



전공 기초 프로젝트

Main themes

- 예외 처리
- 파일 처리
- 상황 흐름의 복잡성

1차 회의

- A06팀으로 배정된 우리는 위의 Main theme 들을 만족하기 위해 Unity-2D-FlatFormer 게임을 제작하기로 했다.

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/53a32b76-b8a8-4e62-ab52-55ac0189eec6/%ED%9A%8C%EC%9D%981.txt>

- 하지만 Unity로 위의 Main theme들을 만족하기 위해선 복잡한 과정의 예외 처리, 혹은 두 사용자 간의 상호 작용을 교수님이 추가하라고 하셔서 이 툴은 다음 번에 사용해보려고 마음 먹고...
- 급하게 주제 수정을 했다.
- 그래도 교수님이 정말 학생들에게 친절하시고 열정 있으시다.

그래서, 크게 두 가지 중 하나를 제안할까 합니다:

A. 아예 완전히 다른 (게임이 아닌, 기왕이면 Unity도 안 쓰는) 주제를 생각해보기

B. (Unity 기반이든 뭐든) '게임'이긴 하되, 규칙이 충분히 복잡한 다른 종류의 게임을 생각해보기

1번째 프로젝트 과목에서 바로 주제를 기각먹는 20학번 박성근...

2차 회의

- 첫 번째 주제 선정에서 기각을 먹고... 충격을 빠진 우리 A06팀
- 각자 주제 선정을 통해 아이디어를 내고, 회의를 통해 정한 뒤 교수님께 정했다.
- 내가 우리 팀에서 스터디룸을 빌려 회의를 한 기억이 떠올라 “학교 스터디룸 예약 프로그램” 주제를 제안했고 조원들 모두 동의해 교수님께 컨펌을 받기로 했다.

네, 'K cube 스터디룸 예약'은 상당히 좋은 주제입니다. 한정된 자원을 공유/선점하며 경쟁하는 시스템이어서, 각 사용자의 입력(조작/선택)에 따라 다른 사용자의 선택지가 영향을 받아 달라지기 때문에, 이런 규칙을 자세히 정할 여지가 충분하기 때문입니다.

세부 내용도 전반적으로 괜찮습니다: 일단 (당연한 사항이지만) 한 지점(?)에 대한 예약 시간이 겹치는 문제를 처리하는 게 핵심이니 이 요소는 꼭 예정대로 넣기 바랍니다. 그리고 시간 제한과 인원 제한도 훈련에 좋은 요소니 꼭 넣되, 기왕이면 인원은 각 지점(방?)마다 다르게 정해져있도록 만들기를 추천합니다. 그리고 그림에 안 나와있어서 언급합니다만, 자신의 예약 내역을 조회하고 취소하는 기능도 필요합니다.

- 다행히 통과를 했다..!

