

WePlan

동아리 스케줄 공유를 통한 공용공간 예약 및 조회 시스템

7조 박종범, 정유환, 이승주, 김동령

목차

WePlan

1. 문제 정의
2. 해결 방안
3. 요구사항 명세
4. 디자인 설계
5. 구현 및 테스트

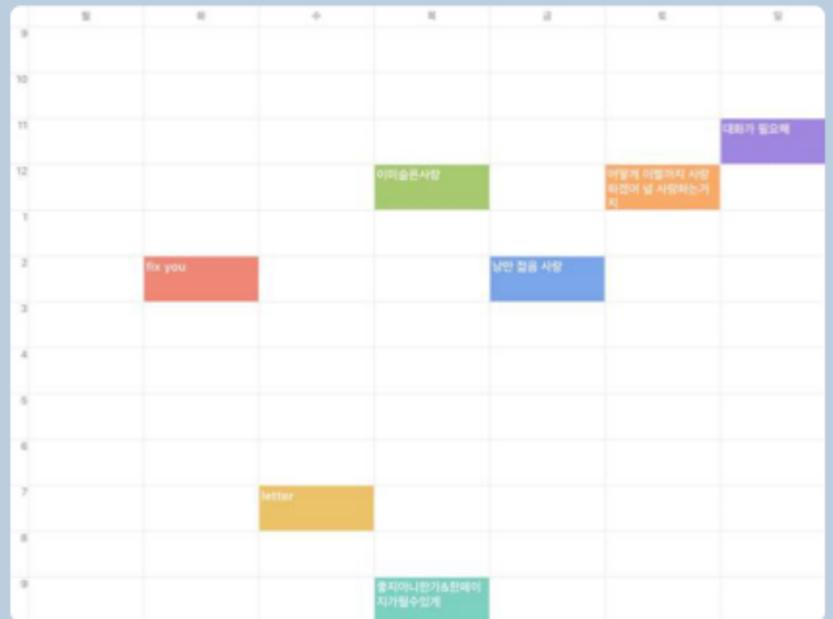
문제 정의

1. 밴드 동아리 회장이 단체톡방에 매번 일정표를 업데이트하여 공지하는 모습

📅 2023년 9월 26일 화요일



음료수 마시는 어피치



동아리방 사용 타임테이블입니다. 해당 시간에는 개인연습이 어려우니 동아리방 사용에 참고해주시기 바랍니다. 타임테이블은 일 1회 업데이트됩니다.

👍 3 👤

2
오후 1:08

📅 2023년 9월 27일 수요일



음료수 마시는 어피치



👍 4 👤

2
오후 10:45

📅 2023년 9월 29일 금요일



음료수 마시는 어피치



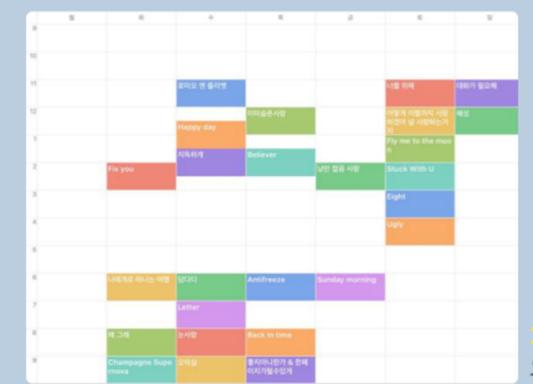
👍 4 👤

2
오후 9:37

📅 2023년 10월 3일 화요일



음료수 마시는 어피치



👍 4 👤

2
오후 8:05



<11월 27일 ~ 12월 3일 테니스장 예약 알림 >

개인정보 확인 설문지를 먼저 작성하신 후 예약 바랍니다.

1.개인정보 확인 설문지

<https://forms.gle/>

(이전 작성자는 작성하지 않으셔도 괜찮습니다.)

2. 테니스장 예약 링크

<https://forms.gle/>

3. 테니스장 예약 확인

<https://docs.google.com/spreadsheets/d/>

[준수사항]

개인 예약 및 동아리 배정된 코트를 예약(배정)된 시간임에도 사용하지 않는 경우, 코트가 비게 되어 예약하지 못한 분들이 사용할 수 있는 기회를 상실하게 됩니다.

예...

전체보기 >

2. 테니스 동아리 회장이 구글폼, 엑셀을 이용하여 동아리원의 테니스장 이용 현황을 관리하는 모습

| 월요일 | | | | | | |
|-------|----|------------------|------------------|------------------|---------|---------|
| | 1번 | 2번 | 3번 | 4번 | 5번(클레이) | 6번(클레이) |
| 7:00 | | | | | | |
| 8:00 | | | | | | |
| 9:00 | | | | | | |
| 10:00 | | | | | | |
| 11:00 | | | | | | |
| 12:00 | | | | | | |
| 13:00 | | | | | | |
| 14:00 | | ■ 13:30~15:30 | | ■ 14:00~16:00 | | |
| 15:00 | | | | | | |
| 16:00 | | | ■ 15:00~17:00 | | | |
| 17:00 | | | | ■ 16:00~18:00 | | |



3. UI/UX가 불편한 아주대학교 학사서비스의 학교 공용 공간 예약 기능

강의실 예약 현황

조회구분 일자별 건물별 조회기간 2023-12-06 시설물구분 건물명 성호관

예약 상세 내역

| 건물명 | 강의실명/호수 수용인원 | 비품 정보 | 불가 사유 | 7 | | 8 | | 9 | | 10 | | 13 | | 14 | | 15 | | 16 | | 17 | | 18 | | 19 | | 20 | | 21 | | 22 | |
|-----|------------------|-------|-------|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| | | | | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 | 00 | 30 |
| 성호관 | 자동녹화/실시간강의... | 비품 | | X | X | X | X | L | L | L | | | | | | | | | | | | | R | R | R | R | | | X | X | X |
| 성호관 | 강의실237(62명) | 비품 | | X | X | X | X | | | | | | | | | | | | | | | | R | R | | | | X | X | X | |
| 성호관 | 멀티강의실304(96명) | 비품 | | X | X | X | X | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | X | X | X | X | X | X |
| 성호관 | 강의실305(75명) | 비품 | | X | X | X | X | L | L | L | L | L | L | L | L | L | L | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 성호관 | 강의실306(86명) | 비품 | | X | X | X | X | L | L | L | L | L | L | L | L | L | L | X | R | R | X | X | X | X | X | X | X | X | X | X | X |
| 성호관 | 자동녹화/실시간강의... | 비품 | | X | X | X | X | L | L | L | L | L | L | L | L | L | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 성호관 | 자동녹화/실시간강의... | 비품 | | X | X | X | X | L | L | L | L | L | L | L | L | L | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 성호관 | 자동녹화/실시간강의... | 비품 | | X | X | X | X | L | L | L | L | L | L | L | L | L | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 성호관 | 물리전공 실습실337(...) | 비품 | | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X | X |
| 성호관 | 강의실201-1(54명) | 비품 | | X | X | X | X | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | L | X |

2 / 3 (Total Record : 24)

T 시험 L 수업 R 타인예약 M 본인예약 X 예약불가기간

9:40 5G 90

학사서비스

강의실/세미나/강당

강의실 예약 List

조회구분
 일자별
 건물별

조회기간 시설물구분
 2023-1... 강의실

건물명
 다산관

예약 상세 내역

| 건물명 | 강의실명/호수 수용인원 |
|-----|---------------|
| 다산관 | 멀티강의실B105(80) |

© 2019 Ajou University. 도움말

MY AJOU 전자출결 학사서비스 메뉴

동아리원

매번 관리자에게 스케줄을 따로 전달해야하는 불편함





동아리 관리자

동아리원들의 스케줄들을 결정 및 취합하여
일일이 공지해야하는 번거로움

문제 해결

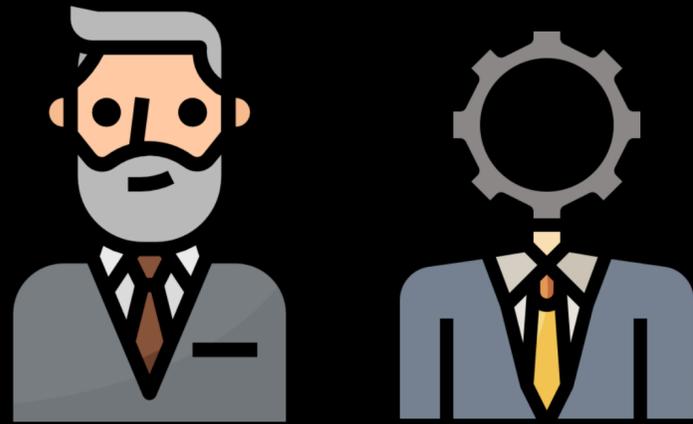
“동아리 공용공간을 동아리원들과
효율적으로 공유하며 활동하기 힘들다.”

“동아리의 일정을 한번에 관리할 수 있는 앱 서비스를 제작하자”

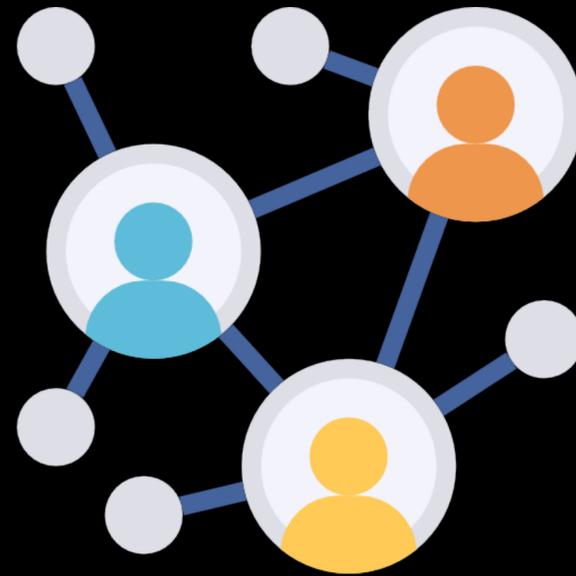
WepPlan

동아리 일정관리 및 공유 서비스

요구사항 명세



1. 유저 / 관리자



2. 각 동아리별 채널



3. 타임 테이블



3. 채널별 스케줄 확인



3. 채널별 스케줄 확인



3. 채널별 스케줄 확인

User Requirements

Requirement Specifications

- 시스템을 이용하는 유저는 관리자와 가입 유저로 구분되어야 한다.
- 시스템은 서비스를 이용하는 모든 유저에 대하여 유효한 인증 절차를 마련하여야 한다.
- 유저는 자동 로그인을 통해 간편하고 빠르게 시스템에 접속할 수 있어야 한다.
- 유저의 인증 정보가 탈취되더라도 탈취자가 해당 정보로 인증할 수 없도록 해야 한다.
- 관리자 유저는 동아리 채널을 생성, 수정, 삭제할 수 있어야 한다.
- 유저는 채널의 스케줄 예약 현황을 타임 테이블 형태로 조회할 수 있어야 한다.
- 유저는 타임 테이블에서 각 스케줄의 상세한 정보를 추가적으로 확인할 수 있어야 한다.
- 유저는 타임 테이블에서 시스템 접속 시점부터 한 달까지의 예약 현황을 조회할 수 있어야 한다.
- 유저는 채널에서 이용 가능한 시간 중 원하는 시간을 선택하여 스케줄을 예약할 수 있어야 한다.
- 관리자는 관리하고 있는 채널의 예약 현황을 별도로 확인할 수 있어야 한다.
- 관리자는 가입 유저의 예약을 승인 혹은 거절할 수 있어야 한다.
- 가입 유저의 예약 신청이 승인 혹은 거절될 경우, 가입 유저는 이를 앱을 통해서 알림을 받아야 한다.
- 유저는 타임 테이블의 변경 사항을 실시간으로 확인할 수 있어야 한다.

