

TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

— * —



BÁO CÁO CUỐI KỲ

MÔN HỌC: THỰC HÀNH CƠ SỞ DỮ LIỆU

Mã lớp: 147780

Chủ đề: Quản lý học tập

Giảng viên hướng dẫn: Nguyễn Hữu Đức

Nhóm sinh viên thực hiện: Nguyễn Trí Đức – 20225812
Lương Văn Khanh – 20225728
Lê Hải Anh - 20225594

Hà Nội, tháng 6 năm 2024

Mục lục

I. Giới thiệu	3
1. Đặt vấn đề	3
2. Công nghệ sử dụng.....	3
II. Cơ sở dữ liệu.....	3
1. Sơ đồ thực thể liên kết.....	3
2. Sơ đồ quan hệ.....	5
3. Phân quyền	6
4. Xây dựng bảng	7
III. Truy vấn	10
IV. Function, procedure, trigger, index	18
V. Web	22
VI. Kết luận	22
1. Điểm mạnh	22
2. Điểm yếu	22
VII. Phân công công việc	23

I. Giới thiệu

1. Đặt vấn đề

Xác định yêu cầu nghiệp vụ: Xây dựng cơ sở dữ liệu quản lý học tập của sinh viên trường Đại học.

Mục đích:

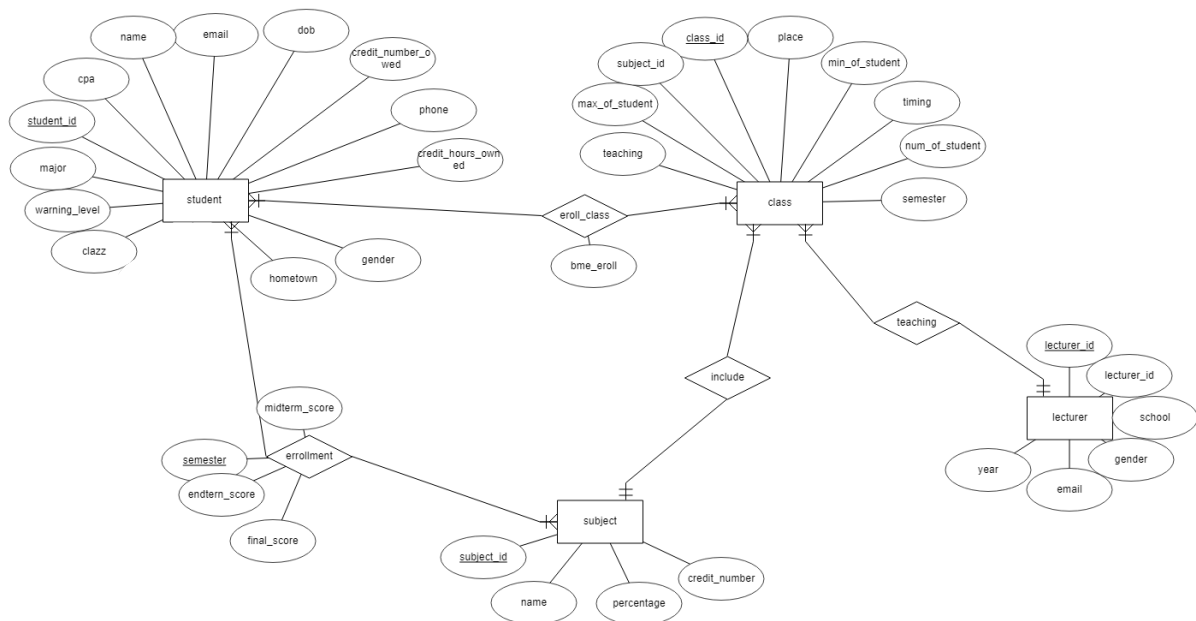
- Giúp nhà trường quản lý, phân công các công việc trong giáo dục một cách dễ dàng và tối ưu.
- Giúp sinh viên nắm bắt thông tin, chủ động trong việc đăng ký học tập, xem điểm và khiếu nại.

2. Công nghệ sử dụng

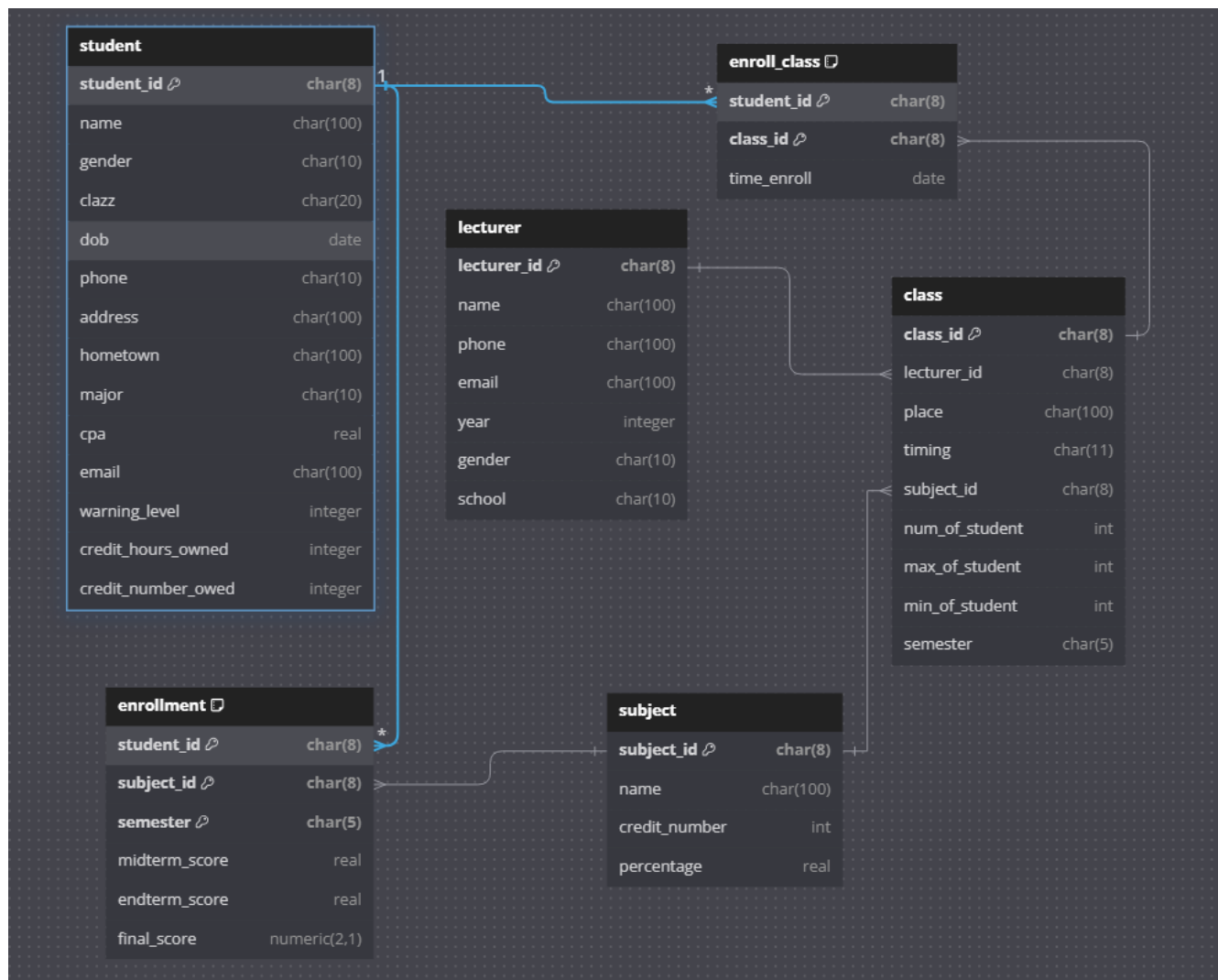
- Hệ quản trị cơ sở dữ liệu: Postgresql
- Web:
 - + Frontend: HTML, CSS
 - + Backend: Nodejs

