

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO BÀI TẬP LỚN**

**PHÁT TRIỂN PHẦN MỀM HƯỚNG DỊCH VỤ**

**ĐỀ TÀI : HỆ THỐNG NHẬP ĐIỂM CHO SINH VIÊN TỪ FILE**

**Giảng viên: TS. Đặng Ngọc Hùng**

**Trần Minh Nghĩa - B20DCCN471 Lương Nhật Tuấn - B20DCCN615**

**Lê Duy Mạnh - B20DCCN423**

**Hà Nội, 2024**

Mục lục

[**I.** **Phân tích** 3](#_Toc166671999)

[**1.1.** **Phân tách quy trình thành các hành động chi tiết** 4](#_Toc166672000)

[**1.2.** **Lọc ra các hành động không phù hợp** 4](#_Toc166672001)

[**1.3.** **Xác định các ứng viên dịch vụ thực thể** 5](#_Toc166672002)

[**1.4.** **Xác định logic cụ thể quy trình** 7](#_Toc166672003)

[**1.5.** **Xác định các nguồn tài nguyên** 7](#_Toc166672004)

[**1.6.** **Liên kết các năng lực dịch vụ với nguồn và phương thức** 8](#_Toc166672005)

[**1.7.** **Xác định ứng viên tổ hợp dịch vụ** 8](#_Toc166672006)

[9](#_Toc166672007)

[**1.8.** **Phân tích các yêu cầu xử lý** 9](#_Toc166672008)

[ **Các chức năng mang tính chất tiện ích** 9](#_Toc166672009)

[ **Các hành động có yêu cầu xử lý quan trọng và chuyên biệt :** 9](#_Toc166672010)

[**1.9.** **Xác định các ứng viên dịch vụ tiện ích (Và liên kết các tài nguyên phương thức)** 9](#_Toc166672011)

[10](#_Toc166672012)

[**1.10.** **Xác định các ứng viên microservice (Liên kết tài nguyên và phương thức)** 10](#_Toc166672013)

[11](#_Toc166672014)

[**1.11.** **Cấu trúc phân cấp các ứng viên dịch vụ sau khi kết hợp và mở rộng** 11](#_Toc166672015)

[**II.** **Thiết kế** 12](#_Toc166672016)

[**2.1.** **Biểu đồ tuần tự** 12](#_Toc166672017)

**Case study: Import điểm từ file**

1. **Phân tích**

Thực hiện mô hình hóa dịch vụ REST cho quy trình “Import điểm cho sinh viên từ file”.

Quy trình này bao gồm:

* Đánh giá bảng điểm do giảng viên nhập cho môn học của học kì mà giảng viên đang dạy
* Kiểm tra file điểm nhập vào có hợp lệ hay không
* Kiểm tra xem môn học có do giảng viên giảng dạy không
* Phê duyệt nhập điểm của giảng viên
  1. **Phân tách quy trình thành các hành động chi tiết**

Quy trình nhập điểm được chia thành các hành động chi tiết:

* Bắt đầu nhập điểm
* Xác định thời gian nhập điểm
* Nếu ngoài thời gian thì không cho phép nhập
* Nhập vào file cần lấy điểm
* Lấy ra kỳ học có trong file
* Xác thực các kỳ học mà giảng viên đang dạy
* Lấy ra môn học trong file
* Xác minh môn học phải do giáo viên giảng dạy
* Nếu không phải bắt đầu từ chối và hiển thị thông báo
* Gửi thông báo từ chối
* Gửi thông báo chấp nhận
* Lấy ra lớp học phần trong file
* Xác minh lớp học phần do giáo viên giảng dạy
* Nếu không phải bắt đầu từ chối
* Gửi thông báo từ chối
* Gửi thông báo chấp nhận
* Lấy ra danh sách sinh viên
* Export điểm tương ứng
* Gửi thông báo chấp nhận
* Thêm bản ghi điểm vào cơ sở dữ liệu
  1. **Lọc ra các hành động không phù hợp**
* Bắt đầu nhập điểm
* ~~Xác định thời gian nhập điểm~~
* Nếu ngoài thời gian thì không cho phép nhập
* Nhập vào file cần lấy điểm
* Lấy ra kỳ học có trong file
* Xác thực các kỳ học mà giảng viên đang dạy
* Lấy ra môn học trong file
* Xác minh môn học phải do giáo viên giảng dạy
* Nếu không phải bắt đầu từ chối và hiển thị thông báo
* Gửi thông báo từ chối
* Gửi thông báo chấp nhận
* Lấy ra lớp học phần trong file
* Xác minh lớp học phần do giáo viên giảng dạy
* Nếu không phải bắt đầu từ chối
* Gửi thông báo từ chối
* Gửi thông báo chấp nhận
* Lấy ra danh sách sinh viên
* Export điểm tương ứng
* Gửi thông báo chấp nhận
* Thêm bản ghi điểm vào cơ sở dữ liệu
  1. **Xác định các ứng viên dịch vụ thực thể**
* Kết quả xác định các hành động bất khả tri:
  + Lấy danh sách kỳ học
  + Gửi thông báo từ chối
  + Gửi thông báo chấp nhận
  + Lấy danh sách môn học
  + Lấy chi tiết lớp học phần
  + Lấy danh sách sinh viên
  + Giảng viên nhập điểm
  + Giảng viên click lưu bảng điểm
  + Thêm bảng điểm vào cơ sở dữ liệu
* Các hành động bất khả tri được phân loại là các ứng viên năng lực dịch vụ sơ bộ và được nhóm lại tương ứng vào các ứng viên năng lực dịch vụ:
  + Ứng viên dịch vụ kỳ học (Term)
    - GetListTerm
    - CheckListTerm
  + Ứng viên dịch vụ môn học (Subject)
    - GetListSubject
    - CheckListSubject
  + Ứng viên dịch vụ lớp học phần(SectionClass)
    - GetListSectionClass
  + Ứng viên dịch vụ Sinh viên (Student)
    - GetListStudent
    - CheckListStudent
    - saveScore
    - checkScore
  1. **Xác định logic cụ thể quy trình**
* Các hành động sau đây được coi là không bất khả tri vì chúng dành riêng cho quy trình nghiệp vụ “Nhập điểm cho sinh viênˮ
  + Bắt đầu nhập điểm
  + Xác minh môn học phải do giáo viên giảng dạy
  + Nếu không phải, bắt đầu từ chối
  + Xác minh lớp học phần phải do giáo viên giảng dạy
  + Nếu không phải, bắt đầu từ chối
  + Xác minh sinh viên thuộc lớp học phần phải do giáo viên giảng dạy
  + Nếu không phải, bắt đầu từ chối
  + Giảng viên nhập điểm
  + Giảng viên click lưu bảng điểm
  + Xác minh điểm giảng viên nhập vào có đáp ứng yêu cầu
  + Nếu bảng điểm giảng viên không đủ điều kiện , Bắt đầu từ chối
  + Thêm bản ghi điểm vào cơ sở dữ liệu
  1. **Xác định các nguồn tài nguyên**

Sau khi xem xét các yêu cầu xử lý của khả năng dịch vụ đã được xác định , các tài nguyên tiềm năng sau đây được xem xét :