User Requirements Requirement Specifications

- 시스템을 이용하는 유저는 관리자과 가입 유저로 구분되어야 한다.

- 시스템은 서비스를 이용하는 모든 유저에 대하여 유효한 인증 절차를 마련하여야 한다.

- 유저는 자동 로그인을 통해 간편하고 빠르게 시스템에 접속할 수 있어야 한다.

- 유저의 인증 정보가 탈취되더라도 탈취자가 해당 정보로 인증할 수 없도록 해야 한다.

- **관리자 유저는 동아리 채널을 생성, 수정, 삭제할 수 있어야 한다.**

- **유저는 채널의 스케줄 예약 현황을 타임 테이블 형태로 조회할 수 있어야 한다.**

- 유저는 타임 테이블에서 각 스케줄의 상세한 정보를 추가적으로 확인할 수 있어야 한다.

- 유저는 타임 테이블에서 시스템 접속 시점부터 한 달까지의 예약 현황을 조회할 수 있어야 한다.

- **유저는 채널에서 이용 가능한 시간 중 원하는 시간을 선택하여 스케줄을 예약할 수 있어야 한다.**

- 관리자는 관리하고 있는 채널의 예약 현황을 별도로 확인할 수 있어야 한다.

- **관리자는 가입 유저의 예약을 승인 혹은 거절할 수 있어야 한다.**

- 가입 유저의 예약 신청이 승인 혹은 거절될 경우, 가입 유저는 이를 앱을 통해서 알림을 받아야 한다.

- 유저는 타임 테이블의 변경 사항을 실시간으로 확인할 수 있어야 한다.

Functional Requirements

System Requirements

- 시스템은 회원가입 시 관리자 권한을 인증하기 위해 관리자 인증번호를 요청하고 이를 확인해야 한다.
- 시스템은 유저가 아이디, 비밀번호를 통해 로그인을 진행하도록 하며, 성공적으로 인증이 되어야 로그인이 되도록 한다.
- 시스템은 토큰을 기반으로 하여 이전에 수행된 로그인에 대해서는 최소 1달 간 유저의 로그인 상태를 유지해야 한다
- 시스템은 보안을 위해 비밀번호를 해싱 알고리즘을 통해 변환한 다음 저장하여야 한다
- 시스템은 채널을 생성할 수 있는 관리자 페이지의 접근 권한 및 UI를 관리자 유저에게만 제공하여야 한다.
- 시스템은 모든 종류의 유저에게 채널의 스케줄 예약 현황을 타임 테이블 형태로 제공하여야 한다.
- 시스템은 채널의 타임 테이블에서, 각 스케줄 블록의 스케줄 상세 정보를 확인할 수 있도록 하는 기능을 별도로 제공한다.
- 시스템은 접속 시점부터 최소 한달까지의 스케줄 정보를 채널의 타임 테이블에 출력해야 한다.
- 시스템은 사용자가 스케줄에 대한 예약을 신청할 수 있는 별도의 UI를 제공하여야 한다.
- 시스템은 스케줄 예약 및 예약 승인시 스케줄 간에 충돌이 일어나지 않도록 한다.
- 시스템은 관리자가 해당 채널의 예약 신청 현황을 조회할 수 있는 UI를 별도로 제공하여야 한다.
- 시스템은 관리자가 스케줄 예약 요청에 대한 승인 혹은 거절할 수 있는 별도의 UI를 제공하여야 한다
- 시스템은 관리자의 승인/거절 결과를 해당 예약을 신청한 가입 유저에게 알림을 발송하여 알린다.

Functional Requirements

System Requirements

- 시스템은 회원가입 시 관리자 권한을 인증하기 위해 관리자 인증번호를 요청하고 이를 확인해야 한다.
- 시스템은 사용자가 아이디, 비밀번호를 통해 로그인을 진행하도록 하며, 성공적으로 인증이 되어야 로그인이 되도록 한다.
- 시스템은 토큰을 기반으로 하여 이전에 수행된 로그인에 대해서는 최소 1달 간 유저의 로그인 상태를 유지해야 한다
- 시스템은 보안을 위해 비밀번호를 해싱 알고리즘을 통해 변환한 다음 저장하여야 한다

- **시스템은 채널을 생성할 수 있는 관리자 페이지의 접근 권한 및 UI를 관리자 유저에게만 제공하여야 한다.**
- **시스템은 모든 종류의 유저에게 채널의 스케줄 예약 현황을 타임 테이블 형태로 제공하여야 한다.**

- 시스템은 채널의 타임 테이블에서, 각 스케줄 블록의 스케줄 상세 정보를 확인할 수 있도록 하는 기능을 별도로 제공한다.
- 시스템은 접속 시점부터 최소 한달까지의 스케줄 정보를 채널의 타임 테이블에 출력해야 한다.

- **시스템은 사용자가 스케줄에 대한 예약을 신청할 수 있는 별도의 UI를 제공하여야 한다.**

- 시스템은 스케줄 예약 및 예약 승인시 스케줄 간에 충돌이 일어나지 않도록 한다.
- 시스템은 관리자가 해당 채널의 예약 신청 현황을 조회할 수 있는 UI를 별도로 제공하여야 한다.

- **시스템은 관리자가 스케줄 예약 요청에 대한 승인 혹은 거절할 수 있는 별도의 UI를 제공하여야 한다**

- 시스템은 관리자의 승인/거절 결과를 해당 예약을 신청한 가입 유저에게 알림을 발송하여 알린다.

Non-Functional Requirements

System Requirements

- 시스템은 사용자 친화적이고 자연스러운 흐름의 UX를 제공해야 한다.
- 시스템은 한 채널의 스케줄을 주 단위의 타임 테이블로 표시하여 제공하여야 한다.
- 시스템은 한 채널의 테이블 내에서 각 스케줄을 각기 다른 색상으로 표기하여야 한다.
- 시스템은 사용자가 특정 스케줄 블록을 선택 시 제공하는 스케줄 상세 정보를 모달 형태로 출력하여야 한다.
- 시스템은 정해진 시간(30초)이 지나면 자동으로 타임 테이블 업데이트를 제공한다.
- 시스템은 사용자의 입력에 1초 이내로 반응하여야 한다.
- 시스템은 데이터 요청에 대하여 3초 이내에 응답하여야 한다.

Non-Functional Requirements

System Requirements

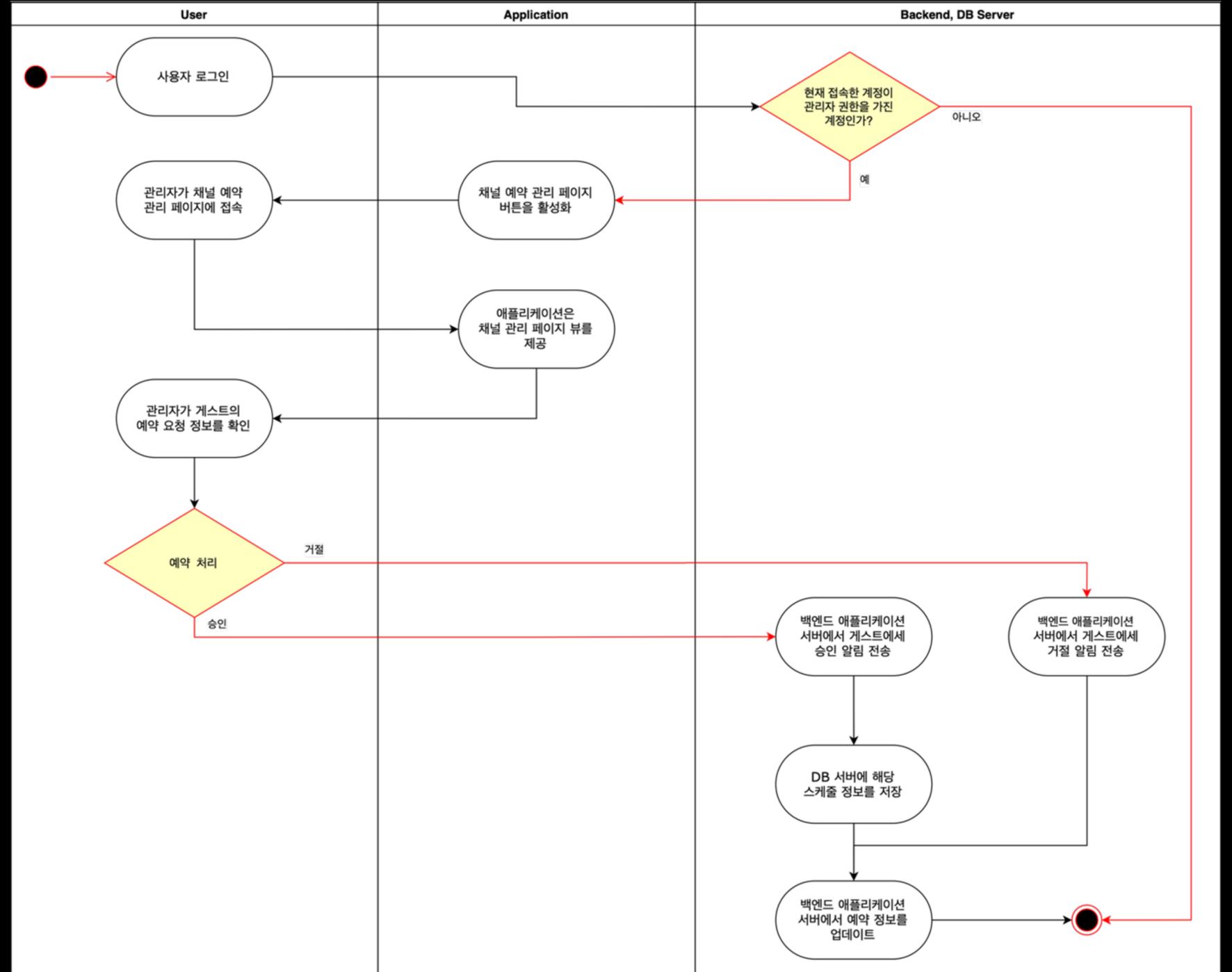
- 시스템은 사용자 친화적이고 자연스러운 흐름의 UX를 제공해야 한다.
- **시스템은 한 채널의 스케줄을 주 단위의 타임 테이블로 표시하여 제공하여야 한다.**
- **시스템은 한 채널의 테이블 내에서 각 스케줄을 각기 다른 색상으로 표기하여야 한다.**
- 시스템은 사용자가 특정 스케줄 블록을 선택 시 제공하는 스케줄 상세 정보를 모달 형태로 출력하여야 한다.
- **시스템은 정해진 시간(30초)이 지나면 자동으로 타임 테이블 업데이트를 제공한다.**
- 시스템은 사용자의 입력에 1초 이내로 반응하여야 한다.
- 시스템은 데이터 요청에 대하여 3초 이내에 응답하여야 한다.