II. Cơ sở dữ liệu

1. Sơ đồ thực thể liên kết:



2. Sơ đồ quan hệ:



3. Phân quyền

- Phân quyền với schema

edudb			
General Security Default privileges SQL			
Privileges			
	Grantee	Privileges	Grantor
	admin	UC	postgres
	lecturer	U	postgres
	postgres	UC	postgres
	student	U	postgres

- Phân quyền với từng bảng

class

General

Columns

Advanced

Constraints

Parameters

Security

SQL

Privileges

	Grantee	Privileges	Grantor
	admin v	daxrtDw	postgres
	lecturer v	r	postgres
	postgres v	rxawDtd	postgres
	student v	raw	postgres

enroll_class

General

Columns

Advanced

Constraints

Parameters

Security

SQL

Privileges

	Grantee	Privileges	Grantor
	admin v	rxwtdaD	postgres
	lecturer v	tr	postgres
	postgres v	raxdtDw	postgres
	student v	dwra	postgres

enrollment

General

Columns

Advanced

Constraints

Parameters

Security

SQL

Privileges

	Grantee	Privileges	Grantor
	admin v	rt	postgres
	postgres v	xdartDw	postgres
	student v	arwd	postgres

lecturer

General

Columns

Advanced

Constraints

Parameters

Security

SQL

Privileges

	Grantee	Privileges	Grantor
	admin v	tdwaDxr	postgres
	lecturer v	r	postgres
	postgres v	rtdDxwa	postgres

student			
General Columns Advanced Constraints Parameters Security SQL			
Privileges +			
	Grantee	Privileges	Grantor
	admin	waxrtDd	postgres
	lecturer	r	postgres
	postgres	wdraDxt	postgres
	student	r	postgres

subject			
General Columns Advanced Constraints Parameters Security SQL			
Privileges +			
	Grantee	Privileges	Grantor
	admin	waxrtDd	postgres
	lecturer	r	postgres
	postgres	wdtaDxr	postgres
	student	r	postgres

4. Xây dựng bảng

- Bảng 'student' (sinh viên)

Thuộc tính:

+ student_id (character) (PK)	Mã số sinh viên, định danh duy nhất của sinh viên
+ name (character)	Tên sinh viên
+ gender (character)	Giới tính sinh viên
+ clazz (character)	Lớp của sinh viên
+ dob (date)	Ngày tháng năm sinh của sinh viên
+ phone (character)	Số điện thoại của sinh viên
+ address (character)	Địa chỉ của sinh viên
+ hometown (character)	Quê quán của sinh viên
+ major (character)	Ngành học của sinh viên
+ cpa (real)	CPA của sinh viên
+ email (character)	Email của sinh viên
+ warning_level (integer)	Mức cảnh cáo của sinh viên
+ credit_hours_owed	Số giờ tín chỉ sinh viên đã sở hữu
+ credit_number_owed	Số tín chỉ nợ của sinh viên

```

1  --CREATE TABLE
2  CREATE TABLE student (
3      student_id CHAR(8) PRIMARY KEY,          -- Mã số sinh viên
4      name CHAR(100),                          -- Tên sinh viên
5      gender CHAR(10),                        -- Giới tính sinh viên
6      clazz CHAR(20),                        -- Lớp của sinh viên
7      dob DATE,                              -- Ngày tháng năm sinh của sinh viên
8      phone CHAR(10),                        -- Số điện thoại của sinh viên
9      address CHAR(100),                    -- Địa chỉ của sinh viên
10     hometown CHAR(100),                  -- Quê quán của sinh viên
11     major CHAR(10),                      -- Ngành học của sinh viên
12     cpa REAL,                            -- CPA của sinh viên
13     email CHAR(100),                    -- Email của sinh viên
14     warning_level INTEGER,              -- Mức cảnh cáo của sinh viên
15     credit_hours_owed INTEGER,          -- Số giờ tín chỉ đã sở hữu
16     credit_number_owed INTEGER          -- Số tín chỉ nợ
17 );
18

```

- Bảng 'subject' (học phần)

Thuộc tính:

+ subject_id (character) (PK)	Mã học phần
+ name (character)	Tên học phần
+ credit_number (integer)	Số tín chỉ
+ percentage (real)	Trọng số điểm (cuối kì)

```

33 CREATE TABLE subject (
34     subject_id CHAR(8) PRIMARY KEY,          -- Mã học phần
35     name CHAR(100),                          -- Tên học phần
36     credit_number INT,                      -- Số tín chỉ
37     percentage REAL,                        -- Trọng số điểm cuối kì
38     check(percentage > 0 and percentage < 1)
39 );

```

- Bảng 'lecturer' (giảng viên)

Thuộc tính:

