**FINAL REPORT**

**TOPIC:** **SCHOOL MANAGEMENT SYSTEM**

**MEMBERS:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Student ID** | **Role** |
| Lê Thanh | ITITIU17070 | Member |
| Trần Minh Ngọc | ITITIU17068 | Member |
| Trang Thanh Mai Duyên | ITITUN17007 | Leader |

TABLE OF CONTENTS

[**1.** **INTRODUCTION** 1](#_Toc40302999)

[**a.** **My Project** 2](#_Toc40303000)

[**b.** **Tasks** 2](#_Toc40303001)

[**2.** **ENTITY-RELATIONSHIP MODELING** 2](#_Toc40303002)

[**a.** **Entity and attribute** 3](#_Toc40303003)

[**b.** **Explaination** 4](#_Toc40303004)

[**3.** **RELATIONAL MODEL** 5](#_Toc40303005)

[**4.** **DATABASE** 5](#_Toc40303006)

[**5.** **GUI** 9](#_Toc40303007)

[**REFERENCES** 9](#_Toc40303008)

1. **INTRODUCTION**
   1. **My Project**

* We have implemented a project on the topic School Management System
  1. **Tasks**
* We met and discussed a few things and decided to divide the project into the following tasks:

• First, we discussed together and came up with ideas about ERD

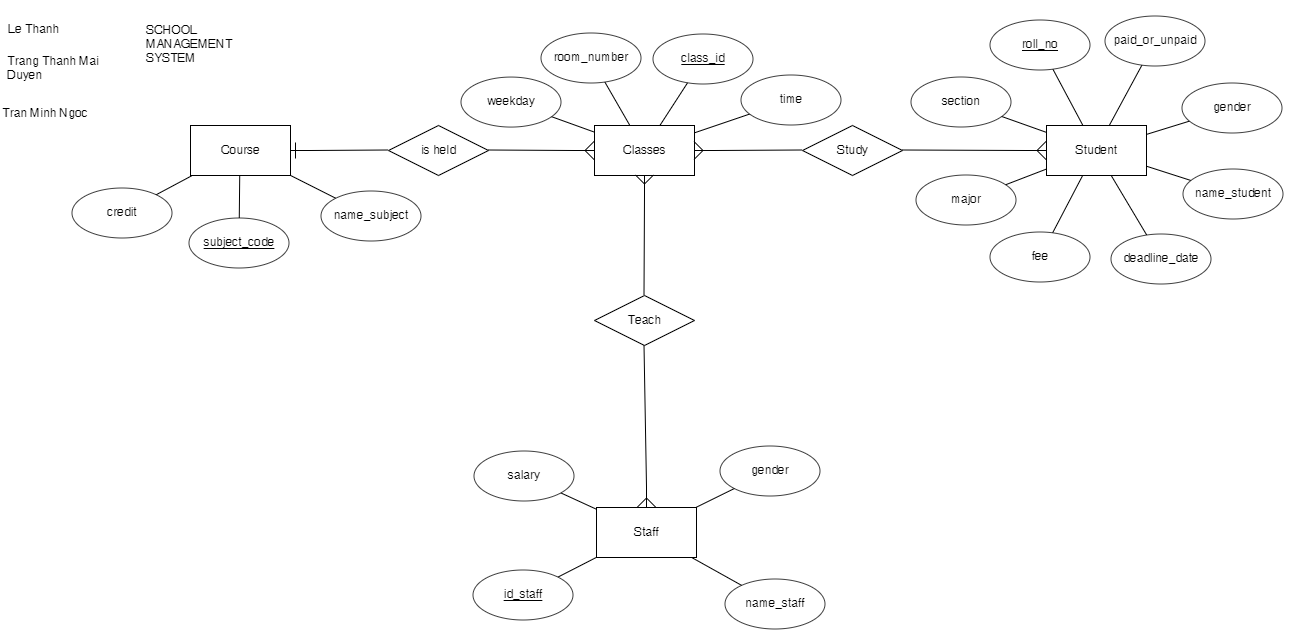
• Next, we base on ERD and draw Relational Model

• Then create the table and insert the information into the tables in the database

• Finally, we designed the interface to retrieve data from the database and display it in the most user-friendly way