Domain Requirements

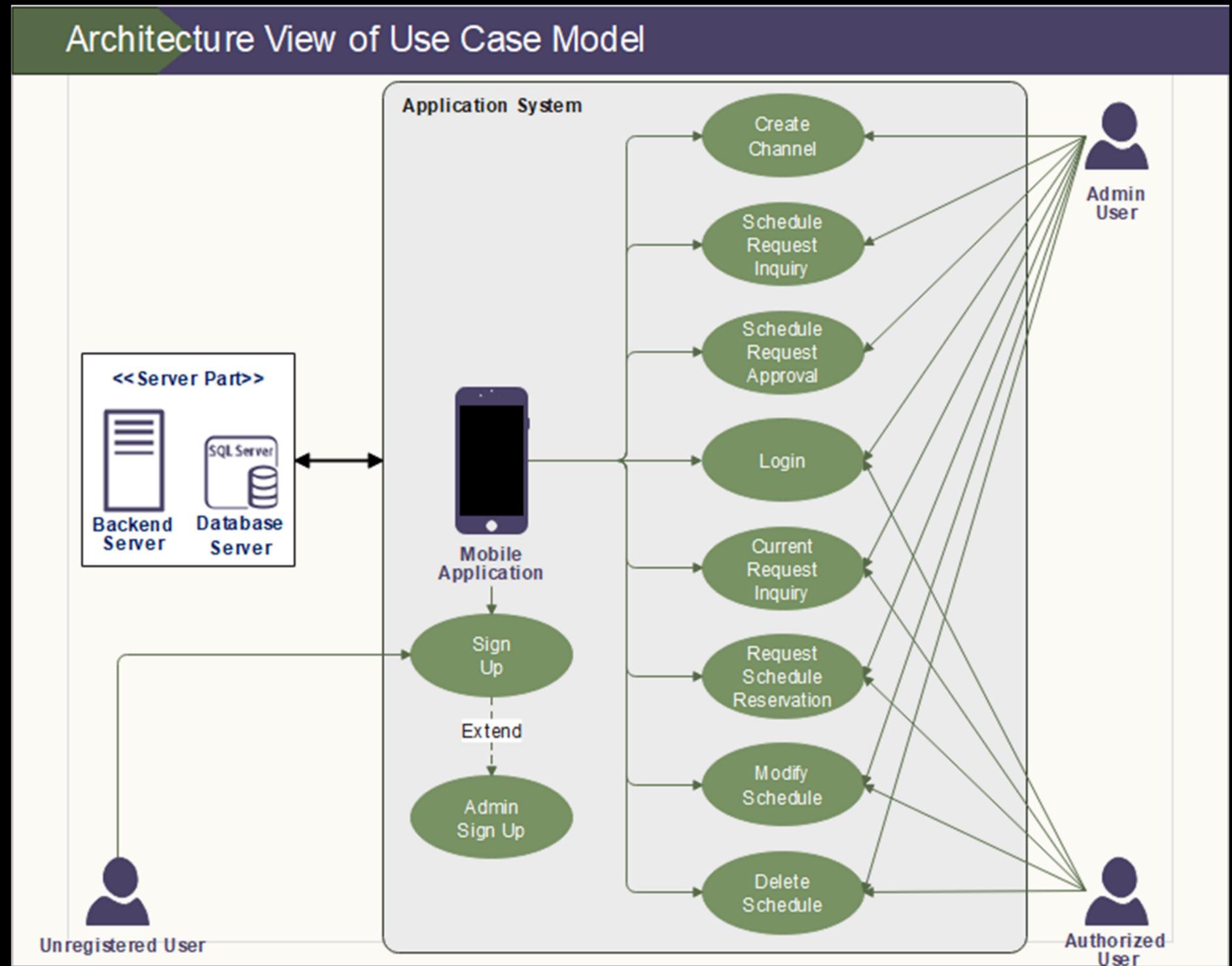
- 예약 시스템은 각 대학별 학사규정 및 규약을 준수해야 한다.
- 예약 시스템은 대학의 요청에 따라 서비스를 변경 또는 제공을 중단할 수 있어야 한다.
- 예약 시스템은 각 대학별로 재학생 인증 절차를 거쳐야 하며, 해당 유저가 재학 상태가 아닐 경우 서비스 제공을 제한할 수 있어야 한다.

Use Case & Activity Diagram

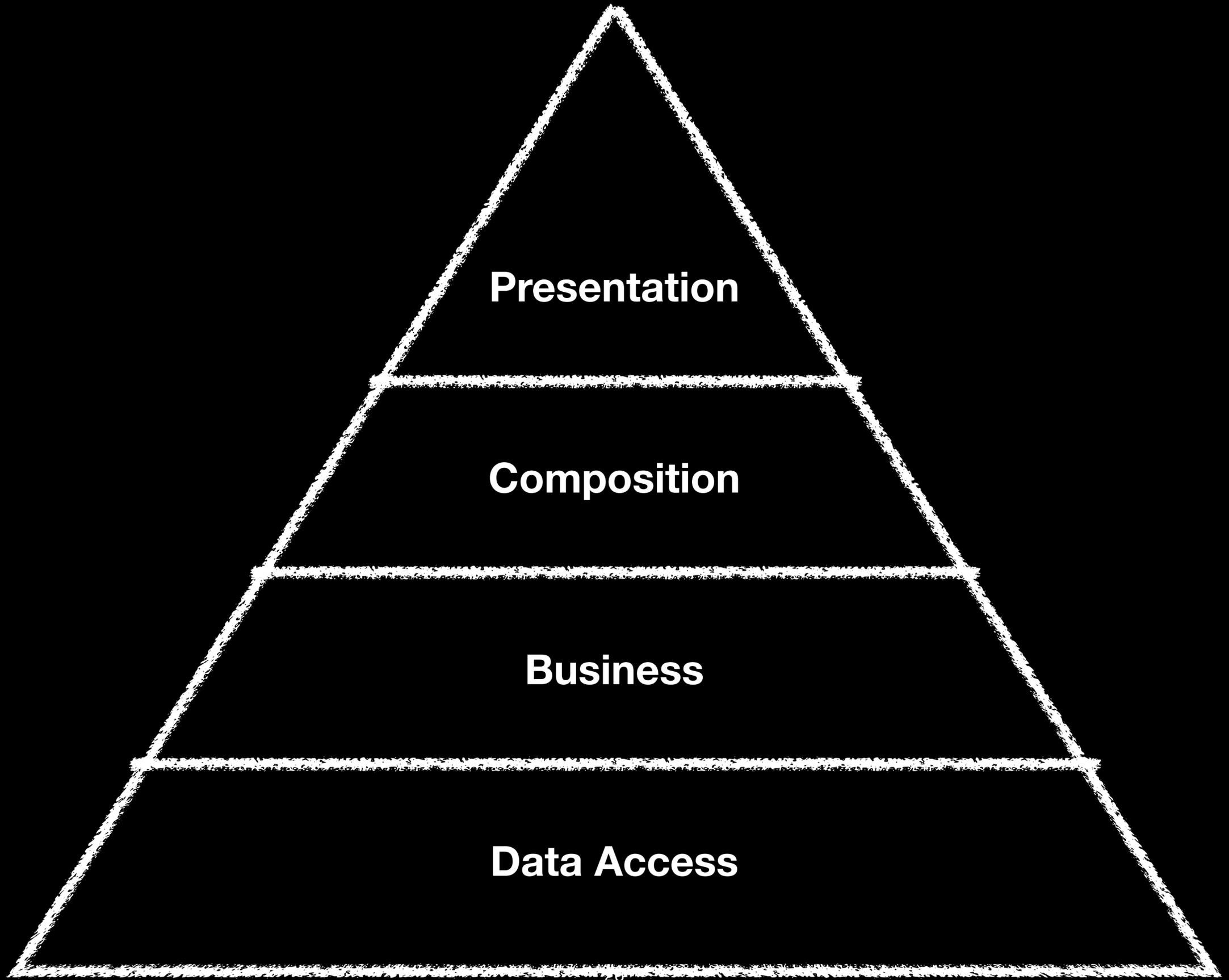
| | |
|-----------------|--|
| Use Case | 6 |
| Case Name | 관리자의 스케줄 승인/거절 |
| Actors | 유저(관리자), 모바일 애플리케이션, 백엔드 애플리케이션 서버, DB 서버 |
| Description | 관리자가 가입 유저가 요청한 스케줄을 승인하거나 거절할 수 있다 |
| Pre-Conditions | 유저는 관리자 권한이 부여된 계정으로 로그인을 성공해야 하며, 현재 승인을 대기 중인 예약이 존재해야 한다. |
| Post-Conditions | 예약의 승인 여부에 따라 해당 예약 정보가 타임 테이블에 반영된다. |



