GLOABAL SW ENGINEERING

WOOZOO - 물물교환 서비스 어플리케이션

32217072 김도익
32203743 임석범
32194074 정민호
32204292 조민혁

Contents

01. 프로젝트 개요

02. 프로젝트 계획 및 역할 분담

03. 프로젝트 구현

04. 실행 화면

05. 향후 계획 및 마무리

1. 프로젝트 개요

1. 프로젝트 개요

- 모바일, 웹 애플리케이션 사용 급증

  ➔ **다양한 서비스 플랫폼 등장**

- 의류, 신발, 악세서리, 생필품과 같은 상품 중심으로 한 온라인 마켓

  시장은 오프라인 시장 압도

  ➔ **쿠팡, 무신사, 당근마켓**

- 이러한 플랫폼은 거래 위주의 플랫폼으로써 존재

- 물물 교환 서비스에 대한 수요 존재, 그러나 해당 시장 활성화 저조

1. 프로젝트 개요

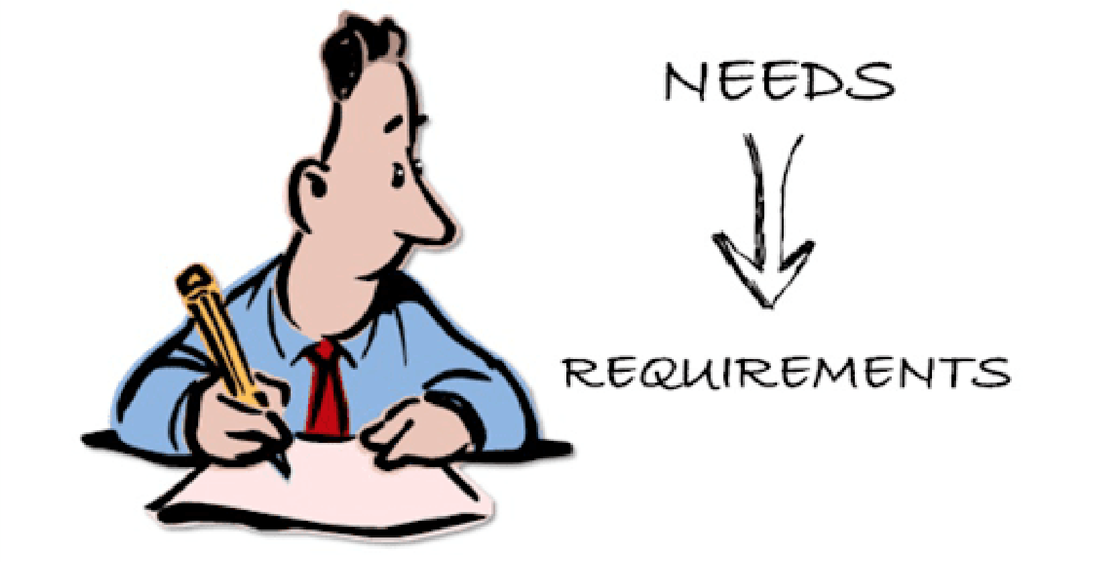
“이에 따라, 물물 교환 서비스 애플리케이션 필요성 증가”

2. 프로젝트 계획 및 역할 분담

2. 프로젝트 계획 및 역할 분담

STEP 1) 요구사항 분석

- 고객 및 이해 관계자와의 기능, 비기능적 요구사항 수집
- 기간: 10월 4일 ~ 10월 11일, 1주
- 산출물: 요구사항 명세서



2. 프로젝트 계획 및 역할 분담

STEP 2) 시스템 설계

- 프로젝트의 전반적인 아키텍처 구체화
- 기간: 10월 12일 ~ 10월 18일, 1주
- 산출물: 설계 문서, UML 다이어그램, 데이터베이스 스키마

