTKSinhvien(), gán cho lớp TKSinhvien.
* Bước 10, giao diện danh sách điểm các môn học của sinh viên -> lớp GDDiemSV, cần:
  + danh sách điểm các môn học của sinh viên: output, submit
* Để có danh sách điểm các môn học của sinh viên, cần xử lí:
  + lấy danh sách điểm các môn học của sinh viên
  + input: sinh viên, học kì
  + output: danh sách các điểm sinh viên (Dangkihoc)
  + đề xuất phương thức getDSDiemSV(), gán cho lớp Dangkihoc
* Bước 12, giao diện điểm chi tiết môn học -> lớp GDDiemMHcuaSV, cần có:
  + thông tin môn học: output
  + danh sách các đầu điểm và điểm: output
* Để có thoongt in điểm chi tiết, cần xử lí:
  + lấy điểm chi tiết môn học của sinh viên
  + input: môn học, sinh viên
  + output: điểm chi tiết của môn học (Dangkihoc)
  + đề xuất phương thức getDiemMHcuaSV(), gán cho lớp Dangkihoc.

Kết quả thu được biểu đồ lớp phân tích modul xem thống kê loại học lực như trong sau.



**Phân tích hoạt động (phân tích)**

Tóm tắt các bước thực hiện để vẽ biểu đồ trạng thái của một modul/chức năng:

* Bước 1: Mỗi trạng thái chờ tương ứng với mỗi lần hệ thống hiển thị 1 giao diện để chờ tương tác của người dùng
* Bước 2: Điều kiện chuyển trạng thái tương ứng với hành động của người dùng trên giao diện tương ứng.

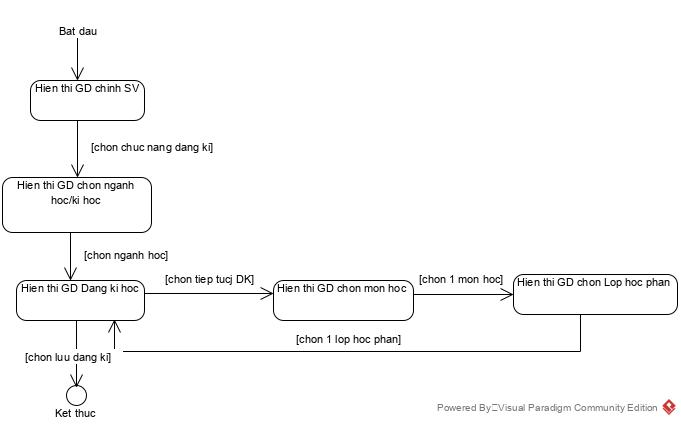
Tóm tắt các bước thực hiện để vẽ biểu đồ tuần tự/giao tiếp pha phân tích:

* Bước 1: Viết kịch bản phiên bản 2 từ biểu đồ lớp của modul: Mỗi bước là một hành động của một lớp nào đó trong biểu đồ lớp, hoặc là của tác nhân.
* Bước 2: Vẽ biểu đồ giao tiếp cho kịch bản chuẩn: các bước tương ứng với các bước trong kịch bản phiên bản 2. Quan hệ giữa các lớp phải thống nhất với quan hệ giữa chúng trong biểu đồ lớp của modul.
* Bước 3: chuyển hóa biểu đồ giao tiếp thành biểu đồ tuần tự cho kịch bản chuẩn. Kiểm tra lại các bước đúng tuần tự và logic nghiệp vụ.

***a. Modul đăng kí học***

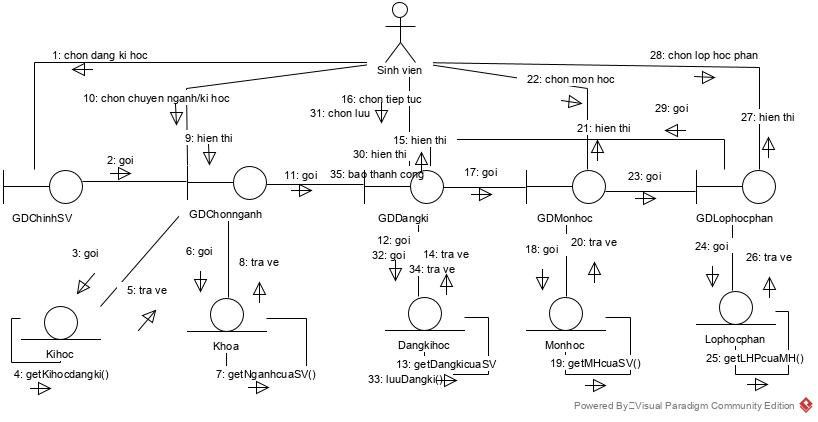
Biểu đồ trạng thái mô tả hoạt động của modul:

* Từ giao diện chính của sinh viên, nếu chức năng đăng kí được chọn thì chuyển sang giao diện chọn ngành học,
* Từ giao diện chọn ngành học, khi ngành được chọn, hệ thống chuyển sang giao diện đăng kí học.
* Tại giao diện đăng kí, nếu click tiếp tục thì chuyển sang giao diện chọn môn học. Nếu chọn một lớn đã đăng kí thì chuyển sang giao diện chọn lớp học phần để chọn lại. Nếu chọn lưu thì lưu và kết thúc.
* Tại giao diện chọn môn học, nếu môn học được chọn thì chuyển sang giao diện chọn lớp học phần.
* Tại giao diện chọn lớp học phần, sau khi chọn lớp học phần thì quay về giao diện đăng kí.



Kịch bản v.2 cho modul đăng kí môn học như sau:

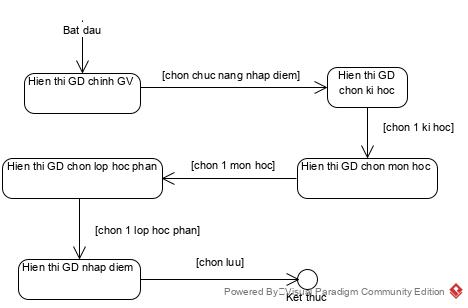
1. Tại giao diện chính của sinh viên sau khi đăng nhập, sinh viên click chọn đăng kí.
2. Lớp GDChinhSV gọi lớp GDChonnganh
3. GDChonnganh gọi lớp Kihoc yêu cầu tìm danh sách các kì học đnag mở đăng kí
4. Lớp Kihoc tìm các kì học đang mở đăng kí
5. Lớp Kihoc trả kết quả lại cho lớp GDChonnganh
6. GDChonnganh gọi lớp Khoa yêu cầu tìm danh sách ngành
7. Lớp Khoa tìm các ngành học của sinh viên
8. Lớp Khoa trả kết quả lại cho lớp GDChonnganh
9. Lớp GDChonnganh hiển thị cho sinh viên
10. Sinh viên click 1 ngành của mình học
11. Lớp GDChonnganh gọi lớp GDDangki
12. Lớp GDDangki gọi lớp Dangkihoc yêu cầu tìm thông tin đã đăng kí
13. Lớp Dangkihoc lấy thông tin đã đăng kí của sinh viên
14. Lớp Dangkihoc trả kết quả lại cho lớp GDDangki.
15. Lớp GDDangki hiển thị cho sinh viên.
16. Sinh viên chọn tiếp tục đăng kí
17. Lớp GDDangki gọi GDMonhoc
18. Lớp GDMonhoc gọi lớp Monhoc yêu cầu tìm các môn học được đăng kí
19. Lớp Monhoc tìm các môn học mà sinh viên được đăng kí
20. Lớp Monhoc trả kết quả lại cho lớp GDMonhoc
21. Lớp GDMonhoc hiển thị cho sinh viên
22. Sinh viên click chọn 1 môn học
23. Lớp GDMonhoc gọi lớp GDLophocphan
24. Lớp GDLophocphan gọi lớp Lophocphan yếu cầu tìm các lớp học phần của môn học
25. Lớp Lophocphan tìm các lớp học phần của môn học
26. Lớp Lophocphan trả kết quả cho lớp GDLophocphan
27. Lớp GDLophocphan hiển thị cho sinh viên
28. Sinh viên click 1 lớp lớp học phần
29. GDLophocphan gọi lại lớp GDDangki
30. GDDangki lại hiển thị (Lặp các bước từ 12 đến 27 đến khi đủ số tín chỉ)
31. Sinh viên click lưu
32. Lớp GDDangki gọi lớp Dangkihoc yêu cầu lưu
33. Lớp Dangkihoc lưu thông tin đăng kí
34. Lớp Dangkihoc trả kết quả lại cho lớp GDDangki
35. Lớp GDDangki báo đăng kí thành công.