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Duyên** | **Ngọc** | **Thanh** |
| Create table | X |  |  |
| Insert data | X |  | X |
| Draw ERD |  |  | X |
| Draw Relational Model |  |  | X |
| Write query retrieving data | X |  |  |
| Find the reference | X | X | X |
| Design GUI |  | X |  |
| Code the GUI function |  | X |  |
| Write proposal | X |  |  |
| Write report |  |  | X |
| Check grammar/edit report | X |  |  |
| Fix bug | X | X |  |

1. **ENTITY-RELATIONSHIP MODELING**

****

1. **Entity and attribute**

* **Course:**

• credit

• subject\_code(PK)

• name\_subject

* **Classes:**

• weekday

• room\_number

• time

* **Student:**

• roll\_no (PK)

• section

• paid\_or\_unpaid

• gender

• major

• fee

• deadline\_date

• name\_student

* **Staff:**

• id\_staff (PK)

• name\_staff

• gender

• salary

1. **Explanation**
   * Entity course is represented for the subject so it has attribute: subject\_code, name\_subject and credit
   * Entity Classes is represented for the classes of each course (1 course can have many classes in different time or room in the schedule, and it cannot be duplicated) so it has attribute: weekday, room\_number, time
   * Entity Student is represented for student(of course) and it have attribute: roll\_no(id student), section, paid\_or\_unpaid( the status to distinguish student that paid and not), gender(male/female), name\_student, deadline\_date(the deadline for student to pay the fee), major
   * Entity Staff is represented for teacher and it have attribute: id\_staff, name\_staff, gender, salary
   * In relationship, we have those relation between 2 entities:

• “Locate” between Course and Classes (1 to many)

• “Study” between Classes and Student (many to many)

• “Teach” between Classes and Staff (many to many)

* + Function of system:

• Get information of staff, student

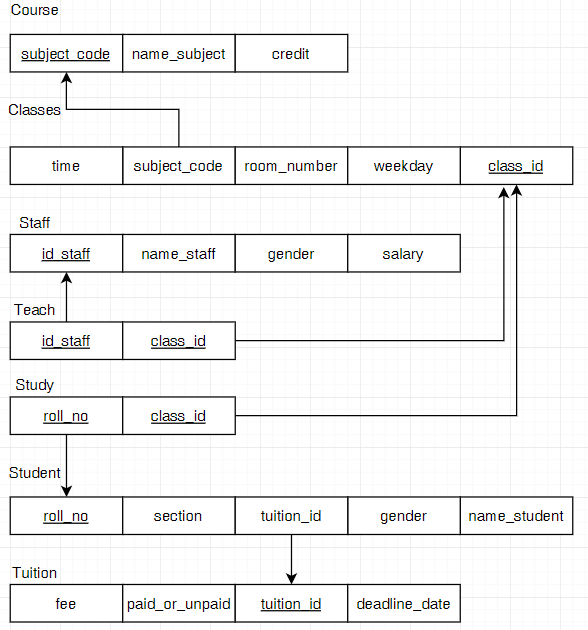
• Get the information about the course

• Get the information about the tuition of student and salary of staff

• Get the schedule of student

• Get the schedule of teacher

1. **RELATIONAL MODEL**

****

* + Our goal when creating more tables (Study, Teach, Tuition) but not to change the amount of information compared to ERD is to make the data more visible and easier to control

**•** Course(subject\_code(PK), name\_subject, credit)

**•** Classes(time, subject\_code(FK), room\_number, weekday, class\_id(PK)

**•** Staff(id\_staff(PK), name\_staff, gender, salary)

**•** Teach([id\_staff(FK),class\_id(FK)](PK))

**•** Study([roll\_no(FK),class\_id(FK)](PK))

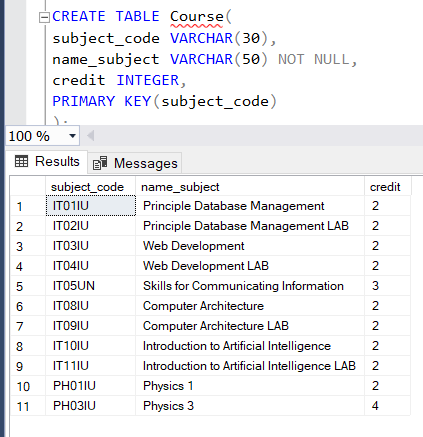
**•** Student(roll\_no(PK), section, tuition\_id, gender, name\_student)

