


# 캡스톤 디자인 I

## 종합설계 프로젝트

프로젝트 명	디클
팀 명	7조
문서 제목	결과보고서

<b>Version</b>	1.0
<b>Date</b>	2024-MAY-13

팀원	권 지아(조장)
	윤 응배
	김 동운
	윤 흥현

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14


**CONFIDENTIALITY/SECURITY WARNING**

이 문서에 포함되어 있는 정보는 국민대학교 전자정보통신대학 컴퓨터공학부 및 컴퓨터공학부 개설 교과목 캡스톤 디자인 I 수강 학생 중 프로젝트 “디클”을 수행하는 팀 “7조”의 팀원들의 자산입니다. 국민대학교 컴퓨터공학부 및 팀 “7조”의 팀원들의 서면 허락없이 사용되거나, 재가공 될 수 없습니다.

### 문서 정보 / 수정 내역


<b>Filename</b>	수행결과보고서-7조디클.doc
원안작성자	윤웅배, 김동윤, 윤홍현
수정작업자	

수정날짜	대표수정자	Revision	추가/수정 항목	내용
2024-05-14	윤웅배	1.0	최초 작성	

 국민대학교 컴퓨터공학부 캠프스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 목 차

1	개요	4
1.1	프로젝트 개요	4
1.2	추진 배경 및 필요성	4
2	개발 내용 및 결과물	5
2.1	목표	5
2.2	연구/개발 내용 및 결과물	6
2.2.1	연구/개발 내용	6
2.2.2	시스템 기능 및 구조 설계도	6
2.2.3	활용/개발된 기술	6
2.2.4	현실적 제한 요소 및 그 해결 방안	6
2.2.5	결과물 목록	6
2.3	기대효과 및 활용방안	6
3	자기평가	7
4	참고 문헌	7
5	부록	7
5.1	사용자 매뉴얼	7
5.2	운영자 매뉴얼	7
5.3	배포 가이드	7
5.4	디클 매뉴얼	7
5.5	디클에 대한 기술 문서	7

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

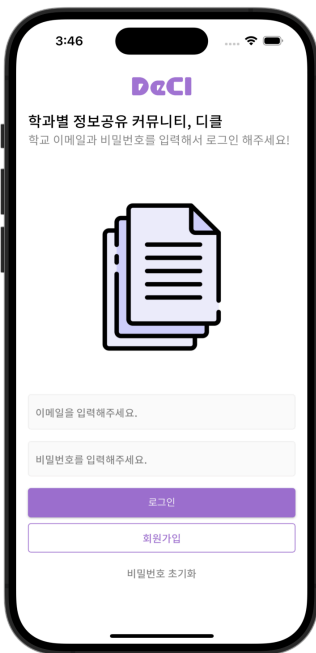
# 1 개요

## 1.1 프로젝트 개요




### 회원가입 기능

- 사용자는 전국 대학교 이메일 주소를 사용하여 가입할 수 있다. 이메일 주소는 데이터베이스에 사전 등록되어 있으며, 등록되지 않은 이메일 주소로는 가입이 불가능하다.
- 사용자는 가입 과정에서 본인의 전공 및 부/복수 전공을 선택할 수 있다. 총 두 개의 전공 정보를 입력할 수 있다.



### 로그인 기능

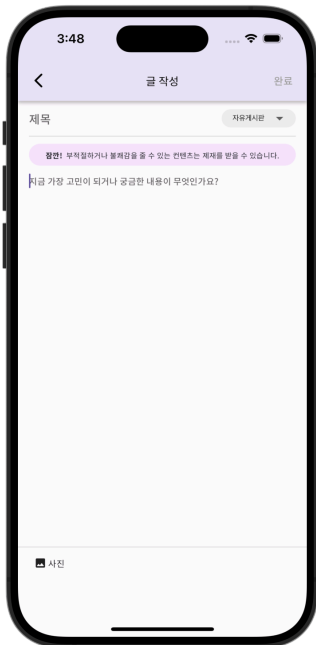
- 사용자는 JWT 토큰을 사용하여 로그인한다. 로그인 시 발급되는 Access Token과 Refresh Token을 사용하여 사용자 인증을 관리한다.
- Access Token이 만료되었을 경우, Refresh Token을 사용하여 새로운 Access Token을 재발급 받을 수 있습니다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14




### 글 목록 불러오기 기능

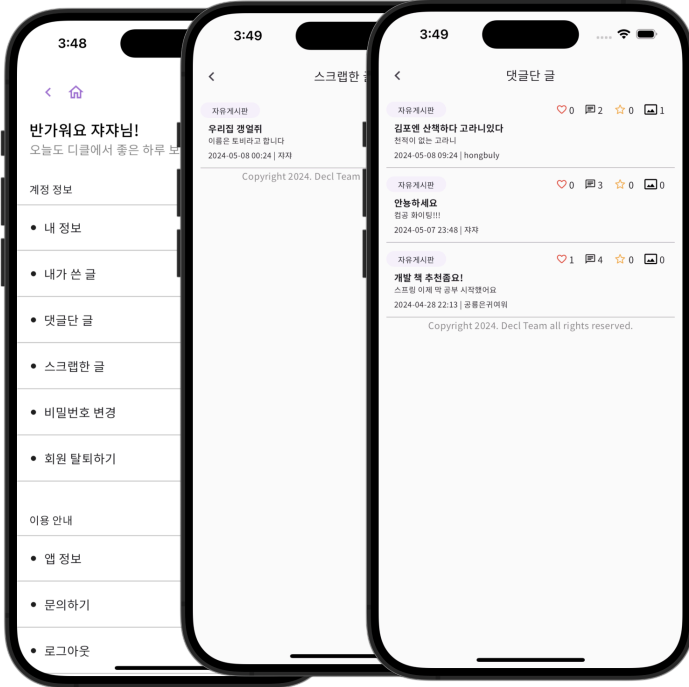
- 사용자는 현재 활성화된 학과 기준 및 커뮤니티 기준으로 작성된 글을 볼 수 있다.
- 자유게시판, 대학원 게시판 등 총 5개의 게시판과 전체 게시판, 좋아요 개수가 10개 이상인 글을 모아보는 인기 게시판이 있다.



### 글 작성 기능

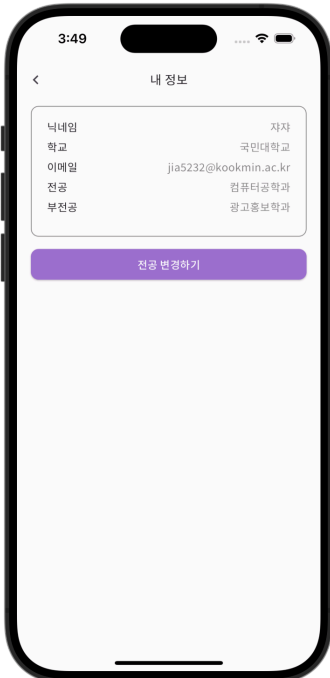
- 사용자는 특정 커뮤니티를 선택하여 글을 작성할 수 있다. 글 작성 시, 제목, 내용, 관련 이미지 등을 추가할 수 있다.