***b. Modul nhập điểm***

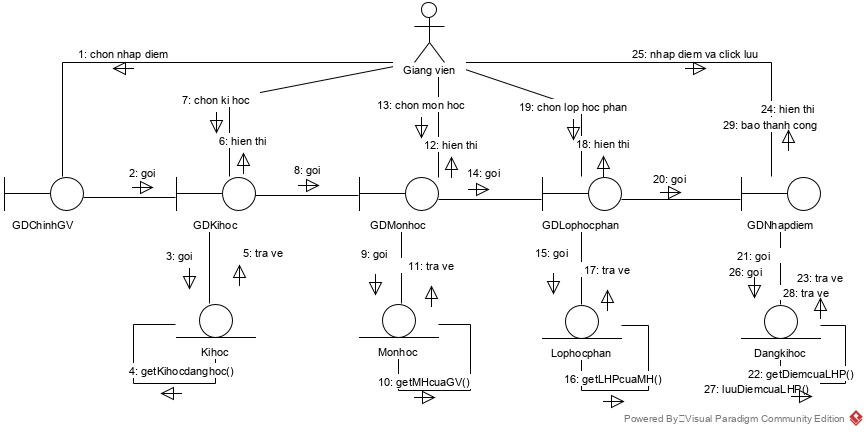
Biểu đồ trạng thái của modul:

* Tại giao diện chính của giảng viên, hệ thống chuyển sang giao diện chọn môn học khi chức năng chọn môn học được chọn.
* Tại giao diện chọn môn học, khi giảng viên chọn 1 môn học, hệ thống sẽ chuyển sang giao diện chọn lớp học phần.
* Tại giao diện chọn lớp học phần, khi giảng viên chọn 1 lớp học phần, hệ thống sẽ chuyển sang giao diện nhập điểm.
* Tại giao diện nhập điểm, sau khi nhập nhập và chọn lưu, hệ thống lưu điểm vào và kết thúc.



Kịch bản v.2 của modul diễn ra như sau:

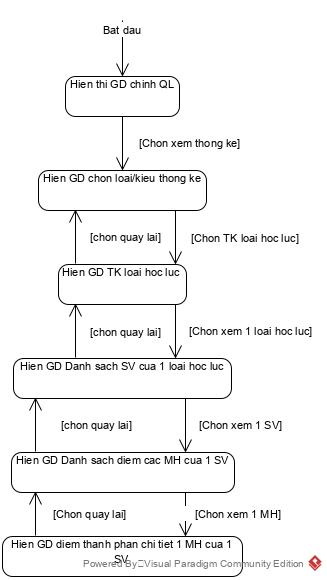
1. Tại giao diện chính của giảng viên, sau khi đăng nhập, giảng viên chọn chức năng nhập điểm.
2. Lớp GDChinhGV gọi GDKihoc
3. Lớp GDKihoc goi lớp Kihoc yêu cầu tìm kì học đang học
4. Lớp Kihoc tìm các kì học đang học
5. Lớp Kihoc trả kết quả cho lớp GDKihoc
6. Lớp GDKihoc hiển thị cho giảng viên
7. Giảng viên chọn 1 kì học.
8. Lớp GDKihoc gọi GDMonhoc.
9. Lớp GDMonhoc gọi lớp Monhoc yêu cầu tìm các môn do giảng viên dạy
10. Lớp Monhoc tìm các môn học do giảng viên dang dạy
11. Lớp Monhoc trả kết quả lại cho lớp GDMonhoc.
12. Lớp GDMonhoc hiển thị cho giảng viên
13. Giảng viên chọn 1 môn học trong danh sách
14. Lớp GDMonhoc gọi lớp GDLophocphan
15. Lớp GDLophocphan gọi lớp Lophocphan yêu cầu tìm lớp học phần của giảng viên
16. Lớp Lophocphan tìm các lớp học phần do giảng viên dạy, của môn học đã chọn
17. Lớp Lophocphan trả kết quả lại cho lớp GDLophocphan
18. Lớp GDLophocphan hiển thị cho giảng viên.
19. Giảng viên chọn 1 lớp học phần trong danh sách
20. Lớp GDLophocphan gọi lớp GDNhapdiem
21. Lớp GDNhapdiem gọi lớp Dangkihoc yêu cầu tìm thông tin điểm đã có
22. Lớp Dangkihoc tìm sinh viên và các đầu điểm đã có của lớp học phần
23. Lớp Dangkihoc trả kết quả lại cho lớp GDNhapdiem
24. Lớp GDNhapdiem hiển kết quả cho giảng viên.
25. Giảng viên nhập thêm đầu điểm và click lưu
26. Lớp GDNhapdiem gọi lớp Dangkihoc yêu cầu cập nhật
27. Lớp Dangkihoc lưu thông tin điểm của lớp học phần
28. Lớp Dangkihoc trả kết quả lại cho lớp GDNhapdiem
29. Lớp GDNhapdiem thông báo thành công.



***c. Modul xem thống kê loại học lực***

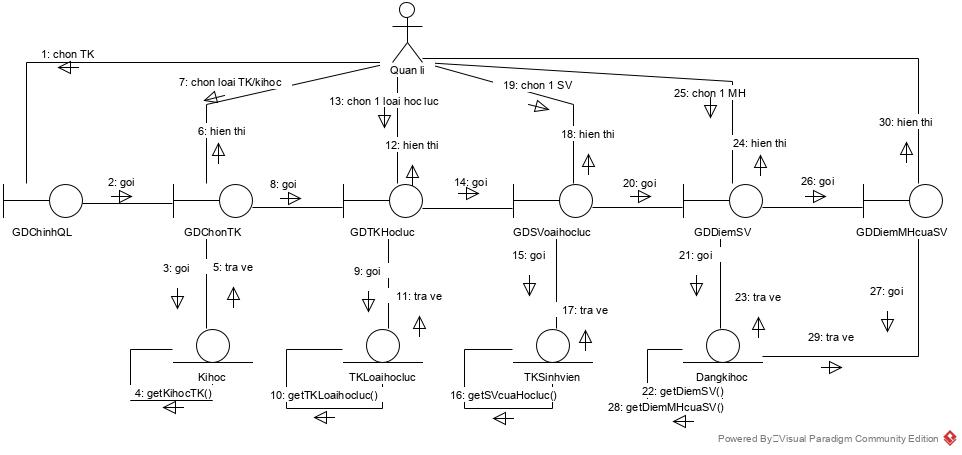
Hoạt động của biểu đồ trạng thái của modul :

* Tại giao diện chính của nhân viên quản lí, khi chọn chức năng xem thống kê, hệ thống sẽ chuyển sang giao diện chọn kiểu thống kê
* Tại giao diện chọn kiểu thống kê, chọn loại thống kê loại học lực, hệ thống chuyển snag giao diện thống kê loại học lực
* Tại giao diện thống kê loại học lực, khi chọn vào 1 loại học lực, hệ thống chuyển snag giao diện thống kê các sinh viên của loại học lực đã chọn
* Tại giao diện thống kê sinh viên, chọn một sinh viên, hệ thống chuyển sang giao diện liệt kê điểm các môn học của sinh viên đã chọn
* Tại giao diện điểm các môn học, chọn 1 môn học, hệ thống sẽ chuyển snag giao diện thông tin điểm chi tiết của môn học của sinh viên đã chọn.
* Tại tất cả các giao diện, nếu chọn quay lại, hệ thống sẽ quay lại giao diện trước đấy.



Kịch bản v.2 của modul diễn ra như sau:

1. Tại giao diện chính của nhân viên quản lí, sau khi đăng nhập, quản lí chọn xem thống kê.
2. Lớp GDChinhQL gọi lớp GDChonTK.
3. Lớp GDChonTK gọi lớp Kihoc yêu cầu tìm các kì học đã kết thúc
4. Lớp Kihoc tìm các kì học đã kết thúc
5. Lớp Kihoc trả kết quả lại cho lớp GDChonTK.
6. Lớp GDChonTK hiển thị cho QL
7. QL chọn thống kê loại học lực
8. Lớp GDChonTK gọi lớp GDTKHocluc
9. Lớp GDTKHocluc hiển thị cho QL
10. QL chọn học kì muốn thống kê
11. Lớp GDTKHocluc gọi lớp TKLoaihocluc yêu cầu lấy dữ liệu
12. Lớp TKLoaihocluc lấy dữ liệu thống kê
13. Lớp TKLoaihocluc trả kết quả cho lớp GDTKHocluc
14. Lớp GDTKHocluc hiển thị cho QL
15. QL chọn 1 loại học lực
16. Lớp GDTKHocluc gọi lớp GDSVLoaihocluc
17. Lớp GDSVLoaihocluc gọi lớp TKSinhvien yêu cầu lấy dữ liệu
18. Lớp TKSinhvien lấy dữ liệu thống kê sinh viên của loại học lực
19. Lớp TKSinhvien trả kết quả lại cho lớp GDSVLoaihocluc
20. Lớp GDSVLoaihocluc hiển thị cho QL
21. QL click 1 sinh viên
22. Lớp GDSVLoaihocluc gọi lớp GDDiemSV
23. Lớp GDDiemSV gọi lớp Dangkihoc yêu cầu lấy dữ liệu điểm
24. Lớp Dangkihoc lấy điểm các môn học của sinh viên
25. Lớp Dangkihoc trả kết quả cho lớp GDDiemSV
26. Lớp GDDiemSV hiển thị cho QL
27. QL chọn 1 môn học
28. Lớp GDDiemSV gọi lớp GDDiemMHcuaSV
29. Lớp GDDiemMHcuaSV gọi lớp Dangkihoc yêu cầu lấy điểm chi tiết
30. Lớp Dangkihoc lấy thông tin điểm chi tiết của 1 môn học của sinh viên.
31. Lớp Dangkihoc trả kết quả ;ại cho lớp GDDiemMHcuaSV
32. Lớp GDDiemMHcuaSV hiển thị cho QL