**•** Tuition(fee, paid\_or\_unpaid, tuition\_id(PK))

1. **DATABASE**
2. **Table**

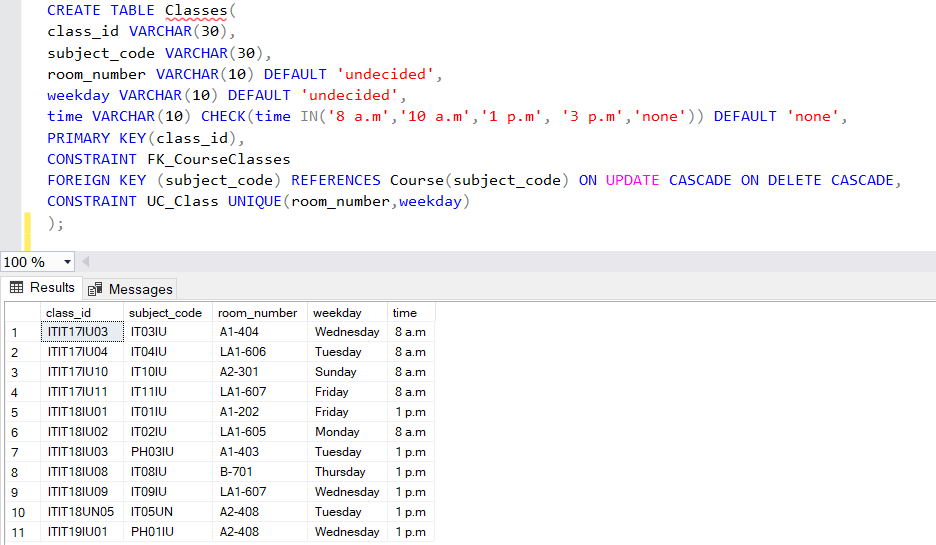
* Create new database name “School\_Management”
* Create tables and insert data:

**• Course**



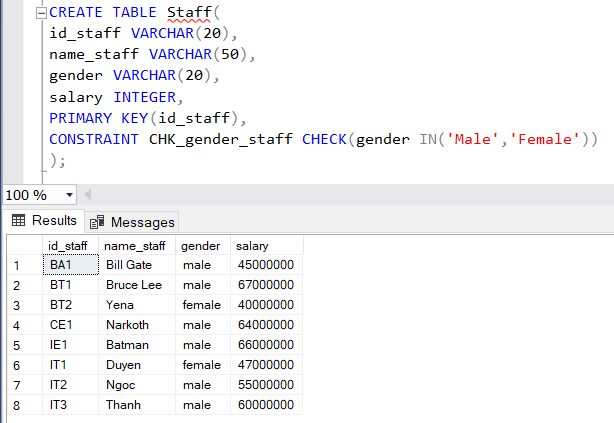
* We declared the attribute subject\_code and name\_subject represented for the id and the name of course in type varchar and set the maximum for it, credit is type integer
* Set the primary key for subject\_code (not be duplicated) and we can use it as foreign key in another table that has relationship “Course”

**• Classes**



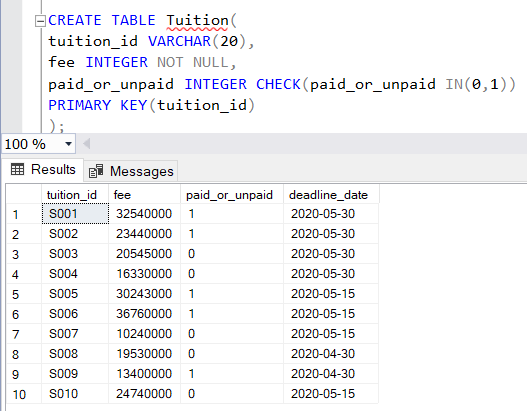
* In this table the subject code we use as the foreign key form table”Course”
* We declared room\_number and weekday in type varchar and set if we don’t enter the value in it, it will display “undecided”
* We declared attribute time and set it static in 4 time (8a.m, 10a.m,1p.m,3p.m) and if we don’t enter the value in it, it will display “none”
* Set attribute weekday and room\_numbet is unique which make the class can not be duplicated in the same day and room
* Set the primary key for attribute class\_if and anther table can use it as foreign key if it has relationship with table “Classes”