\_ /EnterScoresForStudents/

\_ /SectionClass/

\_ /Student/

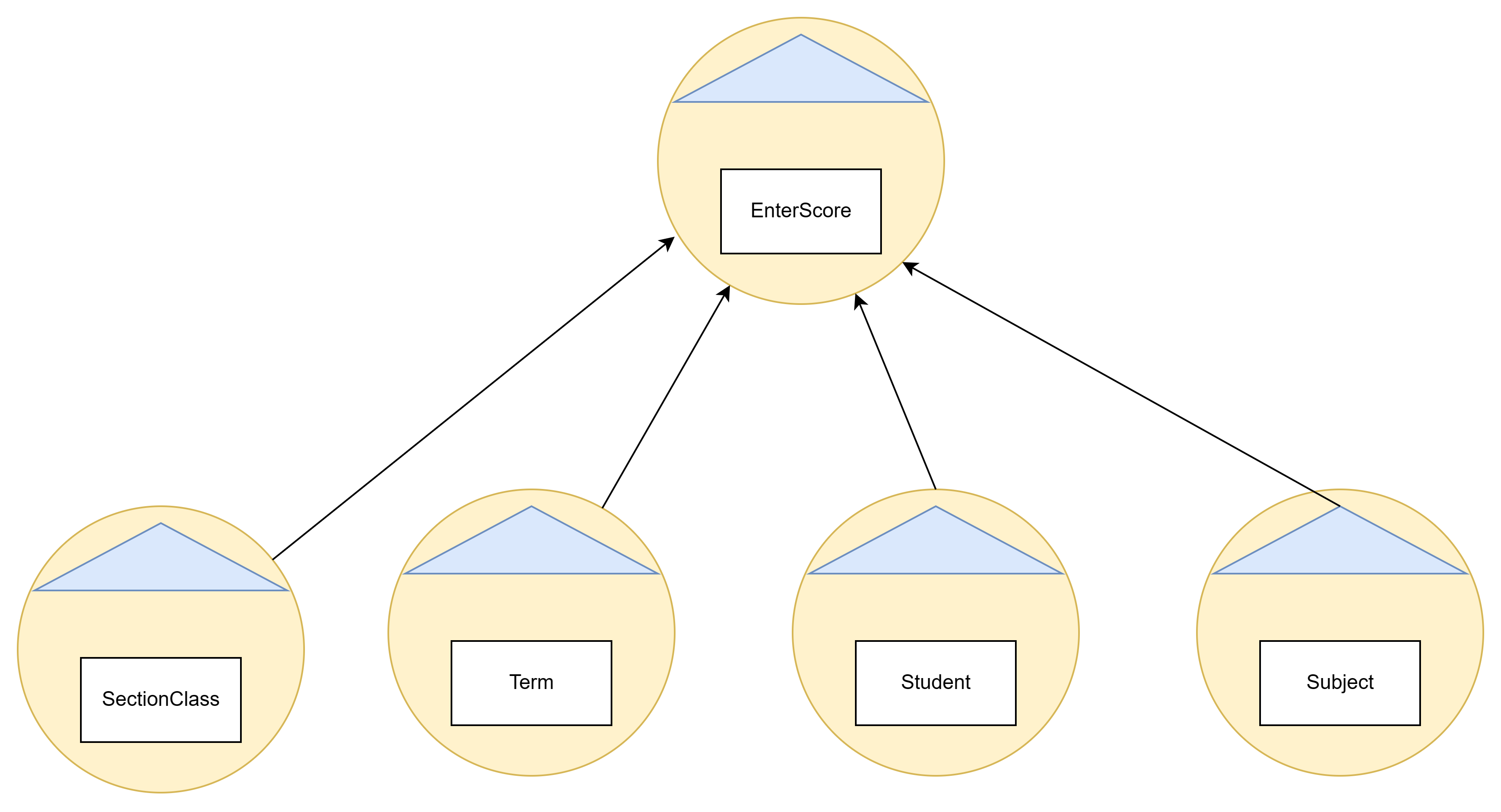
\_ /Subject/

\_ /Term/

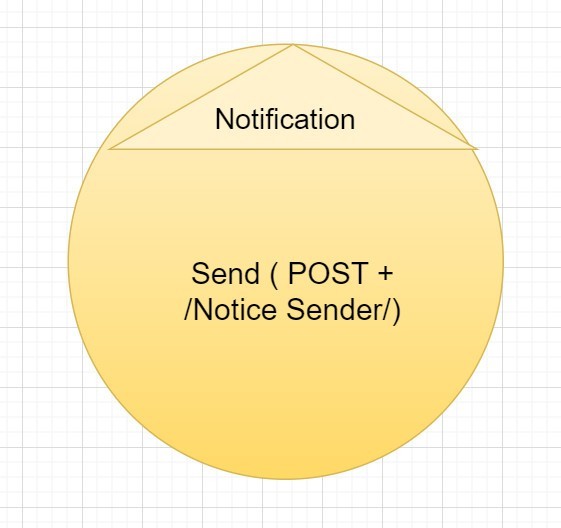
Thiết lập một số ánh xạ sơ bộ giữa các tài nguyên và thực thể kinh doanh được xác định

|  |  |
| --- | --- |
| **Entity** | **Resource** |
| SectionClass | \_ /SectionClass/ |
| Student | \_ /Student/ |
| Subject | \_ /Subject/ |
| Term | \_ /Term/ |