2. 프로젝트 계획 및 역할 분담

STEP 3) 구현

- 시스템의 핵심 기능 개발
- 기간: 10월 21일 ~ 11월 22일, 5주
- 산출물: 기본 기능이 포함된 초기 버전 애플리케이션

2. 프로젝트 계획 및 역할 분담

STEP 4) 테스트 및 배포

- 시스템의 안정성과 완성도 검증
- 기간: 11월 25일 ~ 12월 8일, 3주
- 산출물: 애플리케이션 출시 및 배포 완료

2. 프로젝트 계획 및 역할 분담

STEP 5) 최종 검토 및 피드백 반영

- 최종 오류 수정 및 시스템 안정화
- 기간: 12월 9일 ~ 12월 10일, 2일
- 산출물: 최종 사용자 설명서

2. 프로젝트 계획 및 역할 분담

팀원 별 역할

🤪 정민호: Project Manager(PM), Full-Stack Main

🤪 김도익: UI 디자인, Front-End Sub

🤪 임석범: Data Base 설계, Back-End Sub

🤪 조민혁: 보안 설계 담당, Test Code 작성

3. 프로젝트 구현

3. 프로젝트 구현 : 프론트엔드

A. 프론트엔드 특징

- 모듈화된 구조를 통해 UI, 상태 관리, 데이터 모델을 체계적으로 분리하고 유지보수성과 재사용성을 극대화

- 라우팅과 사용자 경험을 정교하게 설계하여 확장성과 효율성을 강조

3. 프로젝트 구현 : 프론트엔드

B. 프론트엔드 세부 기능

이름	기능
model/item	물물교환 앱의 상품 데이터 모델과 관련된 핵심 로직을 담당하며, 상품의 정보, 카테고리, 거래 장소를 체계적으로 관리하고 유효성 검사와 상태 업데이트를 제공.
model/onboarding	회원 인증 및 초기 사용자 등록 과정을 담당하며, 로그인 및 회원가입 데이터를 관리하고 서버와의 통신 및 유효성 검사를 지원.
model/status	앱의 상태관리와 버전 관리를 담당하며, 사용자 활동 상태, 시스템 버전 상태 등을 관리하고 상태 흐름 제어 및 유지보수를 지원.
model/user	사용자 인증, 프로필 관리, 비밀번호 변경 등의 사용자 관련 데이터를 관리하며, 입력값 유효성 검사와 API 통신을 지원.
model/app_state_model	앱 전역 상태(로그인 상태, 초기화 여부 등)를 관리하며, 초기 상태 설정 및 상태 업데이트 기능 제공.
model/Hive_Box_enum	Hive 데이터베이스에서 사용할 박스와 어댑터를 관리하며, 사용자 정보 및 설정 데이터를 영구 저장하고 효율적으로 접근 가능하도록 지원.
model/secure_model	앱의 보안 모델을 정의하며, 민감한 데이터를 안전하게 처리 및 저장하기 위한 로직을 제공.

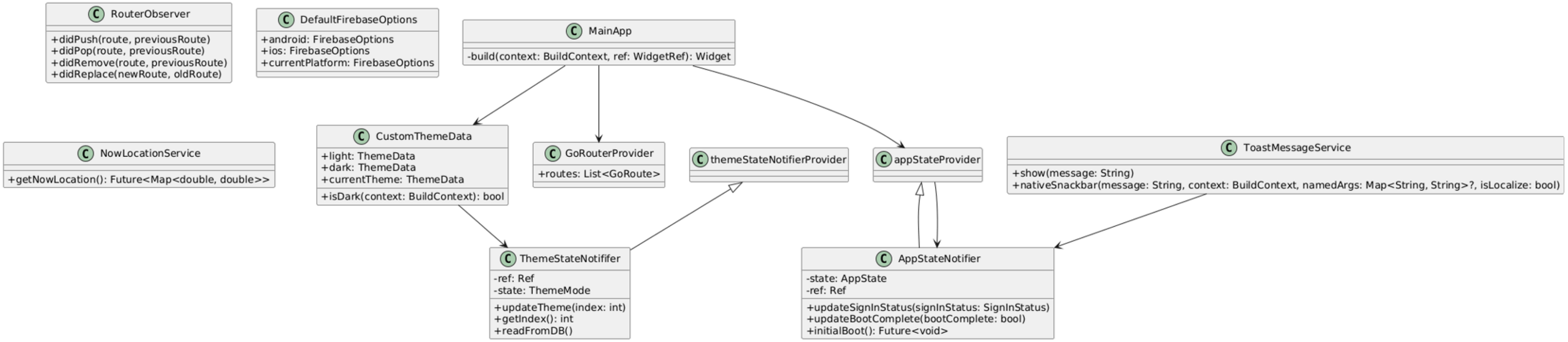
3. 프로젝트 구현 : 프론트엔드

B. 프론트엔드 세부 기능

이름	기능
model/server_state_model	서버 연결 상태를 정의하고, 각 상태에 대응하는 메시지를 제공하여 사용자에게 서버 상태를 명확히 전달.
model/text_field_model	텍스트 필드의 유효성 검사와 입력 데이터를 관리하는 기본 믹스인으로, 다양한 입력값 관리 및 상태 처리 로직 제공.
page/error	앱에서 발생하는 에러 상황을 사용자에게 안내하며, 라우팅 에러 및 서버 에러를 처리하여 문제 해결 방법을 제공.
page/home	물물교환 앱의 핵심 사용자 경험을 제공하는 주요 화면과 기능을 포함하며, 네비게이션 바와 주요 탭을 통해 직관적 화면 전환을 지원.
provider	중앙 상태 관리를 통해 로직과 화면을 분리하며, 비동기 작업 관리 및 상태 업데이트를 통해 앱의 안정성과 성능을 향상.
service	디버깅, 로깅, 이미지 처리, 위치 기반 서비스 등 다양한 기능을 제공하여 앱의 안정성과 사용자 경험을 개선.
theme	앱의 시각적 스타일과 UX를 체계적으로 관리하며, 디자인 일관성과 사용자 친화적 인터페이스를 제공.