•  **Staff**



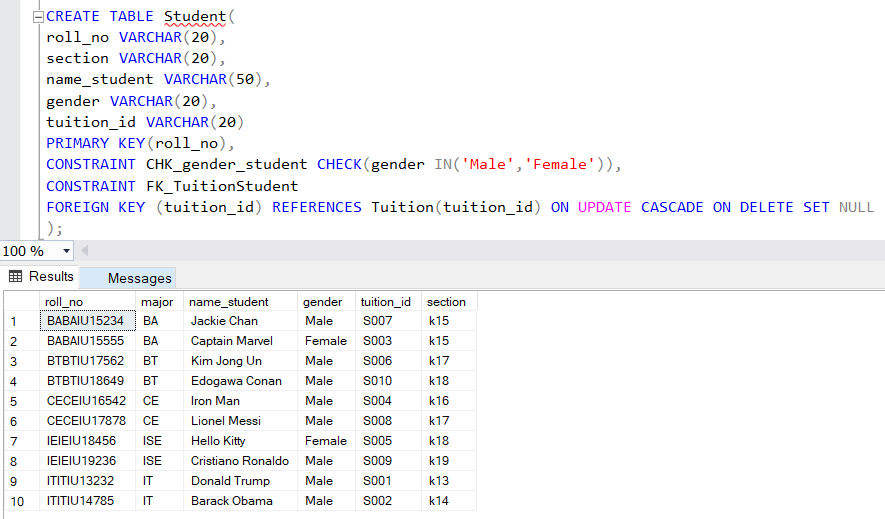
* We declared attribute id\_staff, name\_staff and gender in type varchar , and attribute salary in type integer
* Set the primary key for is\_staff and the another table will use it as foreign key if it has a relationship with “Staff”
* And we set that gender must be “Male” or “Female”

• **Tuition**



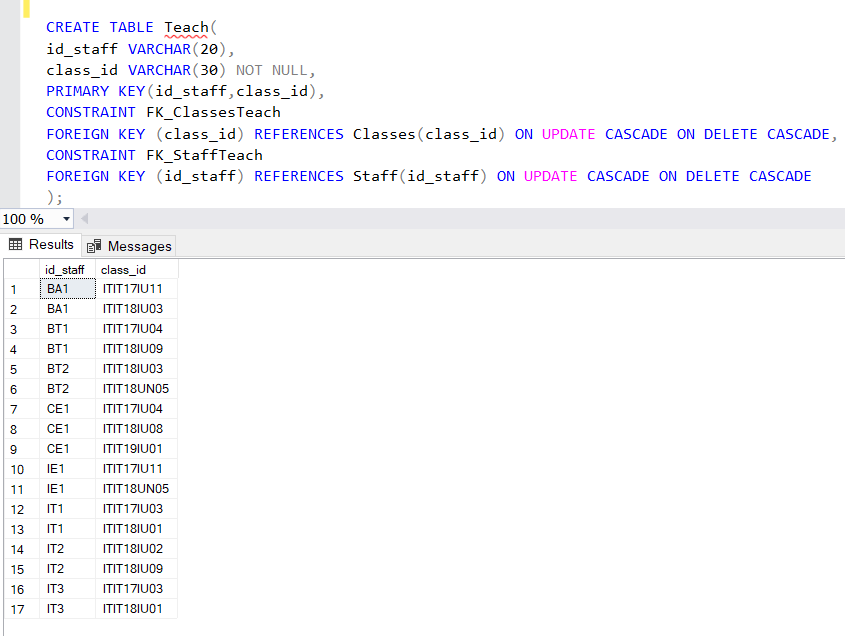
* We declared tuition\_id in type varchar, fee and paid\_or\_unpaid in integer and set if paid it will display “1”, unpaid “0”
* Set the primary key for tuition\_id and it can be used as foreign key in another table which has a relationship with “Tuition”