* 1. **Liên kết các năng lực dịch vụ với nguồn và phương thức**
* Ứng viên dịch vụ Kì học(Term) :
  + GetListTerm (GET +/ Term/)
  + CheckTerm ( POST + /Term/ )
* Ứng viên dịch vụ Môn học(Subject):
  + GetListSubject(GET +/ Subject/)
  + CheckSubject ( POST + /Subject/ )
* Ứng viên dịch vụ Lớp học phần (SectionClass):
  + GetList(GET + /SectionClass/)
  + CheckCapacity(POST +/SectionClass/)
  + CheckDuplicateSchedule(POST + /SectionClass/)
* Ứng viên dịch vụ Sinh viên (Student):
  + CheckSubjectOfSemester (POST + /Student/)
  + saveScore(POST + /Student/SaveScore/)
  + checkScore(POST + /Student/CheckScore/)
* (Tác vụ)EnterScoresForStudents
  + Start (POST + /EnterScoresForStudents/)
  1. **Xác định ứng viên tổ hợp dịch vụ**

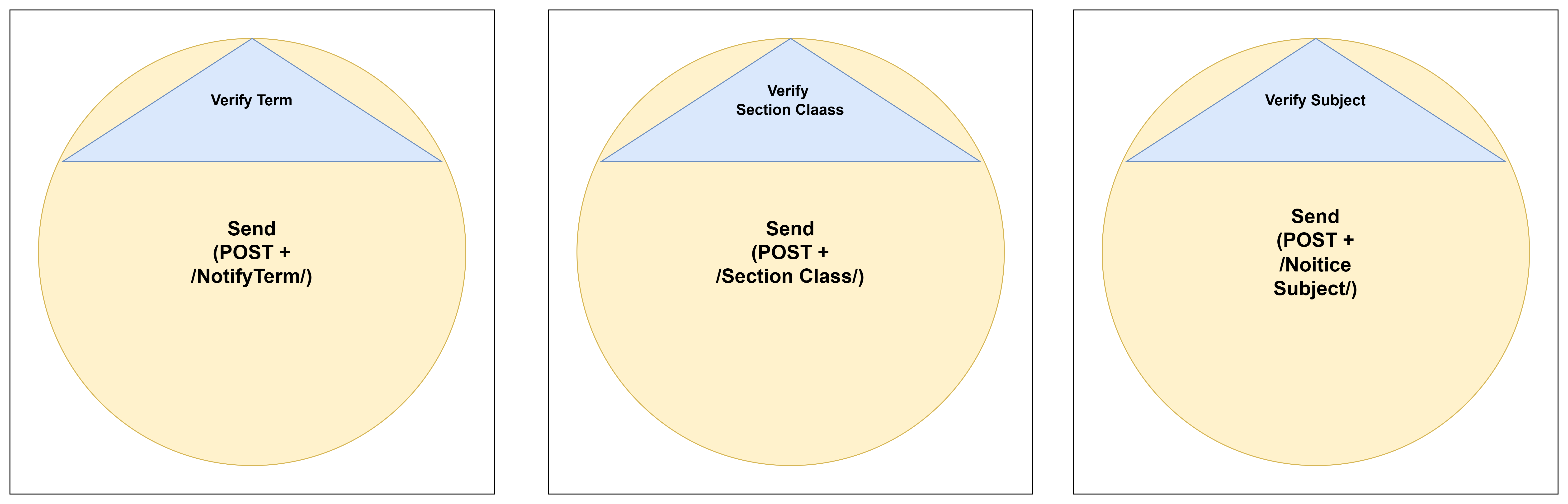


* 1. **Phân tích các yêu cầu xử lý**
* **Các chức năng mang tính chất tiện ích**
  + Gửi thông báo từ chối
  + Gửi thông báo cho phép
  + Gửi thông báo thành công
* **Các hành động có yêu cầu xử lý quan trọng và chuyên biệt :**
  + Kiểm tra xem sinh viên thuộc lớp học phần phải do giáo viên giảng dạy
  + Kiểm tra môn học phải do giáo viên giảng dạy
  + Kiểm tra điểm giảng viên nhập vào có đáp ứng yêu cầu
  1. **Xác định các ứng viên dịch vụ tiện ích (Và liên kết các tài nguyên phương thức)**

