Object-Oriented Programming (SE271)
Final Project Report

1. Title of the Project

> Our team's title of the project is <Studyroom Management Program>.

2. Team Members

- ✓ Hyunwoo Kwak / khw5218@dgist.ac.kr / 201911013
- ✓ Shinwook Park / bnb2011@dgist.ac.kr / 201911063
- ✓ Jaehyeon Park / jhpahk1441@dgist.ac.kr / 201911067
- ✓ Howoo Jang / janghowoo@dgist.ac.kr / 201911144

3. Motivation for our project

> There are several studyrooms in DGIST. There are many Studyrooms such as academic information center(E8), dormitory, E7 and graduate school building. However, there are few managed studyrooms. Multiple problems exist due to unmanaged systems. If one person uses multiple seats or leaves their luggage after use, there is a problem that others cannot use the seats. In addition, if the studyroom is large, it is impossible to check the seats in the studyroom, so there is an inconvenience that you have to go to the seat you want and check it yourself before choosing a seat. In order to solve various problems arising from these unmanaged studyrooms, a studyroom management program was created to reserve and cancel sudyrooms' seats, and to control the overall studyrooms situation through the manager function.

4. Summarize what we planned to do and what have done and haven't done

> The main functions of the initially planned studyroom management program are as follows.

- ✓ Implementation of seat reservation and cancellation
- ✓ The graphic section that shows the overall seating layout of the study room
- ✓ Implement administrator mode to control the overall studyroom's situation

    (ex) Reservation status initialization function, etc.)

- ✓ Get membership information through membership and manage all membership information

> Most of the above four contents were implemented overall. Studyroom users were allowed to reserve and cancel their seats through the GUI. The administrator mode also enables various functions such as initializing the overall seating status of the studyroom, adding the studyroom itself, or changing the seating arrangement in the studyroom.

5. Describe the implement design of our project

> The overall implementation was implemented through a model, view, and control (MVC) design. First of all, the Model section was organized through the following classes. There are Studyroom class, Seat class, Student class, Admin class. The StudyRoom class has a Set container that contains the overall data of the study room and stores the entire capacity, current occupancy, and Seat objects. Seat Class is a class that stores information about individual seats, which provides information such as the study room to which they belong, the unique number of seats, reservation status, reservation status, pre-accommodation time, and free time. Student class is a regular student member class that contains class numbers, passwords that can be used

to cancel reservations, and information about reservations. The Admin class is the administrator class and inherited the Student class.

The control section was constructed through the following classes. There are ControlByStudent class, ControlByAdmin(inherited ControlByStudent class.). The ControlByStudent class is a class that controls the general user and has functions such as seat reservation, extension of reservation, cancellation of reservation, declaration of absence, and release of absence. ControlByAdmin is responsible for the administrator's control functions and implements functions that can add study rooms, change seating arrangements in study rooms, and force cancellation of reservations.

Finally, the View section consists of a main class called UI class. User interface was implemented through UI class.