+ lecturer_id (character) (PK)	Mã giảng viên
+ name (character)	Tên giảng viên
+ phone (character)	Số điện thoại của giảng viên
+ email (character)	Email của giảng viên
+ year (integer)	Năm sinh của giảng viên
+ gender (character)	Giới tính giảng viên
+ school (character)	Trường, viện giảng viên trực thuộc

```

41 CREATE TABLE lecturer (
42     lecturer_id CHARACTER VARYING PRIMARY KEY,      -- Mã giảng viên
43     name CHAR(100),                                -- Tên giảng viên
44     phone CHAR(100),                                -- Số điện thoại của giảng viên
45     email CHAR(100),                                -- Email của giảng viên
46     year INTEGER,                                   -- Năm sinh của giảng viên
47     gender CHAR(10),                                -- Giới tính giảng viên
48     school CHAR(10)                                 -- Trường, viện giảng viên trực thuộc
49 );

```

- Bảng 'class' (lớp học)

Thuộc tính:

+ class_id (character) (PK)	Mã lớp học
+ lecturer_id (character)	Mã giảng viên
+ place (character)	Địa điểm (lớp) học
+ timing (character)	Thời gian (lớp) học
+ subject_id (character)	Mã học phần
+ num_of_student (integer)	Số lượng sinh viên

- + max_of_student (integer) Số sinh viên tối đa
- + min_of_student (integer) Số sinh viên tối thiểu
- + semester (char) Kỳ học

```

19 CREATE TABLE class (
20     class_id CHAR(8) PRIMARY KEY,           -- Mã lớp học
21     lecturer_id CHAR(8),                     -- Mã giảng viên
22     place CHAR(100),                         -- Địa điểm (lớp) học
23     timing CHAR(11),                         -- Thời gian (lớp) học
24     subject_id CHAR(8),                     -- Mã học phần
25     num_of_student INT,                     -- Số lượng sinh viên
26     max_of_student INT,                     -- Số sinh viên tối đa
27     min_of_student INT,                     -- Số sinh viên tối thiểu
28     FOREIGN KEY (subject_id) REFERENCES subject(subject_id),
29     FOREIGN KEY (lecturer_id) REFERENCES lecturer(lecturer_id),
30     semester CHAR(5)                        -- Kỳ học
31 );
32

```

- Bảng 'enrollment' ()

Thuộc tính:

- + student_id (character) (PK) Mã số sinh viên
- + subject_id (character) (PK) Mã học phần
- + semester (character) (PK) Kỳ học
- + midterm_score (real) Điểm giữa kì
- + endterm_score (real) Điểm cuối kì
- + final_score (numeric(2, 1)) Điểm tổng kết

```

33 CREATE TABLE enrollment (
34     student_id CHAR(8) REFERENCES student(student_id), -- Mã sinh viên
35     subject_id CHAR(8) REFERENCES subject(subject_id), -- Mã học phần
36     semester CHAR(5),                                   -- Học kỳ
37     midterm_score REAL,                                 -- Điểm giữa kỳ
38     endterm_score REAL,                                -- Điểm cuối kỳ
39     final_score NUMERIC(2, 1),
40     PRIMARY KEY (student_id, subject_id, semester)
41 );

```

- Bảng 'enroll_class' (đăng kí lớp)

Thuộc tính:

- + student_id (character) (PK) Mã số sinh viên
- + class_id (character) (PK) Mã lớp học
- + time_enroll (date) Thời gian đăng kí

```

74 CREATE TABLE enroll_class (
75     student_id CHAR(8) REFERENCES student(student_id), -- Mã sinh viên
76     class_id CHAR(8) REFERENCES class(class_id),         -- Mã lớp học
77     time_enroll DATE,                                   -- Thời gian đăng ký
78     PRIMARY KEY (student_id, class_id)
79 );

```

III. Truy vấn

1. Thêm học phần :

Query		Query History			
<pre> 1 select * from subject; 2 --Them hoc phan 3 insert into subject(subject_id, name, credit_number, percentage) 4 values('SUBJ0006', 'Database Lab', '2', '0.7'); 5 </pre>					
Data Output		Messages			
	subject_id [PK] character	name character	credit_number integer	percentage real	
1	SUBJ0001	Mathematics 1	3	0.6	
2	SUBJ0002	Physics 1	4	0.7	
3	SUBJ0003	Chemistry 1	3	0.5	
4	SUBJ0004	Biology 1	3	0.65	
5	SUBJ0005	Computer Science 1	4	0.8	
6	SUBJ0006	Database Lab	2	0.7	

2. Xóa học phần

Query		Query History			
<pre> 1 --Xoa hoc phan 2 delete from subject where subject_id = 'SUBJ0006'; 3 select * from subject; </pre>					
Data Output		Messages			
	subject_id [PK] character	name character	credit_number integer	percentage real	
1	SUBJ0001	Mathematics 1	3	0.6	
2	SUBJ0002	Physics 1	4	0.7	
3	SUBJ0003	Chemistry 1	3	0.5	
4	SUBJ0004	Biology 1	3	0.65	
5	SUBJ0005	Computer Science 1	4	0.8	

3. Sửa học phần

Query Query History

```
1 --Sua hoc phan
2 update subject set name = 'Database LAB' where subject_id = 'SUBJ0006';
3 select * from subject;
```

Data Output Messages Notifications

	subject_id [PK] character	name character	credit_number integer	percentage real
1	SUBJ0001	Mathematics 1	3	0.6
2	SUBJ0002	Physics 1	4	0.7
3	SUBJ0003	Chemistry 1	3	0.5
4	SUBJ0004	Biology 1	3	0.65
5	SUBJ0005	Computer Science 1	4	0.8
6	SUBJ0006	Database LAB	2	0.7

4. Giảng viên vừa dạy 'Mathematics 1' vừa dạy 'Physics 1'

Query

Query History

```
1 SELECT lecturer_id, name, phone, email
2 FROM lecturer
3 WHERE lecturer_id IN (SELECT lecturer_id FROM class WHERE subject_id IN (
4     SELECT subject_id FROM subject WHERE name = 'Mathematics 1')
5     INTERSECT
6     SELECT lecturer_id FROM class WHERE subject_id IN (
7     SELECT subject_id FROM subject WHERE name = 'Physics 1'
8 )
9 );
10
```

Data Output

Messages

Notifications

	lecturer_id [PK] character	name character	phone character	email character
1	L0000001	Le Van An	0912345670	an.le@univ.edu.vn

5. Chọn 2 sinh viên có điểm trung bình môn 'Mathematics 1' cao nhất kỳ 20231

Query

Query History

```
1 --Chon 2 sinh vien co diem trung binh mon Mathematics 1 nhat ki 20231
2 select student.student_id, student.name,
3 round ((midterm_score* (1-percentage) + endterm_score*percentage)::numeric, 2) as average_score
4 from student
5 join enrollment using (student_id) as c1
6 join subject using (subject_id)
7 where semester = '20231' and subject.name = 'Mathematics 1'
8 order by (average_score) desc limit 2;
```