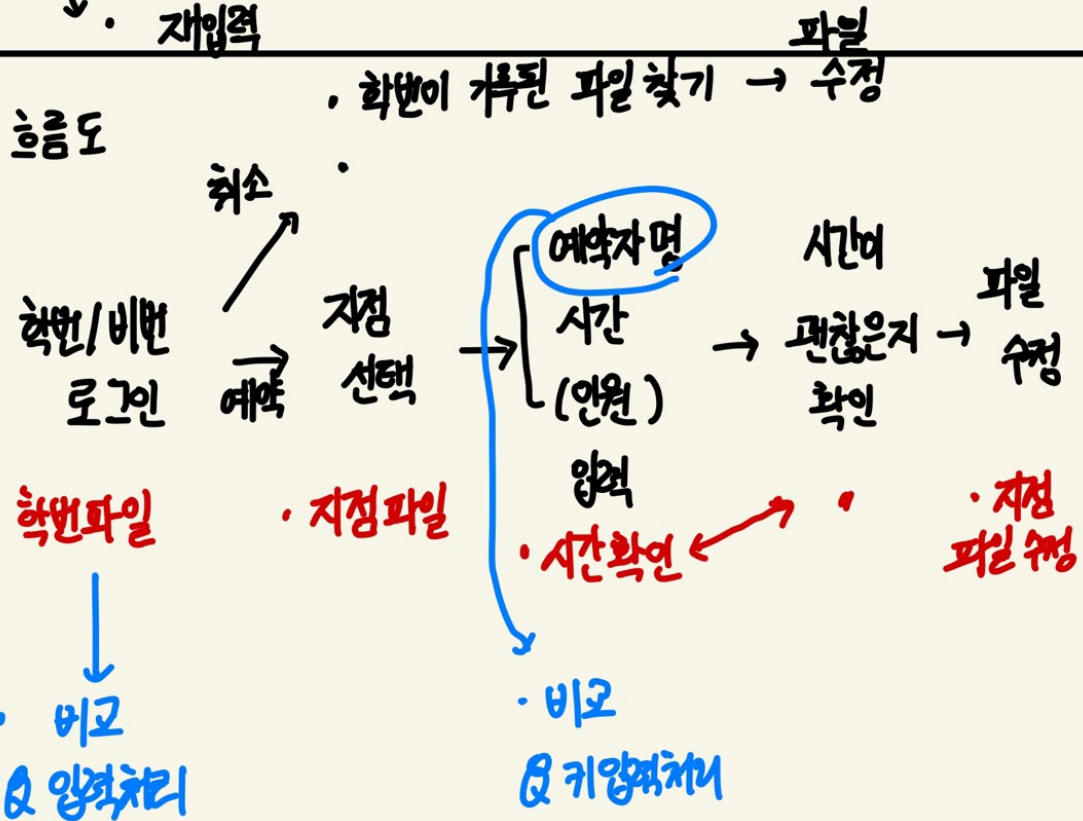
3차 회의

- 스터디룸 예약 프로그램...이지만 사실 서버와 통신 X
 - 방학 때 꼭 프론트엔드, 백엔드 공부하자..
- console 창에서 txt 파일을 읽고 쓰는 프로그램 수준으로 구현했다.
- 내가 기획한 주제인 만큼 내가 잘 마무리 짓고 싶었다.
- 그래서 클래스 선언, 클래스 간의 상호관계 등을 정리해 팀 회의에서 조원들에게 개요에 대한 발표를 진행 했다.

Kcube
스튜디오 예약 프로그램.

가존기 있는 학번/비번
파일에서 찾기

9자리
예외처리
· 기존 Kcube 마다 파일이 하나씩 존재
· 학번 / 비밀번호로 로그인 & 지점선택
· 예약 시 Kcube file(x)에 저장. 예외처리
요소
· 예약자명
· 시간
· 인원
· 시간 정찰 시 오류 메세지 출력.
· 재입력



생각 정리 1

총 클래스 구현

20201129 박성근

1. Main

2. MemberInfo, Student : 회원가입, 로그인 기능 , 학생 정보

3. Reservation, Room : 예약 시스템 구현 , 방 정보

① main 실행 시

1) → MemberInfo 객체 생성 → 아래 생성자는 학번 파일을 바로 불러와서 MemberInfo 시 멤버 변수로 가지는 Student[] 객체 배열에 저장.

2) 1. 2. 3. 4 선택

if(1) => MemberInfo에 구현된 메서드 (회원가입)에 접근

else if(2) => " (로그인) "

