우 성 욱

2025-BD02-04-시스템 설계서

시스템 설계서

[웹 페이지 연락처 프로그램 구현]

|  |  |  |  |
| --- | --- | --- | --- |
| 제/개정 이력 | | | |
|  | | | |
| 문서 명 | 시스템 설계서 | | |
|  | | | |
| 연번 | 작성일 | 제/개정사항 | 작성자 |
| 1 | 2025-02-19 | 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)

# 서론

## 1.1 설계 목적

본 설계서는 연락처 관리 프로그램의 시스템 아키텍처와 데이터베이스 설계, 사용자 인터페이스 설계를 포함한 시스템 설계 문서입니다. 이 시스템은 회원 별로 연락처를 관리하고, 연락처를 다양한 그룹(가족, 친구, 기타)으로 구분하여 관리할 수 있습니다.

## 1.2 설계 범위

* **회원가입 및 로그인  
  -** 사용자는 자신의 계정을 생성하여 로그인하고, 개인화된 서비스를 이용할 수 있습니다.
* **아이디 및 비밀번호 찾기**  
  - 사용자가 로그인 정보를 잊어버렸을 경우, 이메일을 통해 아이디를 찾거나 비밀번호를 재설정 할 수 있는 기능을 제공합니다.
* **연락처 관리**  
  - 사용자는 로그인 후, 자신의 연락처 목록을 확인하고 관리할 수 있습니다.
* **연락처 추가**  
  - 사용자는 새로운 연락처를 추가할 수 있으며, 주소 입력 시 우편번호 검색 기능을 사용하여 주소를 자동으로 입력할 수 있습니다.
* **연락처 수정**  
  - 사용자는 기존에 저장된 연락처의 정보를 수정할 수 있는 기능을 제공합니다.
* **연락처 삭제**  
  - 사용자는 저장된 연락처를 삭제할 수 있는 기능을 제공하여 불필요한 정보를 제거할 수 있습니다.
* **사진 업로드**  
  - 연락처 추가 시, 해당 연락처의 사진을 업로드할 수 있는 기능을 제공합니다.
* **이름 검색 기능**  
  - 사용자는 연락처 목록에서 이름을 기준으로 특정 연락처를 빠르게 검색할 수 있습니다.

# 시스템 아키텍처

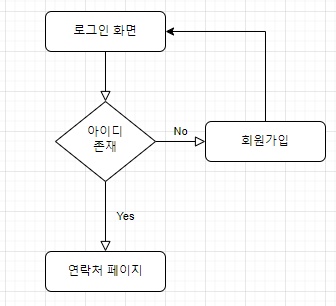
## 시스템 구성도

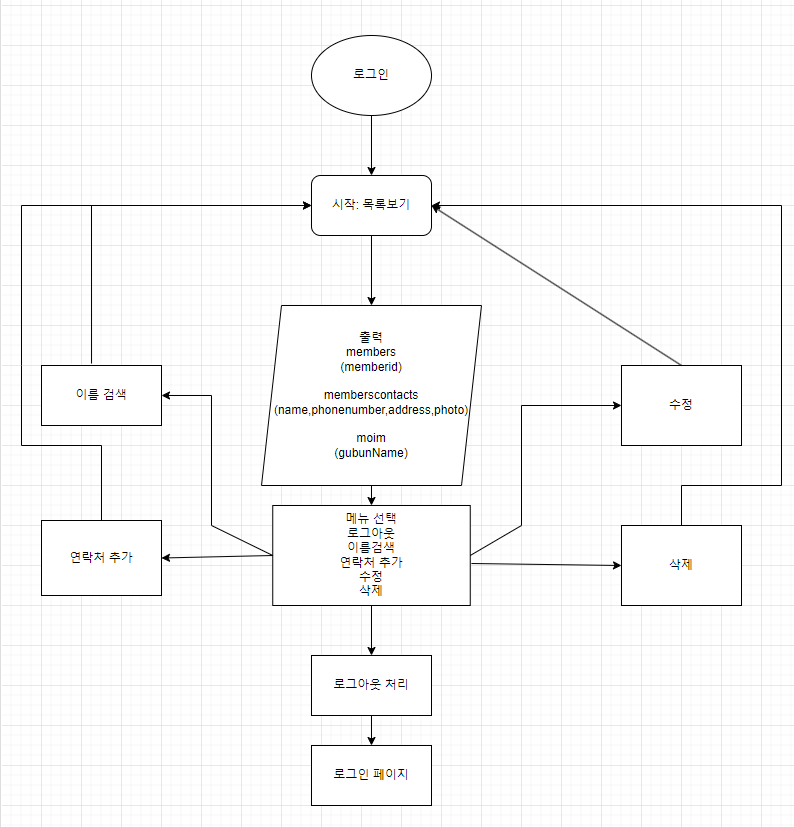
시스템은 클라이언트와 서버로 구성됩니다. 클라이언트는 사용자 인터페이스를 제공하고, 사용자의 요청을 데이터베이스 서버에서 처리합니다.