Data Output

Messages

Notifications

	student_id [PK] character	name character	average_score numeric
1	S0000002	Tran Thi Binh	8.20
2	S0000003	Le Van Cuong	7.48

6. Tính tổng số sinh viên quê Huế

Query Query History

```

1 --Tong so sinh vien que Hue
2 select count (*) as total_student from student
3 where hometown = 'Hue';

```

Data Output Messages Notifications

	total_student bigint
1	1

7. Danh sách sinh viên > 18 tuổi ở lớp C0000004

Query Query History

```

1 --Danh sach sinh vien > 18 tuoi o lop C0000004
2 select student.student_id, name from student
3 inner join enroll_class on student.student_id = enroll_class.student_id
4 where class_id = 'C0000004' and extract (year from now()) - extract(year from dob ) > 18;
5
6

```

Data Output Messages Notifications

	student_id [PK] character	name character
1	S0000004	Pham Thi Dao
2	S0000009	Pham Van Minh

8. Tổng số giảng viên Trường CNTT & TT

Query Query History

```

1 --Danh sach sinh vien hoc Mathematics 1 20231
2 select student.student_id, student.name from student where student_id in
3 (select student_id from enrollment where subject_id in
4 (select subject_id from subject where subject.name = 'Mathematics 1') and semester = '20231');

```

Data Output Messages Notifications

	student_id [PK] character	name character
1	S0000003	Le Van Cuong
2	S0000002	Tran Thi Binh

11. Giảng viên dạy nhiều lớp nhất kỳ 20231

Query Query History

```

1 --Giang vien day nhieu lop nhat ky 20231
2 SELECT lecturer.lecturer_id, lecturer.name, COUNT(class.class_id) AS total_class
3 FROM lecturer
4 JOIN class ON lecturer.lecturer_id = class.lecturer_id
5 WHERE class.semester = '20231'
6 GROUP BY lecturer.lecturer_id, lecturer.name
7 ORDER BY total_class DESC
8 LIMIT 1;
9

```

Data Output Messages Notifications

	lecturer_id [PK] character	name character	total_class bigint
1	L0000001	Le Van An	4

12. Hiển thị tên lớp và số lượng sinh viên tương ứng trong mỗi lớp. Sắp xếp kết quả theo số lượng sinh viên giảm dần

Query Query History

```

1 --Hien thi ten lop va so luong sinh vien tuong ung theo so luong sinh vien giam
2 select class.class_id, count (class.class_id) as total_student
3 from class join enroll_class using (class_id)
4 join student using (student_id)
5 group by class.class_id
6 order by count (class.class_id) desc;

```

Data Output Messages Notifications

	class_id [PK] character	total_student bigint
1	C0000003	2
2	C0000001	2
3	C0000002	2
4	C0000004	2
5	C0000005	2
6	C0000006	1
7	C0000008	1
8	C0000007	1
9	C0000010	1
10	C0000009	1

13. Sinh viên có điểm môn Mathematics 1 > điểm trung bình của cả lớp kì 20231

Query Query History

```

1 --Sinh vien co diem mon Mathematics 1 > diem trung binh cua ca lop ki 20231
2 select student_id, round ((midterm_score* (1-percentage) + endterm_score*percentage) :: numeric, 2) as final_score
3 from subject join enrollment using (subject_id)
4 where (midterm_score* (1-percentage) + endterm_score*percentage) >
5 (select avg(midterm_score* (1-percentage) + endterm_score * percentage) from enrollment join subject using (subject_id)
6 where subject.name = 'Mathematics 1' and semester = '20231')
7 and subject.name = 'Mathematics 1' and semester = '20231';
8

```

Data Output Messages Notifications

	student_id character	final_score numeric
1	S0000002	8.20

14. Đưa ra danh sách học phần, điểm cuối kì cao nhất, thấp nhất, trung bình trong kì 20231

Query Query History

```
1 --Dua ra danh sach hoc phan, diem cuoi ki cao nhat, thap nhat, trung binh trong hoc ki 20231
2 select subject.subject_id, subject.name as subject_name,
3 round (max (percentage * endterm_score + (1 - percentage)*midterm_score):: numeric, 2) as max_score,
4 round (min (percentage * endterm_score + (1 - percentage) *midterm_score):: numeric, 2) as min_score,
5 round (avg(percentage * endterm_score + (1 - percentage) *midterm_score):: numeric,2) as avg_score
6 from subject
7 join enrollment on subject.subject_id = enrollment.subject_id where enrollment.semester = '20231'
8 group by subject.subject_id, subject.name;
9
```

Data Output Messages Notifications



	subject_id [PK] character	subject_name character	max_score numeric	min_score numeric	avg_score numeric
1	SUBJ0001	Mathematics 1	8.20	7.48	7.84
2	SUBJ0002	Physics 1	8.29	7.65	8.00
3	SUBJ0003	Chemistry 1	8.75	8.10	8.53
4	SUBJ0004	Biology 1	8.94	8.25	8.67
5	SUBJ0005	Computer Science 1	8.96	7.64	8.32

15. Sinh viên có nguy cơ bị đuổi học

Query Query History

```
1 --Sinh vien co nguy co bi duoi hoc (warning level = 3 )
2 select student_id, name from student where warning_level = 3;
```

Data Output Messages Notifications



	student_id [PK] character	name character
1	S0000007	Tran Van Kien

16. Sinh viên trượt môn Mathematics 1 trong kỳ 20231

Query Query History

```

1 --Sinh vien truot mon Mathematics 1 ky 20221
2 SELECT student_id, name
3 FROM student
4 WHERE student_id IN (
5     SELECT student_id |
6     FROM enrollment
7     JOIN subject USING (subject_id)
8     WHERE (midterm_score < 3 OR endterm_score < 3 OR (midterm_score * (1 - percentage) + endterm_score) / 2 < 4)
9     AND subject.name = 'Mathematics 1'
10    AND semester = '20221'
11 );
12

```

Data Output Messages Notifications

	student_id [PK] character	name character
1	S0000010	Hoang Thi Nga

17. Sinh viên học lại môn Mathematics 1 nhiều nhất

Query Query History

```

1 --Sinh vien hoc lai mon Mathematics 1 nhieu nhat
2 SELECT s.name, e.student_id, COUNT(*) AS times_learn
3 FROM enrollment e
4 JOIN student s USING (student_id)
5 JOIN subject sj USING (subject_id)
6 WHERE sj.name = 'Mathematics 1'
7 GROUP BY s.name, e.student_id
8 ORDER BY times_learn DESC
9 LIMIT 1;