Use Case Model



디자인 설계



Presentation

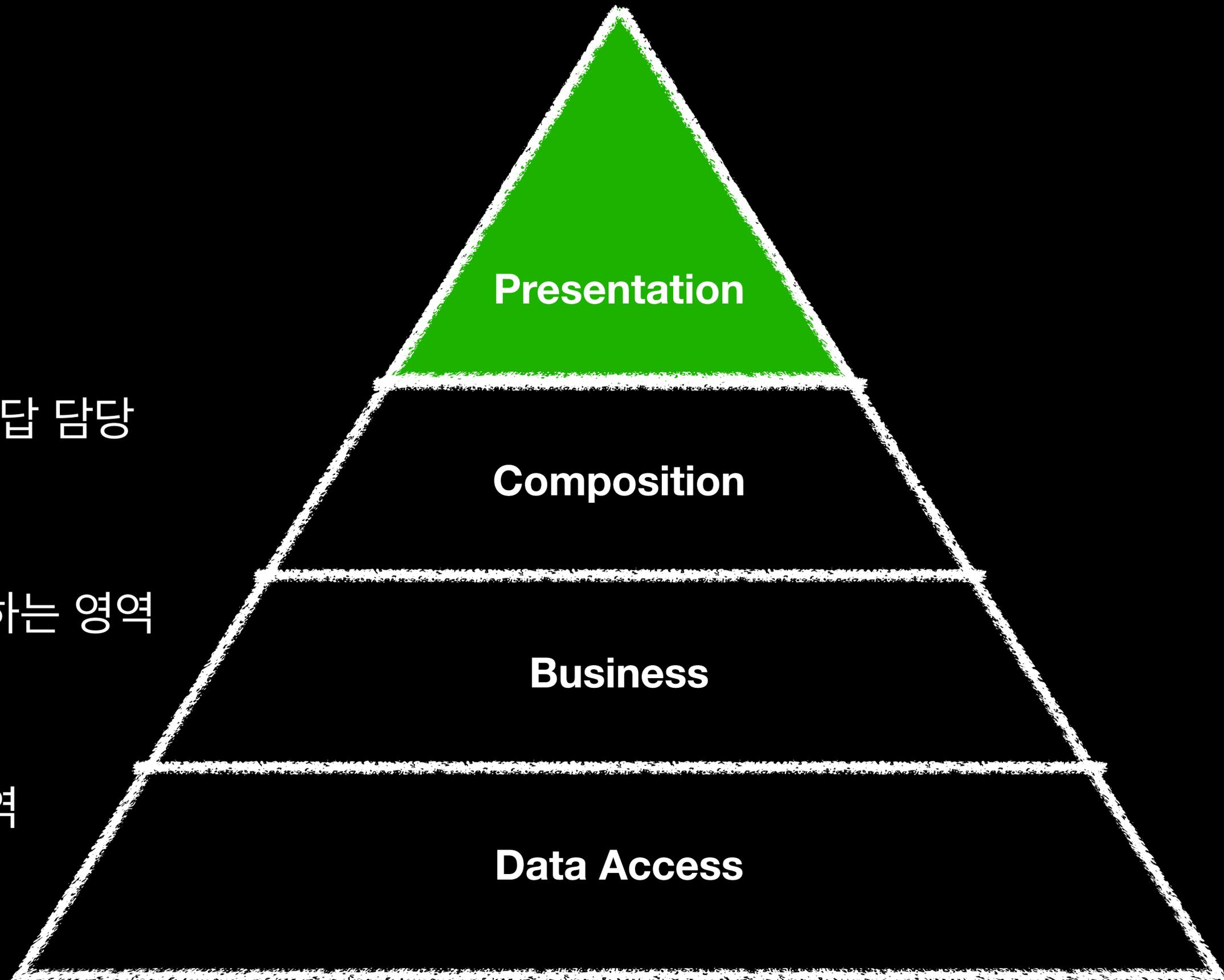
Composition

Business

Data Access

Presentation Software Layer

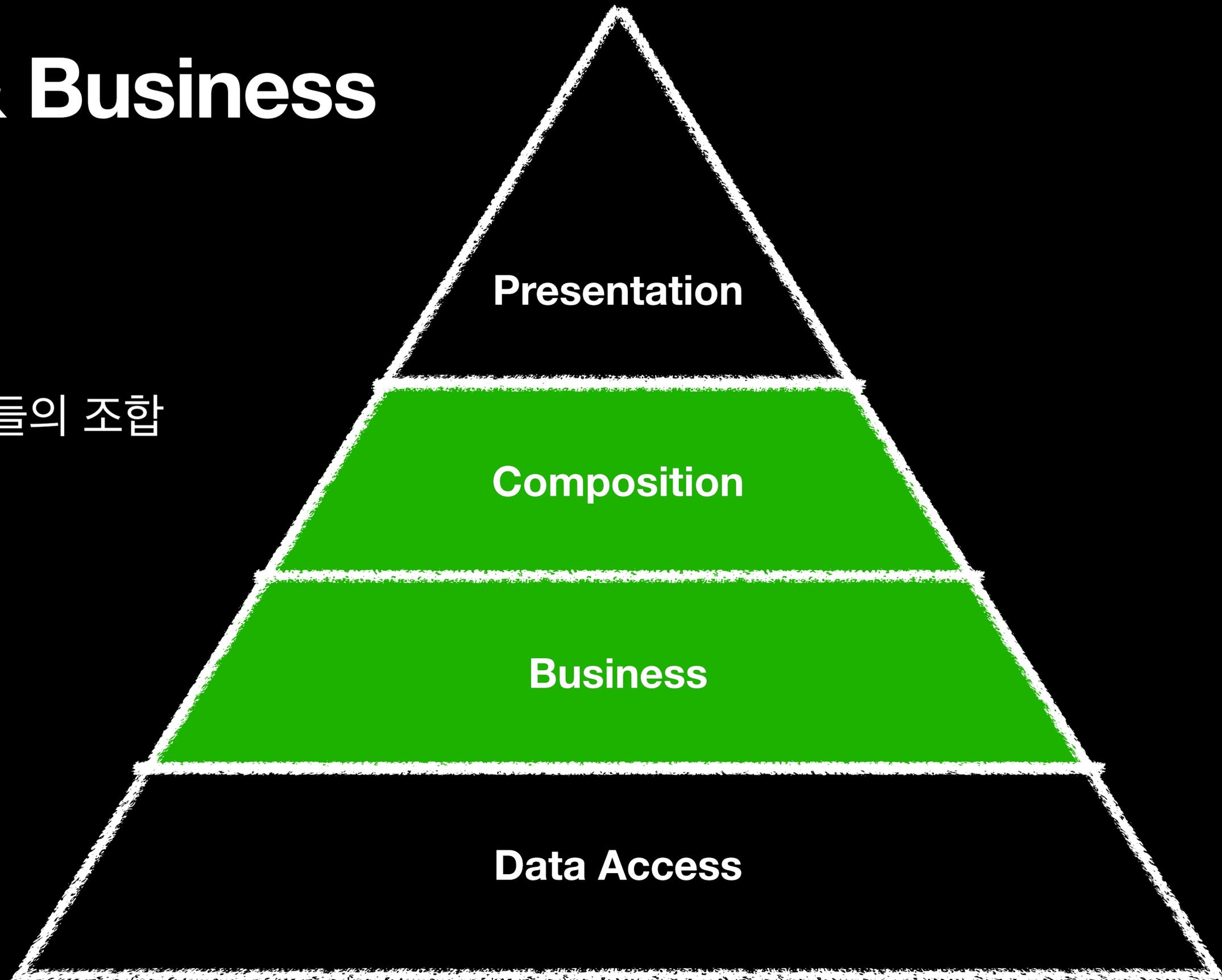
- 외부 의존성이 높은 영역
- 외부 영역에 대한 처리 및 응답 담당
- BackEnd
 - 요청에 대한 응답을 처리하는 영역
- FrontEnd
 - 실제 화면에 보여지는 영역



Composition & Business

Software Layer

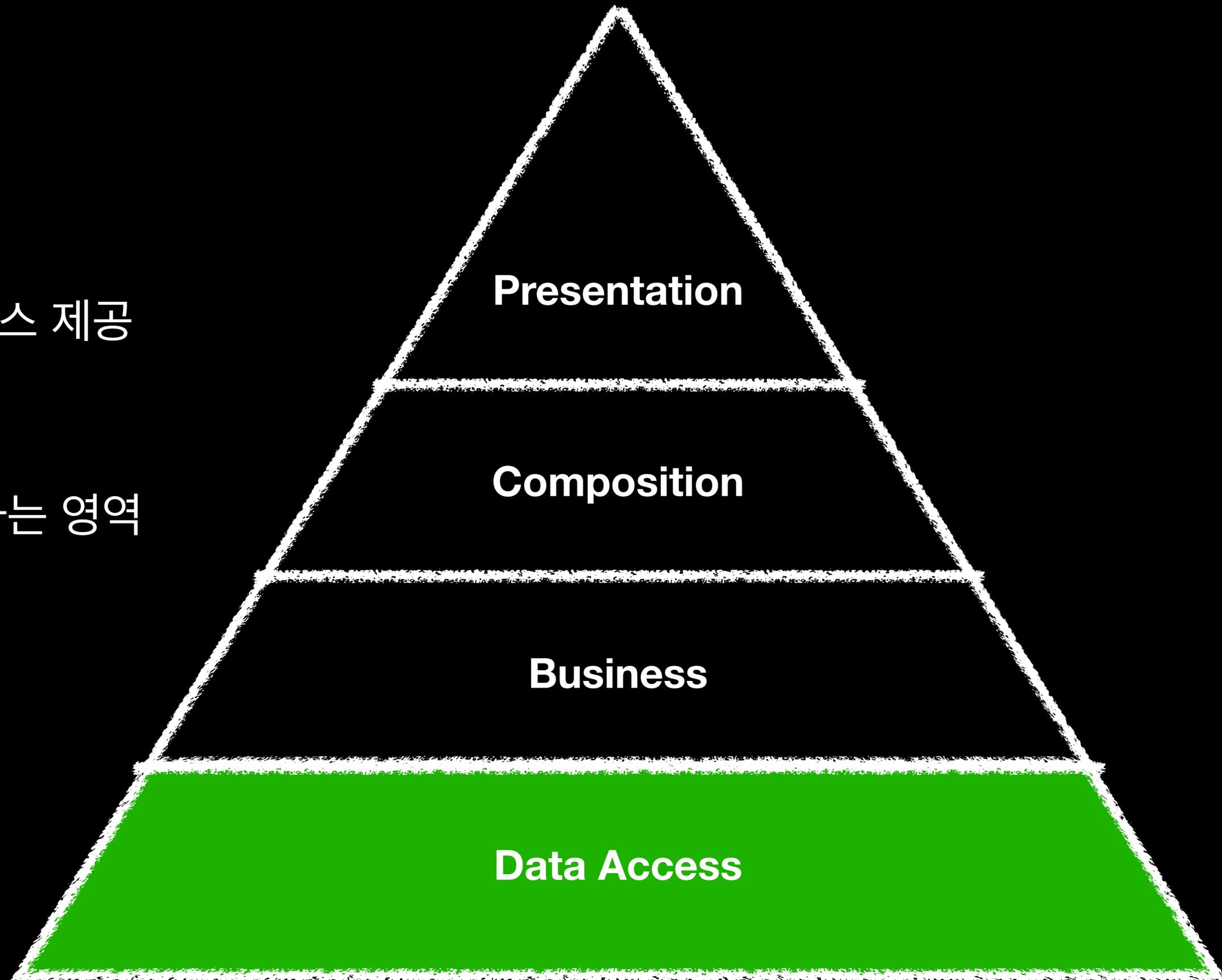
- Composition Layer
 - Business Layer 클래스들의 조합
- Business Layer
 - Business 로직을 담당



Data Access

Software Layer

- DB에 접근 가능한 인터페이스 제공
- BackEnd
 - 실제 mysql DB에 접근하는 영역
- FrontEnd
 - REST API와의 통신
 - 내부저장소



