**PROJECT REPORT**

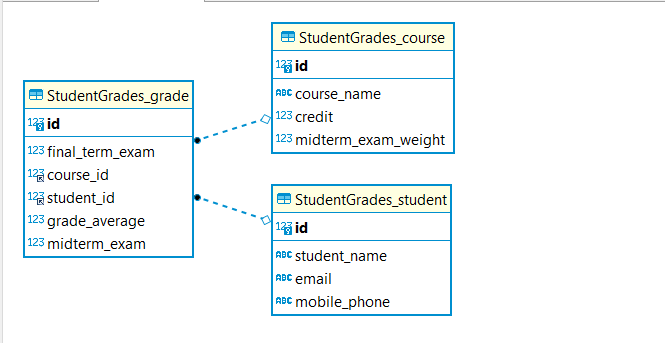
Dự án xây dựng chức năng thêm, sửa, xóa, lấy thông tin sinh viên, khóa học và điểm số từ trong database.

1. **Công nghệ**

Dự án sử dụng các công nghệ sau:

* Ngôn ngữ lập trình: Python version 3.8
* Framework: Django
* Database: SQL
* Docker

1. **Database**



Database lưu trữ dữ liệu gồm 3 bảng:

1. Bảng student:

* Bảng lưu trữ thông tin của sinh viên, bao gồm 4 cột: id, student\_name, email

mobile\_phone

1. Bảng course:

* Bảng lưu trữ thông tin của từng khóa học, bao gồm 4 cột: id, course\_name

Credit, midterm\_exam\_weight. Midterm\_exam\_weight là trọng số của bài thi giữa kỳ, được dùng để tính toán điểm trung bình của từng học sinh

1. Bảng grade:

* Bảng lưu trữ thông tin về điểm số, bao gồm 6 cột: id, final\_term\_exam, course\_id (foreign key, refererence to course table), student\_id (foreign key, reference to student table), grade\_average, midterm\_exam.

1. **Application**

Dựa theo cấu trúc file của Framework Django, cấu trúc file được chỉnh sửa lại bao gồm:

* urls.py
* models.py
* controller.py
* services.py
* repository.py

1. urls.py

File chứa các path dẫn request, trong đó:

* Path **students/** và **students/<int:id>**: Có thể tạo request thêm, sửa, xóa, lấy thông tin sinh viên và theo id sinh viên. Request sẽ được truyền đến hàm get\_all\_students và get\_one\_student tương ứng trong file controller.
* Path **students/<int:id>/grades:** Có thể tạo request thêm, sửa, xóa, lấy thông tin điểm số theo từng sinh viên. Request sẽ được truyền đến hàm grades\_by\_students tương ứng trong file controller
* Path **courses/** và **<courses/<int:id>/:** Có thể tạo request thêm, sửa, xóa, lấy thông tin khóa học và theo id khóa học. Request sẽ được truyền đến hàm get\_all\_courses và get\_one\_course tương ứng trong file controller
* Path **<courses/<int:id>/ grades:** Có thể tạo request thêm, sửa, xóa, lấy thông tin điểm số theo từng khóa học Request sẽ được truyền đến hàm grades\_by\_course tương ứng trong file controller

1. models.py

File models.py được dùng để define các model migrate vào database, gồm 3 class tương ứng với 3 bảng: Student, Course và Grade

1. controller.py

**def transform\_incoming\_data(requestBody)**

Hàm được định nghĩa với mục đích chuyển data từ request body sang dạng dictionary để tiện xử lý. Decode data bytestring sau đó chuyển sang dictionary thông qua function json.loads().

**def get\_all\_students(request: HttpRequest)**

Hàm được định nghĩa để xử lý request truyền vào từ đường dẫn **students/** có thể lấy data toàn bộ sinh viên và thêm sinh viên mới.

* Lấy thông tin sinh viên: Query lấy toàn bộ data bảng student thông qua function query\_student() được define tại repository.py, chuyển sang json format và trả response.
* Tạo sinh viên mới:

+ Gọi hàm **transform\_incoming\_data(requestBody)** để xử lý data gửi từ body request

+ Gọi hàm query\_insert\_student define trong repository.py để insert data vào database

+ Trả lại kết quả record được insert thành công.

**def get\_one\_student (request: HttpRequest, id)**

* Query bảng student trong database theo id truyền vào qua function query\_student(). Nếu không tìm được id trả lại status code 404.
* Lấy thông tin sinh viên: Trả lại thông tin sinh viên đó dạng json
* Update thông tin sinh viên:

+ Chuyển request data thành dạng dictionary

+ Update thông tin vào database qua function query\_update\_student()

+ Trả lại thông tin student được update

* Xóa sinh viên theo ID, gọi method delete cho QuerySet bảng student.

**def grades\_by\_student (request: HtttpRequest, id)**

* Query bảng student trong database theo id truyền vào qua function query\_student(). Nếu không tìm được id trả lại status code 404.
* Lấy thông tin điểm số theo student ID: truyền student ID vào function query\_grade() để query ra toàn bộ điểm số của sinh viên đó và trả lại kết quả
* Post điểm mới cho sinh viên:

+ Transform data thành dạng dictionary

+ Query lấy midterm weight, gọi hàm calculate\_average\_mark (define trong services.py) để tính điểm trung bình

+ Post điểm vào database cùng với điểm trung bình sau khi được tính toán

+ Trả lại điểm mới được post.

Các hàm thêm, sửa, xóa, lấy thông tin khóa học tương tự như các hàm thêm, sửa, xóa, lấy thông tin sinh viên.

1. services.py

Hàm calculate\_average\_mark được dùng để tính điểm trung bình với công thức: midterm + midterm\_weight + final\_term\_exam \* (1-midterm\_weight)

1. repository.py

File bao gồm các hàm query, insert, edit data trong database.