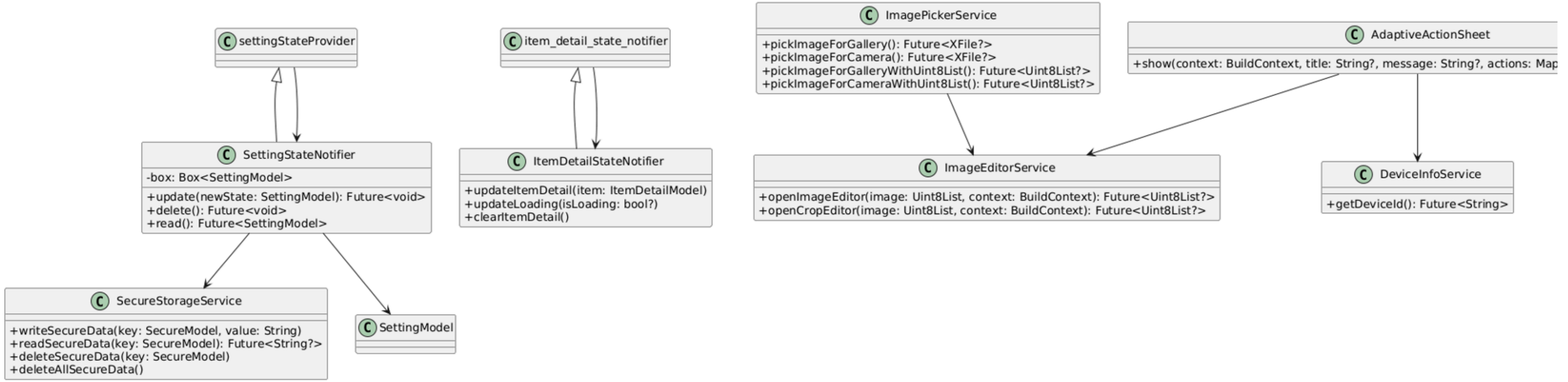
3. 프로젝트 구현 : 프론트엔드

C. 프론트엔드 UML



3. 프로젝트 구현 : 프론트엔드

C. 프론트엔드 UML



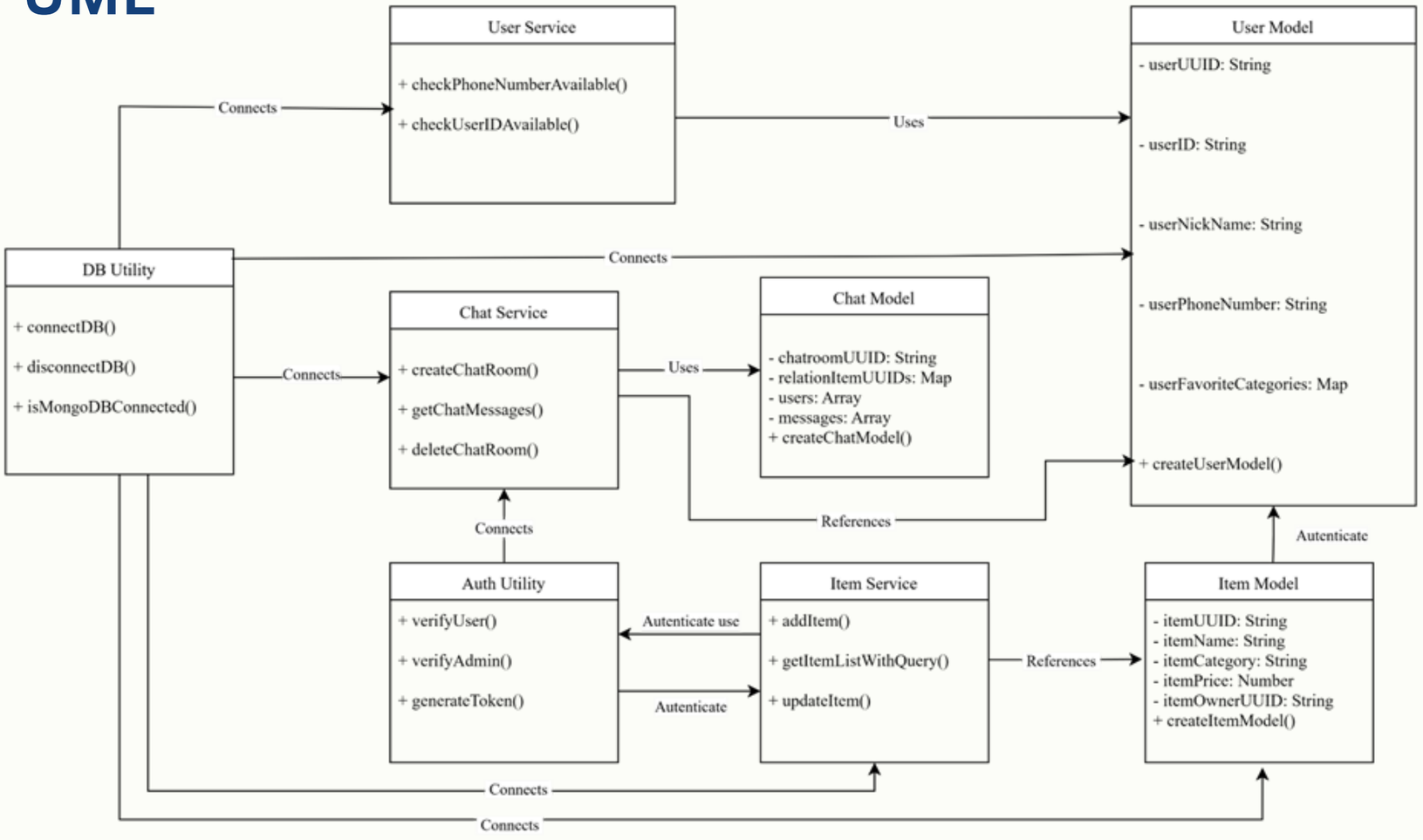
3. 프로젝트 구현 : 백엔드

✓ A. 백엔드 특징

- 모노레포 기반 설계: 코드 재사용성과 개발 생산성을 높임, 서비스별 독립성을 유지하는 멀티 서버 아키텍처를 채택.
- 멀티 서버 아키텍처: 도메인 기반으로 분리된 서버(server_chat, server_item, server_user) 간 역할 분담으로 성능, 안정성, 확장성 강화.
- 공통 모듈 활용: shared 디렉토리를 통해 공통 모델, 유틸리티를 관리해 중복 제거 및 유지보수 효율성 향상.
- CI/CD 파이프라인: GitHub Actions를 활용한 자동화된 빌드, 테스트, 배포 프로세스로 개발 체계화.
- 보안 강화: 환경 변수 암호화 및 무결성 검증(.env.enc)으로 민감 정보 보호 및 안전한 운영 환경 제공.