 국민대학교 컴퓨터공학부 캠프스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14




### 마이페이지

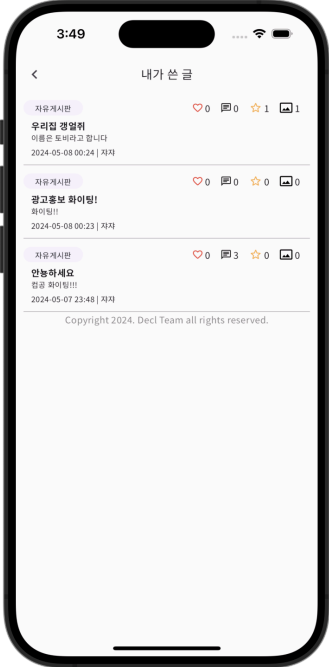
- 로그인된 사용자는 마이페이지를 통해 내 정보 확인, 내가 쓴 글, 댓글 단 글 확인, 로그아웃 등의 기능을 이용할 수 있다.



### 내 정보

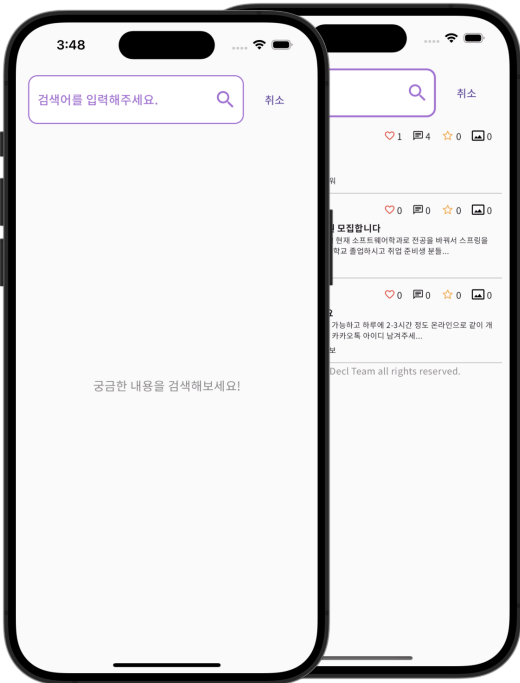
- 사용자는 자신의 닉네임, 학교, 전공 등의 정보를 확인할 수 있으며, 전공 정보를 변경할 수 있는 기능을 제공한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14




### 내가 쓴 글, 댓글 단 글, 스크랩 한 글 페이지

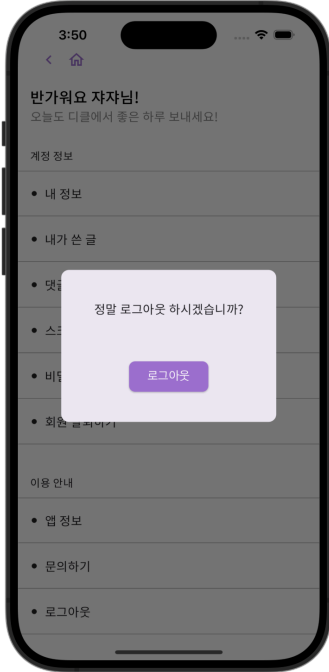
- 사용자는 자신이 작성한 글, 댓글을 단 글, 스크랩한 글을 한눈에 볼 수 있는 기능을 제공받는다.



### 검색 페이지

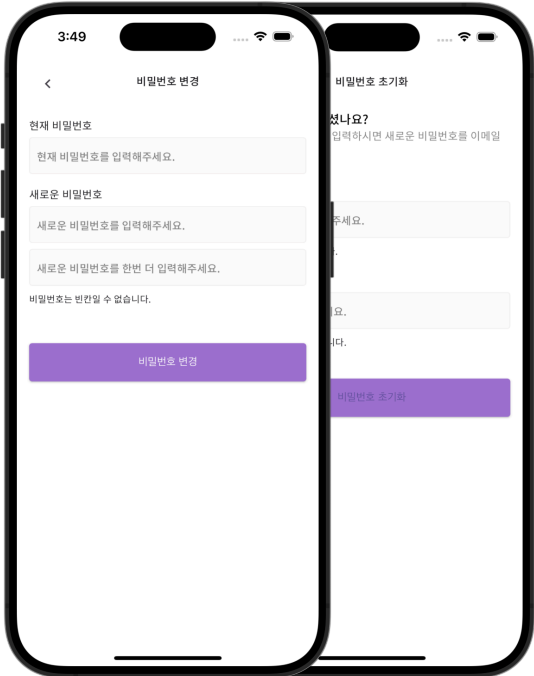
- 사용자는 특정 키워드를 제목이나 내용에 포함한 글들을 검색할 수 있다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14



### 로그아웃 기능

- 사용자는 언제든지 로그아웃을 할 수 있으며, 로그아웃 시 모든 세션이 종료된다.




### 비밀번호 변경 기능

- 사용자는 현재 비밀번호를 확인한 후 새로운 비밀번호로 변경할 수 있다.

### 비밀번호 초기화 기능

- 비밀번호를 분실한 경우, 사용자는 등록된 이메일을 통해 새로운 랜덤 비밀번호를 발급받고 초기화할 수 있다.



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14




### 비속어 필터링

- AI 모델을 학습시켜 사용자가 작성하는 글에 비속어가 포함되어 있을 경우, 경고 메시지를 표시한다.



### 신고 기능

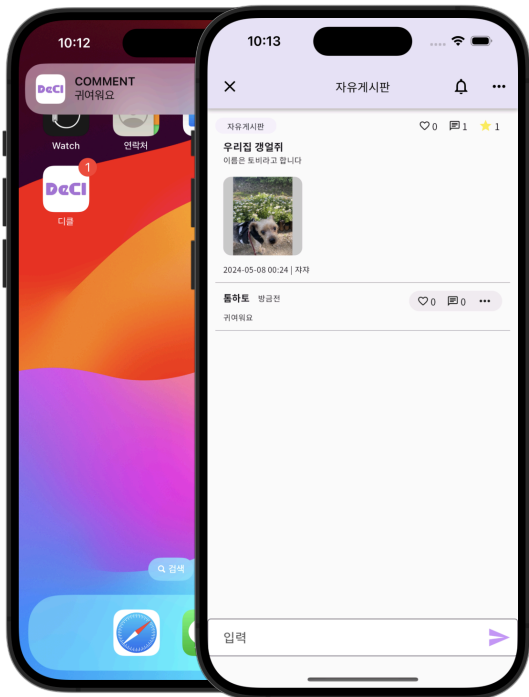
- 사용자는 특정 유저의 글, 댓글, 대댓글이 커뮤니티 가이드라인을 위배한다고 판단될 경우, 신고할 수 있다. 신고가 10회 누적된 유저는 자동으로 정지된다.