**Share this:**

# Thiết kế lớp thực thể

**Quy trình thực hiện**

Đầu vào là biểu đồ lớp thực thể pha phân tích, tiến hành các bước như sau:

* Bước 1: Bổ sung thuộc tính id cho các lớp không kế thừa từ lớp khác
* Bước 2: Bổ sung kiểu dữ liệu cho các thuộc tính. Kiểu dữ liệu phải tương ứng với kiểu dữ liệu của ngôn ngữ lập trình đã chọn.
* Bước 3: Chuyển đổi các quan hệ association sang các dạng aggregation/composition, nếu có: Nếu hai lớp A và B liên kết (asociation) tạo ra lớp C thì có thể chuyển thành:
  + A chứa C, C chứa B;
  + hoặc B chứa C, C chứa A;
  + hoặc C chứa A và B.
* Bước 4: Bổ sung thuộc tính đối tượng của các lớp thực thể, nếu có
  + Giữa hai lớp có quan hệ aggregation/composition thì lớp chứa lớp còn lại nên khai báo tường minh thuộc tính có kiểu là lớp còn lại
  + Số lượng (số ít/số nhiều) của thuộc tính phụ thuộc vào quan hệ số lượng giữa hai thực thể.
* Vẽ biểu đồ thực thể pha thiết kế

Lưu ý khi vẽ biểu đồ lớp thực thể pha thiết kế:

* Tên lớp, tên thuộc tính nên đặt tên theo chuẩn code convention: Tên lớp nên bắt đầu bằng chữ hoa, tên thuộc tính nên bắt đầu bằng chữ thường. Có thể đặt tên gợi nhớ theo Tiếng Anh hoặc Tiếng Việt. Nếu Tiếng Việt thì không được có dấu, không được có dấu cách. Trong tên lớp/thuộc tính không được có các kí tự đặc biệt.
* Thuộc tính cần khai báo tường minh kiểu dữ liệu.

##### Áp dụng

***Bước 1***

Các lớp được bổ sung thuộc tính id: trừ các lớp TK, Sinhvien, Thanhvien, Giaovu, Giangvien, NVQuanli, NVKhaothi.

***Bước 2***

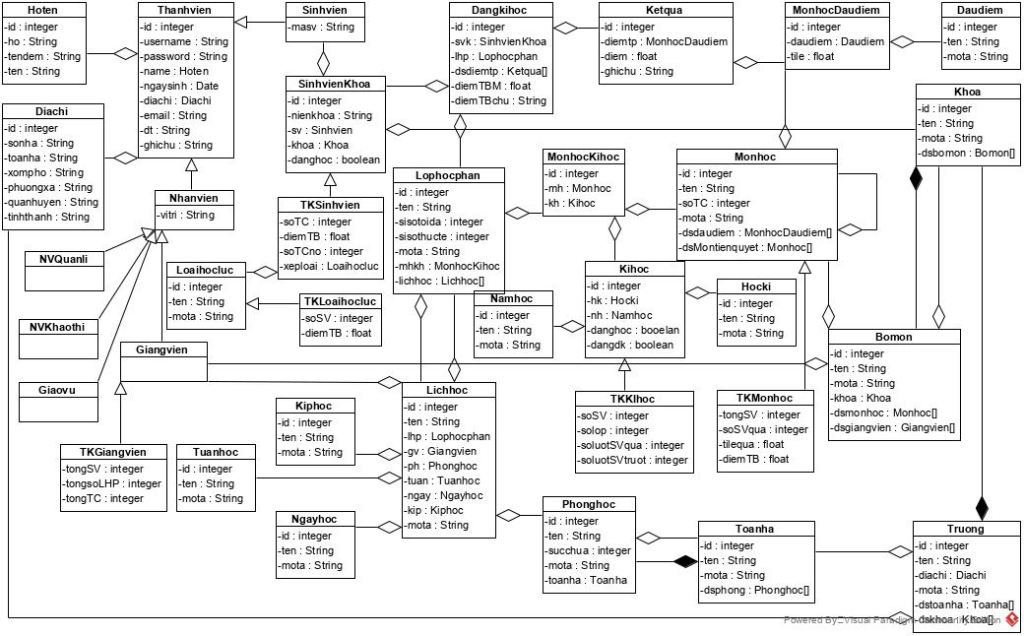
Thuộc tính của các lớp được bổ sung thuộc tính theo kiểu của ngôn ngữ lập trình Java.

***Bước 3***

* Quan hệ Sinhvien – Khoa -> SinhvienKhoa chuyển thành SinhvienKhoa chứa Sinhvien và Khoa.
* Quan hệ Monhoc – Daudiem -> MonhocDaudiem chuyển thành MonhocDaudiem chứa Monhoc và Daudiem.
* Quan hệ Namhoc – Hocki -> Kihoc chuyển thành Kihoc chứa Namhoc và Hocki
* Quan hệ Monhoc – Kihoc -> MonhocKihoc chuyển thành MonhocKihoc chứa Monhoc và Kihoc.
* Quan hệ SinhvienKhoa – Lophocphan -> Dangkihoc chuyển thành Dangkihoc chứa SinhvienKhoa và Lophocphan.
* Quan hệ Dangkihoc – MonhocDaudiem -> Ketqua chuyển thành Dangkihoc chứa Ketqua, Ketqua chứa MonhocDaudiem.

***Bước 4***

Bổ sung các thuộc tính thành phần của quan hệ hợp thành/gắn chặt, ta thu được biểu đồ lớp thực thể pha thiết kế.



# Thiết kế CSDL

**Quy trình thực hiện**

Đầu vào là biểu đồ lớp thực thể pha thiết kế, tiến hành các bước như sau:

* Bước 1: Với mỗi lớp thực thể, đề xuất một bảng dữ liệu tương ứng. Nên đặt tên bảng theo cách gợi nhớ tương ứng. Ví dụ, nếu lớp thực thể là Student, thì có thể đặt tên bảng tương ứng là tblStudent.
* Bước 2: Với mỗi lớp thực thể, bỏ qua các thuộc tính kiểu đối tượng, lấy các thuộc tính kiểu cơ bản đưa sang làm thuộc tính của bảng tương ứng, và chuyển kiểu dữ liệu theo hệ quản trị cơ sở dữ liệu tương ứng.
* Bước 3: Xét quan hệ số lượng giữa các lớp thực thể, quan hệ số lượng giữa hai lớp thực thể cũng là quan hệ số lượng giữa hai bảng tương ứng với hai lớp đấy. Khi đó:
  + Nếu hai bảng có quan hệ là 1-1 thì nên gộp lại, tuy nhiên, trong một số trường hợp đặc biệt thì có thể giữ nguyên.
  + Nếu hai bảng có quan hệ 1-n thì giữ nguyên.
  + Nếu hai bảng có quan hệ n-n thì nên bổ sung ít nhất một bảng trung gian để tách thành một số quan hệ 1-n. Thực tế, nếu đến thời điểm này vẫn còn quan hệ n-n giữa các bảng thì chứng tỏ trong biểu đồ lớp thực thể, vẫn còn quan hệ n-n, cho nên tốt nhất là nên quay lại sửa từ biểu đồ lớp thực thể pha phân tích.
* Bước 4: Bổ sung khóa cho các bảng:
  + Khóa chính (Primary key – PK): Bảng nào có thuộc tính id thì thiết lập thuộc tính đó làm khóa chính cho bảng.
  + Khóa ngoại (Foreinger key – FK): Nếu hai bảng tblA và tblB có quan hệ 1-n (1 tblA có n tblB), thì trong bảng tblB phải có khóa ngoài tham chiếu đến khóa chính của bảng tblA. Khó ngoại có thể đặt tên là aId hoặc idA.
* Bước 5: Loại bỏ các thuộc tính gây dư thừa dữ liệu. Có hai loại thuộc tính gây dư thừa dữ liệu:
  + Thuộc tính bị trùng lặp: thường là thuộc tính của cùng một đối tượng, không phải thuộc tính khóa, nhưng xuất hiện ở hai bảng khác nhau.
  + Thuộc tính dẫn xuất: là thuộc tính có thể tính toán cơ học từ các thuộc tính khác trong CSDL. Ví dụ một số thuộc tính xuất thân từ các lớp thực thể thống kê thì thường là thuộc tính dẫn xuất. Sau khi loại bỏ các thuộc tính dư thừa, nếu xuất hiện bảng nào không còn thuộc tính nào nữa ngoài 1 khóa ngoại thì có thể loại bỏ luôn bảng đấy.

##### Áp dụng

***Bước 1***

Mỗi lớp thực thể đề xuất bảng tương ứng:

* Lớp Truong -> bảng tblTruong
* Lớp Khoa -> bảng tblKhoa
* Lớp Toanha -> bảng tblToanha
* ….