BackEnd

Channel
Controller

Member
Controller

Schedule
Controller

Presentation

Channel
Creation/Find
Service

Member
SignIn/SignUp
Service

Schedule
Create/Find
Service

JWT
Create/Manage
Service

**Composition
&
Business**

Channel Repo

Member Repo

Schedule Repo

JWT Repo

Data Access

Channel
DB

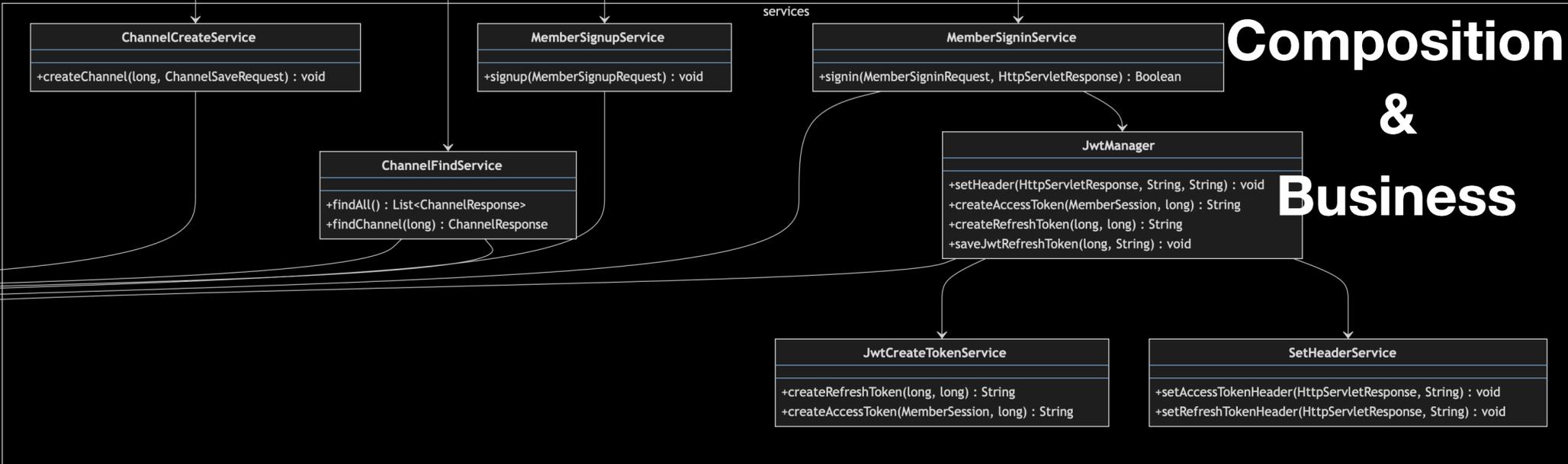
Member
DB

Schedule
DB

JWT
Repository

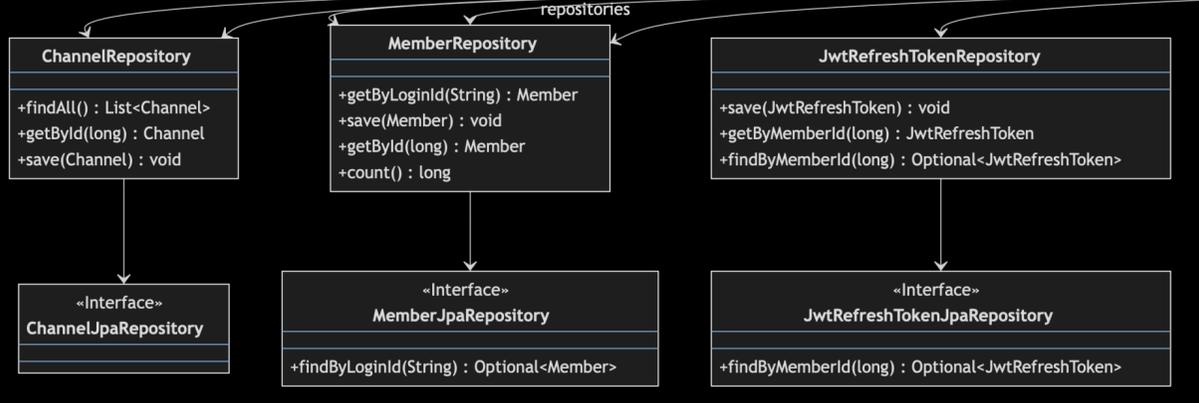
Database

Presentation



Composition & Business

Data Access



FrontEnd

Login/Register
SignApp View

TimeTable View

Reservation
Management
View

Presentation

Channel
ViewModel

Schedule
ViewModel

Auth
Service

Business

Channel
Creation/Find
Service

Member
SignIn/SignUp
Service

My Reservations
Service

Reservation
Requests
Service

Admin API

Guest API

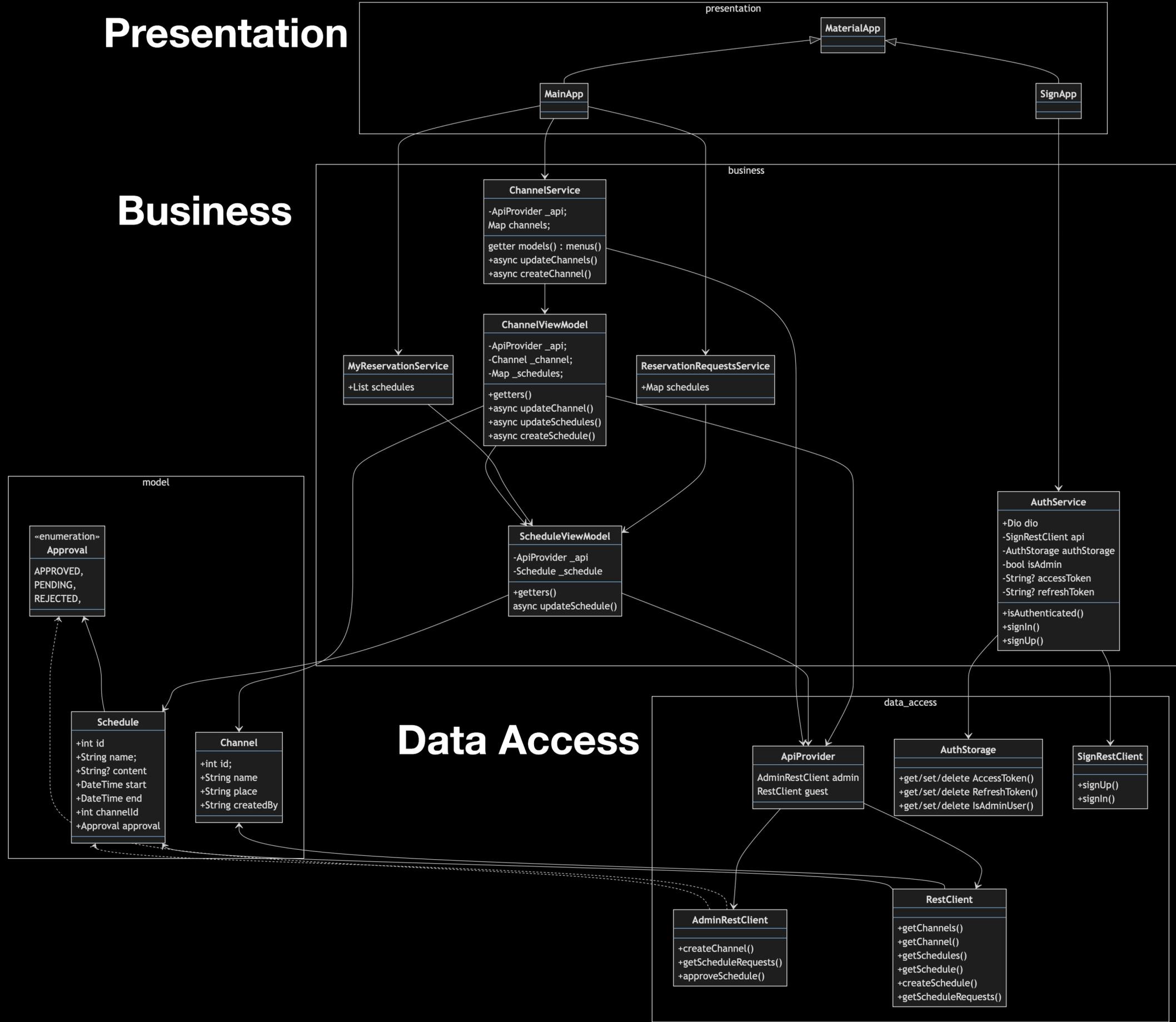
Token Storage

Data Access

Presentation

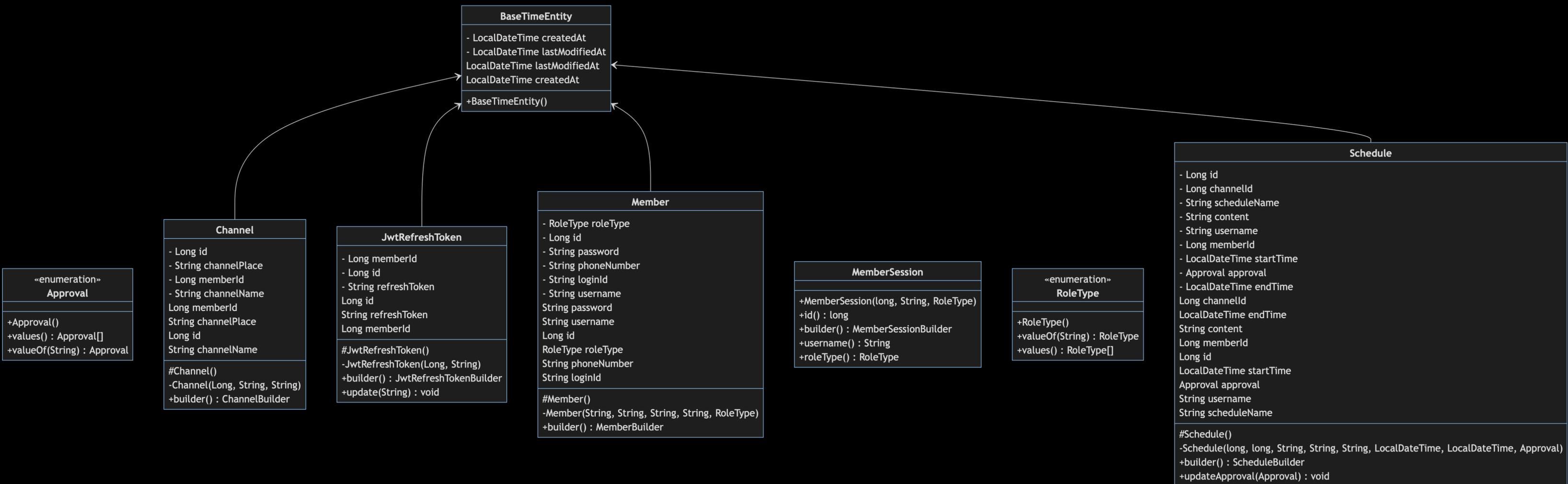
Business

Data Access

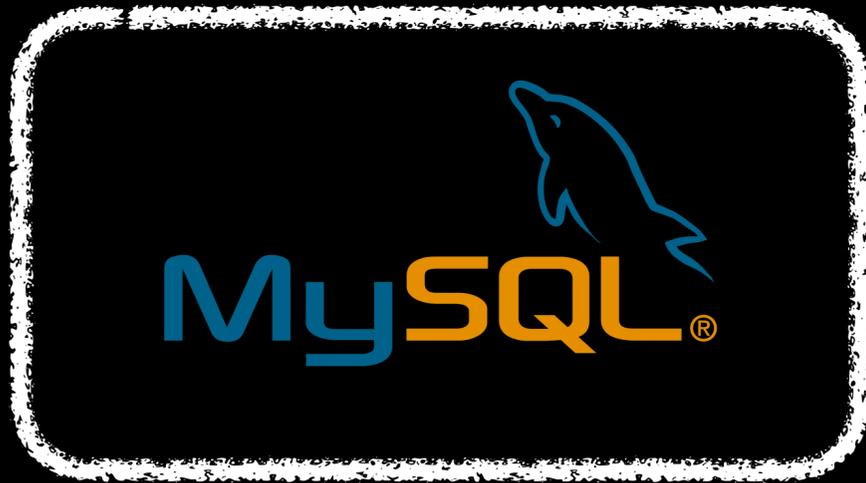


Domain Class Diagram

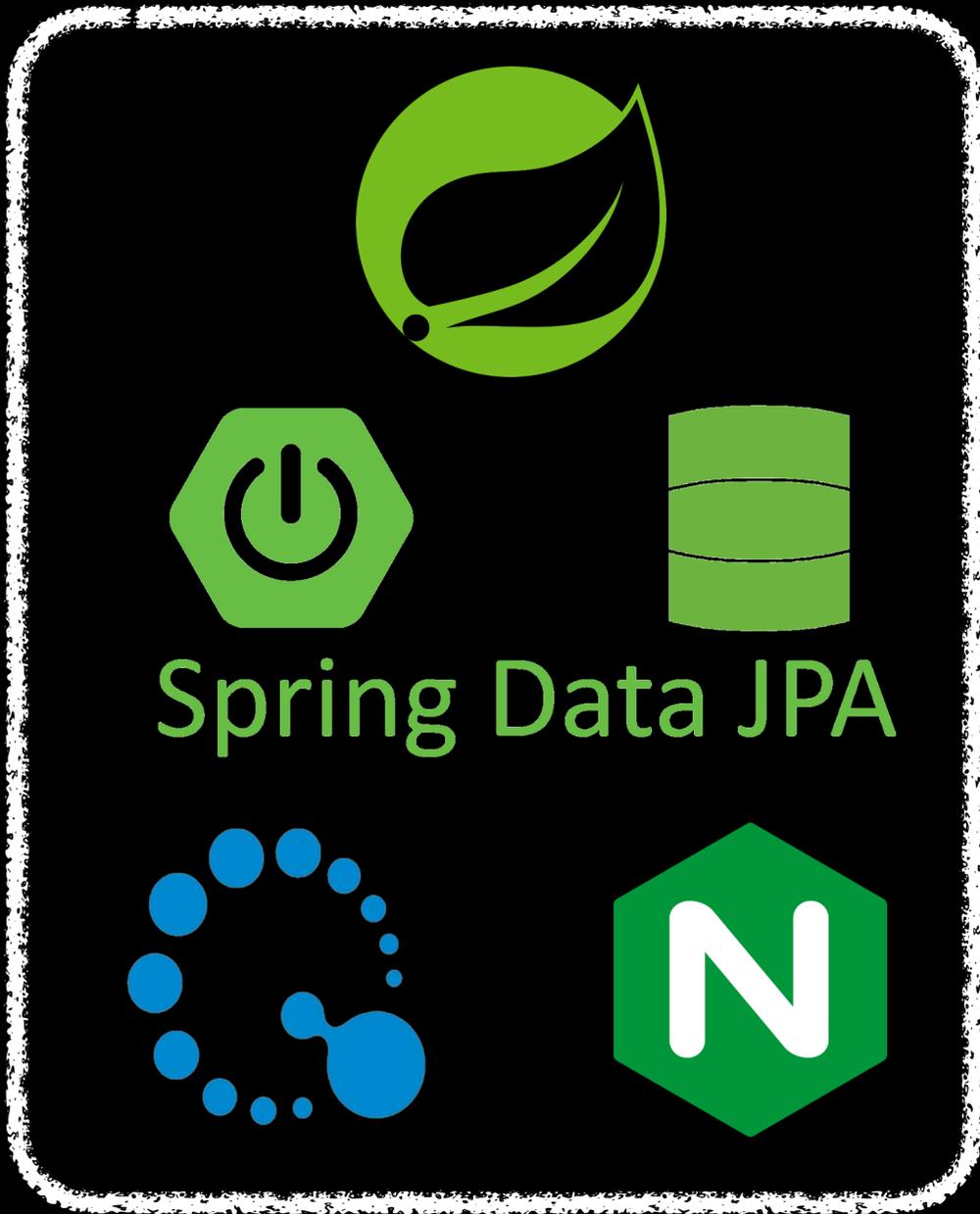
Relations between Model Entities



DB



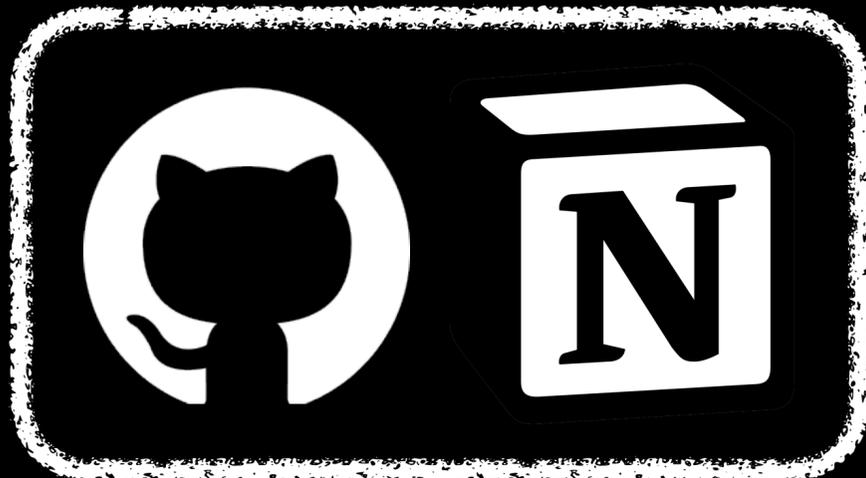
BackEnd



FrontEnd



협업툴



구현 및 테스트

테스트 방식

단위 테스트, 통합 테스트, QA 테스트

통합 테스트

특정 기능이 제대로 동작하는 지 테스트

ex) 사용자가 스케줄을 예약한다, 사용자 회원가입

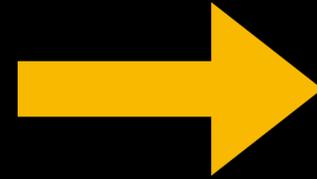
단위 테스트

특정 기능을 단위로 나누어 테스트

ex) 입력 형식 확인, 해당 시간대에 이미 예약이 있는지 확인, DB에 데이터를 저장

통합 테스트

사용자 회원가입

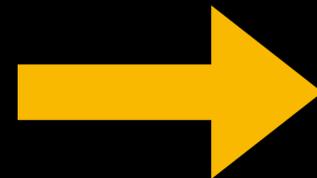


입력 형식 검증

중복된 회원이 존재하는지 확인

DB에 데이터 저장

스케줄 예약 신청



입력 형식 검증

해당 시간대에 예약 가능한지 확인

DB에 데이터 저장

단위 테스트

Tests in 'softwar7.presentation': 25 total, 25 passed

5.26 s

[Collapse](#) | [Expand](#)

| Test Name | Status | Duration |
|--------------------------------------|--------|----------|
| ChannelControllerTest 2.44 s | | |
| 로그인 한 관리자가 채널을 생성 | passed | 1.26 s |
| 권한없는 사용자가 특정 채널을 수정 | passed | 233 ms |
| 채널 목록 조회 | passed | 176 ms |
| 로그인 한 관리자가 특정 채널을 수정 | passed | 150 ms |
| 권한 없는 사용자가 특정 채널 삭제 | passed | 154 ms |
| 로그인한 관리자가 채널을 삭제 | passed | 155 ms |
| 채널 생성 실패 - 잘못된 형식 | passed | 156 ms |
| 채널 ID로 특정 채널을 조회 | passed | 151 ms |
| MemberControllerTest 924 ms | | |
| 로그인 성공 | passed | 305 ms |
| 로그인 실패 | passed | 267 ms |
| 회원가입 실패 - 잘못된 형식 | passed | 32 ms |
| 게스트 회원가입 성공 | passed | 148 ms |
| 회원가입 실패 - 중복 회원 | passed | 25 ms |