 국민대학교 컴퓨터공학부 캠프톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14




### 차단 기능

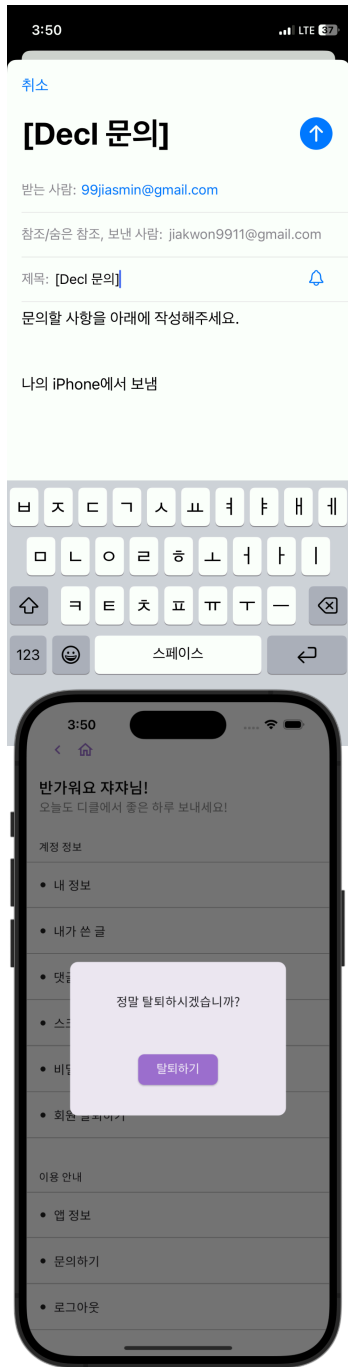
- 사용자는 특정 유저를 차단할 수 있으며, 차단된 유저의 글, 댓글, 대댓글은 보이지 않는다.



### 실시간 댓글 알림

- 사용자는 자신이 작성한 글 또는 댓글에 대댓글이 달리는 경우 실시간으로 알림을 받는다. 알림을 클릭하면 해당 글로 이동할 수 있다.

 국민대학교 컴퓨터공학부 캠프스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14




### 문의

- 사용자는 고객센터 메일로 문의 사항을 작성할 수 있으며, 문의 사항에 대한 답변을 이메일로 받을 수 있다.

### 회원 탈퇴

- 사용자는 자신의 계정을 삭제할 수 있는 기능을 제공받는다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 1.2 추진 배경 및 필요성


대학 생활은 단순히 학문적 지식을 쌓는 곳을 넘어, 다양한 사람들과의 소통과 협업을 통해 사회에 첫발을 내딛기 위한 중요한 과정이다. 이 과정에서 학과나 전공별 소통은 학생들이 직면한 공통의 고민을 해결하고, 필요한 정보를 공유하는 데 있어 필수적인 역할을 한다. 하지만 현실은 대학생들이 각 학과나 전공별로 소통할 수 있는 창구가 매우 부족하다는 것이다. 많은 학생들이 같은 학교 내의 사람들과의 소통에 부담을 느끼거나, 그 한계를 경험하고 있다.

기존에 존재하는 서비스들, 예를 들어 에브리타임은 학교별로 소통하는 기능을 제공하지만, 실질적으로 학과나 전공별 깊이 있는 소통에는 한계가 있다. 이러한 서비스들은 대체로 학교 생활 전반에 대한 정보를 공유하는 데 초점을 맞추고 있으며, 특정 학과나 전공에 대한 세부적인 정보나 고민 상담 등을 지원하는 데는 부족함이 있다.

이러한 상황에서, 전국의 대학생들이 자신의 학과나 전공에 대해 자유롭게 정보를 주고받고, 고민이나 경험을 공유할 수 있는 새로운 플랫폼의 필요하다고 생각되었다.

따라서, 우리는 전국의 대학생들이 사용할 수 있는 학과별 커뮤니티 어플리케이션을 개발하기로 결정했다. 이 어플리케이션은 사용자가 자신의 학과를 선택하여 가입함으로써, 같은 학과나 전공을 가진 학생들과의 소통을 가능하게 한다. 또한, 학과별 맞춤 정보 공유, 스터디 그룹 형성, 취업 정보 공유, 그리고 학과 관련 고민 상담 등 다양한 기능을 제공하여, 학생들이 학과 생활의 질을 향상시킬 수 있도록 지원한다.


이러한 서비스는 학생들에게 더욱 풍부하고 다양한 학과 생활을 경험할 수 있는 기회를 제공함으로써, 학과 내 소통의 폭을 넓히고, 학생들 간의 유대감을 강화하는 데 기여할 것이다. 최종적으로는 학과 커뮤니티를 강화하고, 대학 생활 전반에 긍정적인 영향을 미치는 것을 목표로 하고 있다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 2 개발 내용 및 결과물

### 2.1 목표

디클은 전국 대학생들이 학과를 중심으로 모여 소통할 수 있는 학과별 커뮤니티 어플이다. 기존 에브리타임과 같은 커뮤니티는 소속 학교를 중심으로 모이므로, 같은 전공 분야에 있는 다양한 학교의 학생들과는 소통이 어렵다는 문제가 있다. 또한 에브리타임 내의 학과별 게시판도 활성화 되어있는 일부 학과들 외에는 사실상 사용하는 사람이 적다. 따라서 전국의 모든 대학교 재학생들이 학과별로 모여 정보를 공유하고 자유롭게 소통할 수 있는 커뮤니티를 개발하고자 한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 2.2 연구/개발 내용 및 결과물

### 2.2.1 연구/개발 내용

#### 1. 학교 이메일 인증을 통한 회원가입 및 로그인

전국 401개 대학교의 이메일을 수집하여 SUFFIX를 데이터베이스에 저장했다. 따라서 이메일을 통해 회원가입 시, 데이터베이스에서 조회가 가능한 401개 대학에 포함되어 있다면 정상적으로 인증 메일을 발송해 회원가입 절차를 따르게 된다.

#### 2. 전공 선택

커리어넷(<https://www.career.go.kr/cnet/front/base/major/FunivMajorList.do>) 에서 조회 가능한 학과들을 디클 서비스에 활용한다.

#### 3. 전공 관심사에 따른 카테고리 분류


게시글 객체를 데이터베이스에 저장할 때, 카테고리 필드를 추가하여 쿼리를 통해 원하는 카테고리의 게시글을 검색하도록 구현했다.

#### 4. 게시글 CRUD, 마이페이지

스프링 부트 및 JPA를 활용해 데이터베이스와 객체간 매핑 문제를 해결했고 Controller-Service-Repository 레이어로 나누어 CRUD 및 마이페이지 기능을 구현했다.

#### 5. 알림 기능(추가)

SSE 기술을 활용해 사용자가 접속 시 실시간 알림을 받을 수 있도록 한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 2.2.2 시스템 기능 요구사항

### 1. 회원가입 관련(완료)

- 1.1. 학교 이메일 인증 기능(완료)
- 1.2. 회원가입 기능(완료)
- 1.3. JWT 기반 API 요청(완료)
- 1.4. 아이디 및 비밀번호 찾기(완료)

### 2. 게시글 CRUD(완료)

- 1.1. 게시글 페이지네이션 조회(완료)
- 1.2. 게시글 작성 및 이미지 업로드(완료)
- 1.3. 게시글 삭제(완료)
- 1.4. 게시글 수정 및 이미지 수정(완료)
- 1.5. 게시글 검색 기능(완료)

### 3. 게시글 상세(완료)

- 3.1. 댓글 CRUD(완료)
- 3.2. 대댓글 CRUD(완료)
- 3.3. 게시글 좋아요 기능(완료)
- 3.4. 스크랩 기능(완료)