3. 프로젝트 구현 : 백엔드

B. 백엔드 UML



3. 프로젝트 구현 : 백엔드

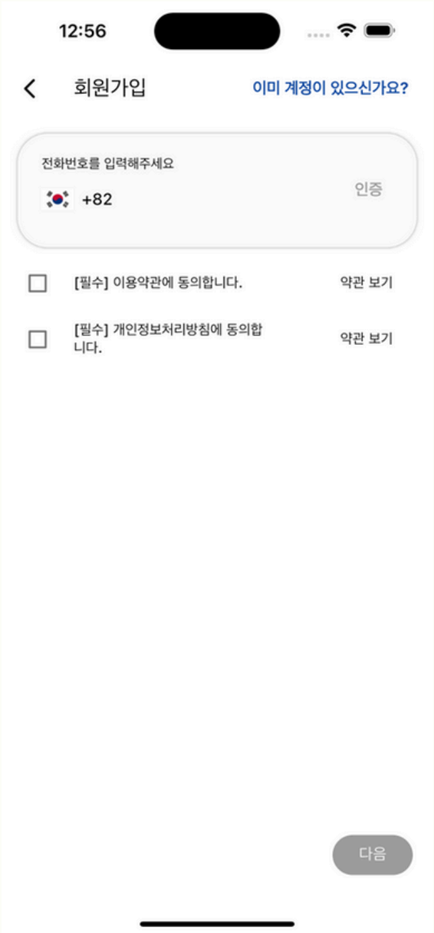
C. 백엔드 세부 사항 및 기능

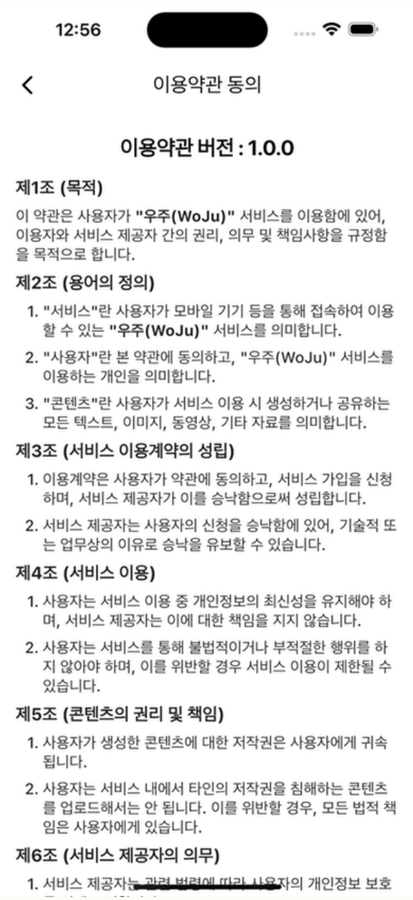
이름	기능
service_chat	사용자 간 실시간 소통을 위한 채팅방 생성, 메시지 전송, 조회 및 삭제 기능 제공. 중복 채팅방 검증, 메시지 읽음 상태 관리, 데이터베이스(chatDB) 저장 및 관리.
service_item	사용자 물품의 등록, 수정, 조회, 삭제, 매칭 기능 제공. 필터링, 정렬, 선호도 기반 추천 지원 및 데이터 검증 수행. 데이터베이스(itemDB)와 통합.
service_user	회원가입, 로그인, 정보 수정 및 인증 관리. 비밀번호 암호화, JWT 토큰 발급 및 사용자 데이터 검증 수행. 데이터베이스(userDB) 연계 및 인증 로직 관리.
Auth Utility	사용자 및 관리자의 인증과 권한 검증. JWT 기반 토큰 생성 및 관리, 만료된 토큰과 부적절한 요청 처리. 서비스별 공통 인증 로직 제공.
DB Utility	MongoDB와의 연결 및 상태 관리. 데이터베이스 연결 및 해제, 테스트 환경 지원, 연결 상태 확인 및 환경 변수 기반 설정 관리.
Crypto Utility	데이터 보호와 암호화 관리. bcrypt로 비밀번호 해시화 및 AES-256-CBC로 민감한 데이터 암호화/복호화. 데이터 보안 유지 및 유출 방지.
Logger Utility	서버 이벤트와 오류 로깅. HTTP 요청 로그 기록, 디버깅 및 운영 모니터링 지원. 로그 메시지의 타임스탬프, IP 포함 및 morgan 라이브러리 활용.
Response Model	API 호출의 일관된 응답 구조 제공. 실패 원인을 명확히 전달하기 위한 FailureReason Enum 정의. 사용자에게 명확한 피드백 제공.