2 → true → '예약 클래스 Reservation' 접근 ①

false → 다시 입력

① → 생성자 '스터디룸 현황' 파일 바로 불러오기. Room 배열 ②

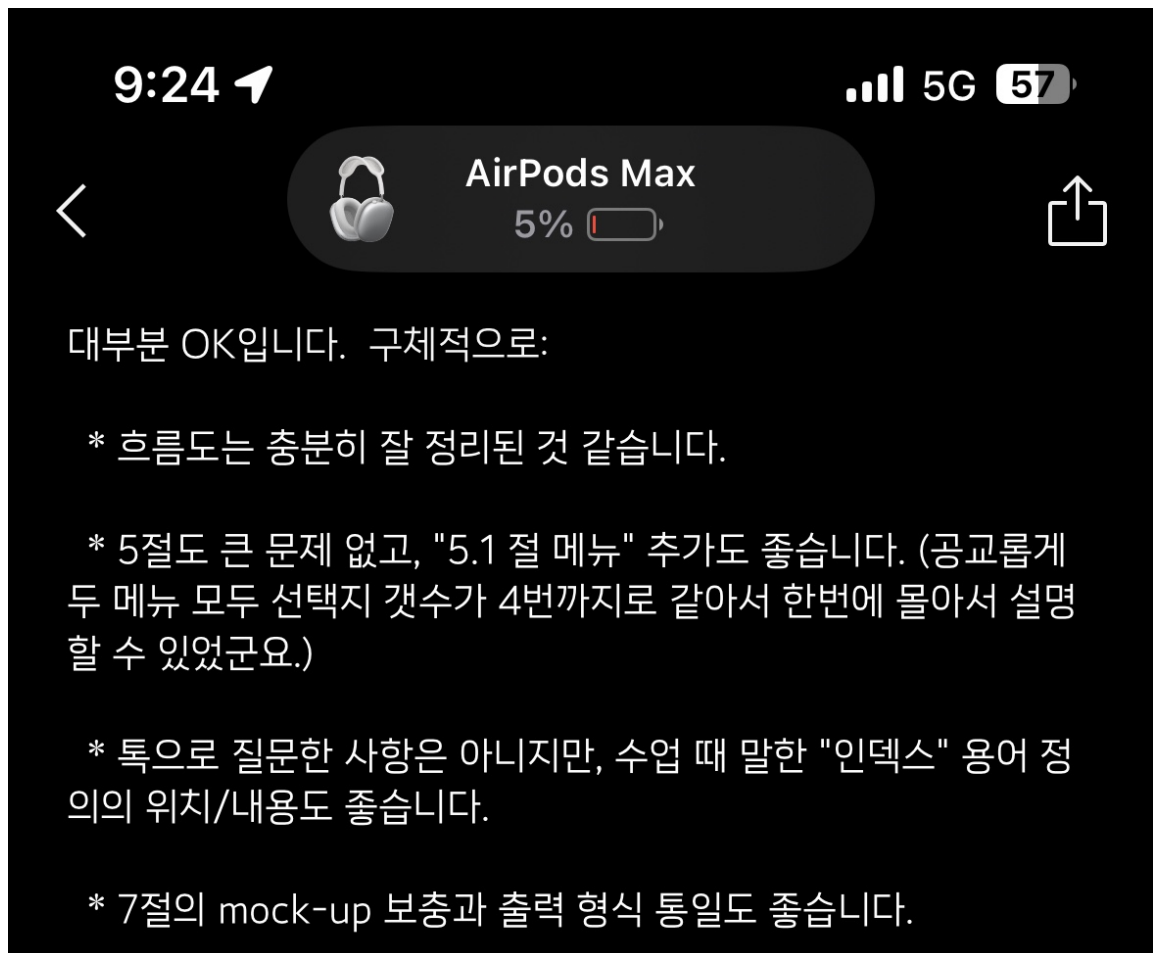
로그아웃 : 그냥 break;
현황판 : 배열 Room 출력
② 예약 : 입력 후 " 비교
확인 : " 비교 후 정보 찾기.
취소 : " "

실제 조원들에게 소개한 자료

4차 회의 → 기획서 1차 마무리 전 회의까지

- 주제도 통과하고 내용 기획에서도 틀을 짰 상태에서
- 기획서를 작성하기 시작했다.

- 많은 예시들을 참고하며 개발 기획서에 대한 입문을 시작하게 된 것 같다.
- 기본적인 용어 정의, 예외 처리에 대해 생각하는 것이 어려웠다.
- 당연히 하라는 대로 하겠지? 라는 생각을 버리라고 교수님께서 강조를 하셨다.
- 코드 구현도 미리 진행을 했다. (안그러면 시험 기간 때 해야된다..)
 - 전체적인 맥락으로 회원가입/로그인 클래스 , 예약 클래스 2가지로 분류하여
 - 3:3으로 나누어 코드를 구성했다.
- 아직 코드 구현까지는 기간이 남아 전체적인 틀만 마무리하고 기획서 작성을 완료했다.
- 조원들에게 전체적인 틀을 설명한 뒤 각자 맡은 역할을 부여하니 진행 속도가 다른 팀에 비해 월등히 빨랐다.
- 수 많은 컨펌(4회 정도)을 교수님께 받고 나서 완성된 1차 기획서(최종판)!



에어팟 맥스 배터리리는 무시...하자 ㅎㅎ

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/653ea8ee-c30b-4d23-9a04-e0098bbae28a/%EA%B8%B0%ED%9A%8D%EC%84%9C06%ED%8C%80.docx>

- 아직까지 수정 사항이 조금 남아있다고 하지만... 1차 기획서 제출 이후 → 구현 전까지 미비된 상황을 찾아보고자 한다!

1차 기획서 제출 후

- 중간고사 직후까지 과제가 주어졌다.
- “검사 보고서”, “구현물” 제출이었다.
- 기존, “StudentInfo 클래스”, “ReservationInfo 클래스” 을 3:3으로 나누어 구현을 했기 때문에, 검사 보고서를 작성할 땐 상대의 클래스에 대한 검증을 하고자 한다.

