



시스템 설계서

[웹 페이지 연락처 프로그램 구현]

2025-BD02-04-시스템 설계서

우 성 욱

제/개정 이력

문서 명	시스템 설계서
------	---------

[illegible]

목 차

1. 서론	3
1.1 설계 목적	3
1.2 설계 범위	3
2. 시스템 아키텍처	4
2.1 시스템 구성도	4
2.2 주요 시스템 구성 요소	4
2.3 시스템 설계도	4
3. 데이터베이스 설계	6
3.1 데이터베이스 개요	6
3.2 ERD (Entity Relationship Diagram)	6
3.3 테이블 설계	7
3.4 테이블 관계	10
4. 사용자 인터페이스 설계	11
4.1 화면 설계	11

1. 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.

2. 시스템 아키텍처

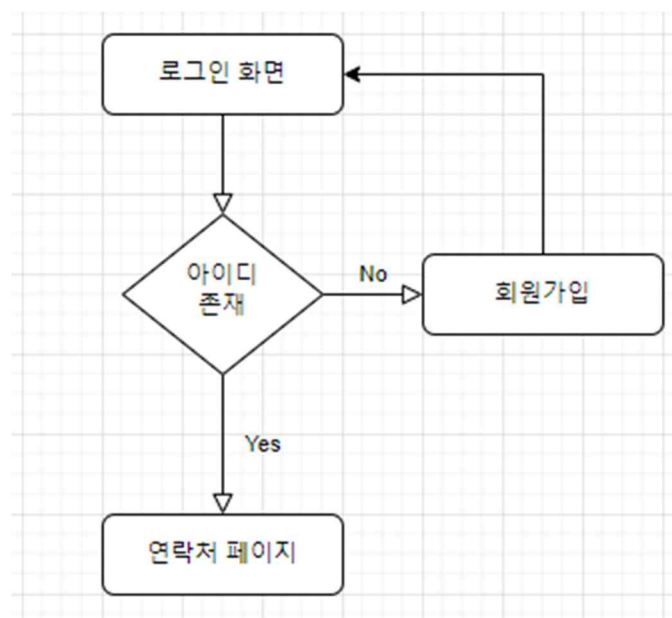
2.1 시스템 구성도

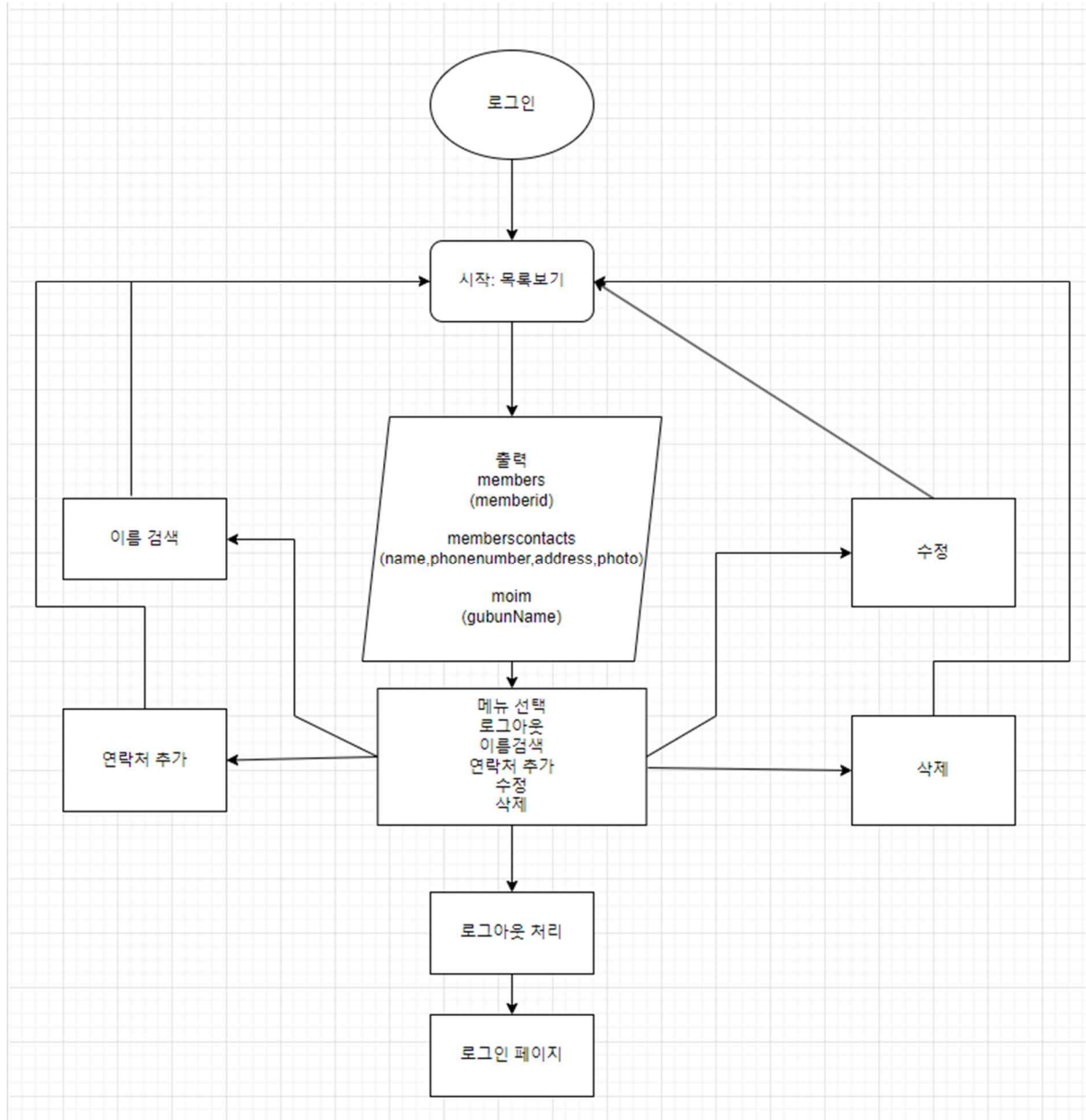
시스템은 클라이언트와 서버로 구성됩니다. 클라이언트는 사용자 인터페이스를 제공하고, 사용자의 요청을 데이터베이스 서버에서 처리합니다.