### 4. 마이 페이지(완료)


- 4.1. 내가 작성한 글 조회(완료)
- 4.2. 내가 댓글단 글 조회(완료)
- 4.3. 내가 스크랩한 글 조회(완료)
- 4.4. 전공 변경 기능(완료)
- 4.5. 비밀번호 변경(완료)
- 4.6. 회원 탈퇴(완료)

### 5. 알림 기능(완료)

### 6. 신고 기능(완료)

- 6.1. 신고 누적 시 게시글, 댓글, 대댓글 차단 기능(완료)
- 6.2. 신고된 게시글, 댓글, 대댓글이 있는 사용자 Create 기능 일시정지(완료)

### 7. 차단 기능(완료)

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

### 2.2.3 시스템 비기능(품질) 요구사항

#### 1. 고가용성(달성)

AWS 두 가용영역(ap-northeast-2a, ap-northeast-2b)에 배포한 이미지를 ECS Task로 실행하여 고가용성을 달성함

#### 2. 확장성(달성)

컨테이너의 CPU와 Memory 사용률이 70%를 넘어갈 시, Autoscaling으로 추가적인 컨테이너가 실행되어 확장성을 달성함

#### 3. 보안(달성)

- 1) 로직 서버를 AWS private subnet에 배치하여 외부에서 로직 서버에 함부로 접근할 수 없게 함.
- 2) 사용자 암호를 해시하여 데이터베이스에 저장함.
- 3) JWT 토큰을 사용해 사용자 인증 메커니즘을 구현함.
- 4) HTTPS 설정을 통해 보안성을 높임.

#### 4. 성능(달성)

- 1) 게시글을 불러오는 API 호출 시, 페이지네이션 및 batch size 설정으로 발생하는 쿼리 횟수를 줄여 성능 최적화를 달성함.
- 2) 코틀린 코루틴을 활용해 응답 시간을 최적화함.


#### 5. 유지관리(달성)

- 1) 프론트엔드와 백엔드 협업 시, Swagger를 통해 API 문서 자동화를 달성함.
- 2) 발생하는 API에 대해 AOP를 적용해 로그를 확인하고 AWS CloudWatch를 통해 조회 가능함.
- 3) github action을 활용해 CI/CD를 구축함

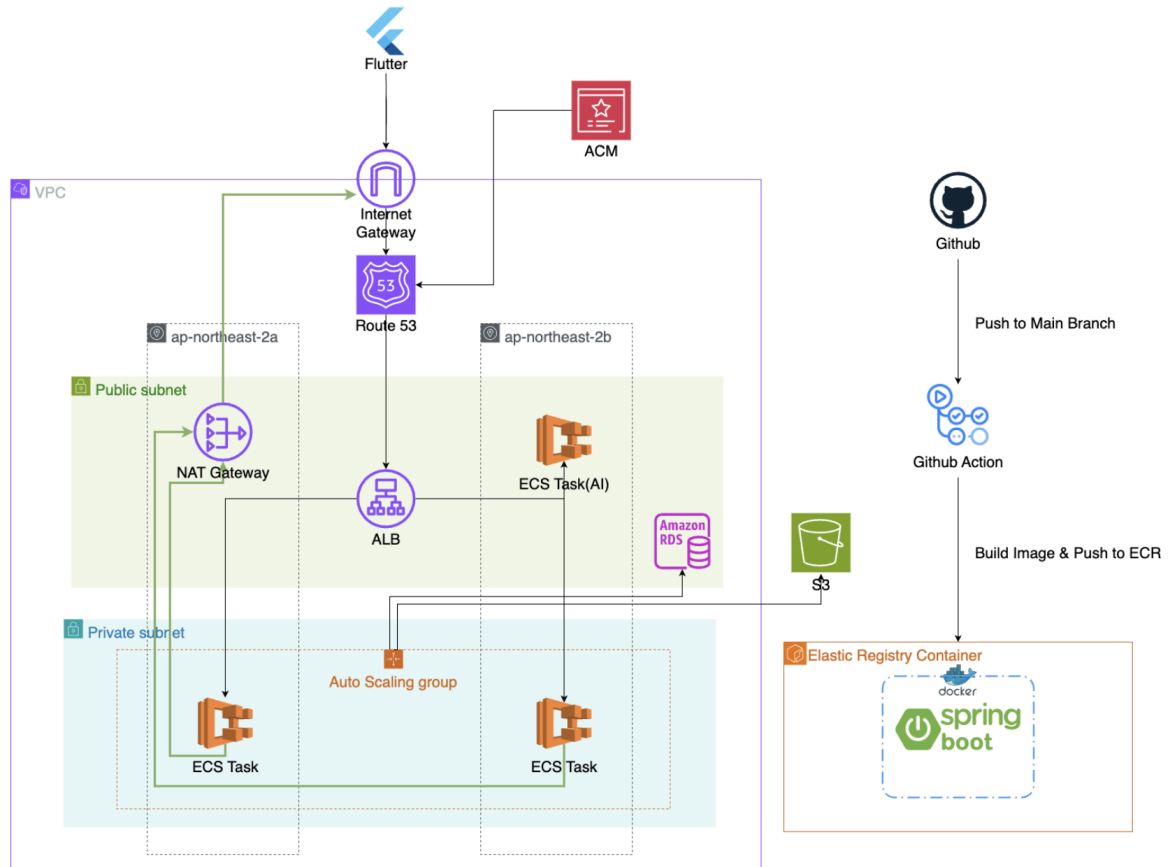
#### 6. 품질(달성)

- 1) 테스트 코드 작성을 통해 코드 품질을 높임.
- 2) 코드 리뷰 및 리팩토링을 통해 코드 가독성을 높임.




 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 2.2.4 시스템 구조 및 설계도



보안성을 고려하기 위해 VPC를 public subnet과 private subnet으로 구분했다. public subnet에는 private subnet의 트래픽을 처리할 수 있는 NAT Gateway와 사용자의 데이터를 저장하기 위한 데이터베이스를 배치했다. 사용자로부터 요청이 발생하면 로드밸런서에 정의한 Listener Rule에 따라 적절한 Target Group으로 트래픽을 포워딩 한다. 마지막으로 HA와 Scalability를 고려하기 위해 두 개의 가용영역에 Autoscale 기능을 활용하여 각각 ECS Task를 실행한다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 2.2.5 활용/개발된 기술

### AI



Python



Scikit learn

### Frontend



Dart



Flutter



Riverpod



Dio

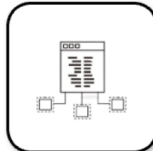
### Backend



Kotlin



Spring Boot



JPA



LINE

Kotlin jdsl



Swagger



MySQL

### Infrastructure



Terraform



ECS



Fargate



S3



Docker




Github action

#### AI

비속어 필터링 기능 구현을 위해 파이썬의 사이킷런 라이브러리를 활용했다. 욕설 유무에 해당하는 y값이 0,1 임을 고려해 로지스틱 모델로 기능을 구현했다.