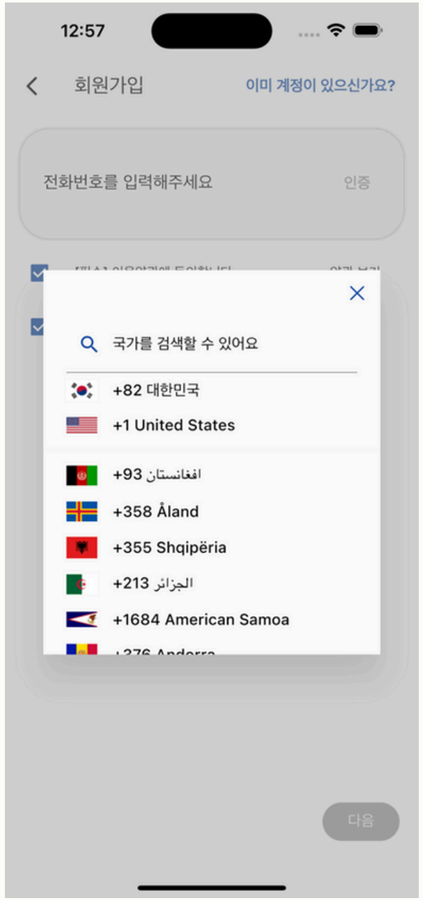
4. 실행화면

4. 실행 화면

A-1. 회원 가입

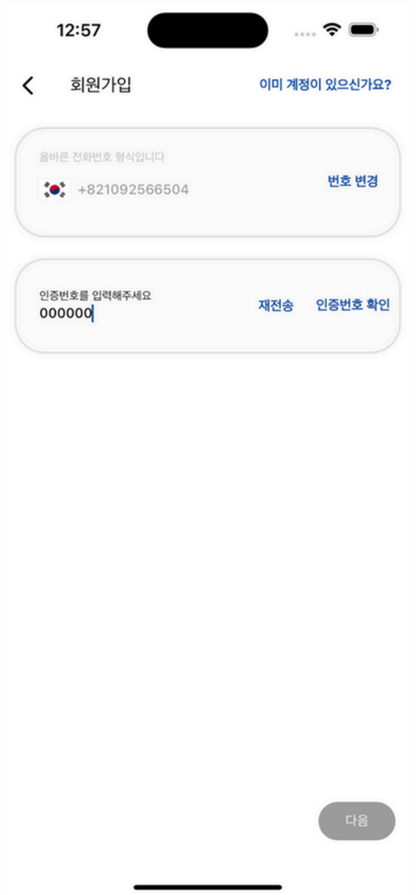


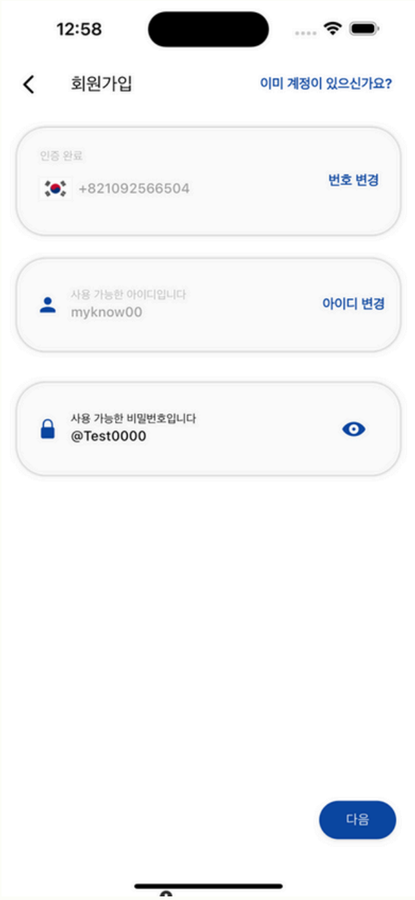


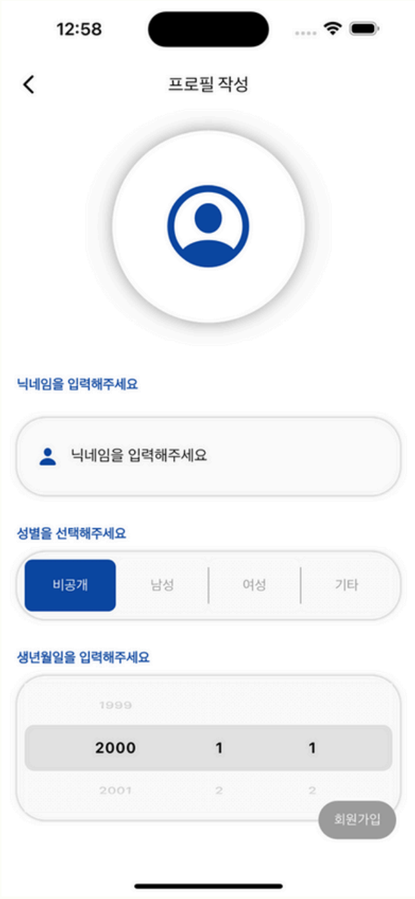


4. 실행 화면

A-2. 회원 가입

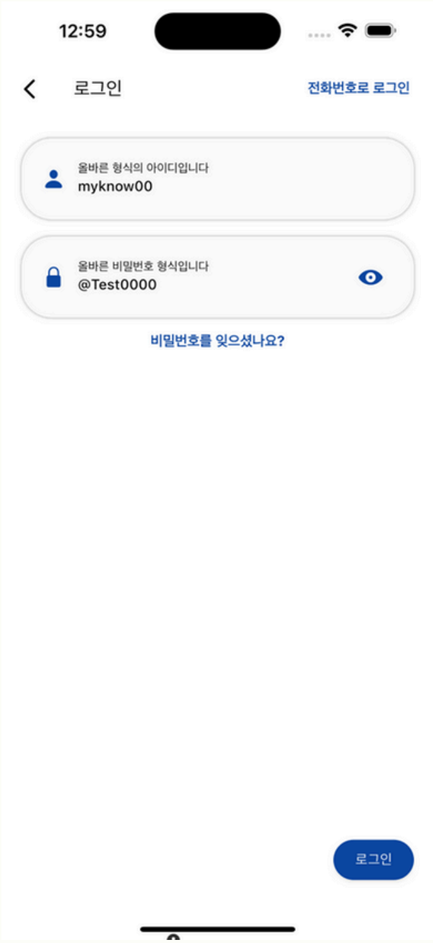


