우 성 욱

2025-BD02-05-프로그램소스

프로그램 소스

[웹 페이지 연락처 프로그램 구현]

|  |  |  |  |
| --- | --- | --- | --- |
| 제/개정 이력 | | | |
|  | | | |
| 문서 명 | 프로그램 소스 | | |
|  | | | |
| 연번 | 작성일 | 제/개정사항 | 작성자 |
| 1 | 2025-02-20 | 최초 작성 | 우 성 욱 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

목 차

[1. 서론 3](#_Toc190882793)

[1.1 설계 목적 3](#_Toc190882794)

[1.2 설계 범위 3](#_Toc190882795)

[2. 시스템 아키텍처 4](#_Toc190882796)

[2.1 시스템 구성도 4](#_Toc190882797)

[2.2 주요 시스템 구성 요소 4](#_Toc190882798)

[3. 데이터베이스 설계 5](#_Toc190882799)

[3.1 데이터베이스 개요 5](#_Toc190882800)

[3.2 ERD (Entity Relationship Diagram) 5](#_Toc190882801)

[3.3 테이블 설계 6](#_Toc190882802)

[4. 사용자 인터페이스 설계 8](#_Toc190882803)

[4.1 화면 설계 8](#_Toc190882804)

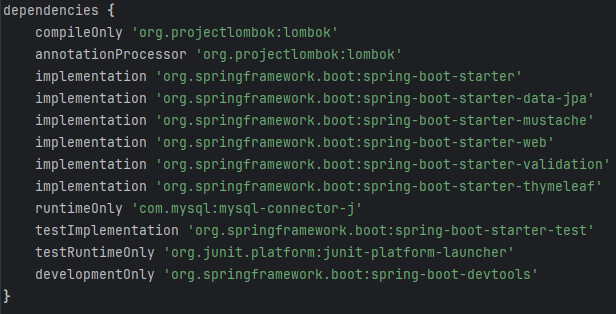
# 프로젝트 설정

## 환경 설정

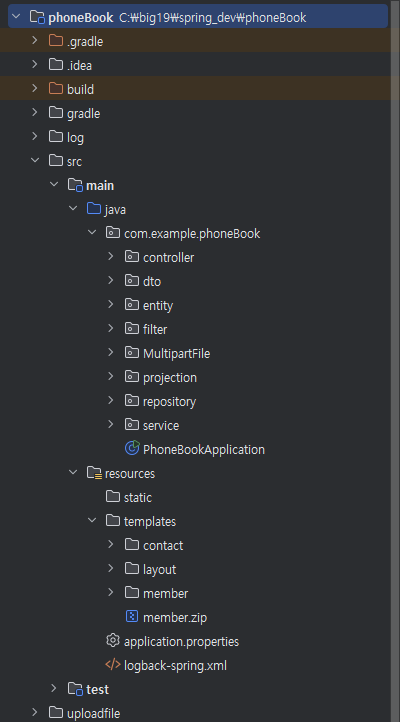
* 서버: Spring Boot를 사용하여 서버 설정
* 데이터베이스: Ubuntu 22.04에 Docker(v. 27.4.1)로 MySQL 8.0.17 설치
* 라이브러리: Gradle을 사용하여 의존성 관리

## 의존성 관리

* build.gradle에 필요한 라이브러리 추가



## 디렉토리 구조



# 회원 관리

## 회원 가입(회원 정보 입력, DB저장)

* MemberDTO: 회원 가입 정보를 담는 DTO 생성
* MemberEntity: DB와 DTO를 연계하는 entity 클래스
* MemberService: 회원 가입 비즈니스 로직을 처리하는 서비스 클래스
* MemberRepository: JPA 및 Crud를 사용하여 회원 정보를 DB에 저장하는 리포지토리 클래스

## 로그인 기능(아이디/비밀번호 확인)

* MemberService: SHA-256과 Salt를 사용한 비밀번호 암호화 방식을 만들어 아이디 및 비밀번호 로그인 확인
* 세션 관리: 세션 기반 로그인 설정

# 연락처 관리

## 연락처 추가(이름, 전화번호, 주소, 그룹, 사진 입력)

* MembersContactsDTO: 연락처 정보를 담는 DTO 클래스
* MembersContactsEntity: DB와 DTO를 연계하는 Entity 클래스
* PhoneService: 연락처를 추가하는 비즈니스 로직을 처리하는 서비스
* PhoneRepository: 연락처 정보를 DB에 저장하는 리포지토리 클래스