#### Frontend

크로스 플랫폼 앱 개발 플랫폼으로 플러터를 채택했다. 이를 통해 단일한 코드 베이스를 바탕으로 Android, iOS뿐 아니라 PC와 웹 호환성까지 고려했다. 상태관리 라이브러리는 Riverpod, 백엔드와 통신을 위해 Dio 라이브러리를 채택했다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

### Backend

객체지향성을 고려하고 다양한 라이브러리를 지원하는 스프링 부트를 프레임워크로 채택했다. 개발 언어는 Java와 Kotlin 중 Null safety, 함수형 프로그래밍 지원 및 간결성과 가독성 측면에서 우수한 Kotlin을 채택했다. 데이터베이스와 객체간 매핑 문제를 해결하기 위해 JPA를 사용했고 동적 쿼리 작성을 위해 Kotlin JDSL을 사용했다. QueryDSL은 동적 쿼리 작성을 위해 메타데이터를 사용해야하는데 해당 라이브러리는 메타데이터를 생성하지 않고도 동적 쿼리를 작성할 수 있다.

### Infrastructure

배포를 위해 가장 많이 사용하는 AWS를 채택했다. 수동으로 배포하는 대신, 클라우드 및 온프레미스 인프라를 코드로 관리할 수 있는 테라폼을 활용했다. 이를 통해 AWS 리소스 생성, 수정, 삭제를 코드 레벨에서 관리 가능한 이점을 챙겼다. 빌드한 이미지를 실행할 도구로 서버리스 기반 Fargate를 사용했고 container orchestration 도구는 ECS를 사용했다. EKS가 가지는 여러 이점이 존재하지만 비용 및 오버 엔지니어링이라 판단했기 때문이다. 계시글과 함께 업로드하는 이미지는 S3, 도커 이미지 저장소는 ECR Private Registry를 사용했다. 마지막으로 CI/CD를 위해 github action를 사용했는데 이미 github 기반으로 협업하므로 추가적인 외부 플랫폼의 도움을 받지 않아도 되는 장점이 가장 크게 작용했기 때문이다.

## 2.2.6 현실적 제한 요소 및 그 해결 방안

### 비용

좋은 아키텍처를 사용할수록 그만큼의 비용을 지불해야한다. 서버 비용은 매달 15만원으로 추산했고 본격적으로 배포한 4월, 5월의 서버비용은 최대 30만원정도로 예상했다. 학교에서 캡스톤디자인을 위해 인당 10만원씩 총 40만원의 지원을 해주어 해당 금액으로 서버 비용 및 로고 비용을 충당했다.


## 2.2.7 결과물 목록

구글 플레이스토어




앱스토어 (ios)



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 2.3 기대효과 및 활용방안

- 1) 전공 관련 정보 공유 강화  
 학과별로 모인 유저들이 전공 관련 정보를 쉽게 공유할 수 있어, 학업에 도움이 되는 정보를 쉽게 얻을 수 있다.
- 2) 네트워킹 및 협업 기회 제공  
 같은 학과 출신의 사용자끼리의 소통은 졸업 후의 진로 결정이나 협업 기회를 제공한다.
- 3) 자기 계발 및 취미 활동 공유  
 학과 이외의 관심사나 취미에 관한 정보 또한 공유하고 함께 활동할 수 있는 기회를 제공한다

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

### 3 자기평가

#### 자 체 평 가 서

팀명	7조	성명	윤홍현
프로젝트제목	디클 (Decl)		



자체  
평가

프로젝트 수행 소감:

하이브리드 앱 개발을 위해 **Flutter**를 배운 것은 새로운 기술을 배우는 것에 대한 두려움을 줄이는 데 도움이 되었습니다. 특히 백엔드와의 연결에서...

또한 평소에 팀프로젝트를 하면, 적극적이지 않고 자신이 해야 할 일들을 다 하지 못하는 팀원들과 프로젝트를 진행하여, 거의 대부분을 제가 프로젝트를 했기에, 팀으로 일하는 것을 싫어했습니다. 하지만 이번 캡스톤에서는 저에게 필요한 점들을 짚어주고, 도움을 받을 수 있는 팀원이 함께 하여, 기술을 빠르게 습득하고 개발을 하는데 도움이 되었습니다. 이러한 경험을 통해 같이 일하는 팀원의 책임감이나 능력이 중요하다는 점을 다시한번 느꼈습니다. 저에게 이 점을 적용하여 함께 일하고 싶고, 함께 성장하고 싶은 팀원이 될 수 있도록 노력해야겠다고 느꼈습니다.

개선요구사항:


프로젝트를 하면서, 저 자신에게 개선해야겠다고 느낀 점은 깃 사용이었습니다. 오류를 만났을 때 수정을 해야하는데, 깃 커밋을 쪼개지 않고 계속 작업을 하는 버릇이 있었습니다. 앞으로 이러한 습관을 개선해야겠다고 느꼈습니다.

프로젝트에서 개선할 점으로는 코드를 간략하게 하고, 로딩시간을 줄이기 위한 방법을 더 모색해야겠다고 느꼈습니다.

발전적 제언:

하이브리드 앱 개발 말고 네이티브 앱 개발에서만 할 수 있는 것들이 있기 때문에, 네이티브 앱 개발 중 **IOS**에 대해 더 공부해 봐야겠다고 느꼈습니다.

프로젝트에서 발전해야 할 점으로는 대학교 어플이니까 시간표와 식사표가 같이 있는것이 사용자들이 더 앱을 자주 사용하게 만들 수 있는 기능이기에 추가적으로 있으면 좋겠다고 생각했습니다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 자 체 평 가 서

팀명	7조	성명	윤용배
프로젝트제목	디클		



국민대학교  
컴퓨터공학부  
캡스톤 디자인 I

결과보고서

프로젝트 명

디클

팀 명

7조

Confidential Restricted

Version 1.0

2024-JUN-14

자체  
평가

프로젝트 수행 소감:

이전 프로젝트를 수행하며 개선하거나 새로 적용해보고 싶은 기술들이 여럿 있었습니다. 대표적으로 **Kotlin + Spring, CI/CD**, 도메인 주도 개발, 테스트 커버리지, **Terraform** 활용뿐만 아니라 브랜치 전략과 깃허브를 활용한 협업을 경험해보고 싶었습니다. 캡스톤 프로젝트에서 그 기회가 닿아 새로운 기술들을 적용해볼 수 있었습니다. 피상적으로 알고만 있던 지식들을 실제 코드로 구현하는 과정에서 미처 알지 못했던 여러 문제들을 겪었고 이를 해결하는 과정에서 성장할 수 있어 좋았습니다. 특히 테라폼을 활용해 **AWS**에 배포하는 경험을 통해 **AWS**에서 제공하는 서비스에 대해 많은 공부를 할 수 있어 좋았습니다. 개발부터 앱스토어에 배포하는 전체 프로세스를 체험하고 나아가 이를 유지보수하며 실제 현업에 가까운 경험을 할 수 있는 소중한 기회라고 생각합니다. 개발 외적으로도 좋은 팀원들과 의견 공유와 토의하는 과정에서도 개인적 성장을 이루었다 생각합니다.


개선요구사항:

모든 부분에서 클린 코드 원칙을 지키며 코드를 작성하고 싶었지만 더 나은 코드 작성 방법을 알지 못해 어쩔 수 없이 원칙을 어겼던 적이 있습니다. 기회가 된다면 이런 부분도 리팩토링 하고 싶습니다.

발전적 제언:

협업 프로세스를 개선해 매주 회의를 갖는 대신 매일 짧은 회의를 가져 모든 팀원이 동일한 정보를 팔로우업 할 수 있도록 한다.