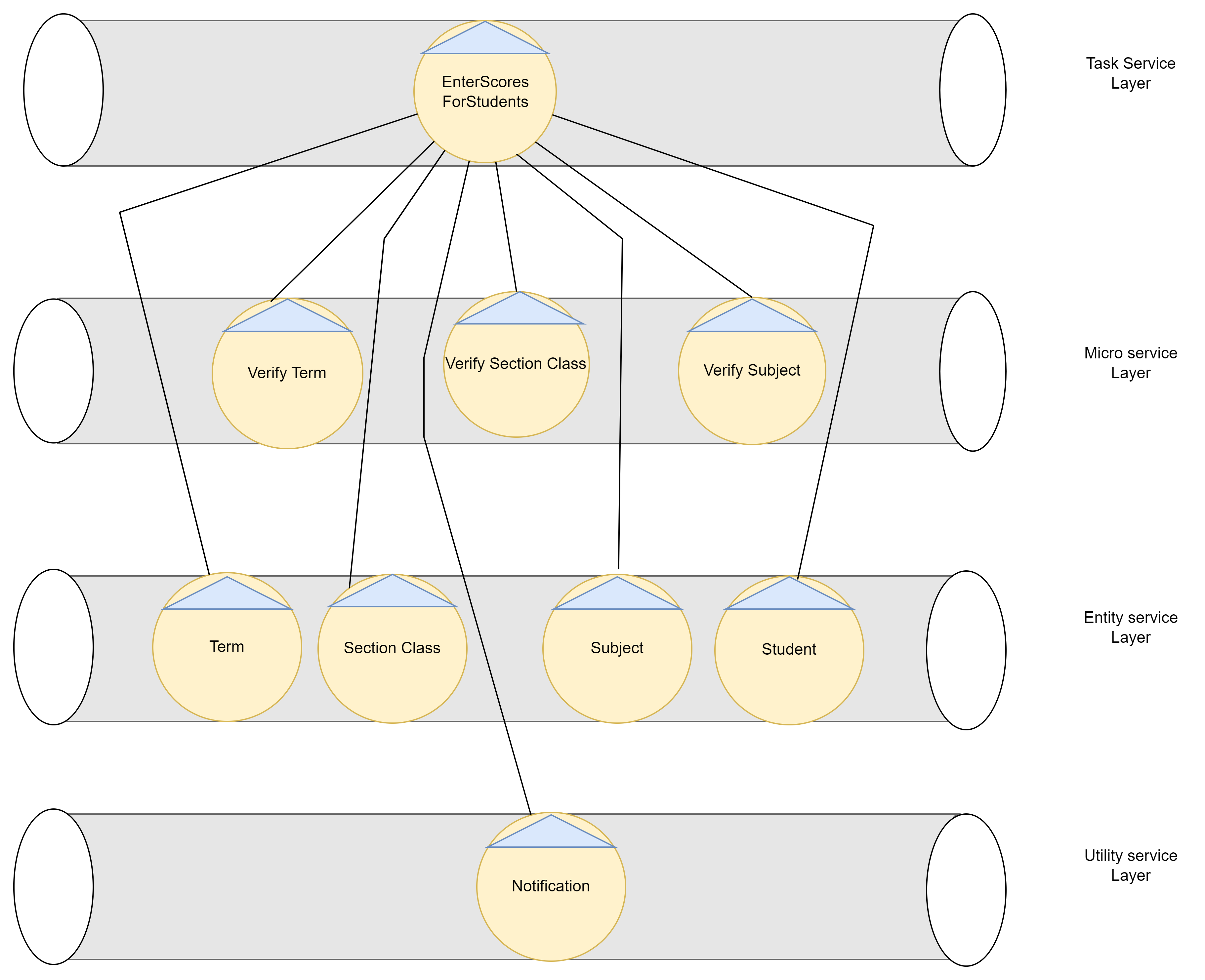


* Ứng viên dịch vụ thông báo
  1. **Xác định các ứng viên microservice (Liên kết tài nguyên và phương thức)**

**Để cô lập việc xử lý hành động** “Kiểm tra xem danh sách các kỳ học có đủ yêu cầu nằm trong thời gian hợp lệ hay không ˮ **⇒ đề xuất một ứng viên microservice có tên “Verify Termˮ** , và hành động “ Kiểm tra lớp học phần mới được chọn có lịch học trùng với lịch học của lớp học phần đã chọn ˮ ⇒ **đề xuất một ứng viên có tên “Verify Section Classˮ** và hành động “Kiểm tra xem môn học có nằm trong học kỳ hợp lệ hay không **có tên “Verify Subjectˮ**



* 1. **Cấu trúc phân cấp các ứng viên dịch vụ sau khi kết hợp và mở rộng**



1. **Thiết kế**
   1. **Biểu đồ tuần tự**