***Bước 2***

Đưa thuộc tính không phải đối tượng của lớp thực thể thành thuộc tính của bảng tương ứng:

* tblTruong có các thuộc tính: id, tên, mô tả
* tblKhoa: id, tên, mô tả
* tblToanha: id, tên, mô tả
* ….

***Bước 3***

Chuyến quan hệ số lượng giữa các lớp thực thể thành quan hệ số lượng giữa các bảng:

* 1 tblTruong – n tblKhoa
* 1 tblTruong – n tblToanha
* 1 tblToanha – n tblPhonghoc
* ….

***Bước 4***

Bổ sung các thuộc tính khóa. Khóa chính được thiết lập với thuộc tính id của các bảng tương ứng: trừ các bảng tblTK, tblSinhvien, tblThanhvien, tblGiangvien, tblNVQuanli, tblNVKhaothi, tblGiaovu.

Khóa ngoại được thiết lập cho các bảng:

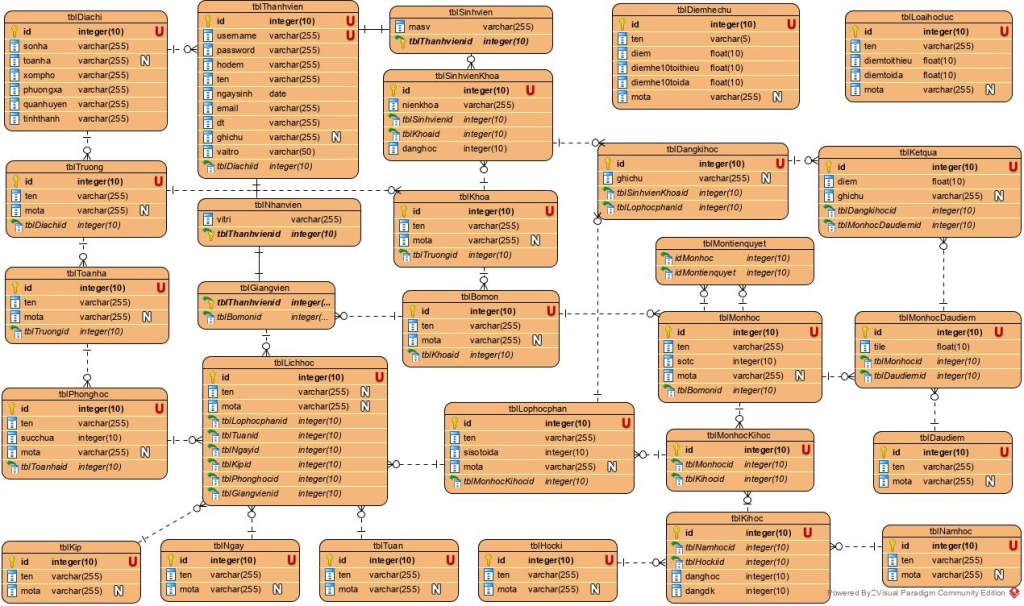
* 1 tblTruong – n tblKhoa -> bảng tblKhoa có khóa ngoại tblTruongid
* 1 tblTruong – n tblToanha -> bảng tblToanha có khóa ngoại tblTruongid
* 1 tblToanha – n tblPhonghoc -> bảng tblPhonghoc có khóa ngoại tblToanhaid
* ….

***Bước 5***

Các thuộc tính dẫn xuất:

* điểm TBM, điểm TB chữ trong bảng tblDangkihoc
* Các thuộc tính của các lớp thống kê -> loại bỏ hết các bảng thống kê.

Kết quả thu được CSDL toàn hệ thống.



**Thiết kế tĩnh: giao diện và biểu đồ lớp cho từng chức năng**

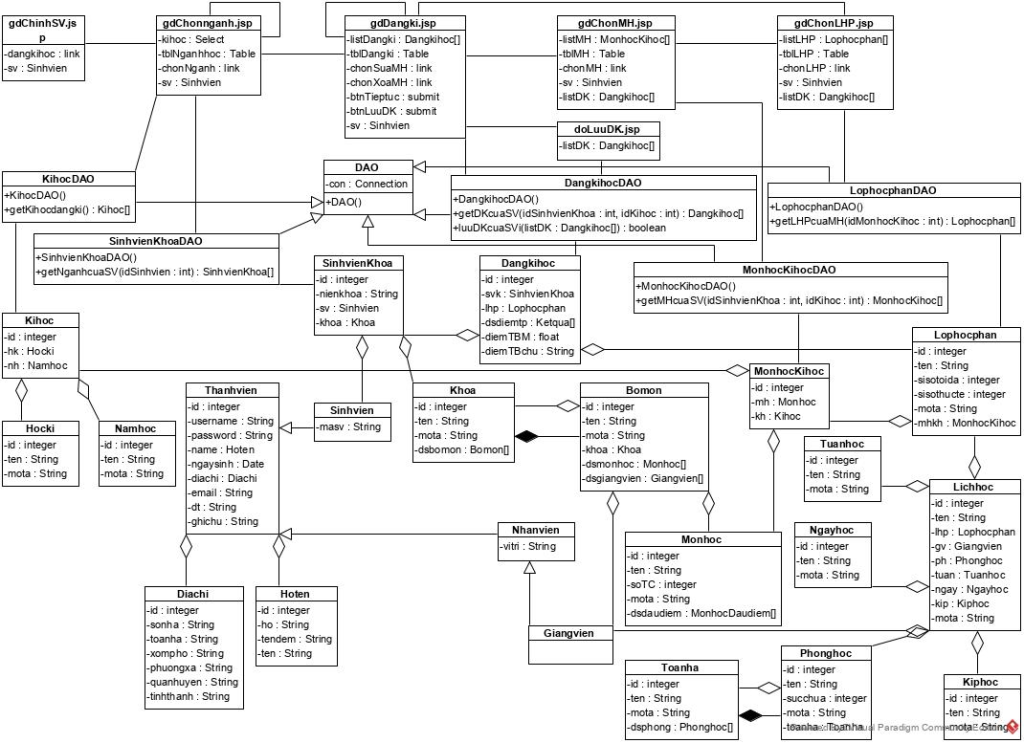
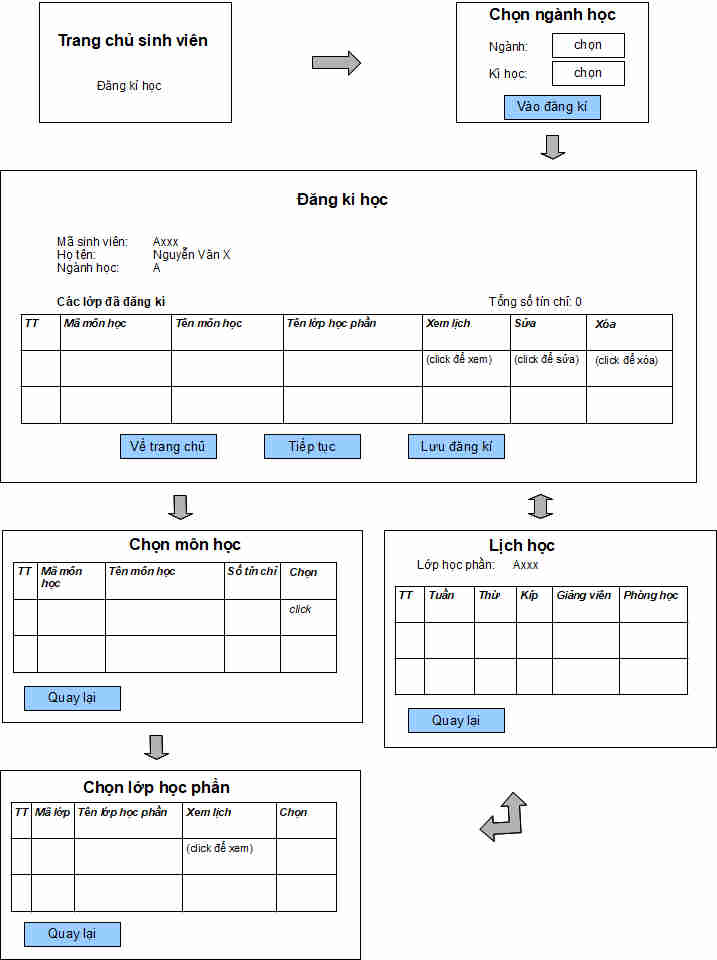
**Các bước thực hiện**

Đầu vào là biểu đồ lớp pha phân tích, tiến hành các bước như sau:

* Bước 1: Thiết kế giao diện cho các giao diện xuất hiện trong biểu đồ.
* Bước 2: Đề xuất các lớp giao diện tương ứng với ngôn ngữ lập trình và nền tảng hệ thống. Ví dụ hệ thống chạy trên ứng dụng máy bàn, hệ thống chạy trên nền web, hệ thống chạy trên nền tảng thiết bị di động.
  + Đề xuất tên lớp tương ứng. Có thể tách, gộp các lớp giao diện so với pha phân tích
  + Thiết kế các thuộc tính đã liệt kê trong pha phân tích.
  + Bổ sung các *thuộc tính ẩn* để chứa thông tin cần thiết để xử lí và trao đổi qua lại giữa các lớp giao diện.
* Bước 3: Đề xuất các lớp truy xuất dữ liệu (DAO), nếu có.
  + Nên thiết kế dạng Interface hoặc kế thừa từ lớp trừu tượng để có thể dùng chung kết nối vào CSDL cho toàn hệ thống để tiết kiệm tài nguyên.
  + Thường, nếu lớp thực thể trong pha phân tích cần phương thức, hành động nào thì nên đề xuất lớp DAO cho lớp thực thể tương ứng, và gán phương thức đang xem xét cho lớp vùa đề xuất.
  + Với mỗi phương thức của lớp DAO, cần lưu ý thiết kế tên gợi nhớ, các tham số đầu vào, các tham số đầu ra.
* Bước 4: Bổ sung các lướp thực thể liên quan đến modul. Giữ nguyên quan hệ giữa các thực thể như các quan hệ giữa chúng trong biểu đồ lớp thực thể.