| 관리자 회원가입 성공 | passed | 147 ms |
| ScheduleControllerTest 1.90 s | | |
| 사용자의 스케줄 신청 목록 조회 | passed | 372 ms |
| 스케줄 등록을 요청한 스케줄을 승인 | passed | 166 ms |
| 스케줄 예약 요청 - 중복된 스케줄 | passed | 165 ms |
| 스케줄 등록을 요청한 스케줄을 거절 | passed | 147 ms |
| 일정에 맞는 스케줄 조회 | passed | 161 ms |
| 권한 없는 사용자가 스케줄 삭제 | passed | 147 ms |
| 스케줄 업데이트 | passed | 154 ms |
| 스케줄 예약 요청 | passed | 149 ms |
| 스케줄 등록을 요청한 스케줄 목록 조회 | passed | 150 ms |
| 단일 스케줄 조회에 성공 | passed | 146 ms |
| 스케줄 삭제 성공 | passed | 144 ms |

Tests in 'softwar7.application': 41 total, 41 passed

2.53 s

[Collapse](#) | [Expand](#)

| | |
|---------------------------------------|---------------|
| ChannelCreateServiceTest | 622 ms |
| 형식에 맞는 데이터로 채널을 생성 | passed 622 ms |
| ChannelDeleteServiceTest | 56 ms |
| 채널 삭제 권한이 없는 관리자가 채널 삭제 | passed 27 ms |
| 서버에 등록된 채널 ID로 채널 삭제 | passed 21 ms |
| 서버에 등록되지 않은 채널 ID로 채널 삭제 | passed 8 ms |
| ChannelFindServiceTest | 38 ms |
| 채널 ID 리스트로 전체 채널 조회 | passed 19 ms |
| 서버에 등록된 채널 ID로 단일 채널 조회 | passed 12 ms |
| 서버에 등록되지 않은 채널 ID로 단일 채널 조회 | passed 7 ms |
| ChannelUpdateServiceTest | 29 ms |
| 서버에 등록되지 않은 채널 ID로 채널 정보 업데이트 | passed 6 ms |
| 서버에 등록된 채널 ID로 형식에 맞는 데이터로 채널 정보 업데이트 | passed 16 ms |
| 채널 수정 권한이 없는 관리자가 채널 정보 업데이트 | passed 7 ms |
| JwtCreateTokenServiceTest | 148 ms |
| 회원 정보를 가지고 있는 AccessToken을 생성 | passed 144 ms |
| 회원 ID를 가지고 있는 RefreshToken을 생성 | passed 4 ms |
| JwtManagerTest | 74 ms |
| 이미 존재하는 회원의 토큰 관련 정보 DB에서 업데이트 | passed 48 ms |
| 회원 정보를 가지고 있는 AccessToken을 발급 | passed 4 ms |
| 로그인한 회원의 토큰 관련 정보를 DB에 저장 | passed 12 ms |
| HTTP 헤더 필드에 로그인 토큰 정보 삽입 | passed 6 ms |
| 회원 ID를 가지고 있는 RefreshToken을 발급 | passed 4 ms |
| SetHeaderServiceTest | 7 ms |
| HTTP 헤더 필드 RefreshToken에 토큰 삽입 | passed 3 ms |
| HTTP 헤더 필드 AccessToken에 토큰 삽입 | passed 4 ms |
| MemberSigninServiceTest | 1.04 s |
| 로그인시 관리자 권한이면 true를 반환한다. | passed 262 ms |
| 등록된 ID 및 패스워드로 로그인 | passed 260 ms |
| 등록되지 않은 ID 및 패스워드로 로그인 | passed 255 ms |
| 로그인시 게스트 권한이면 false를 반환한다. | passed 261 ms |
| MemberSignupServiceTest | 281 ms |

QA 테스트

코드로 테스트하기 어려운 부분과 사용자 관점에서 서비스를 테스트

테스트하기 어려운 부분

QA 테스트

- 클라이언트 앱의 반응 시간, 토큰의 만료 시간 등 변하는 시간과 관련된 로직
- 제어할 수 없는 부분을 최소화하여 최대한 코드 레벨에서 테스트 할 수 있도록 해야 한다
- 그래도 테스트하기 어려운 부분에 대해서는 QA 테스트를 진행



TestFlight

모든 테스터

앱 테스트에 초대된 모든 사람과 각 테스터와 관련된 지표를 확인할 수 있습니다. [추가 정보](#)