• **Student**



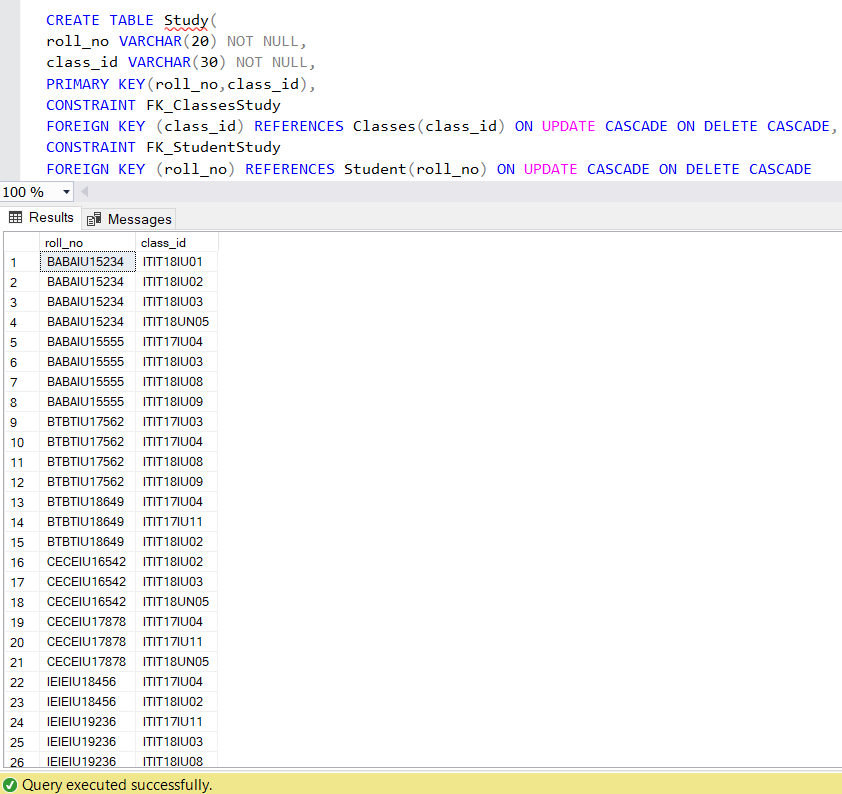
* We declared attribute roll\_no, section, name\_student, gender in type varchar
* Set the primary key for roll\_no and it can be used as foreign key in another table which has a relationship with “Student”
* And we set gender must input as “Male” or Female”
* Attribute tuition\_id we use as foreign key from table “Tuition”

• **Teach**



* Attribute class\_id we use as foreign key from table “Classes”
* Attribute id\_staff we use as foreign key from table “Staff”

**• Study**



* Attribute roll\_no we use as foreign key from table “Student”
* Attribute class\_id we use as foreign key from table “Classes”

1. **Explaination**

* In our system, a course has 3 attributes: subject\_code(PK), name\_subject, credit.
* A course can be held in many classes, which has 4 attributes class\_id(PK) ,room\_number, weekday, time and those cannot be duplicated by using PK for class\_id, and UNIQUE (room\_number,weekday,time).
* A student is studying in classes (a student can attend to many classes, a class can be attended by many students), and the student has 8 main attributes: roll\_no(PK), section,major ,name\_student, gender, fee, paid\_or\_unpaid, and deadline\_date. The deadline\_date, and paid\_or\_unpaid, which shows the payment status of student (use Boolean in JAVA, CHECK constraint in SQL), are for displaying the student data who had paid or unpaid the tuition fee up to date. This will support the function: the student cannot see his/her timetable, if he/she had not paid his/her tuition fee before the deadline date.
* Staff has 4 basic attributes: id\_staff,name\_staff,gender and salary, he/she teaches classes (a staff can teach in many classes, a class can be taught by many students)

1. **GUI**

**REFERENCES**

* SQL:
  + - <https://www.w3schools.com/sql/>
* GUI (java swing):
  + - <https://www.javatpoint.com/java-joptionpane>
    - <https://www.youtube.com/watch?v=QKsfHqu4Pps>
    - <https://www.youtube.com/watch?v=WIMojkwMTa0>