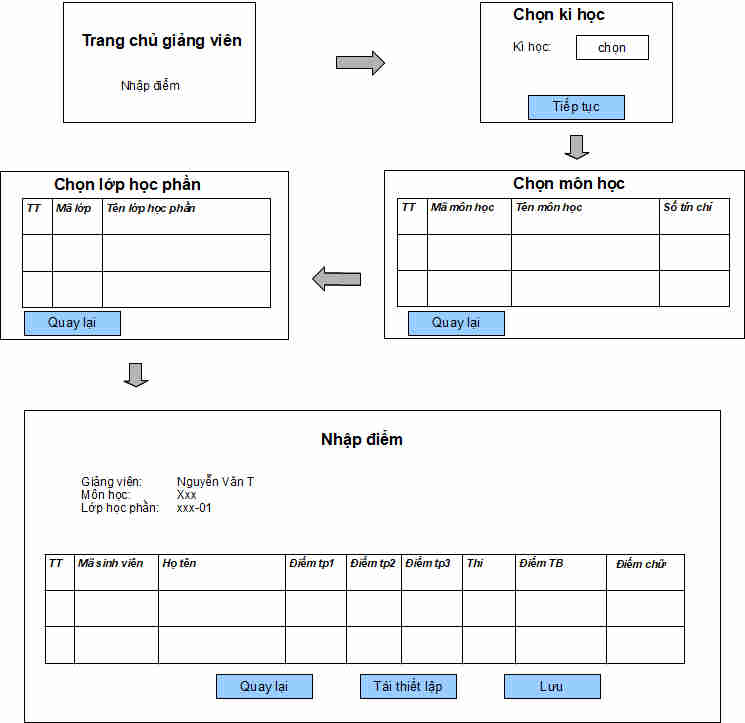
***a. Thiết kế tĩnh modul đăng kí học***

* Tầng giao diện có các trang jsp: gdChinhSV, gdChonnganh, gdDangki, gdChonMH, gdChonLHP, và doLuuDK.
* Tầng thao tác với dữ liệu có các lớp DAO, KihocDAO, SinhvienKhoaDAO, MonhocKihocDAO, LophocphanDAO, DangkihocDAO.
* Tầng thực thể gồm tất cả các thực thể liên quan. Hình 4.3: Thiết kế giao diện cho modul đăng kí học

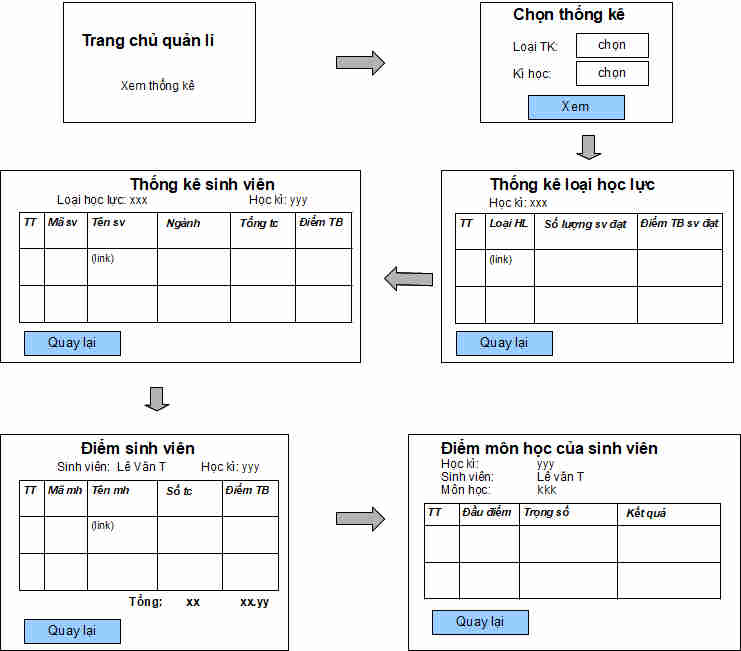


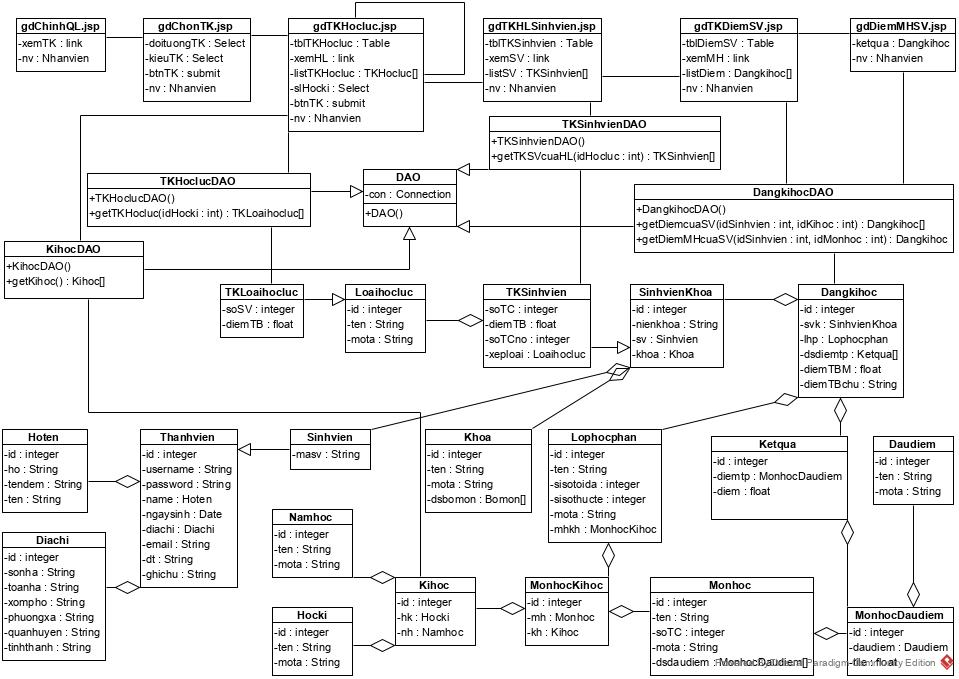
***b. Thiết kế tĩnh modul nhập điểm***

* Các lớp giao diện là các trang jsp: gdChinhGV, gdMonhoc, gdLophocphan, gdDiem, doLuudiem.
* Các lớp tầng truy cập dữ liệu DAO: DAO, KihocDAO, MonhocKihocDAO, LophocphanDAO, DangkihocDAO.
* Các lớp thực thể liên quan



***c. Thiết kế tĩnh modul xem thống kê loại học lực***

* Tầng giao diện gồm các trang jsp: gdChinhQL, gdChonTK, gdTKHocluc, gdTKHLSinhvien, gdTKDiemSV, gdDiemMHSV.
* Tầng xử lí truy cập dữ liệu: DAO, KihocDAO, TKHoclucDAO, TKSinhvienDAO, DangkihocDAO.
* Các lớp thực thể liên quan. 



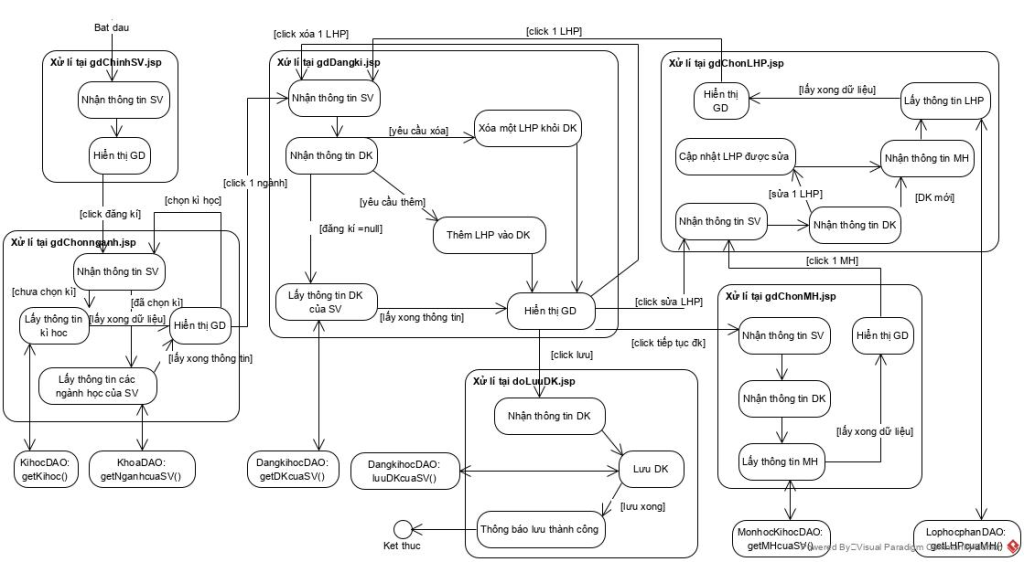
**Thiết kế động: biểu đồ hoạt động và biểu đồ tuần tự cho từng modul**