테스터(4)

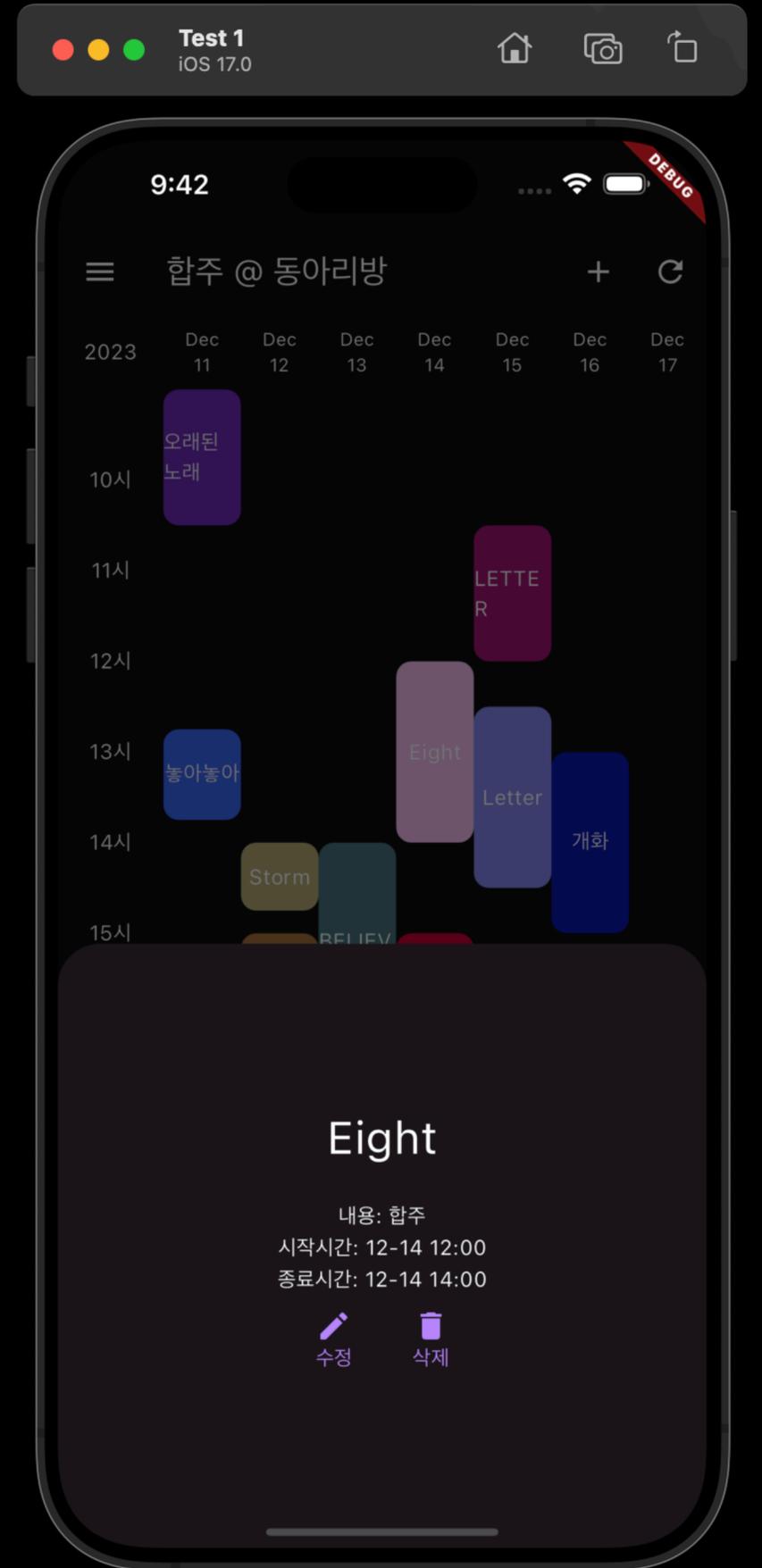
필터 추가

| <input type="checkbox"/> 테스터 | 진행 상태 ^ | 그룹 | 세션 | 충돌 | 피드백 | 기기 |
|---|----------------------------------|----|-----|----|-----|---------------------------------------|
| <input type="checkbox"/> yhwjd@naver.com 정유환 · 내부 | ✔ 설치됨 1.0.0 (22) 2023년 12월 6일 | WE | 2 | | | iPhone 8 iOS 15.1 |
| <input type="checkbox"/> insoyafear@gmail.com KimDongryeong · 내부 | ✔ 설치됨 1.0.0 (23) 2023년 12월 6일 | WE | 15 | | | iPhone 13 Pro iOS 17.1.1 |
| <input type="checkbox"/> leesj2222@naver.com 이승주 · 내부 | ✔ 설치됨 1.0.0 (23) 2023년 12월 6일 | WE | 8 | | | iPhone 14 iOS 17.1.2 |
| <input type="checkbox"/> parkjb825@icloud.com 박종범 · 내부 | ✔ 설치됨 1.0.0 (23) 2023년 12월 6일 | WE | 115 | | | iPhone 12 Pro Max 및 다른 기기 iOS 17.2 |

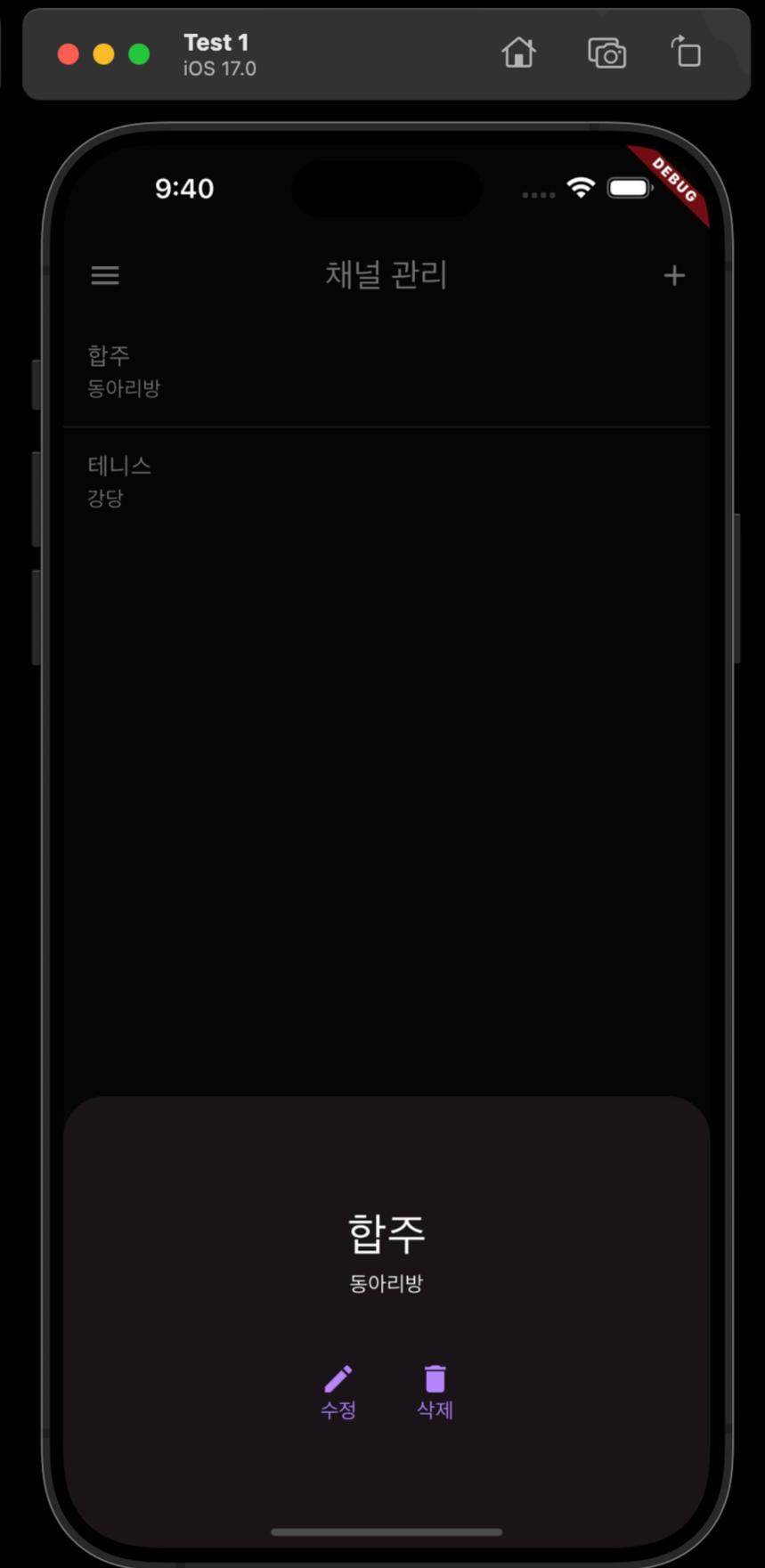
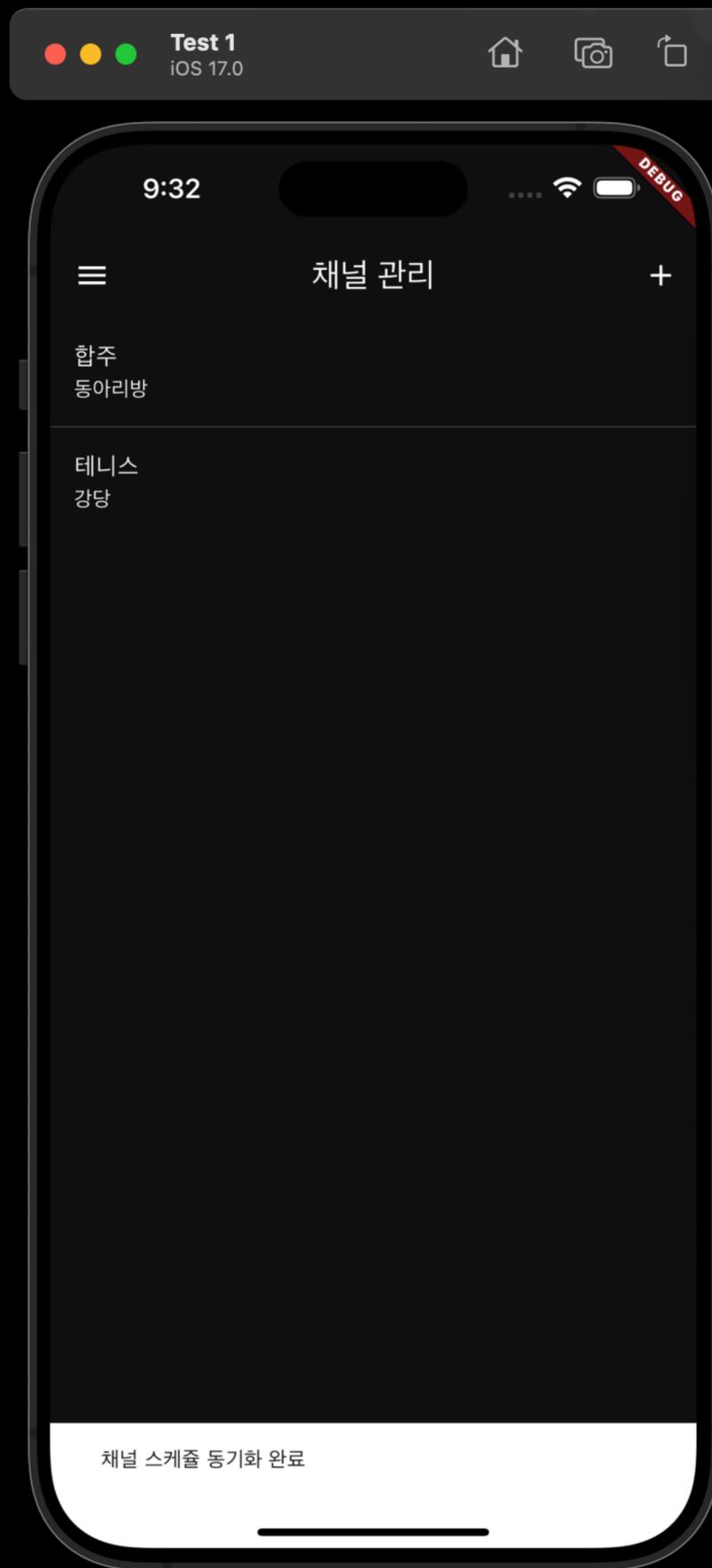
▼ 버전 1.0.0