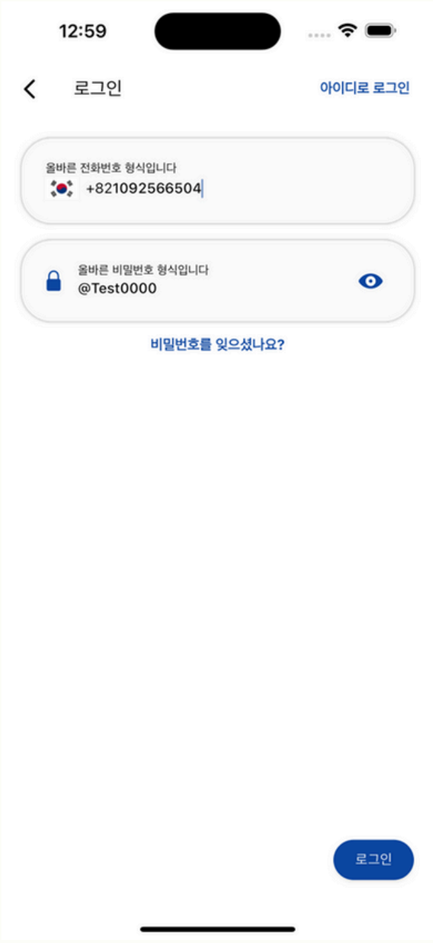




4. 실행 화면

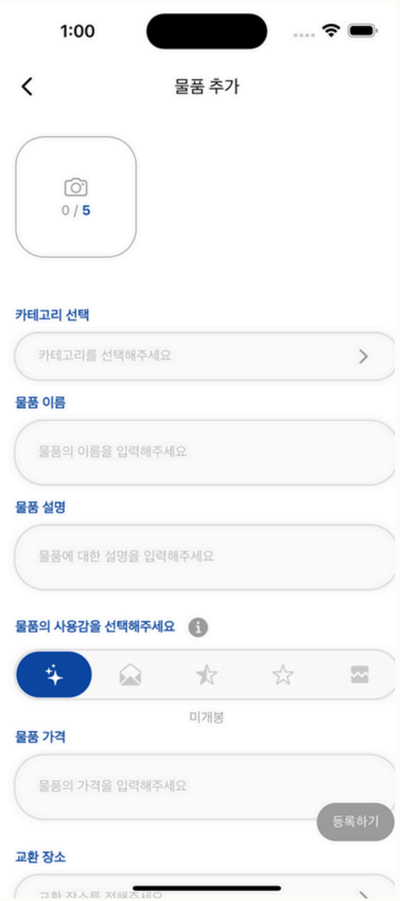
B. 로그인


12:59
로그인
전화번호로 로그인
올바른 형식의 아이디입니다
myknow00
올바른 비밀번호 형식입니다
@Test0000
비밀번호를 잊으셨나요?
로그인

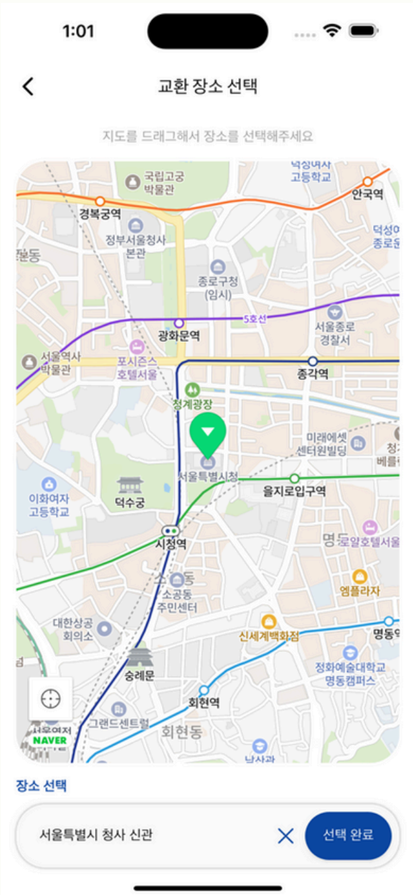

12:59
로그인
아이디로 로그인