 국민대학교 컴퓨터공학부 캠프스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 자 체 평 가 서

팀명	Decl	성명	김동윤
프로젝트제목	Decl		



자체  
평가

프로젝트 수행 소감:

이번 프로젝트는 제가 백엔드 개발자로 참여하는 두 번째 프로젝트였습니다. 백엔드 개발에 대한 이해도가 여전히 부족하다고 느끼는 가운데, 이번 프로젝트를 통해 처음으로 다루어 보는 프레임워크와 프로그래밍 언어에 도전했습니다. 프로젝트 시작 전, 저는 새로운 기술 습득에 많은 시간을 할애했습니다. 이 과정에서 기본 개념을 익히는 것뿐만 아니라, 다양한 개발 도구와 라이브러리에 대한 이해도를 높이기 위해 노력했습니다.

프로젝트를 진행하면서, 같은 백엔드 포지션에 있는 팀원으로부터 많은 것을 배울 수 있었습니다. 그들의 경험과 지식 덕분에 개발 과정 중 발생할 수 있는 다양한 문제들을 쉽게 해결할 수 있었습니다. 특히, 새로운 프레임워크와 언어를 사용함에도 불구하고, 프로젝트를 성공적으로 마무리할 수 있었던 것은 큰 성취감을 안겨주었습니다. 이 경험은 저에게 많은 자신감을 주었고, 앞으로의 개발자 생활에 긍정적인 영향을 미칠 것입니다.

그러나, 프로젝트의 개발 부분 외에 인프라스트럭처 설계와 프로젝트 세팅에 직접 참여하지 못한 것은 아쉬움으로 남습니다. 이 부분은 프로젝트의 성공적인 진행을 위해 매우 중요한 요소임에도 불구하고, 저는 이번 기회에 그에 대한 경험을 충분히 쌓지 못했습니다. 하지만, 이러한 아쉬움도 팀원으로부터 관련 지식을 배울 수 있는 기회가 되었습니다. 이 경험을 바탕으로, 다음 프로젝트에서는 프로젝트 세팅과 인프라스트럭처 설계를 주도적으로 진행할 수 있는 능력을 갖추고자 합니다.

또한, 이번 프로젝트를 통해 처음으로 앱스토어와 플레이스토어에 앱을 출시하는 경험을 할 수 있었습니다. 실제 사용자들이 저희 앱을 다운로드하고 사용하는 것을 보며, 개발자로서 큰 뿌듯함을 느꼈습니다. 이번 경험은 저에게 실제 사용자를 대상으로 한 애플리케이션 개발의 중요성과 그에 따른 책임감을 일깨워 주었습니다.

앞으로 저는 더 많은 사용자를 확보하고, 높은 트래픽을 관리하며, 애플리케이션의 지속적인 유지보수를 진행하는 경험도 쌓고 싶습니다. 이번 프로젝트가 저에게 많은 것을 가르쳐 준 만큼, 다음 프로젝트에서는 이러한 경험을 바탕으로 더욱 성장한 모습을 보여주고 싶습니다. 프로젝트를 통해 배운 교훈과 경험은 제 개발자 생활의 소중한 자산이 될 것이며, 앞으로 더 좋은 개발자로 성장하는 데 큰 도움이 될 것입니다.

개선요구사항:

1. 인프라스트럭처 설계 지식 강화 및 실습: 프로젝트 초기 단계에서 인프라스트럭처 설계에 더 적극적으로 참여하고 싶습니다. 이를 위해 클라우드 서비스(AWS, GCP 등)에




대한 심화 학습을 진행하며, 인프라스트럭처를 구성하는 다양한 서비스들(예: 컴퓨팅, 스토리지, 네트워킹 등)의 조합과 최적화 방법을 습득할 계획입니다. 또한, 실제 프로젝트에서 인프라 설계를 담당하여, 이론과 실습을 병행하는 경험을 쌓고자 합니다.

2. **대용량 데이터 처리 최적화:** 대규모 트래픽과 대용량 데이터 처리는 현대 애플리케이션 개발에서 중요한 요소입니다. 이에 대응하기 위해, 데이터베이스 쿼리 최적화와 도메인 구조 개선에 주력할 예정입니다. **SQL** 쿼리 튜닝, 인덱싱 전략 수립, **NoSQL** 데이터베이스의 적절한 활용 등을 통해 데이터 조회 성능을 개선하고자 합니다. 또한, 캐싱, 데이터 파티셔닝, 비동기 처리 등의 기법을 도입하여 시스템의 전반적인 성능을 향상시킬 계획입니다.
3. **코드 품질 개선 및 리팩토링:** 코드의 중복을 최소화하고, 가독성과 유지보수성을 높이는 것은 지속 가능한 개발을 위해 필수적입니다. 이를 위해, 리팩토링에 좀 더 시간을 할애하며, 디자인 패턴, 함수형 프로그래밍 등의 원칙을 적용하여 더 깔끔하고 효율적인 코드를 작성할 계획입니다. 코드 리뷰 문화를 활성화하여 팀 내에서의 코드 품질 향상을 도모하고, 지속적인 학습을 통해 새로운 기술과 방법론을 접하며 개발 역량을 강화할 예정입니다.

발전적 제언:

1. **테스트 코드 작성 및 테스트 커버리지 향상:** 프로젝트의 안정성을 확보하기 위해 모든 기능에 대한 충분한 테스트 코드를 작성하는 것이 중요합니다. 이를 통해 코드 변경 시 발생할 수 있는 예기치 않은 오류를 사전에 방지하고, 높은 테스트 커버리지를 유지하여 신뢰성 있는 서비스를 제공할 수 있습니다. 테스트 주도 개발(**TDD**) 방식을 도입하여 개발 과정에서부터 테스트를 중심으로 코드를 작성하는 방법도 고려해볼 수 있습니다.
2. **주기적인 코드 리뷰 실시:** 코드 리뷰는 코드의 품질을 향상시키고, 개발자간의 지식 공유에도 매우 효과적인 방법입니다. 주기적으로 코드 리뷰를 실시하여 서로의 코드에 대한 구체적이고 건설적인 피드백을 제공함으로써, 개선된 코드 작성을 도모하고 개발자의 역량을 함께 향상시킬 수 있습니다. 특히, 신규 기능 개발이나 큰 변경 사항이 있는 경우에는 반드시 코드 리뷰를 통해 검증하는 과정을 거치는 것이 좋습니다.
3. **정기적인 진행 상황 공유 및 계획 토론:** 프로젝트의 진행 상황을 정기적으로 공유하고, 팀원 간의 의견을 나누며, 향후 계획에 대해 논의하는 것은 프로젝트를 계획대로 성공적으로 마치는 데 필수적입니다. 이러한 정기적인 회의를 통해 프로젝트의 목표 달성 상황을 점검하고, 문제가 발생했을 때 신속하게 대처할 수 있는 방안을 모색하며, 미래에 발생할 수 있는 잠재적 문제에 대해 사전에 대비할 수 있습니다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 자 체 평 가 서

팀명	7조 디클(DeCl)	성명	권지아
프로젝트제목	디클		



자체  
평가

프로젝트 수행 소감:

기획부터 개발과 배포까지 매주 2회씩 회의를 하며 정성을 쏟아 준비한 프로젝트가 끝나 정말 뿌듯하다.