```

Data Output Messages Notifications

	name character	student_id character	times_learn bigint
1	Hoang Thi Nga	S0000010	2

18. Danh sách môn học có ít nhất 3 giảng viên phụ trách kỳ 20231

Query

Query History

```

1
2 --Danh sach mon hoc co it nhat 3 giang vien phu trach ki 20231
3 SELECT subject_id, name
4 FROM subject
5 WHERE subject_id IN (
6     SELECT subject_id
7     FROM class
8     WHERE semester = '20231'
9     GROUP BY subject_id
10    HAVING COUNT(*) >= 3
11 );

```

Data Output

Messages

Notifications

≡+

▼

▼

	subject_id [PK] character	name character
1	SUBJ0001	Mathematics 1 ...
2	SUBJ0002	Physics 1 ...
3	SUBJ0003	Chemistry 1 ...
4	SUBJ0004	Biology 1 ...

19. Danh sách lớp được mở cho môn Mathematics kì 20231

Query Query History

```

1  --Danh sach lop duoc mo cho mon Mathematics 1 ki 20231
2  SELECT DISTINCT class_id
3  FROM class
4  JOIN enrollment USING (subject_id)
5  WHERE subject_id IN (
6      SELECT subject_id
7      FROM subject
8      WHERE name = 'Mathematics 1'
9      AND class.semester = '20231'
10 );
11
12

```

Data Output Messages Notifications

	class_id [PK] character
1	C0000001
2	C0000006
3	C0000011
4	C0000016

20. Liệt kê các lớp học ở tòa D3

Query Query History

```

1  --Liet ke cac lop hoc o toa D3
2  SELECT class_id, subject_id
3  FROM class
4  WHERE place LIKE '%D3%';
5

```

Data Output Messages Notifications

	class_id [PK] character	subject_id character
1	C0000001	SUBJ0001
2	C0000002	SUBJ0002
3	C0000007	SUBJ0002
4	C0000012	SUBJ0002
5	C0000017	SUBJ0002

21. Đưa ra danh sách sinh viên và tổng số tín sinh viên đăng ký trong kì 20231

Query Query History

```

1  --Dua ra danh sach sinh vien va tong so tin chi sinh vien dang ky trong ki 20231
2  SELECT t1.student_id, t1.name, t2.total_credits
3  FROM (SELECT student_id, name FROM student) AS t1
4  JOIN (
5      SELECT student_id, SUM(credit_number) AS total_credits
6      FROM enrollment
7      JOIN subject USING (subject_id)
8      WHERE semester = '20231'
9      GROUP BY student_id
10 ) AS t2 USING (student_id);
11

```

Data Output Messages Notifications

	student_id [PK] character	name character	total_credits bigint
1	S0000002	Tran Thi Binh	6
2	S0000003	Le Van Cuong	6
3	S0000004	Pham Thi Dao	11
4	S0000006	Nguyen Thi Hoa	8
5	S0000008	Le Thi Lan	3
6	S0000009	Pham Van Minh	3
7	S0000010	Hoang Thi Nga	3
8	S0000001	Nguyen Tran An	8

22. Xem thời khóa biểu của sinh viên 'S0000002' trong kì 20231

Query Query History

```

1  --Xem thời khóa biểu của sinh viên 'S0000002' trong kì 20231
2  CREATE or replace VIEW student_schedule AS
3  SELECT sj.name, c.place, c.timing, l.name AS lecturer_name
4  FROM student s
5  JOIN enroll_class ec ON s.student_id = ec.student_id
6  JOIN class c ON ec.class_id = c.class_id
7  JOIN subject sj ON c.subject_id = sj.subject_id
8  JOIN lecturer l ON c.lecturer_id = l.lecturer_id
9  WHERE s.student_id = 'S0000002' AND c.semester = '20231';
10 select * from student_schedule;

```

Data Output Messages Notifications

	name character	place character	timing character	lecturer_name character
1	Physics 1	D3 102	10:00-11:30	Le Van An
2	Chemistry 1	D3 107	13:00-14:30	Pham Thi Huong

23. Danh sách học phần giảng viên Lê Văn An dạy trong kì 20231

Query Query History

Execute script F5

```

1 --Danh sach hoc phan giang vien Le Van An trach trong hoc ki 20231
2 create view lecturer_subject as
3 select class_id, subject_id, timing, place from class
4 where lecturer_id in
5 (select lecturer_id from lecturer where name = 'Le Van An') and semester = '20231';
6 SELECT * FROM lecturer_subject;

```

Data Output Messages Notifications

	class_id character	subject_id character	timing character	place character	
1	C0000001	SUBJ0001	08:00-09:30	D3 101	...
2	C0000002	SUBJ0002	10:00-11:30	D3 102	...
3	C0000019	SUBJ0004	08:00-09:30	D7 119	...
4	C0000022	SUBJ0002	10:00-11:30	E1 122	...

24. Danh sách giảng viên dạy môn mã SUBJ0001 trong kì 20231

Query Query History

```

1 --Danh sach giảng viên dạy môn SUBJ0001 trong kì 20231
2 CREATE or replace VIEW list_of_lecturer AS
3 SELECT l.lecturer_id, l.name, l.phone, l.email
4 FROM lecturer l
5 JOIN class c ON l.lecturer_id = c.lecturer_id
6 WHERE c.semester = '20231' AND c.subject_id = 'SUBJ0001';
7
8 SELECT * FROM list_of_lecturer;
9

```

Data Output Messages Notifications



	lecturer_id character	name character	phone character	email character	
1	L0000001	Le Van An	0912345670	an.le@univ.edu.vn	...
2	L0000002	Tran Thanh Binh	0912345671	binh.tran@univ.edu.vn	...
3	L0000003	Pham Xuan Cuong	0912345672	cuong.pham@univ.edu.vn	...
4	L0000008	Le Thi Lan	0912345677	lan.le@univ.edu.vn	...