그 검사보고서 같은 경우는 오늘 강의 바탕으로 봤을 때,

1. 성근, 현준, 창근
-> 예약 시스템에서 꾸러기 짓하기

2. 경범, 우현, 정우
-> 회원가입, 로그인 메서드에서 꾸러기 짓하기

: 단, 기획서 바탕으로, 강의 자료(교수님이 수정해서 새로운 파일 올리심)
참고해서 입력하기

+ 각자 어떤 걸 입력해서 이런 결과가 나온다...

정도는 메모장이나, 워드 같은 곳에 정리해서 올리죠. 같은 거 생각해서 입력하면 개인 시간이 아까우니.

오후 9:12

팀원들이 모두 git을 쓴다면... 좋겠다... 하면서 나도 공부 열심히 할 수 있을 거 같은데... 더더욱 중요성 협업툴의 중요성을 느끼는 중

- 코드를 작성한 거에 대한 피드백, 검사를 진행한다면 “이런 오류는 코드를 작성하면서 이미 처리했겠지..”라는 오산이 생길 수 있을 거 같아 위와 같은 방법을 대입하고자 한다.
- 우선 각 팀에서 작성한 코드를 합쳐 → “K-CUBE 예약 시스템”의 이름을 가진 전체 파일을 생성하고자 한다.

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/d6a8a11f-736b-440e-a9cc-7d0a8c59c101/StudyRoom\(1%EC%B0%A8\).zip](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/d6a8a11f-736b-440e-a9cc-7d0a8c59c101/StudyRoom(1%EC%B0%A8).zip)

- 위 파일에서 검사 보고서를 작성하며 코드를 수정하고자 한다.

검사보고서 + 코드 수정 과정

- 위에서 말한 과정대로 검사보고서를 작성하는 과정에서 기획서와 다르게 코드가 구현된 부분을 수정했다.
- 기획서에서 정한 문법에 따라 예외 상황을 찾고, 검사를 통해 출력 or 입력 반응이 옳은 지 파악 후 수정을 했다.
- Git을 쓰기로 했다. 코드를 수정할 때마다 올리는 것도, 문서에서 수정하는 것을 올리는 것도 번거러워 팀원 1명이 레퍼지토리를 생성해 모두 연결하게 되었다.
- 기획서에서 파일 수정을 사용자가 할 수 있다고 명시한 바램에, 파일 형식에서 오류가 있을 때 불러 올 때 오류를 출력하고 종료하는 코드를 추가했다.
- 메인메뉴, 예약메뉴, 검사보고서로 나누어서 작성해, 합친 후 최종파일을 제출하고자 한다.

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/76fab79e-c3e9-418a-bc46-ee5665e82278/%EA%B2%80%EC%82%AC_%ED%85%8C%EC%8A%A4%ED%8A%B8\(%EB%A9%94%EC%9D%B8%EB%A9%94%EB%89%B4%ED%95%AD%EB%AA%A9\)_\(1\).docx](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/76fab79e-c3e9-418a-bc46-ee5665e82278/%EA%B2%80%EC%82%AC_%ED%85%8C%EC%8A%A4%ED%8A%B8(%EB%A9%94%EC%9D%B8%EB%A9%94%EB%89%B4%ED%95%AD%EB%AA%A9)_(1).docx)

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/1b740783-265d-46f4-8cb7-67d7b3ef69dc/%EC%A0%84%EA%B8%B0%ED%94%84_%EA%B2%80%EC%82%AC%EB%B3%B4%EA%B3%A0%EC%84%9C\(Team2\).docx](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/1b740783-265d-46f4-8cb7-67d7b3ef69dc/%EC%A0%84%EA%B8%B0%ED%94%84_%EA%B2%80%EC%82%AC%EB%B3%B4%EA%B3%A0%EC%84%9C(Team2).docx)

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/5b453360-0dfa-48f0-9eaf-d6860833cd39/%EA%B2%80%EC%82%AC%EB%B3%B4%EA%B3%A0%EC%84%9C\(%ED%8C%8C%EC%9D%BC\).docx](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/5b453360-0dfa-48f0-9eaf-d6860833cd39/%EA%B2%80%EC%82%AC%EB%B3%B4%EA%B3%A0%EC%84%9C(%ED%8C%8C%EC%9D%BC).docx)

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/52b44de9-e2aa-446d-9b37-31885b04fc4d/MemberInfo.java>