**Quy trình tiến hành các bước như sau:**

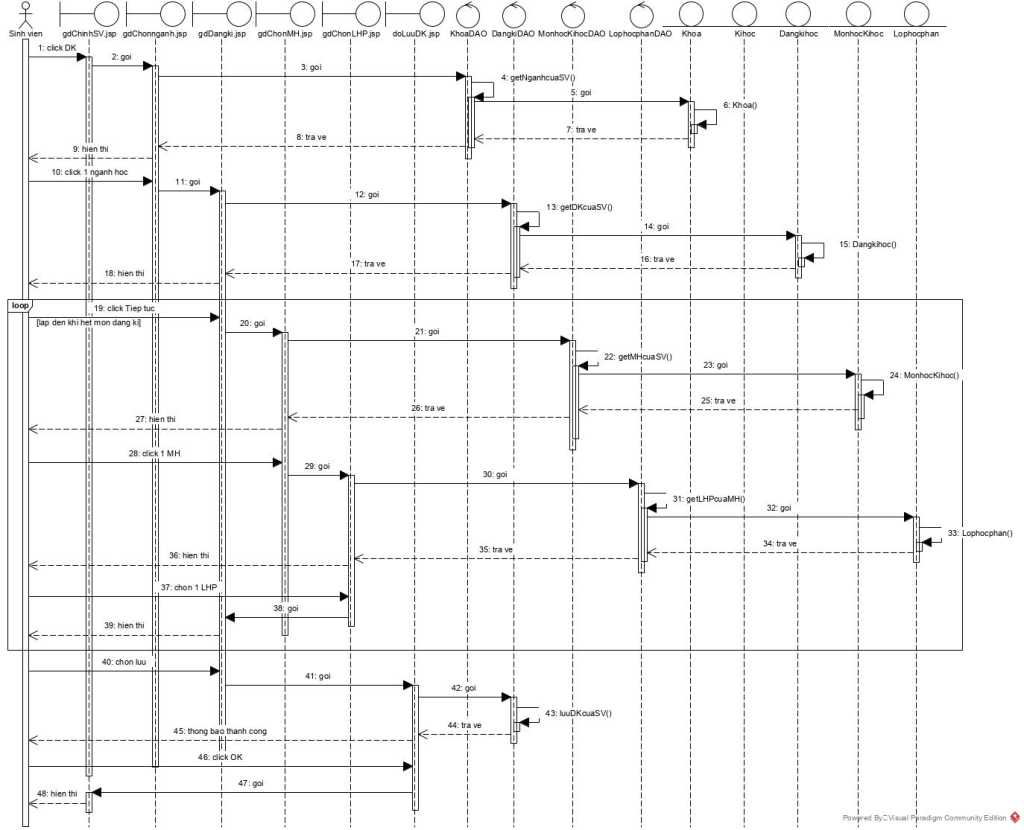
* Bước 1: Vẽ biểu đồ hoạt động của modul: Mỗi hành động tương ứng một phương thức đã thiết kế trong biểu đồ lớp.
* Bước 2: Viết kịch bản phiên bản 3 và vẽ biểu đồ tuần tự cho modul. Những lưu ý về yêu cầu khác biệt của biểu đồ tuần tự pha thiết kế so với biểu đồ tuần tự pha phân tích:
  + Tên các hành động của các lớp phải tương ứng với các phương thức đã thiết kế trong biểu đồ lớp thiết kế
  + Biểu diễn được phạm vi và khoảng thởi gian hoạt động của từng phương thức trong biểu đồ.

Lưu ý là trong các kịch bản thiết kế cho tiết ở mục này, các bước đóng gói thuộc tính con của lớp thực thể đã được lược bỏ đi cho gọn.

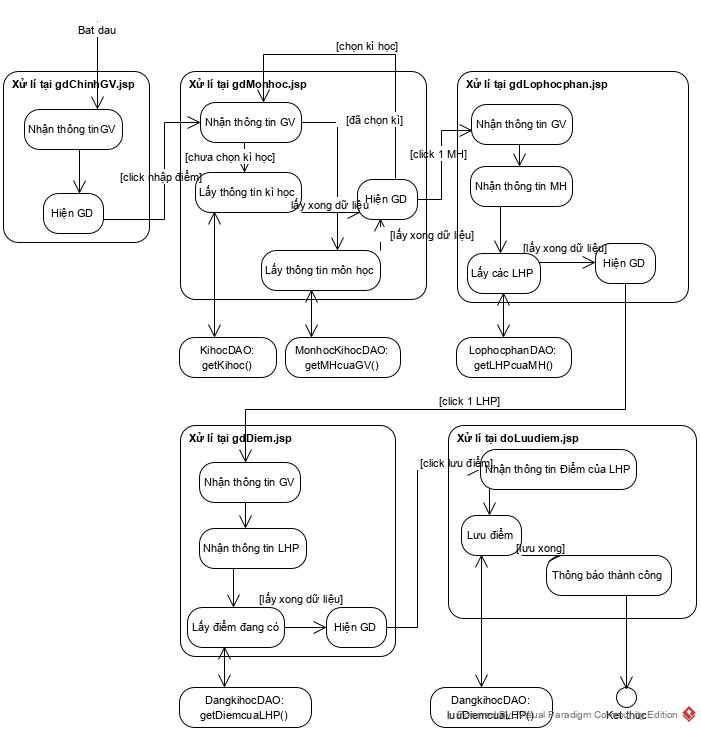
***a. Thiết kế động modul đăng kí học***