기획 단계에서는 서비스 컨셉 및 세부 기획에서부터, 우리 서비스가 유사한 다른 서비스(에브리타임, 캠퍼스픽)와 어떻게 차별화될 수 있는지, 그리고 학기 일정을 고려할 때 팀의 개발 목표를 어디까지 두는 것이 좋을지를 신중하게 고려하여 결정하였고, 업무를 나누어 팀원들 각자의 재능과 흥미에 맞는 역할을 분배하고자 노력하였다.

그리고 개발 단계에서 Flutter를 활용한 프론트엔드 개발을 맡아 수행했다. 로그인, 회원가입, 회원 정보 관리(마이페이지) 및 커뮤니티의 게시판 메인 페이지와 카테고리 필터링, 검색 기능을 맡았고, 화면단 구성부터 페이지네이션 등 동적인 요소들까지 모두 개발하였다. 또한 백엔드와 api 스펙을 적절히 협의하여 프론트엔드 코드와 연동하였다.


또한 AI를 사용한 비속어 필터링 API 개발을 맡아 수행했다. 파이썬과 Scikit-learn, 그리고 로지스틱 회귀 모델을 활용하여 한국어 비속어를 감지하는 머신러닝 모델을 학습시키고, 이 모델을 Fastapi와 연결하여 비속어를 작성하려는 사용자에게 경고 알림을 보내는 api를 개발하였다.

마지막으로 배포 단계에서는 배포를 위해 ios와 안드로이드의 네이티브 코드들을 적절히 수정하여 빌드하고, 이를 앱스토어와 구글 플레이스토어에 올려 여러 번의 심사 과정을 거쳐 배포에 성공했다. 이 과정에서 스토어 등록에 필요한 여러 이미지 및 로고 제작, 영상 제작, 문서 제작, 리젝 대응 등 세부 업무가 많았지만 학기 중 시간을 효율적으로 활용하여 마감 기한 내에 성공적으로 배포를 완료할 수 있었다.


이외에도 팀장으로서 중간 발표 및 최종 발표를 위한 PPT 자료와 발표, 포스터 자료, 팀 보고서 등 문서 작업을 수행하고, 팀의 개발 및 배포 일정을 관리하며 우리 팀의 프로젝트가 기한 내에 원활히 수행될 수 있도록 노력했다.

프로젝트의 진행 과정에서 어려운 문제들을 만날 때 상당히 힘들기도 했지만 모든 팀원들과 함께 노력하여 결국에는 전부 해결해낼 수 있었다. 그리고 이 과정에서 함께 끝까지 노력한다면 어떤 문제라도 이겨낼 수 있다는 자신감을 가질 수 있었고, 팀장으로서의 책임감, 한 학기동안 한명도 빠지지 않고 항상 열심히 참여해준 팀원들에게 감사하는 마음을 키울 수 있었다.

개선요구사항:


 국민대학교 컴퓨터공학부 캠프스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

	<p>추후 실사용자들의 의견을 받아 UI/UX를 더욱 편리하게 개선시킬 수도 있을 것 같다.          또한 업데이트가 있을 경우 강제로 업데이트를 진행하게 할 수 있는 기능을 추가하여          업데이트하는 것도 좋을 것 같다.</p> <p>발전적 제언:</p> <p>캠프스톤 기간이 끝난 이후에도 서비스를 계속 운영하여 실제 사용자들을 받아보고, 서비스를          발전시켜본다면 더욱 좋은 기회가 될 수 있을 것 같다.</p>
--	--

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 4 참고 문헌

번호	종류	제목	출처	발행년도	저자	기타
1	기술문서	Kotlin JDSL	<a href="https://kotlin-jdsl.gitbook.io/docs">https://kotlin-jdsl.gitbook.io/docs</a>		Line	
2	기술문서	Flutter – Dart API docs	<a href="https://api.flutter.dev/index.html">https://api.flutter.dev/index.html</a>		Google	
3	기술문서	Spring	<a href="https://docs.spring.io/spring-framework/reference/index.html">https://docs.spring.io/spring-framework/reference/index.html</a>		VMWare	
4	기술문서	Kotlin	<a href="https://kotlinlang.org/docs/home.html">https://kotlinlang.org/docs/home.html</a>		JetBrains	
5	기술문서	Spring Boot	<a href="https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/">https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/</a>		VMWare	
6	기술문서	Spring Data JPA	<a href="https://docs.spring.io/spring-data/jpa/reference/index.html">https://docs.spring.io/spring-data/jpa/reference/index.html</a>		VMWare	
7	기술문서	Terraform	<a href="https://developer.hashicorp.com/terraform/docs">https://developer.hashicorp.com/terraform/docs</a>		HashiCorp	
8	기술문서	Kotest	<a href="https://kotest.io/docs/framework/framework.html">https://kotest.io/docs/framework/framework.html</a>		Kotest Team	

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 5 부록

### 5.1 사용자 매뉴얼 - frontend

#### (1) 가상 시뮬레이터로 앱을 실행하는 방법

- Prerequisite
  - Flutter 3.13.0
  - Dart 3.1.0
  - 안드로이드 스튜디오
  
- 시뮬레이터 (혹은 에뮬레이터) 실행  
 안드로이드 스튜디오에서 device manager -> virtual -> create device -> 실행
  
- 로컬 실행하기
  - 프론트엔드 파일 경로로 진입  

```
cd frontend
```
  - 패키지 설치  

```
flutter pub get
```
  - 프로젝트 실행  

```
flutter run
```

#### (2) 안드로이드 : apk 설치 방법

파일 경로 frontend -> android -> app > build.gradle

60~73번째 줄 release key 관련 코드 (signingConfigs 와 buildTypes) 주석처리

- APK 생성  


```
flutter build apk --release --target-platform=android-arm64
```

#### (3) IOS : 앱스토어에서 설치 방법

앱스토어 검색창에 학과별 커뮤니티 검색 후 설치

- 앱스토어 URL  
<https://apps.apple.com/kr/app/%EB%94%94%ED%81%B4/id6499332881>



 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 5.2 운영자 매뉴얼 - Backend

- Prerequisite
  - Java 17
  - docker compose
  
- 로컬 MYSQL 설치하기(M1 기준)
  - 백엔드 파일 경로로 진입
 

```
cd backend
```
  
  - `docker-compose`를 데몬으로 실행
 

```
docker-compose up -d
```
  
  - `backend/src/main/resources`에 `application-security.yml` 파일 생성 후 아래 내용 작성
 

```
aws:
  access-key: <YOUR_AWS_ACCESS_KEY>
  secret-key: <YOUR_AWS_SECRET_KEY>

s3:
  bucket: <YOUR_BUCKET_NAME>
  region: "ap-northeast-2"
```
  
  - `backend`에서 아래 명령어 실행
 

```
./gradlew bootRun --args='--spring.profiles.active=local'
```


## 5.3 배포 가이드

- (1) 안드로이드 앱 APK 빌드 방법

```
flutter build apk --release --target-platform=android-arm64
```


- (2) IOS 앱 배포

XCode 와 App Store Connect에 로그인하여, 신규앱 등록을 거쳐야 한다.