중간고사 후 중간발표

- git을 쓸거면 최신화가 확실히 되었는 지, 제출 전 마지막엔 항상 확인하는 작업이 필요 하겠다고 느꼈다.
- 더블체크의 중요성...을 절실히 느껴버렸다.
- 뭐든지 담당하는 인원이 1명이 아닌, 2~3명이 되어야 한다는 것을 느끼고, 발표 대본은 각자 쓰고 내가 병합 후 수정 과정을 통해 다시 인원들에게 배포했다.
- 각자 구현한 결과, 검사 보고서 작성 부분을 재검토하라고 했고, 이를 통해 검사보고서, 기획서에 수정사항이 조금 남아있다는 것을 찾았다.
 - ⇒ 이는 발표과정에서 명시했다.
- 발표 초안

전기프 발표 초안

- 발표자료

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/02e455f8-cdf3-43ab-a3fc-4707b084968f/A06%ED%8C%80_%EC%A4%91%EA%B0%84_%EB%B0%9C%ED%91%9C_%EC%9E%90%EB%A3%8C_\(1\).pdf](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/02e455f8-cdf3-43ab-a3fc-4707b084968f/A06%ED%8C%80_%EC%A4%91%EA%B0%84_%EB%B0%9C%ED%91%9C_%EC%9E%90%EB%A3%8C_(1).pdf)

- 발표대본

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/332c89cc-0330-4829-b8fa-074b558f0f59/%EB%B0%9C%ED%91%9C%EB%8C%80%EB%B3%B8\(%EC%B5%9C%EC%A2%852\).pdf](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/332c89cc-0330-4829-b8fa-074b558f0f59/%EB%B0%9C%ED%91%9C%EB%8C%80%EB%B3%B8(%EC%B5%9C%EC%A2%852).pdf)

확장안 및 2차 기획서

- 2차 확장안 토의 결과
 - 관리자 추가
 - 관리자 로그인 방법
 1. StudentInfo.txt 파일 형식, 경로 오류 시

2. StudentInfo.txt 파일 오류가 없어 로그인 후 ReservationInfo.txt에서 오류가 생기더라도 파일 형식 출력 가능

- 일반 사용자는 로그인 성공하더라도, ReservationInfo.txt 오류 시 프로그램 종료

■ My Idea

05.02 - 2020 11293 박성근

S: StudentInfo
R: ReservationInfo

관리자

1. 일반 사용자가 파일 형식 오류 직면할때 → (문제어려움?)

- 기존: 에러메시지 1 출력 후 exit(0)으로 종료
- 확장안: 에러메시지 "관리자에게 문의하세요" 출력 후 메뉴창으로 이동

2. StudentInfo.txt, **ManagerInfo.txt** 추가
(기존 학생) (관리자 정보파일) ⇒ 프로그램 밖에서 수정불가.

3. 메뉴 수정

(기존)

1. .
2. .
3. .
4. .

→ OR

- Idea1: 기존 로그인 창에서 관리자 id, pwd를 입력받을지
- Idea2: 관리자 메뉴를 따로 만들지.

4. 관리자 기능

- 예약 불가 구역 or 시간 설정 가능
ex) "교시 K-Cube 사용을 금합니다" → 기존 예약인은 취소되었고 console 출력.
↓
Reservation Info 파일 수정.
- 학생에게 경고 기능 부여 가능
⇒ 2회 이상 시 예약 불가 → StudentInfo.txt에 수정 표시
- 잘못된 파일 형식 출력가능.

■ TeamMates Idea

1. 스터디룸 호실을 추가, 수정, 삭제 구현

2. 특정 사용자 정보 추가, 수정, 삭제 구현(자퇴, 편입 등의 사유)
3. 관리자가 공지를 입력하면 → 다음 프로그램 실행 시 맨 처음 화면에서 공지 출력
4. 스터디룸 건물 선택
5. 로그인 한 사람이 자기가 언제 몇 명 예약할 지를 입력

■ 종합 IDEA 1차(수정 05.04)

• 관리자

- 스터디룸 호실을 추가, 수정, 삭제 구현
- 전체 예약 내역 조회
 - 현황판 출력과 동일한 기능 → 대신 o,x가 아닌 학번이 나오도록 출력하기
- 특정 시간, 호실 스터디룸 예약 불가 명시
- 특정 사용자 삭제 가능 (자퇴, 편입 등의 이유)
- 관리자가 공지를 입력하면 → 다음 프로그램 실행 시 맨 처음 화면에 공지 출력
 - ex : “금일 6교시는 건물 점검으로 인해 사용이 불가합니다.” 출력