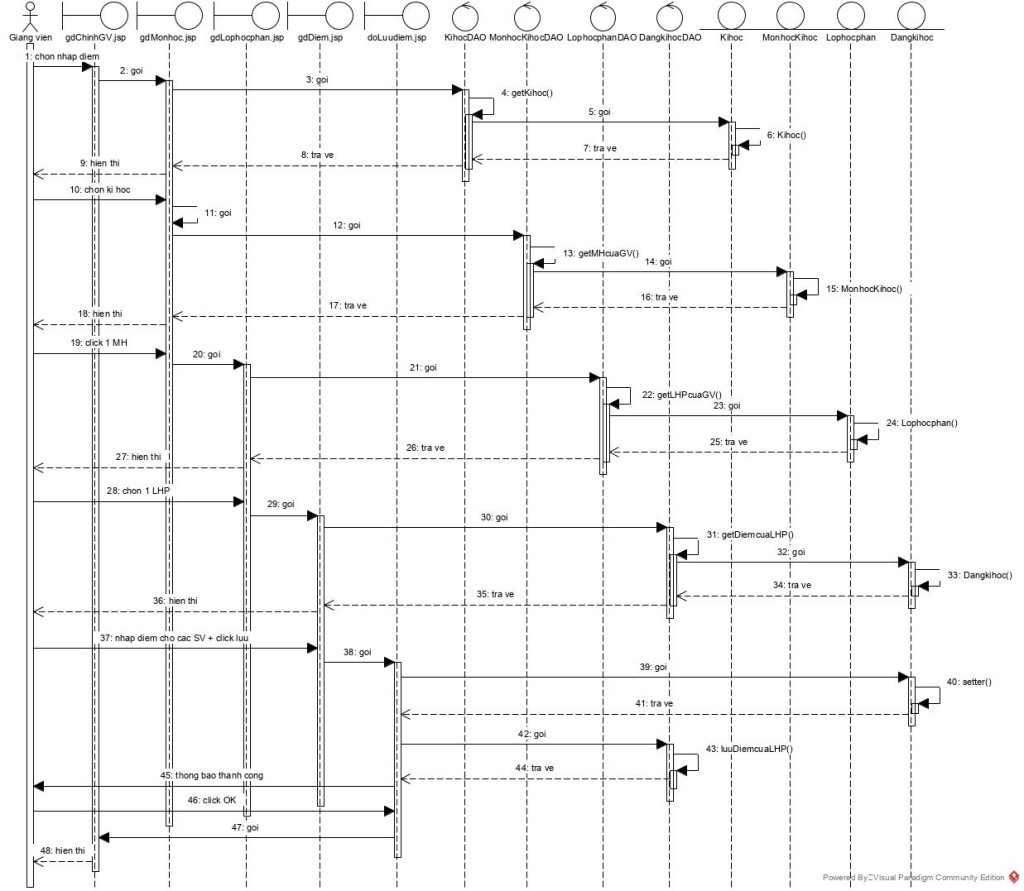
1. Tại giao diện chính của sinh viên, sau khi đăng nhập, sinh viên click chọn chức năng đăng kí học.
2. Trang gdChinhSV.jsp gọi trang gdChonnganh.jsp.
3. Trang gdChonnganh.jsp gọi lớp KhoaDAO yêu cầu tìm danh sách ngành học của sinh viên.
4. Lớp KhoaDAO gọi hàm getNganhhoccuaSV()
5. Hàm getNganhhoccuaSV() gọi lớp Khoa để đóng gói thông tin
6. Lớp Khoa đóng gói thông tin thực thể
7. Lớp Khoa trả kết quả về cho hàm getNganhhoccuaSV()
8. Hàm getNganhhoccuaSV() trả kết quả cho trang gdChonnganh,jsp
9. Trang gdChonnganh.jsp hiển thị cho sinh viên
10. Sinh viên chọn 1 ngành học để đăng kí
11. Trang gdChonnganh.jsp gọi trang gdDangki.jsp
12. Trang gdDangki.jsp gọi lớp DangkihocDAO yêu cầu lấy thông tin đăng kí đã có của sinh viên
13. Lớp DangkihocDAO gọi hàm getDangkicuaSV()
14. Hàm getDangkicuaSV() gọi lớp Dangkihoc để đóng gói thông tin
15. Lớp Dangkihoc đóng gói thông tin thực thể
16. Lớp Dangkihoc trả kết quả lại cho hàm getDangkicuaSV()
17. Hàm getDangkicuaSV() trả kết quả lại cho trang gdDangki.jsp
18. Trang gdDangki.jsp hiển thị cho sinh viên.
19. Sinh viên click tiếp tục để đăng kí.
20. Trang gdDangki.jsp gọi trang gdChonMH.jsp
21. Trang gdChonMH.jsp gọi lớp MonhocKihocDAO yêu cầu lấy danh sách môn học mà sinh viên được đăng kí
22. Lớp MonhocKihocDAO gọi hàm getMonhoccuaSV()
23. Hàm getMonhoccuaSV() thực hiện và gọi lớp MonhocKihoc để đóng gói
24. Lớp MonhocKihoc đóng gói thông tin thực thể
25. Lớp MonhocKihoc trả kết quả lại cho hàm getMonhoccuaSV()
26. Hàm getMonhoccuaSV() trả kết quả lại cho trang gdChonMH.jsp
27. Trang gdChonMH.jsp hiển thị cho sinh viên.
28. Sinh viên click chọn 1 môn học
29. Trang gdChonMH.jsp gọi trang gdChonLHP.jsp
30. Trang gdChonLHP.jsp gọi lớp LophocphanDAO yêu cầu tìm danh sách lớp học phần của môn học đã chọn.
31. Lớp LophocphanDAO gọi hàm getLHPcuaMH()
32. Hàm getLHPcuaMH() thực hiện và gọi lớp Lophocphan
33. Lớp Lophocphan đóng gói thông tin thực thể
34. Lớp Lophocphan trả lại kết quả cho hàm getLHPcuaMH()
35. Hàm getLHPcuaMH() trả kết quả cho trang gdChonLHP.jsp
36. Trang gdChonLHP.jsp hiển thị cho sinh viên
37. Sinh viên chọn 1 lớp học phần để đăng kí
38. Trang gdChonLHP.jsp gọi lại trang gdDangki.jsp
39. Trang gdDangki.jsp hiển thị cho sinh viên. (có thể lặp lại các bước 18 đến 38 cho đến khi đủ số tín chỉ tối thiểu)
40. Sinh viên click chọn lưu đăng kí
41. Trang gdDangki.jsp gọi trang doLuuDK.jsp
42. Trang doLuuDK.jsp gọi lớp DangkihocDAO yêu cầu lưu
43. Lớp DangkihocDAO gọi hàm luuDKcuaSV()
44. Lớp DangkihocDAO trả lại lớp doLuuDK.jsp
45. Lớp doLuuDK.jsp thông báo thành công
46. Sinh viên click OK của thông báo
47. Trang doLuuDK.jsp gọi lại trang gdChinhSV.jsp
48. Trang gdChinhSV.jsp hiển thị.



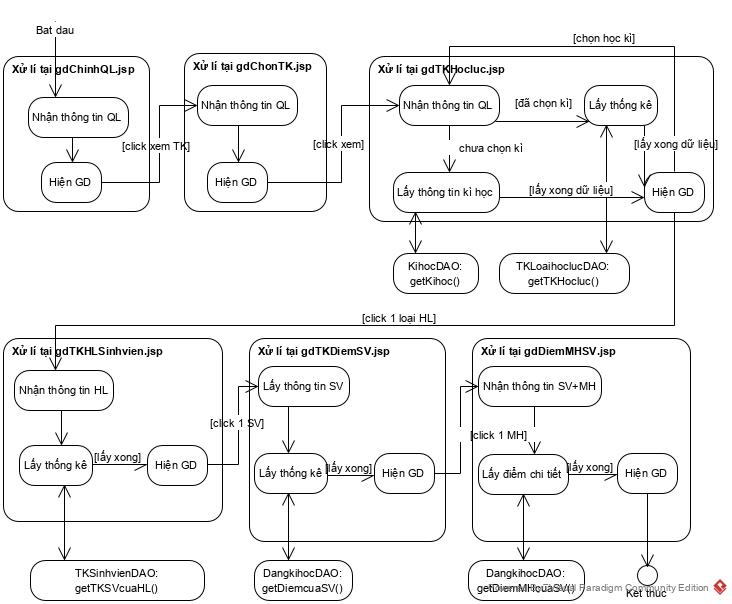
***b. Thiết kế động modul nhập điểm***



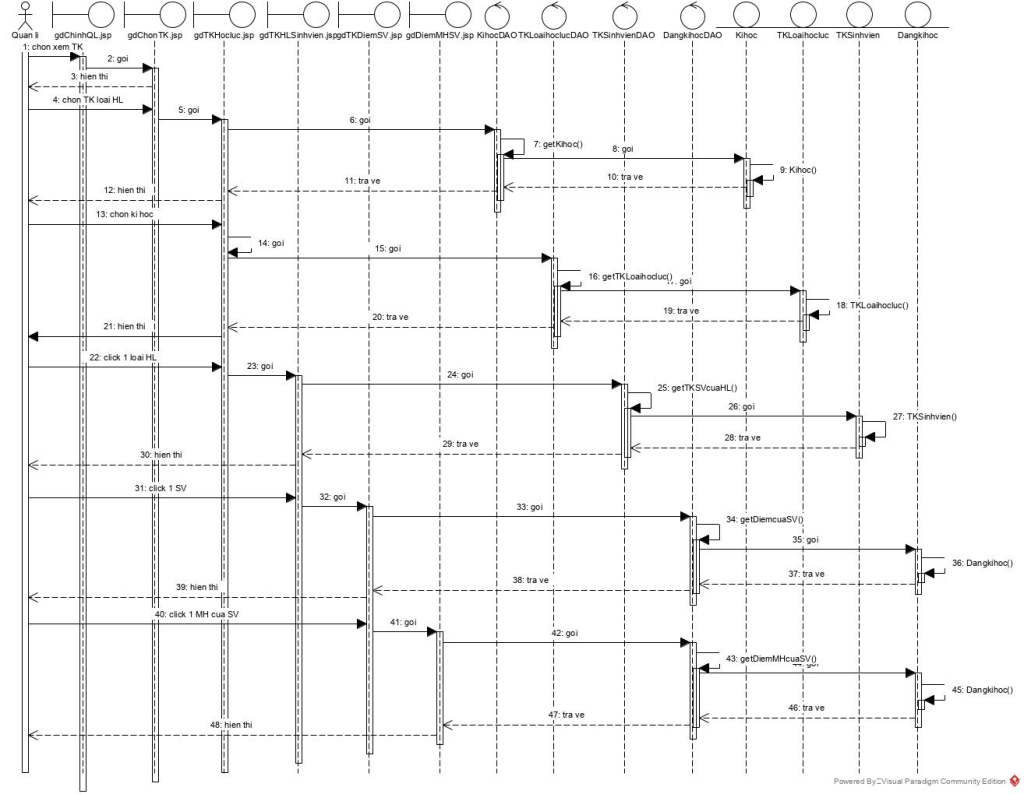
1. Tại trang gdChinhGV.jsp, sau khi đăng nhập, giảng viên chọn chức năng nhập điểm
2. Trang gdChinhGV.jsp gọi trang gdMonhoc.jsp
3. Trang gdMonhoc.jsp gọi lớp KihocDAO yêu cầu lấy thông tin các học kì đang học.
4. Lớp KihocDAO gọi hàm getKihoc()
5. Hàm getKihoc() thục hiện và gọi lớp Kihoc đóng gói thông tin
6. Lớp Kihoc đóng gói thông tin thực thể
7. Lớp Kihoc trả kết quả cho hàm getKihoc()
8. Hàm getKihoc() trả kết quả cho trang gdMonhoc.jsp
9. Trang gdMonhoc.jsp hiển thị cho giảng viên
10. Giảng viên chọn 1 kì học
11. Trang gdMonhoc.jsp submit gọi chính nó xử lí
12. Trang gdMonhoc.jsp gọi lớp MonhocKihocDAO yêu cầu lấy danh sách môn học do giảng viên dạy trong kì được chọn.
13. Lớp MonhocKihocDAO gọi hàm getMHcuaGV()
14. Hàm getMHcuaGV() thực hiện và gọi lớp MonhocKihoc để đóng gói thông tin
15. Lớp MonhocKihoc đóng gói thông tin thực thể
16. Lớp MonhocKihoc trả kết quả lại cho hàm getMHcuaGV()
17. Hàm getMHcuaGV() trả kết quả lại cho trang gdMonhoc.jsp
18. Trang gdMonhoc.jsp hiển thị cho giảng viên
19. Giảng viên click chọn 1 môn học
20. Trang gdMonhoc.jsp gọi trang gdLophocphan.jsp
21. Trang gdLophocphan.jsp gọi lớp LophocphanDAO yêu cầu lấy danh sách các lớp học phần của môn học đã chọn, do giảng viên đó giảng dạy
22. Lớp LophocphanDAO gọi hàm getLHPcuaGV()
23. Hàm getLHPcuaGV() thực hiện và gọi lớp Lophocphan để đóng gói thông tin
24. Lớp Lophocphan đóng gói thông tin thực thể
25. Lớp Lophocphan trả kết quả lại cho hàm getLHPcuaGV()
26. Hàm getLHPcuaGV() trả kết quả lại cho trang gdLophocphan.jsp
27. Trang gdLophocphan.jsp hiển thị cho giảng viên
28. Giảng viên click chọn 1 lớp học phần.
29. Trang gdLophocphan.jsp gọi trang gdDiem.jsp
30. Trang gdDiem.jsp gọi lớp DangkihocDAO yêu cầu lấy danh sách sinh viên đăng kí và kết quả đã có của lớp học phần đã chọn
31. Lớp DangkihocDAO gọi hàm getDiemcuaLHP()
32. Hàm getDiemcuaLHP() thực hiện và gọi lớp Dangkihoc để đóng gói thông tin
33. Lớp Dangkihoc đóng gói thông tin thực thể
34. Lớp Dangkihoc trả kết quả cho hàm getDiemcuaLHP()
35. Hàm getDiemcuaLHP() trả kết quả cho trang gdDiem.jsp
36. Trang gdDiem.jsp hiển thị cho giảng viên
37. Giảng viên nhập các đầu điểm cho sinh viên và click lưu
38. Trang gdDiem.jsp gọi doLuudiem.jsp
39. Trang doLuudiem.jsp gọi lớp Dangkihoc yêu cầu set lại accs thuộc tính điểm mới nhập.
40. Lớp Dangkihoc gọi các hàm set thuộc tính điểm tương ứng
41. Lớp Dangkihoc trả đối tượng về cho trang doLuudiem.jsp
42. Trang doLuudiem.jsp gọi lớp DangkihocDAO yêu cầu lưu điểm cho lớp học phần
43. Lớp DangkihocDAO gọi hàm luuDiemcuaLHP()
44. Lớp DangkihocDAO trả kết quả lại cho trang doLuudiem.jsp
45. Trang doLuudiem.jsp hiện thống báo lưu thành công
46. Giảng viên click vào OK của thông báo
47. Trang doLuudiem.jsp gọi trang gdChinhGV.jsp
48. Trang gdChinhGV.jsp hiển thị