 국민대학교 컴퓨터공학부 캠프스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 5.4 디클 매뉴얼

1.1 의 프로젝트 개요를 참고하여 앱의 기능들과 화면 구성을 알아보시길 바랍니다.

 국민대학교 컴퓨터공학부 캡스톤 디자인 I	결과보고서		
	프로젝트 명	디클	
	팀 명	7조	
	Confidential Restricted	Version 1.0	2024-JUN-14

## 5.5 테스트 케이스

대분류	소분류	기능	테스트 방법	기대 결과	테스트 결과
회원가입	학교 이메일 인증 기능	학교 이메일 인증	회원가입에서 학교 이메일(아이디) 입력란에 이메일을 적고, 오른쪽 전송버튼을 클릭한다.	메일로 인증코드가 전송된다.	성공
회원가입	회원가입 기능	회원가입	회원가입에서 이메일 인증을 포함하여 모든 폼을 채운후 하단 회원가입 버튼을 클릭한다.	회원가입이 완료되었다는 메시지와 함께 로그인이 가능해진다.	성공
회원가입	JWT 기반 API 요청	JWT 기반 API 요청	로그인을 한다.	AccessToken, RefreshToken을 저장하여, 자동으로 로그인된다.	성공
회원가입	아이디 및 비밀번호 찾기	아이디 및 비밀번호 찾기	로그인 화면 하단에 아이디 및 비밀번호 찾기를 한다.	아이디 및 비밀번호를 찾을 수 있다.	성공
게시글 CRUD	게시글 페이지네이션 조회	게시글 페이지네이션 조회	메인화면에서 스크롤하여 게시글을 계속 불러온다.	게시글이 빠르게 화면에 나오며, 스크롤하면 게시글을 계속 불러온다.	성공
게시글 CRUD	게시글 작성 및 이미지 업로드	게시글 작성 및 이미지 업로드	메인페이지에서 우측 하단 플러스 버튼을 클릭 -> 게시글 작성 및 하단 이미지 버튼 클릭, 이미지 선택 후 X 버튼을 클릭하여 잘못 선택한 이미지 삭제 가능	게시글에 이미지를 포함하여 업로드된다. 메인페이지에서 작성한 게시글이 보인다.	성공
게시글 CRUD	게시글 삭제	게시글 삭제	해당 게시글 우측 상단 더보기 버튼 클릭 -> 삭제하기, 또는 수정하기 화면에서 우측 하단 삭제하기 버튼 클릭	게시글이 삭제되고, 메인페이지에서 삭제된 게시글이	성공



				없어져 보이지 않게 된다.	
게시글 CRUD	게시글 수정 및 이미지 수정	게시글 수정 및 이미지 수정	해당 게시글 우측 상단 더보기 버튼 클릭 -> 수정하기, 제목, 내용, 이미지를 바꿀 수 있음	게시글을 수정하여 수정된 내용이 반영된 글을 볼 수 있게 된다.	성공
게시글 CRUD	게시글 검색 기능	게시글 검색 기능	메인 페이지에서 우측 하단 플러스 버튼 위에 돋보기 버튼 클릭 -> 검색창에 원하는 글 검색	검색창에 입력된 글과 연관된 글 리스트가 나온다.	성공
게시글 상세	댓글 CRUD	댓글 CRUD	댓글 생성: 해당 게시글의 하단 바를 클릭하면 키보드가 올라옴 -> 내용 작성 후 하단 우측 종이비행기 버튼 클릭 댓글 수정: 댓글 라인 우측에 더보기 버튼 클릭 -> 수정하기 클릭 -> 해당 댓글 색상이 변경됨 -> 입력창에 수정하려는 글로 작성 후 종이비행기 클릭 댓글 삭제: 댓글 라인 우측에 더보기 버튼 클릭 -> 삭제하기 클릭	댓글을 등록, 읽기, 수정, 삭제가 가능하다.	성공
게시글 상세	대댓글 CRUD	대댓글 CRUD	대댓글 생성: 댓글 라인의 메시지 아이콘 클릭 -> 내용 작성 후 종이비행기 클릭 대댓글 수정: 대댓글 라인 우측에 더보기 버튼 클릭 -> 수정하기 클릭 -> 해당 댓글 색상이 변경됨 -> 입력창에 수정하려는 글로 작성 후 종이비행기 클릭 대댓글 삭제: 댓글 라인 우측에 더보기 버튼 클릭 -> 삭제하기 클릭	대댓글을 등록, 읽기, 수정, 삭제가 가능하다.	성공
게시글 상세	게시글 좋아요 기능	게시글 좋아요 기능	게시글 우측 상단 하트 버튼 클릭	하트 클릭 애니메이션이 재생되며 좋아요 수가 올라간다.	성공
게시글 상세	게시글 스크랩 기능	게시글 스크랩 기능	게시글 우측 상단 별 버튼 클릭	색상이 노란색으로 변하고 스크랩 수가 올라간다.	성공



게시글 상세	비속어 필터링	비속어 필터링 기능	위 표를 참고하여 게시글 작성/수정, 댓글 작성/수정, 대댓글 작성/수정하기	비속어가 포함되어 있어 불이익을 받을 수 있다는 안내를 띄움	성공
마이 페이지	내가 작성한 글 조회	내가 작성한 글 조회	메인페이지에서 우측 상단 사람 버튼 클릭 -> 내가 작성한 글 클릭	내가 작성한 글 조회가 된다.	성공
마이 페이지	내가 댓글단 글 조회	내가 댓글단 글 조회	메인페이지에서 우측 상단 사람 버튼 클릭 -> 내가 댓글단 글 클릭	내가 댓글단 글 조회가 된다.	성공
마이 페이지	내가 스크랩한 글 조회	내가 스크랩한 글 조회	메인페이지에서 우측 상단 사람 버튼 클릭 -> 내가 스크랩한 글 클릭	내가 스크랩한 글 조회가 된다.	성공
마이 페이지	전공 변경 기능	전공 변경 기능	메인페이지에서 우측 상단 사람 버튼 클릭 -> 전공 변경하기 클릭	전공이 변경된다.	성공
마이 페이지	비밀번호 변경	비밀번호 변경	메인페이지에서 우측 상단 사람 버튼 클릭 -> 회원 탈퇴하기 클릭	비밀번호가 변경이 된다.	성공
마이 페이지	회원 탈퇴	회원 탈퇴	메인페이지에서 우측 상단 사람 버튼 클릭 -> 회원 탈퇴하기 클릭	회원 탈퇴가 된다.	성공
알림기능	알림기능	알림기능	앱 알림 허용	내 게시글에 댓글, 대댓글이 달리거나, 내 댓글에 대댓글이 달리면 푸시 알림이 온다.	성공
신고 기능	신고 기능	신고 기능	신고하려는 게시글, 댓글, 대댓글 더보기에서 신고하기 버튼 클릭 -> 신고 사유 선택	신고가 된다.	성공
차단 기능	차단기능	차단기능	차단하려는 게시글, 댓글, 대댓글 더보기에서 차단하기 버튼 클릭	차단이 되어 해당 유저의 글이나 댓글, 대댓글이 보이지 않는다.	성공

## 5.6 디클에 대한 기술 문서

4. 의 참고문헌에 포함되어 있다.