• 기능 수정 및 개선

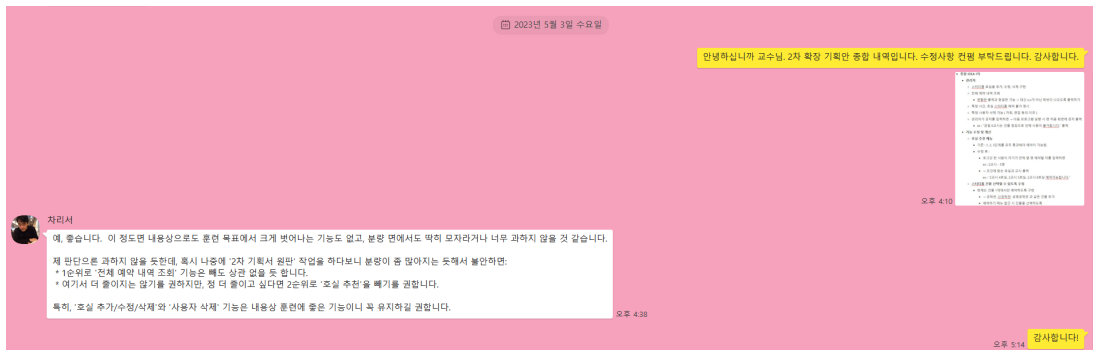
◦ 호실 추천 메뉴

- 기존 : 1, 2, 3단계를 모두 통과해야 예약이 가능함.
- 수정 후 :
 - 로그인 한 사람이 자기가 언제 몇 명 예약할 지를 입력하면
ex : 2교시 - 5명
 - → 조건에 맞는 호실과 교시 출력
ex : “2교시 4호실, 2교시 5호실, 2교시 8호실 예약가능합니다.”

◦ 스터디룸 건물 선택할 수 있도록 수정

- 현재는 건물 1개에서만 예약하도록 구현
 - → 공학관, 신공학관, 생명공학관 과 같은 건물 추가
 - 예약하기 메뉴 접근 시 건물을 선택하도록

05.03 컨펌 결과



- 통과했다.

05.04 역할 분담

- 기획서 수정 사안
 - 관리자 기능 추가
 - 인원 : 박성근, 장현준, 이창근
 - 할 일 목록
 - 관리자 용어 정리 ← 정의, 역할, 기능
 - 관리자 로그인 시스템 기획
 - 기획서 목차 수정 (about 관리자)
 - MockUp 작성
 - 스터디룸 건물 선택
 - 인원 : 최우현, 박경범, 최정우
 - 할 일 목록
 - 데이터 파일 목차, 용어 {수정 && 추가}
 - MockUp & 기획서 {수정 && 추가}
- 기획서 2차 확장안 목표 기간
 - 5월 8일 수업시간까지 진행 → 수업 시간 내 오류 점검

05.19 2차 기획서 확장안 제출

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/a67bc399-de23-49bb-a41f-a49e3700a732/A06%ED%8C%80_2%EC%B0%A8_%EA%B8%B0%ED%9A%8D%EC%84%9C_%EC%9B%90%ED%8C%90.pdf

설계 문서

- 기획대로 동작하게 만들려면 어떻게 만들어야 하는가? (구현 작전)
 - 어떤 자료형, 자료 구조, 클래스, 클래스 멤버, 인스턴스, 인스턴스 멤버, 상속 관계들이 필요한가?
 - 어떤 서브 루틴, 프로시저, 함수들이 필요한가
 - 전역 변수, 지역 변수, 지역 함수 구분
 - 불변 자료, 가변 자료 구분
 - 객체를 덮어쓸 지, 새롭게 찍어낼 지 구분
 - 분기 지점, 조건 명시
 - 재귀 호출 or 루프 호출 명시
 - 루프 도는 기준, 탈출 조건
 - 루프 탈출 후 남아있는 흔적은 어떻게 관리
 - 재귀 호출은 꼬리 재귀인가, 아닌가
 - 분리된 모듈들 중에서 무엇이 블랙 박스이고, 아닌 지 구분
- 초안

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/eda02422-f845-4a3b-9cc3-91349a8598e4/%EC%84%A4%EA%B3%84_%EB%AC%B8%EC%84%9C_%EA%B5%AC%EC%84%B1%EC%95%88.docx

05.31 설계 문서 원판 파일 제출

- 교수님께 컨펌 받은 후, 수정 이후 설계 문서 제출을 완료했다.