***c. Thiết kế động cho modul xem thống kê loại học lực***



1. Tại trang gdChinhQL.jsp, sau khi đăng nhập, nhân viên quản lí chọn chức năng xem thống kê
2. Trang gdChinhQL.jsp gọi trang gdChonTK.jsp
3. Trang gdChonTK.jsp hiển thị cho NV quản lí
4. NV quản lí click xem thống kê loại học lực
5. Trang gdChonTK.jsp gọi trang gdTKHocluc.jsp
6. Trang gdTKHocluc.jsp gọi lớp KihocDAO yêu cầu lấy danh sách các kì đã học
7. Lớp KihocDAO gọi hàm getKihoc()
8. Hàm getKihoc() thực hiện và gọi lớp Kihoc đóng gói thông tin
9. Lớp Kihoc đóng gói thông tin thực thể.
10. Lớp Kihoc trả kết quả về cho hàm getKihoc()
11. Hàm getKihoc() trả kết quả về cho trang gdTKHocluc.jsp
12. Trang gdTKHocluc.jsp hiển thị cho NV quản lí
13. NV quản lí click chọn 1 kì học để xem thống kê
14. Trang gdTKHocluc.jsp submit vào chính nó để xử lí
15. Trang gdTKHocluc.jsp gọi lớp TKLoaihoclucDAO yêu cầu lấy thông tin thống kê
16. Lớp TKLoaihoclucDAO gọi hàm getTKHocluc()
17. Hàm getTKHocluc() thực hiện và gọi lớp TKLoaihocluc đóng gói thông tin
18. Lớp TKLoaihocluc đóng gói thông tin thực thể
19. Lớp TKLoaihocluc trả kết quả cho hàm getTKHocluc()
20. Hàm getTKHocluc() trả kết quả cho trang gdTKHocluc.jsp
21. Trang gdTKHocluc.jsp hiển thị cho NV quản lí
22. NV quản lí click chọn 1 loại học lực
23. Trang gdTKHocluc.jsp gọi trang gdTKHLSinhvien.jsp
24. Trang gdTKHLSinhvien.jsp gọi lớp TKSinhvienDAO yêu cầu lấy thông tin thống kê
25. Lớp TKSinhvienDAO gọi hàm getTKSVcuaHL()
26. Hàm getTKSVcuaHL() thwucj hiện và gọi lớp TKSinhvien đóng gói thông tin
27. Lớp TKSinhvien đóng gói thông tin thực thể
28. Lớp TKSinhvien trả kết quả lại cho hàm getTKSVcuaHL()
29. Hàm getTKSVcuaHL() trả kết quả lại cho trang gdTKHLSinhvien.jsp
30. Trang gdTKHLSinhvien.jsp hiển thị cho NV quản lí
31. NV quản lí click 1 sinh viên
32. Trang gdTKHLSinhvien.jsp gọi trang gdTKDiemSV.jsp
33. Trang gdTKDiemSV.jsp gọi lớp DangkihocDAO yêu cầu lấy thông tin điểm các môn học của sinh viên
34. Lớp DangkihocDAO gọi hàm getDiemcuaSV()
35. Hàm getDiemcuaSV thực hiện và gọi lớp Dangkihoc đóng gói thông tin
36. Lớp Dangkihoc đóng gói thông tin thực thể
37. Lớp Dangkihoc trả kết quả cho hàm getDiemcuaSV()
38. Hàm getDiemcuaSV() trả kết quả lại cho trang gdTKDiemSV.jsp
39. Trang gdTKDiemSV.jsp hiển thị cho NV quản lí
40. NV quản lí click vào một môn học
41. Trang gdTKDiemSV.jsp gọi trang gdDiemMHSV.jsp
42. Trang gdDiemMHSV.jsp gọi lớp DangkihocDAO yêu cầu lấy thông tin điểm chi tiết của môn học
43. Lớp DangkihocDAO gọi hàm getDiemMHcuaSV()
44. Hàm getDiemMHcuaSV() thực hiện và gọi lớp Dangkihoc đóng gói thông tin
45. Lớp Dangkihoc đóng gói thông tin thực thể
46. Lớp Dangkihoc trả lại cho hàm getDiemMHcuaSV()
47. Hàm getDiemMHcuaSV() trả kết quả cho trang gdDiemMHSV.jsp
48. Trang gdDiemMHSV.jsp hiển thị cho NV quản lí



**Share this:**

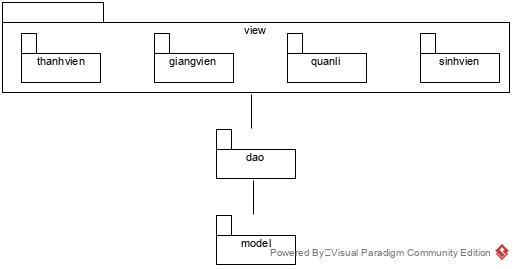
**Thiết kế triển khai**

Quy trình tiến hành các bước như sau:

* Bước 1: Vẽ biểu đồ triển khai của hệ thống: Các máy sẽ cài đặt hệ thống. Mỗi máy đặt các modul hay package nào. Các modul/package tương tác với nhau thế nào.

Áp dụng cho hệ thống, các package được thiết kế theo cấu trúc trong Hình 4.15:

* Các lớp thực thể đặt chung trong gói model
* Các lớp DAO đặt chung trong gói dao.
* Các trang jsp để trong gói view. Gói view được chia nhỏ thành các gói nhỏ hơn tương ứng với các giao diện cho các người dùng khác nhau:
  + Các trang thao tác cho đăng nhập, quản lí mật khẩu đặt trong gói thanhvien.
  + Các trang cho chức năng liên quan đến giảng viên đặt trong gói giangvien
  + Các trang cho chức năng liên quan đến nhân viên quản lí đặt trong gói quanli
  + Các trang cho chức năng liên quan đến sinh viên đặt trong gói sinhvien



**Share this:**