## 주요 시스템 구성 요소

* 클라이언트: mustache
* 서버: Java Spring Framework
* DB: MySQL 8.0.17

## 시스템 설계도



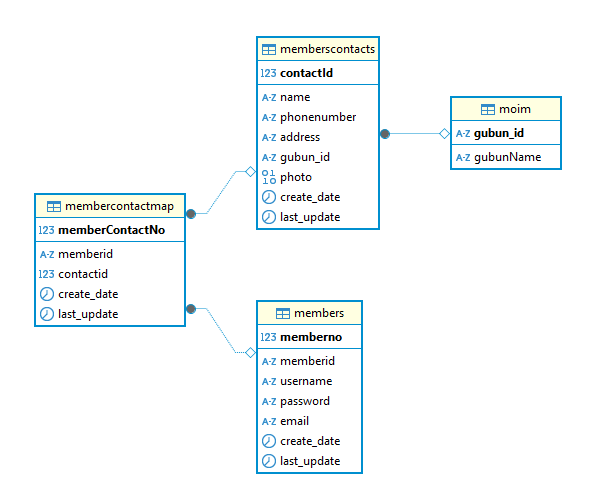


# 데이터베이스 설계

## 데이터베이스 개요

이 시스템의 데이터베이스는 4개의 주요 테이블(members, moim, memberscontacts, membercontactmap) 로 구성됩니다

## ERD (Entity Relationship Diagram)



* members 테이블은 회원 정보를 저장하며, 각 회원은 여러 개의 연락처를 가질 수 있습니다.
* moim 테이블은 연락처 그룹 정보를 저장합니다.
* memberscontacts 테이블은 연락처의 세부 정보를 저장합니다.
* membercontactmap 테이블은 각 회원이 어떤 연락처를 가졌는지 의 관계를 나타냅니다.

## 테이블 설계

* members 테이블

회원 정보를 저장하는 목적을 가진 테이블입니다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 컬럼명 | Data Type | Not null/null | Key | 설명 |
| memberNo | int(11) | not null | Primary Key | 회원 고유 번호 |
| memberId | varchar(255) | not null | Unique Key | 회원 ID |
| userName | varchar(255) | not null | - | 회원 이름 |
| password | varchar(255) | not null | - | 비밀번호 |
| email | varchar(255) | not null | - | 이메일 주소 |
| create\_date | timestamp | - | - | 생성일 |
| last\_update | timestamp | - | - | 마지막 수정일 |

* moim 테이블

연락처 그룹 정보를 저장하는 목적으로 가진 테이블입니다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 컬럼명 | Data Type | Not null/null | Key | 설명 |
| gubun\_id | varchar(255) | not null | Primary Key | 그룹 ID |
| gubunName | enum(‘가족’,’친구’,’기타’) | not null | - | 그룹 이름 (가족, 친구, 기타) |

* memberscontacts 테이블

회원이 관리하는 연락처 정보를 저장하는 목적으로 가진 테이블입니다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 컬럼명 | Data Type | Not null/null | Key | 설명 |
| contactid | int(11) | not null | Primary Key | 연락처 ID |
| name | varchar(255) | not null | - | 연락처 이름 |
| phonenumber | varchar(255) | not null | - | 전화번호 |
| address | varchar(255) | not null | - | 주소 |
| guubn\_id | varchar(255) | not null | Foreign Key | 연락처 그룹  (moim 테이블 참조) |
| photo | longblob | not null | - | 연락처 사진 |
| create\_date | timestamp | - | - | 생성일 |
| last\_update | timestamp | - | - | 마지막 수정일 |

* membercontactmap 테이블

회원과 연락처의 관계를 정의합니다.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 컬럼명 | Data Type | Not null/null | Key | 설명 |
| memberContactNo | int(11) | not null | Primary Key | 관계 ID |
| memberid | varchar(255) | not null | Foreign Key | 회원 ID  (members 테이블 참조) |
| contactid | int(11) | not null | Foreign Key | 연락처 ID  (memberscontacts 테이블 참조) |
| create\_date | timestamp | - | - | 생성일 |
| last\_update | timestamp | - | - | 마지막 수정일 |

## 테이블 관계

members ↔ membercontactmap: 1:N 관계 (한 회원은 여러 연락처를 가질 수 있음)

moim ↔ memberscontacts: 1:N 관계 (한 그룹은 여러 연락처를 포함할 수 있음)

memberscontacts ↔ membercontactmap: 1:N 관계 (한 연락처는 여러 회원에 의해 참조될 수 있음)

# 사용자 인터페이스 설계

## 화면 설계

사용자 인터페이스는 다음과 같은 화면을 제공합니다

* 회원 가입 화면: 회원 정보 입력 폼
* 연락처 목록 화면: 등록된 연락처를 표시
* 연락처 추가 화면: 신규 연락처를 등록할 수 있는 폼
* 연락처 수정 화면: 기존 연락처를 수정할 수 있는 폼
* 연락처 삭제: 기존 연락처를 삭제할 수 있는 폼