25. Danh sách sinh viên theo thứ tự giảm dần cpa

Query		Query History	
1	--Danh sach sinh vien theo thu tu giam dan cpa		
2	CREATE VIEW chart AS		
3	SELECT student_id, name, cpa FROM student		
4	ORDER BY cpa DESC;		
5			
6	SELECT * FROM chart;		
Data Output		Messages	Notifications
<div> <div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> </div> </div>			
	student_id character 🔒	name character 🔒	cpa real 🔒
1	S0000008	Le Thi Lan	3.9
2	S0000004	Pham Thi Dao	3.8
3	S0000003	Le Van Cuong	3.7
4	S0000002	Tran Thi Binh	3.6
5	S0000001	Nguyen Tran An	3.5
6	S0000010	Hoang Thi Nga	3.5
7	S0000006	Nguyen Thi Hoa	3.3
8	S0000005	Hoang Van Duc	3.2
9	S0000009	Pham Van Minh	3.1
10	S0000007	Tran Van Kien	2.3

26. Danh sách sinh viên ở lớp C0000001 kì 20231

```
1  --Danh sach sinh vien lop C0000001 ki 20231
2  CREATE OR REPLACE VIEW list_of_student AS
3  SELECT s.student_id, s.name
4  FROM student s
5  WHERE s.student_id IN (
6      SELECT ec.student_id
7      FROM enroll_class ec
8      JOIN class c USING (class_id)
9      WHERE c.semester = '20231' AND c.class_id = 'C0000001'
10 );
11
12 SELECT * FROM list_of_student;
```

	student_id 	name 
1	S0000006	Nguyen Thi Hoa
2	S0000001	Nguyen Tran An

27. Danh sách lớp được mở kì 2022

```
1 --Danh sach lop mo ki 20232
2 create or replace view lists_of_class as
3 select distinct class_id, subject_id
4 from class c join enrollment using (subject_id) where c.semester = '20232'
5 order by (class_id) desc;
6 SELECT * FROM lists_of_class;
```













	class_id character 	subject_id character 
1	C0000021	SUBJ0001
2	C0000020	SUBJ0005
3	C0000017	SUBJ0002
4	C0000012	SUBJ0002
5	C0000009	SUBJ0004
6	C0000005	SUBJ0005
7	C0000003	SUBJ0003

28. Danh sách sinh viên được học bổng kỳ 2022 (cpa > 3,6 và tổng tín học trong kỳ 20221 > 15 tín)

[Query](#) [Query History](#)

```
1 --Danh sach sinh vien co the apply hoc bong ( cpa>3.6 va tong tin hoc trong ki 20231 > 15 tin )
2 --do data van con it nen neu chay thi se khong co sinh vien nao, thu voi so tin chi >= 2
3 select name , student_id, major, cpa from student where student_id in
4 (select student_id from enrollment join subject using (subject_id)
5  where semester = '20231' group by (student_id)
6  having count (credit_number)>=2) and cpa > 3.6 order by (cpa) desc;
```

[Data Output](#) [Messages](#) [Notifications](#)

	name character	student_id [PK] character	major character	cpa real
1	Pham Thi Dao	S0000004	SE	3.8
2	Le Van Cuong	S0000003	CS	3.7

29. Sinh viên có tuổi lớn nhất:

[Query](#) [Query History](#)

```
1 --sinh vien lon tuoi nhat
2 SELECT student_id, name, major, cpa,
3        (extract(year from now()) - extract(year from dob)) AS age
4 FROM student
5 WHERE (extract(year from now()) - extract(year from dob)) >= ALL(
6        SELECT extract(year from now()) - extract(year from dob)
7        FROM student
8    );
9
```

[Data Output](#) [Messages](#) [Notifications](#)

	student_id [PK] character	name character	major character	cpa real	age numeric
1	S0000001	Nguyen Tran An	CS	3.5	24

30. Giảng viên sắp về hưu

Query

Query History

1

--giang vien sap ve huu (>50)

2

SELECT lecturer_id, name, extract(year from now()) - year AS age, school

3

FROM lecturer

4

WHERE extract(year from now()) - year >= 50;

Data Output

Messages

Notifications

≡+

📄

▼

📋

▼

🗑️

🗄️

⬇️

📈

	lecturer_id [PK] character	name character	age numeric	school character
1	L0000001	Le Van An	61	CS
2	L0000003	Pham Xuan Cuong	51	SE
3	L0000004	Pham Thi Huong	52	CS
4	L0000009	Pham Trung Minh	55	SE

IV. Function, procedure, trigger, index

1. Khi insert hay update điểm ở bảng enrollment thì sẽ tính xem môn đấy là qua hay không qua để cập nhập tín nợ của sinh viên

Query	Query History
1	--1 cap nhat so tin chi no cua sinh vien
2	create or replace function update_credit_number_owed()
3	returns trigger as
4	\$\$
5	declare
6	number integer;
7	credit integer := 0;
8	per real;
9	begin
10	select credit_number, percentage into number, per
11	from subject
12	where subject_id = new.subject_id;
13	if (new.midterm_score * (1 - per) + per * new.termscore) < 4
14	or new.midterm_score < 3
15	or new.termscore < 4
16	then
17	credit := number;
18	end if;
19	update student
20	set credit_number_owed = credit_number_owed + credit
21	where student_id = new.student_id;
22	return new;
23	end;
24	\$\$
25	language plpgsql;
26	
27	create or replace trigger update_credit
28	after update or insert
29	on enrollment
30	for each row
31	execute procedure update_credit_number_owed();
32	

2. Cập nhật số sinh viên của từng lớp khi có người đăng ký thêm vào hay xóa lớp đi

```
Query Query History
1 --2 cap nhat so luong sinh vien cua mot lop khi co nguoi dang ky hoac xoa lop
2
3 create or replace function update_num_of_student()
4 returns trigger as
5 $$
6 begin
7     if old.class_id is not null then
8         update class
9         set num_of_student = num_of_student - 1
10        where class_id = old.class_id;
11    end if;
12    if new.class_id is not null then
13        update class
14        set num_of_student = num_of_student + 1
15        where class_id = new.class_id;
16    end if;
17    return new;
18 end;
19 $$
20 language plpgsql;
21
22 create or replace trigger update_student
23 after insert or delete on enroll_class
24 for each row
25 execute procedure update_num_of_student();
```

3. Sau 1 kỳ khi điểm tất cả các môn đã được giảng viên nhập sẽ chạy thủ tục để cập nhập điểm tổng kết của sinh viên

```
Query Query History
1 --3 thu tuc cap nhat diem tong ket cua sinh vien
2 create or replace function final_grade (id_student char(10), id_subject char (10), semes char(10))
3 returns real as
4 $$
5 declare score real;
6 begin
7     select into score midterm_score * (1-percentage) + endterm_score * percentage
8     from enrollment join subject using (subject_id)
9     where student_id = id_student and subject_id = id_subject and semester = semes;
10    return score;
11 end;
12 $$
13 language plpgsql;
14
15 create or replace procedure update_final_grade() as
16 $$
17 begin
18     update enrollment set final_score = final_grade (student_id, subject_id, semester);
19 end;
20 $$
21 language plpgsql;
22
```