2.2 주요 시스템 구성 요소

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

2.3 시스템 설계도



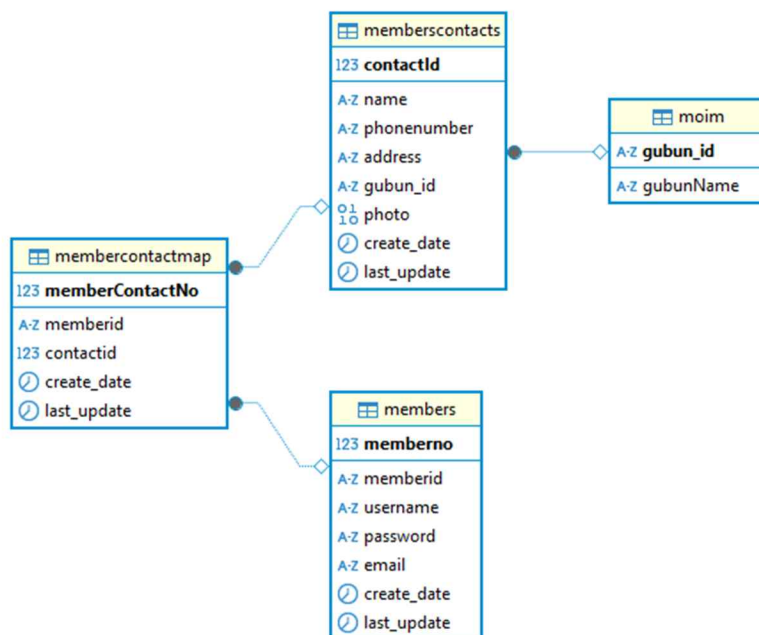


3. 데이터베이스 설계

3.1 데이터베이스 개요

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

3.2 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 가졌는지의 관계를 나타냅니다.

3.3 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

Properties

Data

연티티 관계도

172.24.32.100

Databases

phone

테이블명:

members

Partitioned

Engine:

InnoDB

Auto Increment:

4

Charset:

utf8mb4

Collation:

utf8mb4_unicode_ci

Description:

Columns

Constraints

Foreign Keys

References

Triggers

Indexes

Partitions

Statistics

DDL

Virtual

컬럼명	#	Data Type	Not Null	Auto Increment	Key	디폴트	Extra	Expression	Comment
123 memberno	1	int(11)	[v]	[v]	PRI		auto_increment		
AZ memberid	2	varchar(255)	[v]	[]	UNI				
AZ username	3	varchar(255)	[v]	[]					
AZ password	4	varchar(255)	[v]	[]					
AZ email	5	varchar(255)	[v]	[]					
create_date	6	timestamp	[]	[]		CURRENT_TIMESTAMP	DEFAULT_GENERATED		
last_update	7	timestamp	[]	[]		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP		

- 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)

The screenshot shows a database management interface for a table named 'moim'. The table is configured with the following properties:

- Table Name:** moim
- Engine:** InnoDB
- Auto Increment:** 0
- Charset:** utf8mb4
- Collation:** utf8mb4_unicode_ci
- Partitioned:** No

The table structure is as follows:

컬럼명	#	Data Type	Not Null	Auto Increment	Key	디폴트	Extra	Expression	Comment
A-Z gubun_id	1	varchar(255)	[v]	[]	PRI				
A-Z gubunName	2	enum('가족','친구','기타')	[v]	[]					

- 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

컬럼명	#	Data Type	Not Null	Auto Increment	Key	디폴트	Extra	Expression	Comment
contactid	1	int(11)	[v]	[v]	PRI		auto_increment		
name	2	varchar(255)	[v]	[]					
phonenum	3	varchar(255)	[v]	[]					
address	4	varchar(255)	[v]	[]					
guubn_id	5	varchar(255)	[]	[]	MUL				
photo	6	longblob	[v]	[]					
create_date	7	timestamp	[]	[]		CURRENT_TIMESTAMP	DEFAULT_GENERATED		
last_update	8	timestamp	[]	[]		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP		

- 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

컬럼명	#	Data Type	Not Null	Auto Increment	Key	디폴트	Extra	Expression	Comment
memberContactNo	1	int(11)	[v]	[v]	PRI		auto_increment		
memberid	2	varchar(255)	[]	[]	MUL				
contactid	3	int(11)	[]	[]	MUL				
create_date	4	timestamp	[]	[]		CURRENT_TIMESTAMP	DEFAULT_GENERATED		
last_update	5	timestamp	[]	[]		CURRENT_TIMESTAMP	DEFAULT_GENERATED on update CURRENT_TIMESTAMP		

3.4 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)

4. 사용자 인터페이스 설계

4.1 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