a. Provide a class diagram which includes classes with their major member functions and other functions
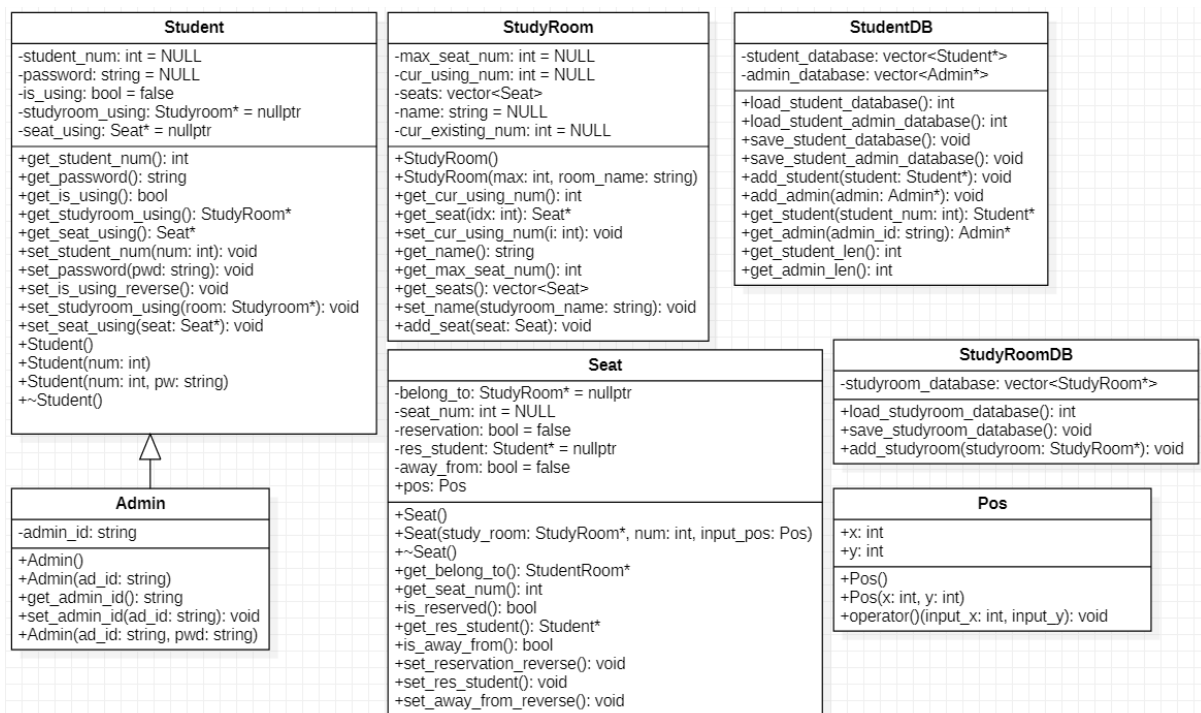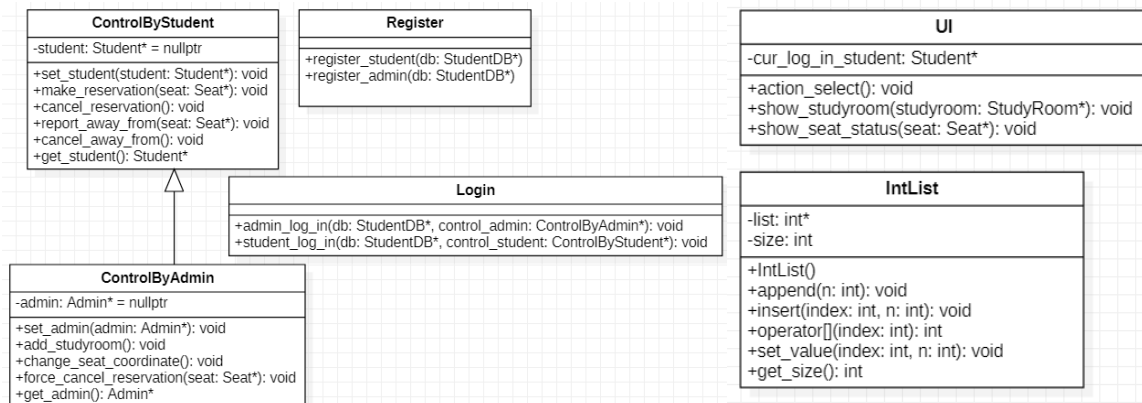

Figure 1. Model Section

Figure 2. Control Section

Figure 3. View Section

b. Provide details about what you are each in charge of.

- ✓ Hyunwoo Kwak(25%): Responsible for final report preparation, member information management, and model section

- ✓ Shinwook Park(25%): Responsible for the Final Report Preparation, Member Information Management Functions and Control Section

- ✓ Jaehyeon Park(25%): Responsible for ppt and presentation, model section and overall code merging

- ✓ Howoo Jang(25%): Responsible for ppt and presentation, view section

6. Explain how to compile your code and how to test

First, implement the attributes and method functions of the classes as originally planned. We then ignore the relationship between each function and class and roughly implement the skeleton. After the overall task is completed, consider all of the user's choices within the main function and construct the entire program using the method functions and attributes in the class that were previously roughly implemented.

When we implemented, the behavior of students and managers was different and managers were able to act after the student's options were prioritized, so we implemented them by considering the options from the student's perspective. We then implemented the managers' forced cancellation of reservations and seat verification functions using previously implemented method functions.

Finally, we debugged each option by setting a breakpoint until it worked as desired and implementing the action for the next option if the options worked as desired. By repeating the previous steps, most method functions and attributes in the class were organically connected through the main function and finally completed the program.