Query Query History

```
1 select * from enrollment;
```

Data Output Messages Notifications

	student_id [PK] character	subject_id [PK] character	semester [PK] character	midterm_score real	endterm_score real	final_score numeric (2,1)
1	S0000001	SUBJ0002	20231	8	7.5	[null]
2	S0000003	SUBJ0003	20231	9	8.5	[null]
3	S0000002	SUBJ0004	20231	8.5	9	[null]
4	S0000001	SUBJ0005	20231	7	7.8	[null]
5	S0000002	SUBJ0001	20231	8.5	8	[null]
6	S0000006	SUBJ0002	20231	7.8	8.5	[null]
7	S0000008	SUBJ0003	20231	8	8.2	[null]
8	S0000009	SUBJ0004	20231	9.2	8.8	[null]
9	S0000006	SUBJ0005	20231	8.8	9	[null]
10	S0000003	SUBJ0001	20231	7	7.8	[null]
11	S0000004	SUBJ0002	20231	8.2	8	[null]
12	S0000010	SUBJ0003	20231	8.5	9	[null]
13	S0000004	SUBJ0004	20231	7.8	8.5	[null]
14	S0000004	SUBJ0005	20231	9	8.2	[null]
15	S0000010	SUBJ0001	20222	3	3.5	[null]
16	S0000010	SUBJ0001	20221	1	2.5	[null]

Query Query History

```
1 call update_final_grade();
2 select * from enrollment;
```

Data Output Messages Notifications

	student_id [PK] character	subject_id [PK] character	semester [PK] character	midterm_score real	endterm_score real	final_score numeric (2,1)
1	S0000001	SUBJ0002	20231	8	7.5	7.7
2	S0000003	SUBJ0003	20231	9	8.5	8.8
3	S0000002	SUBJ0004	20231	8.5	9	8.8
4	S0000001	SUBJ0005	20231	7	7.8	7.6
5	S0000002	SUBJ0001	20231	8.5	8	8.2
6	S0000006	SUBJ0002	20231	7.8	8.5	8.3
7	S0000008	SUBJ0003	20231	8	8.2	8.1
8	S0000009	SUBJ0004	20231	9.2	8.8	8.9
9	S0000006	SUBJ0005	20231	8.8	9	9.0
10	S0000003	SUBJ0001	20231	7	7.8	7.5
11	S0000004	SUBJ0002	20231	8.2	8	8.1
12	S0000010	SUBJ0003	20231	8.5	9	8.8
13	S0000004	SUBJ0004	20231	7.8	8.5	8.3
14	S0000004	SUBJ0005	20231	9	8.2	8.4
15	S0000010	SUBJ0001	20222	3	3.5	3.3
16	S0000010	SUBJ0001	20221	1	2.5	1.9

4. Hàm đầu vào với MSSV và kì học sẽ trả về gpa của sinh viên trong kỳ đó

Query

Query History

```
1  --4
2  create or replace function s_gpa(s_id char(10), se char(10))
3  returns float
4  language plpgsql
5  as
6  $$
7  declare
8      gpa numeric(2);
9      total_credit integer;
10 begin
11     select into total_credit sum(credit_number)
12     from enrollment join subject using (subject_id)
13     where student_id = s_id and semester = se;
14     select into gpa
15     sum((midterm_score * (1-percentage) + endterm_score * percentage) * credit_number) / total_credit
16     from enrollment join subject using (subject_id)
17     where student_id = s_id and semester = se;
18     return gpa / 10 * 4;
19 end;
20 $$
```

Query

Query History

```
1  select s_gpa('S0000001', '20231');
```

Data Output

Messages

Notifications

+

📄

▼

📋

▼

🗑️

🔄

📥

📏

	s_gpa double precision	🔒
1	3.2	

5. Khi sinh viên muốn đăng ký 1 lớp , sinh viên sẽ kiểm tra xem lớp đó còn chỗ hay không, vì vậy hàm sau để sinh viên kiểm tra xem 1 lớp đã đầy hay chưa để đăng ký vào

```
Query Query History
1  --5
2  create or replace procedure check_full_class(cla_id char(10)) as
3  $$
4  declare
5      current_student integer;
6      max_student integer;
7  begin
8      select into current_student num_of_student from class
9      where class_id = cla_id;
10     select into max_student max_of_student from class
11     where class_id = cla_id;
12     if current_student >= max_student then
13         raise notice 'Class is full. Please register other class';
14     end if;
15     if current_student < max_student then
16         raise notice 'You can register this class';
17     end if;
18 end;
19 $$
20 language plpgsql;
21
```

```
Query Query History
1  call check_full_class('C00000008');
```

Data Output Messages Notifications

NOTICE: Class is full. Please register other class
CALL

Query returned successfully in 40 msec.

```
Query Query History
1  call check_full_class('C00000001');
```

Data Output Messages Notifications

NOTICE: You can register this class
CALL

Query returned successfully in 79 msec.

6. Thủ tục cho sinh viên đăng ký lớp học phần

```

130 --6 them lop hoc phan sinh vien
131 CREATE OR REPLACE PROCEDURE them_lop_hoc_phan(c_id CHAR(8))
132 AS $$
133 DECLARE
134     s_id CHAR(8);
135     class_full BOOLEAN;
136 BEGIN
137     SELECT upper(current_user) INTO s_id;
138
139     IF EXISTS (SELECT 1 FROM enroll_class ec WHERE ec.student_id = s_id AND ec.class_id = c_id) THEN
140         RAISE EXCEPTION 'Sinh viên đã đăng ký lớp học này!';
141     END IF;
142
143     SELECT num_of_student >= max_of_student INTO class_full
144     FROM class c
145     WHERE c.class_id = c_id;
146
147     IF class_full THEN
148         RAISE EXCEPTION 'Lớp học đã đủ số lượng sinh viên!';
149     ELSE
150         INSERT INTO enroll_class (student_id, class_id, time_enroll)
151         VALUES (s_id, c_id, NOW());
152     END IF;
153 END;
154 $$ LANGUAGE plpgsql;
155