| 빌드 ▼ | 진행 상태 | 그룹 | 초대 | 설치 | 세션 | 총돌 | 피드백 |
|--|--------------------------|----|----|----|----|----|-----|
|  23 내부 | ✔ 테스트 중 90일 후 무효화 | WE | 4 | — | — | — | — |
|  22 내부 | ✔ 테스트 중 90일 후 무효화 | WE | 4 | — | — | — | — |
|  21 내부 | ✔ 테스트 중 90일 후 무효화 | WE | 4 | 2 | 12 | — | — |
|  20 내부 | ✔ 테스트 중 90일 후 무효화 | WE | 4 | — | — | — | — |
|  19 내부 | ✔ 테스트 준비 완료 90일 후 무효화 | | 4 | — | — | — | — |
|  18 내부 | ✔ 테스트 중 88일 후 무효화 | WE | 4 | 4 | 43 | — | — |
|  17 내부 | ✔ 테스트 중 82일 후 무효화 | WE | 4 | 5 | 18 | — | — |
|  16 내부 | ✔ 테스트 중 82일 후 무효화 | WE | 4 | 1 | — | — | — |
|  13 내부 | ✔ 테스트 중 81일 후 무효화 | WE | 4 | 5 | — | — | — |
|  11 내부 | ✔ 테스트 중 78일 후 무효화 | WE | 4 | 5 | — | — | — |
|  9 내부 | ✔ 테스트 중 78일 후 무효화 | WE | 4 | 5 | — | — | — |

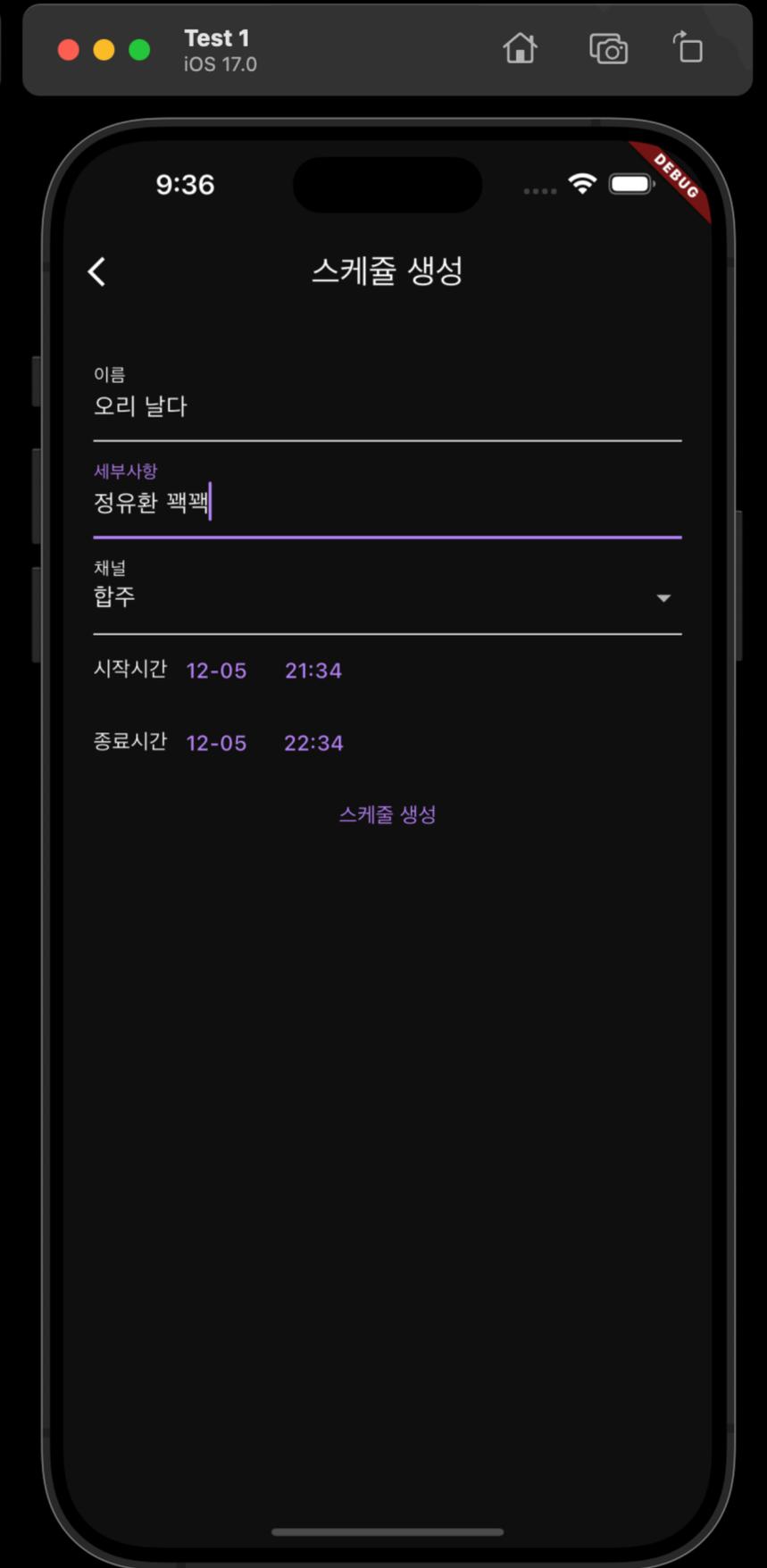
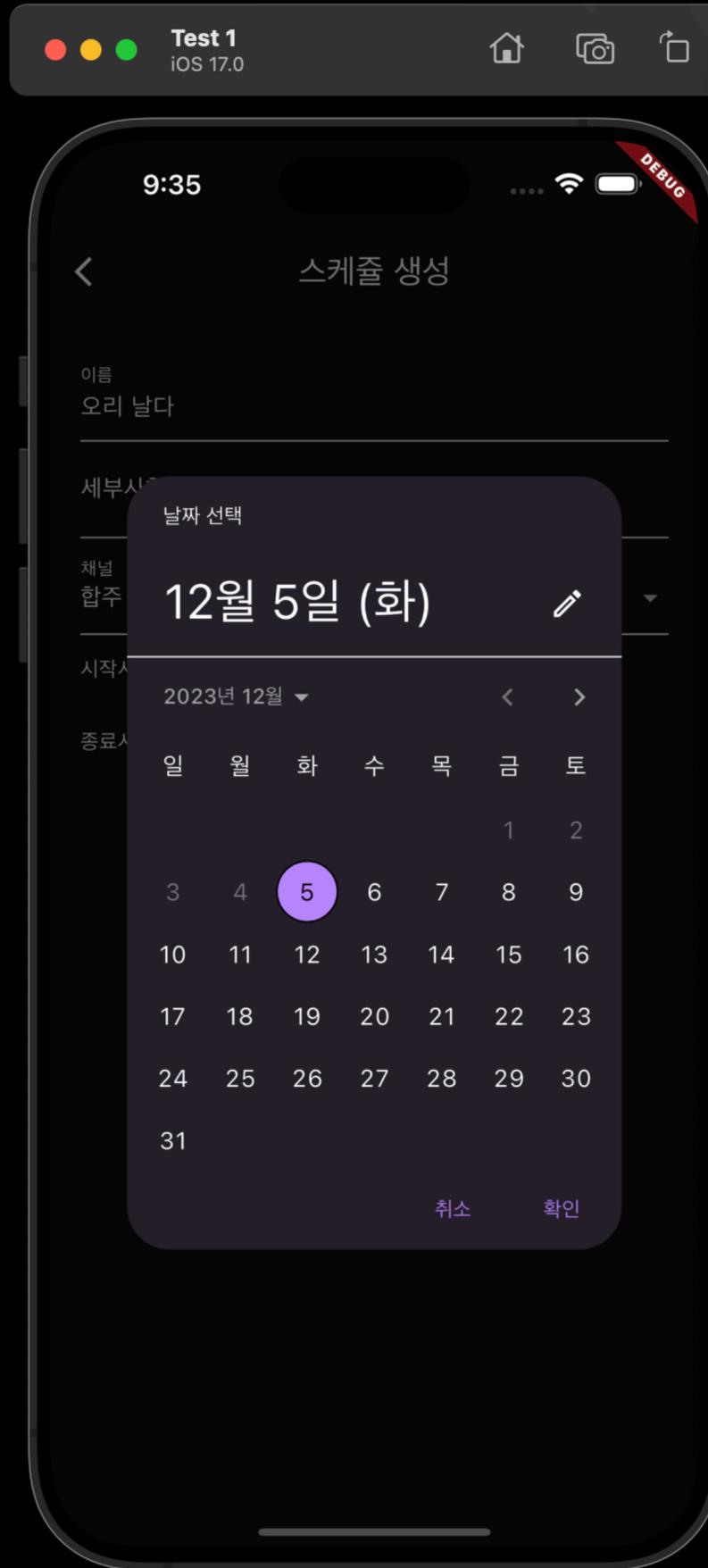
타임테이블



채널 생성/관리 페이지



스케줄 생성 페이지



Rest Docs + Swagger UI API Documentation

회원



POST

/api/signin 로그인



POST

/api/signup 회원가입



채널



POST

/api/admin/channels 채널 생성



GET

/api/guest/channels 모든 채널 조회



DELETE

/api/admin/channels/{channelId} 채널 삭제



PATCH

/api/admin/channels/{channelId} 채널 수정



GET

/api/guest/channels/{channelId} 단일 채널 조회



스케줄



GET

/api/guest/schedules 해당 일정의 스케줄 조회



GET

/api/admin/schedules/requests 스케줄 등록 요청 목록 조회



POST

/api/admin/schedules/requests 스케줄 등록 요청 승인



GET

/api/guest/schedules/requests 게스트의 요청 스케줄 목록 조회



POST

/api/guest/schedules/requests 스케줄 생성



GET

/api/guest/schedules/{scheduleId} 단일 스케줄 조회



DELETE

/api/guest/schedules/{scheduleId} 스케줄 삭제



PATCH

/api/guest/schedules/{scheduleId} 스케줄 수정



감사합니다