7. Results section

a. Describe your experiment setup and metrics of success

*Metrics of success

- ✓ Is the program working well?
- ✓ Do users, administrators work well login,logout?
- ✓ Are the reservation, cancellation, and seat information checking going well?
- ✓ Does the program prevent duplicate bookings?
- ✓ Do all the member functions work well?
- ✓ Are administrators functions work well?

*Experiment setup

-Situation

- ✓ Is the program working well?(Yes)
- ✓ Do all the member functions work well?(Yes)
1. Register to user and administrator as below.

- ✓ Do users, administrators work well login,logout? (Yes)

2. User(201911144, Janghowoo, E7, seat number:8)

✓ User chooses E7 studyroom.


```
User: 201911144              1. E7      2. 뒤로 가기

1. 독서실 선택    2. 로그아웃     입력: 1
입력: 1
```

✓ User reserves seat given 8 number


```
현재 선택한 독서실: E7            현재 선택한 독서실: E7
  01    02    03    04          01    02    03    04

  05    06    07    08          05    06    07    08
                                                  예약됨

        09    10    11
                                      09    10    11
1. 예약   2. 예약 취소   3. 좌석 정보 확인   4. 뒤로 가기
입력: 1
예약할 좌석의 번호를 입력하십시오: 8
```

- Is the reservation going well? (Yes)

✓ User cancels seat.


```
현재 선택한 독서실: E7            현재 선택한 독서실: E7
  01    02    03    04

  05    06    07    08            01    02    03    04
              예약됨

        09    10    11            05    06    07    08

1. 예약   2. 예약 취소   3. 좌석 정보 확인   4. 뒤로 가기
입력: 2
비밀번호를 입력해 주십시오: 1234
```

- Is cancellation going well?(Yes)

✓ User(201911067, ParkJaeHyeon,E7,seat number:8)


```
현재 선택한 독서실: E7
  01    02    03    04

  05    06    07    08
              예약됨


        09    10    11

1. 예약   2. 예약 취소   3. 좌석 정보 확인   4. 뒤로 가기
입력: 1
예약할 좌석의 번호를 입력하십시오: 8
이미 예약된 좌석입니다.
```
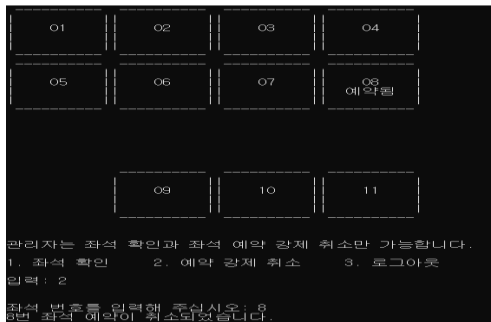
- Does the program prevent duplicate bookings? (Yes)

✓ User checks seat information

```
좌석 번호 : 8
예약 상태 : 예약됨
예약자(Student Number) : 201911144
```

- Is seat information checking going well? (Yes)

3. Administrator (201911144, Janghowoo, cancel seat number:8)

✓ Administraor cancels seat number 8.


```
01       02       03       04

05       06       07       08
                           예약됨



         09       10       11

관리자는 좌석 확인과 좌석 예약 강제 취소만 가능합니다.
1. 좌석 확인      2. 예약 강제 취소      3. 로그아웃
입력 : 2

좌석 번호를 입력해 주십시오 : 8
8번 좌석 예약이 취소되었습니다.
```

- Are administrators functions work well? (Yes)

b. Present results with discussion

You can see that user and administrator membership work well. Based on this, you can see that the function of users choosing a study room, finding and booking a seat is working well. You can also check the seat reservation information and confirm that the cancellation of the reservation works well. From the Administrator function, you can see that the ability to cancel a user's seat for reading room management works well.

One regrettable thing was that it failed to implement certain functions in optimizing the existing study room management program with MVC design.

Finally, we think that we can increase the efficiency of the place by producing a program based on this program that allows users to make reservations for places where user can study at school as well as E7.

8. Conclusion of your work and what you learned

✓ Conclusion: Through this program, students can choose and use after checking the study room seats in the dormitory, E7. Furthermore, it was possible to improve existing inconveniences such as preventing indiscriminate use of study rooms and preventing monopolization of study rooms by one person.

✓ What we learn: It was discovered that the object can be created through the Class and that each class can be flexibly connected through the inheritance relationship. Furthermore, we have learned that class design is also important because it is easy to design using the Hi-architectural Design format. Finally, through this project, we learned that everything can be represented as objects.