```

7. Thủ tục cho sinh viên xóa lớp học phần đã đăng ký

```

156 --7 xoa lop hoc phan
157 CREATE OR REPLACE PROCEDURE xoa_lop_hoc_phan(c_id CHAR(8))
158 AS $$
159 DECLARE
160     s_id CHAR(8);
161 BEGIN
162     SELECT upper(current_user) INTO s_id;
163
164     DELETE FROM enroll_class
165     WHERE student_id = s_id AND class_id = c_id;
166
167     IF NOT FOUND THEN
168         RAISE EXCEPTION 'Sinh viên chưa đăng ký lớp học này!';
169     END IF;
170 END;
171 $$ LANGUAGE plpgsql;
172

```

8. Hàm cho sinh viên in ra thời khóa biểu cho 1 kỳ học

```

173 --8 timetable print
174 CREATE OR REPLACE FUNCTION print_timetable(semester CHAR(5))
175 RETURNS TABLE(name VARCHAR, place VARCHAR, timing VARCHAR) AS $$
176 BEGIN
177     RETURN QUERY
178     SELECT sj.name::VARCHAR, c.place::VARCHAR, c.timing::VARCHAR
179     FROM student s
180     JOIN enroll_class ec ON s.student_id = ec.student_id
181     JOIN class c ON ec.class_id = c.class_id
182     JOIN subject sj ON c.subject_id = sj.subject_id
183     WHERE s.student_id = upper(current_user)
184     AND c.semester = print_timetable.semester;
185
186 IF NOT EXISTS (SELECT 1 FROM student s
187                 JOIN enroll_class ec ON s.student_id = ec.student_id
188                 JOIN class c ON ec.class_id = c.class_id
189                 JOIN subject sj ON c.subject_id = sj.subject_id
190                 WHERE s.student_id = upper(current_user)
191                 AND c.semester = print_timetable.semester) THEN
192     RAISE EXCEPTION 'Không tìm thấy thời khóa biểu cho sinh viên hiện tại!';
193 END IF;
194 END;
195 $$ LANGUAGE plpgsql;
196

```

Index:

Query Query History


```

1  --Index
2  select * from subject;
3  do $$
4  begin
5      for i in 6..1000 loop
6          INSERT INTO subject(subject_id, name, credit_number) VALUES
7              (concat('SU', i), CONCAT('Subject ', i), mod(i, 3));
8      end loop;
9  end;
10 $$;
11 EXPLAIN ANALYZE SELECT * FROM subject WHERE credit_number >= 2;
12 create index index_credit_number on subject(credit_number);
13 EXPLAIN ANALYZE SELECT * FROM subject WHERE credit_number >= 2;
14

```

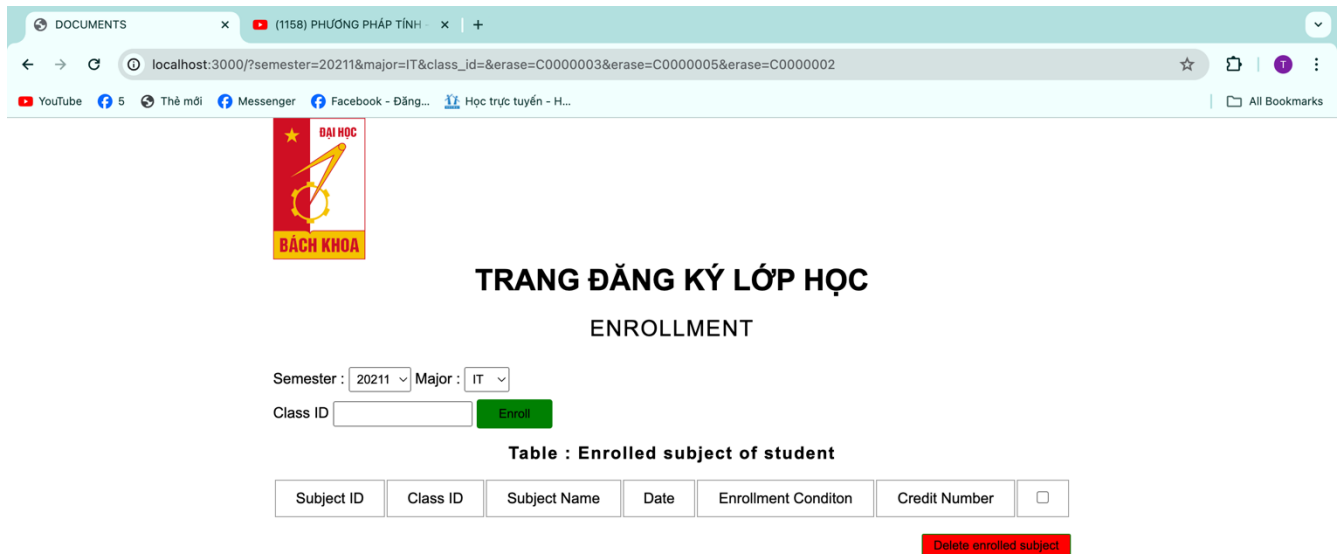
Data Output Messages Notifications



	QUERY PLAN	
	text	
1	Bitmap Heap Scan on subject (cost=6.74..29.91 rows=334 width=448) (actual time=0.024..0.054 rows=337 loops=1)	
2	Recheck Cond: (credit_number >= 2)	
3	Heap Blocks: exact=19	
4	-> Bitmap Index Scan on index_credit_number (cost=0.00..6.66 rows=334 width=0) (actual time=0.020..0.020 rows=337 loop...	
5	Index Cond: (credit_number >= 2)	
6	Planning Time: 0.587 ms	
7	Execution Time: 0.082 ms	

V. Web

Demo Web



VI. Kết luận

1. Điểm mạnh

- Về cơ sở dữ liệu
 - + Tối ưu câu lệnh truy vấn.
 - + Có phân quyền truy cập cho users.
 - + Có demo web và kết nối cơ sở dữ liệu.
 - + Hoàn thiện các yêu cầu cơ bản về thiết kế, thực hiện truy vấn.
- Về nhóm: Các thành viên tích cực làm và xây dựng project.

2. Điểm yếu

- Trang web còn đơn giản.

VII. Phân công công việc

Họ và tên	MSSV	Nhiệm vụ
Lương Văn Khanh	20225728	Thiết kế CSDL. Viết truy vấn, vẽ sơ đồ thực thể và quan hệ, function
Nguyễn Trí Đức	20225812	Thiết kế CSDL. Viết truy vấn, function. Codeweb, kết nối database.
Lê Hải Anh	20225594	Thiết kế CSDL. Viết truy vấn, function, trigger. Viết báo cáo.