올바른 전화번호 형식입니다
+821092566504
올바른 비밀번호 형식입니다
@Test0000
비밀번호를 잊으셨나요?
로그인

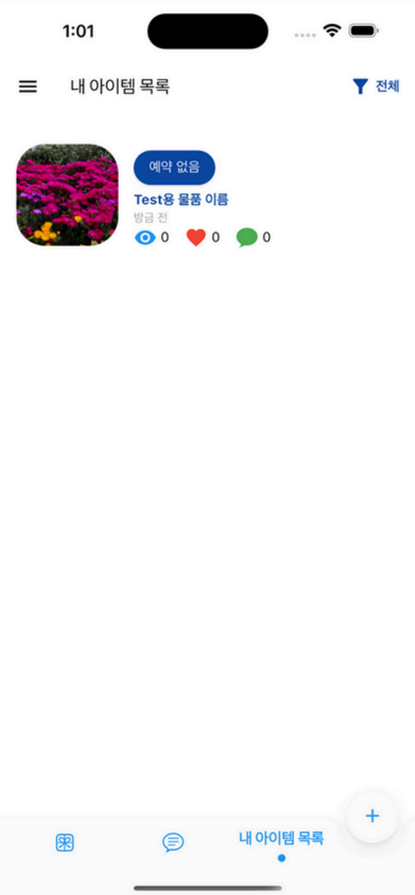
4. 실행 화면

C. 물품 추가


1:00
물품 추가
0 / 5
카테고리 선택
카테고리를 선택해주세요
물품 이름
물품의 이름을 입력해주세요
물품 설명
물품에 대한 설명을 입력해주세요
물품의 사용감을 선택해주세요
미개봉
물품 가격
물품의 가격을 입력해주세요
등록하기
교환 장소