- 이 과정에서 수정된 기획서 내역 또한 마무리 작업 후 제출했다.
- 원판 파일 작성 후

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/c6fdc4a0-3f8e-4920-9397-2e3adf61e413/%EC%84%A4%EA%B3%84_%EB%A%B8%EC%84%9C_%EC%9B%90%ED%8C%90.pdf

- 2차 기획서 확장안 수정 후

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e487825b-0476-48f4-88cc-64f5ddd479e1/A06%ED%8C%80_2%EC%B0%A8_%EA%B8%B0%ED%9A%8D%EC%84%9C\(%EC%A0%9C%EC%B6%9C\).dOCX](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e487825b-0476-48f4-88cc-64f5ddd479e1/A06%ED%8C%80_2%EC%B0%A8_%EA%B8%B0%ED%9A%8D%EC%84%9C(%EC%A0%9C%EC%B6%9C).dOCX)

06.01

- 제출 목록 : 2차 검사보고서 , 2차 구현물
- 2차 검사보고서 기획안
 - 1차 검사보고서 형식과 동일하게 2차 확장된 코드 내에서도 진행할 예정
 - 단, 기획서 명시에서 다르게 된 부분을 소거한 뒤, 새로운 조건에 대해서는 추가한 뒤 검사를 진행한다.
 - Part 분배를 통해 검사보고서를 완성한다.
- 2차 구현물
 - 설계 문서에서 대략적인 코드에 대한 작성을 완료했다.
 - 검사보고서를 통해 발견된 오류 , 기획서와의 대조를 통한 차이점 인식 시, 공지사항에 적은 뒤 코드를 수정하도록 한다.

06.02

- 팀 미팅 결과
 - 스터디룸 수정 : 2명 - 최우현, 최정우

- 사용자 수정, 공지 사항 : 1명 : 이창근
 - 수정된 예약 part : 박경범
 - 코드 : 박경범, 장현준
-

06.14

- 2차 검사보고서

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/5d8b2156-f6d2-4d71-85d2-450c866f6bfd/A06%ED%8C%80_2%EC%B0%A8_%EA%B8%B0%ED%9A%8D%EC%84%9C_%EC%B5%9C%EC%A2%85_%EC%88%98%EC%A0%95_1%ED%8C%90_\(1\).pdf](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/5d8b2156-f6d2-4d71-85d2-450c866f6bfd/A06%ED%8C%80_2%EC%B0%A8_%EA%B8%B0%ED%9A%8D%EC%84%9C_%EC%B5%9C%EC%A2%85_%EC%88%98%EC%A0%95_1%ED%8C%90_(1).pdf)

- 설계문서 수정판

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/05d4da67-79c2-492b-9647-745845fd4541/A06%ED%8C%80_%EC%84%A4%EA%B3%84_%EB%AC%B8%EC%84%9C_%EC%88%98%EC%A0%95_1%ED%8C%90_\(1\).pdf](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/05d4da67-79c2-492b-9647-745845fd4541/A06%ED%8C%80_%EC%84%A4%EA%B3%84_%EB%AC%B8%EC%84%9C_%EC%88%98%EC%A0%95_1%ED%8C%90_(1).pdf)

- 2차 검사보고서

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/c84770e6-5168-43cd-8fb0-8f3d8922bdcb/A06%ED%8C%80_2%EC%B0%A8_%EA%B2%80%EC%82%AC_%EB%B3%B4%EA%B3%A0%EC%84%9C_\(1\).pdf](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/c84770e6-5168-43cd-8fb0-8f3d8922bdcb/A06%ED%8C%80_2%EC%B0%A8_%EA%B2%80%EC%82%AC_%EB%B3%B4%EA%B3%A0%EC%84%9C_(1).pdf)