## 연락처 수정

* PhoneService: 연락처를 수정하는 비즈니스 로직을 처리하는 서비스
* PhoneRepository: 연락처 정보를 DB에 수정하여 저장하는 리포지토리 클래스

## 연락처 삭제

* PhoneService: 연락처를 삭제하는 비즈니스 로직을 처리하는 서비스
* PhoneRepository: 연락처 정보를 DB에 삭제하는 리포지토리 클래스

## 연락처 검색(이름)

* JPA 쿼리 메서드: MemberContactFourthRepository에 findByName 메서드를 추가하여 검색 기능 구현

# 파일 처리

## 사진 업로드

* 파일 업로드: Spring Boot의 MultipartFile을 사용하여 파일 업로드 처리
* 파일 저장: 서버의 특정 디렉토리에 사진 파일을 저장하고, 파일 경로를 DB에 저장

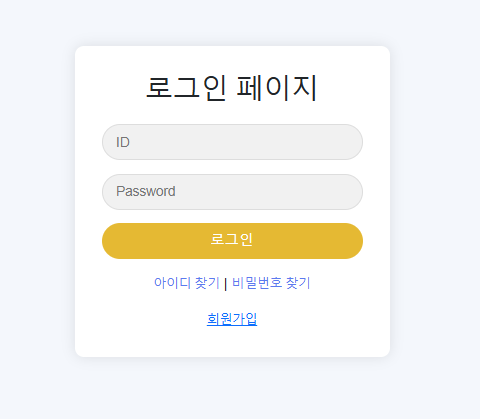
# 보안

## 비밀번호 암호화(회원가입으로 인한 비밀번호 저장 시 암호화)

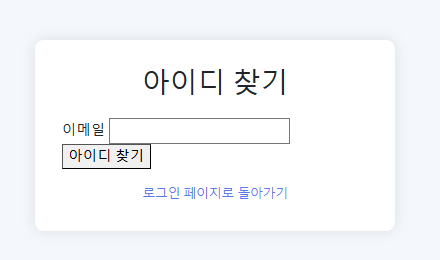
* SHA-256과 Salt를 사용한 비밀번호 암호화 방식 사용

# UI/UX

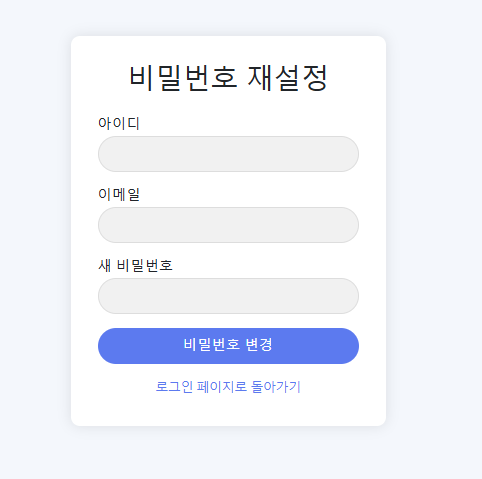
## 로그인 화면



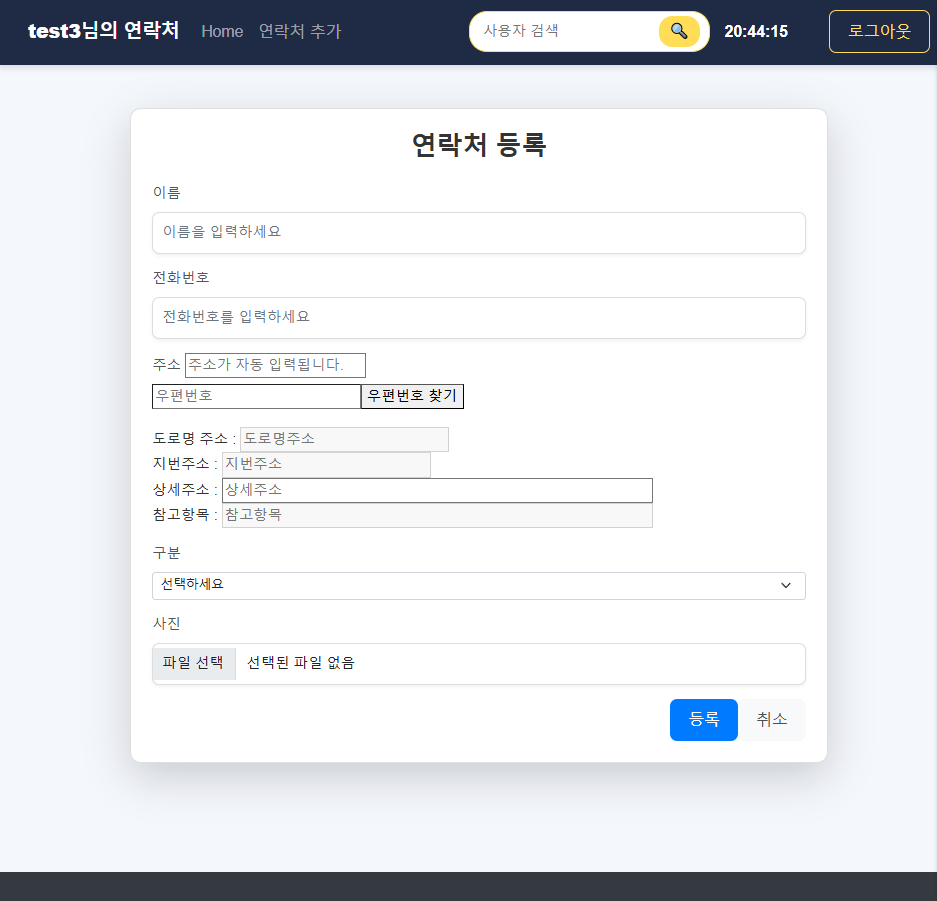
## 아이디 찾기



## 비밀번호 재설정



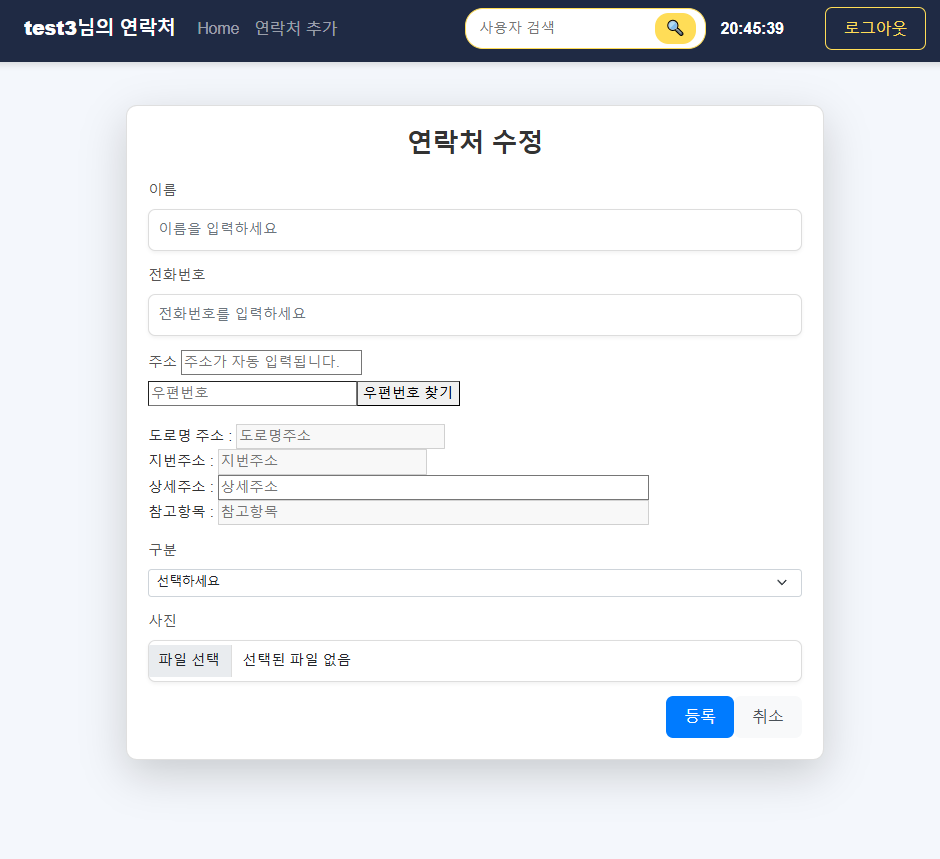
## 연락처 등록



## 연락처 목록



## 연락처 수정



## 연락처 검색