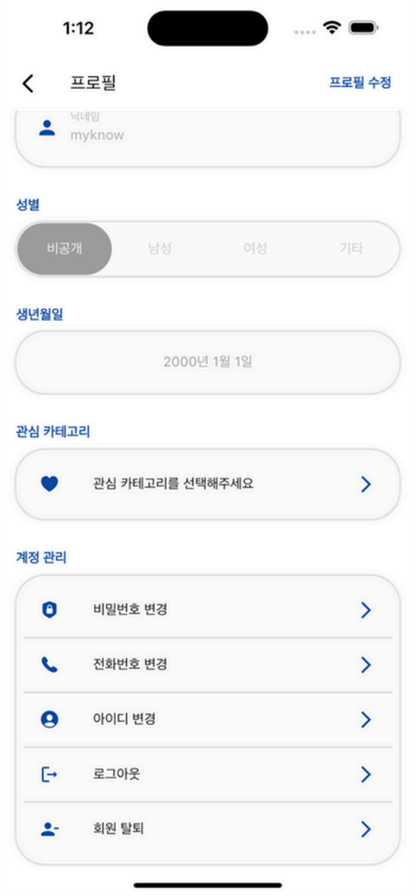
1:00
카테고리 선택
카테고리를 짧게 누르면 선택, 길게 누르면 설명을 볼 수 있어요
전자제품
가구
생활용품
패션
도서
유아용품
뷰티
스포츠용품
취미용품
앨범
자동차용품
티켓


1:01
교환 장소 선택
지도를 드래그해서 장소를 선택해주세요
경복궁역
안국역
광화문역
종각역
을지로입구역
시청역
회현역
장소 선택
서울특별시 청사 신관
선택 완료


1:01
내 아이템 목록
전체
예약 없음
Test용 물품 이름
방금 전
0
0
0
내 아이템 목록

4. 실행 화면

D. 프로필 작성 및 관심 분야 설정

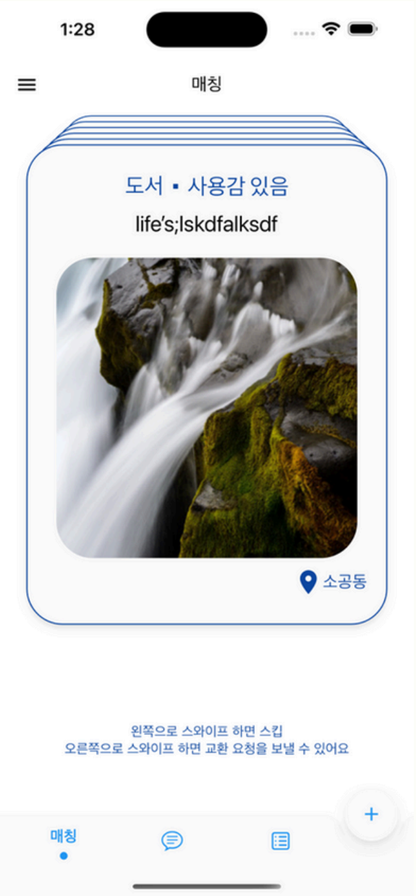


4. 실행 화면

E. 교환 매칭

1:23
매칭
상대방과 교환 할 내 아이템을 선택해주세요
테스트용 물품 이름 1
전자제품
테스트용 물품 이름 2
생활용품
매칭

1:24
아이템 상세 정보
대표 이미지
도서 • 사용감 있음
life's;lskdfalksdf
4시간 전
alskdfajlsdfalksdfjalsdfj
7 0 0
교환 장소
메뉴 열기


1:28
매칭
도서 • 사용감 있음
life's;lskdfalksdf
소공동
왼쪽으로 스와이프 하면 스킵
오른쪽으로 스와이프 하면 교환 요청을 보낼 수 있어요
매칭

5. 향후 계획 및 마무리

5. 향후 계획 및 마무리

- 팀워크와 협업의 경험
  - 팀원 간 역할 분담 및 코드 리뷰를 통해 협업 효율성 극대화.
  - 공통 모듈 재사용과 모노레포 기반 구조로 유지보수성과 확장성 확보.
  - CI/CD 파이프라인과 Git을 활용하여 자동화된 테스트 및 버전 관리 수행.
- 기술적 성장과 문제 해결
  - 코드 작성 과정에서 발견된 이슈를 협의를 통해 해결하며 개발 역량 향상.
  - 버전 관리 및 테스트 자동화를 경험하며 실무 수준의 프로젝트 관리 능력 강화.
- 프로젝트의 의의
  - 단순한 애플리케이션 개발을 넘어 협업, 기술적 성장, 사용자 서비스 설계를 경험.
  - 향후 다양한 분야에서 활용 가능한 개발 역량을 확장하는 기반 마련.
- 향후 계획
  - AI 기반 검색 기능: 사용자가 최적의 물품을 빠르게 찾을 수 있도록 지원.
  - AI 이미지 분류: 물품 이미지를 자동으로 분류하고 적합한 카테고리 추천.
  - AI 챗봇 고객센터: LLM을 활용한 사용자 지원으로 불편함 최소화.

Thank You