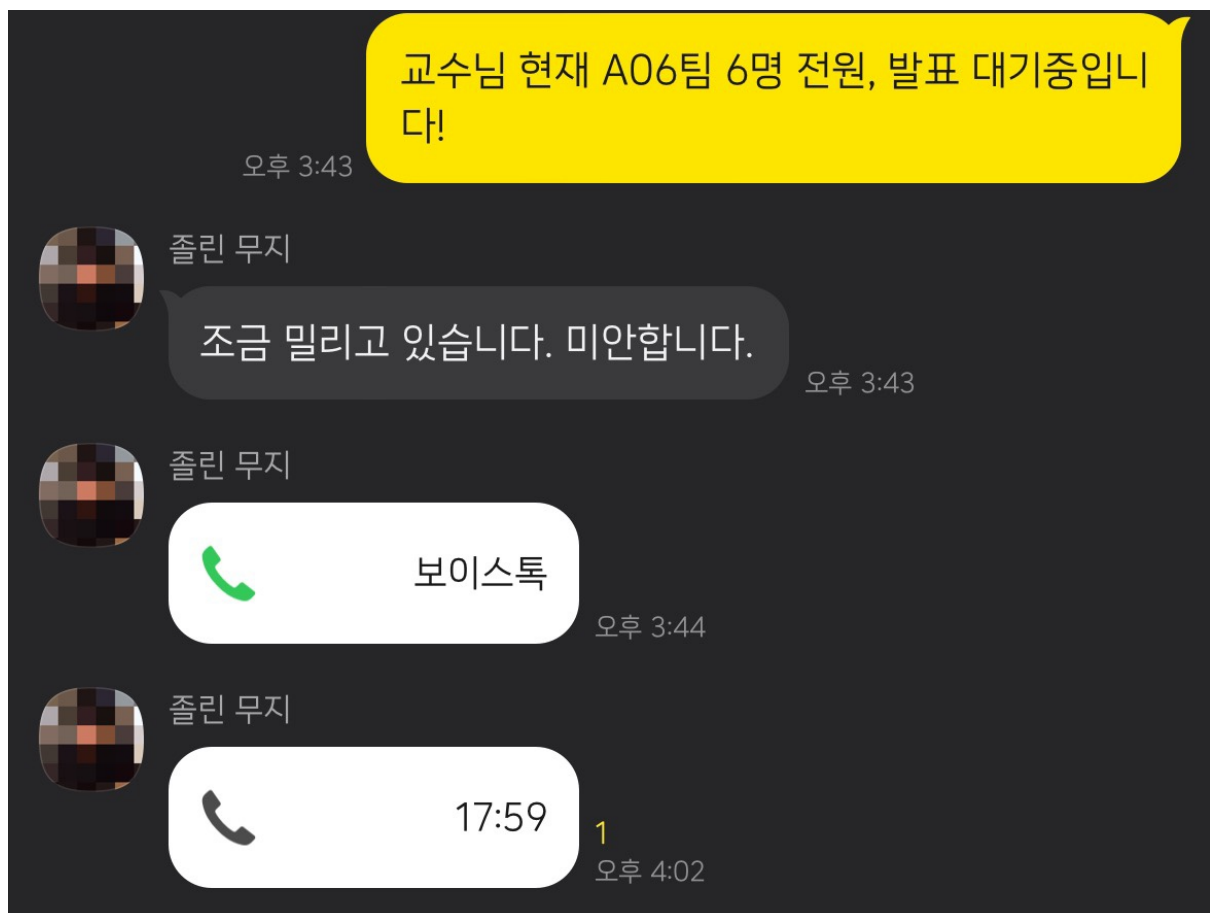
- 구현물 최종

<https://s3-us-west-2.amazonaws.com/secure.notion-static.com/a976bbee-ce20-41b2-9764-ed0a0fcfc466/StudyRoom.zip>

- 기말발표 자료

[https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e9f8a0ed-6d66-421b-bbaa-56b1b9764c8e/%EA%B8%B0%EB%A7%90%EB%B0%9C%ED%91%9C%EC%9E%90%EB%A3%8C\(%EC%B5%9C%EC%A2%85\)_\(1\).pdf](https://s3-us-west-2.amazonaws.com/secure.notion-static.com/e9f8a0ed-6d66-421b-bbaa-56b1b9764c8e/%EA%B8%B0%EB%A7%90%EB%B0%9C%ED%91%9C%EC%9E%90%EB%A3%8C(%EC%B5%9C%EC%A2%85)_(1).pdf)

06.19



- 발표 후기

- 데이터 삭제 혹은 변경을 하게 되면 검사보고서에 기존 이력들이 정상 작동하는 지 꼭 명시를 할 것.
- 준비한 발표대본을 토대로 전원 발표를 무사히 진행했고 마무리를 했다.
- 한 학기(15주) 동안 느꼈던 팀프로젝트에 대한 평은 추후에 남기도록 하겠다.

전공기초프로젝트 후기

• 감정

- 누군가에게 무언가를 지시하는 것은 상당히 어렵고, 신경이 많이 쓰이는 과정인 것을 느꼈다.
- 프로젝트의 방향성을 제시하는 이는 다른 이들보다 더욱 많은 것을 알아야 한다는 것을 깨달았다.
- 어떠한 프로젝트를 기획하고, 완성을 향해 나아가는 과정이 재미있었다.

• 앞으로의 방향성

- git을 무조건 써야겠다.
- online 공유 문서 활용, git에서의 프로젝트 진행 방향 설정 등 다양한 능력들이 기획하는 것에 대해 필요하다는 것을 깨달았다.
- 다음 프로젝트 할 땐, git-flow, trunk-based 전략과 같이 체계적으로 진행하고자 한다.