포팅 매뉴얼

사용 프로그램 버전

> 💡 **프로젝트에서 사용한 프로그램의 버전을 정리합니다.**

- OS
  - Ubuntu 20.04.6 LTS
- Frontend
  - Node.js : 20.11.1
  - Vue : 3.4.15

    - Tailwind CSS : 3.4.1
    - Flowbit : 2.2.1
    - Extension
        - ESLint
        - HTML CSS Support
        - Prettier
        - JS code snippets
        - Live Server
        - Vetur
        - Vue 3 Snippet
        - Vue language feature(Volar)
        - Auto Rename Tag
        - Auto Close Tag
        - Code Spell Checker
- Backend
    - InteliJ Ultimate : 2023.3.2
    - Java : 17
    - Spring Boot : 3.2.1
        - Lombok : 1.18.30
        - Spring Web : 3.2.1
        - Spring Boot DevTools : 3.2.1
        - Spring Data JPA : 3.2.1
        - Spring Security : 6.2.1
    - Gradle : 8.5
    - MySQL : 8.0.35
    - Redis : 7.2
- Infra
    - Portainer : 2.19.4

    - NginxProxyManager : 2.11.1
- OpenAPI
    - Openvidu : 2.29.0
    - OpenCV : 4.9.0

배포 환경

> 💡 **서버 구성에 사용된 코드를 정리합니다**

서버 구성(AWS - EC2)

1. EC2 환경 설치

```
# root로 전환
sudo su

# 시스템 패키지 리스트 업데이트
apt-get install update

# curl 설치
apt install curl

# vim 설치 (yml 파일의 원활한 작성 및 수정을 위함)
apt-get install vim

# 네트워크 도구 설치
apt-get install net-tools

# 도커 및 도커 컴포즈 설치
apt-get install docker.io docker-compose
```

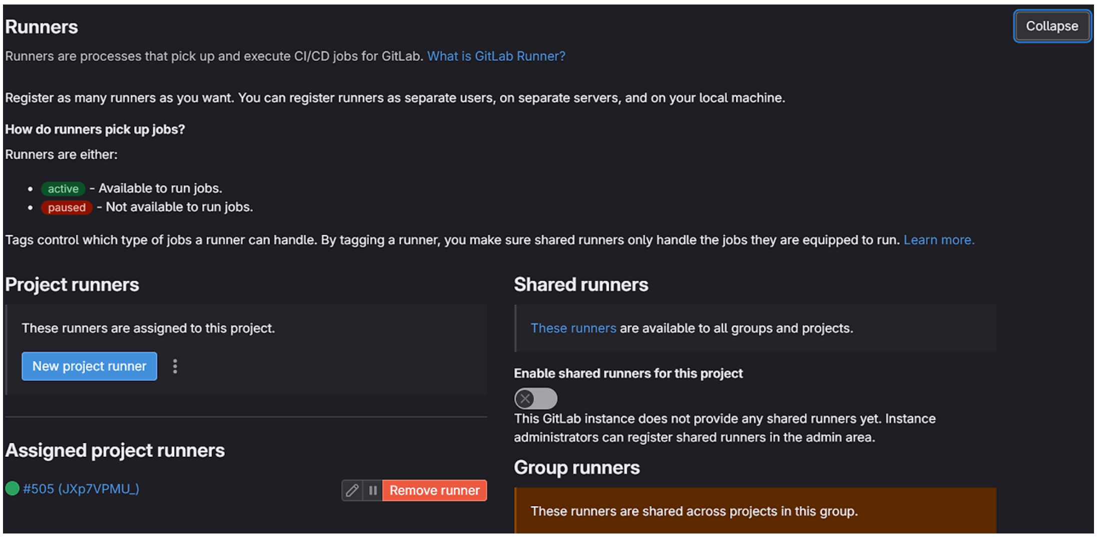
```
# ec2 public ip 확인
curl ifconfig.me

# private ip 확인
hostname -I | awk '{print $1}'
```

배포 자동화(CI/CD)

1. gitlab-runner 설치

    - gitlab 레포지토리에서, Settings - CI/CD - Runners 접속 후 좌측 하단에 New project runner 버튼 클릭



Runners

Runners are processes that pick up and execute CI/CD jobs for GitLab. What is GitLab Runner?

Register as many runners as you want. You can register runners as separate users, on separate servers, and on your local machine.

How do runners pick up jobs?

Runners are either:

- active - Available to run jobs.
- paused - Not available to run jobs.

Tags control which type of jobs a runner can handle. By tagging a runner, you make sure shared runners only handle the jobs they are equipped to run. Learn more.

Project runners

These runners are assigned to this project.

Assigned project runners

#505 (JXp7VPMU_)

Shared runners

These runners are available to all groups and projects.

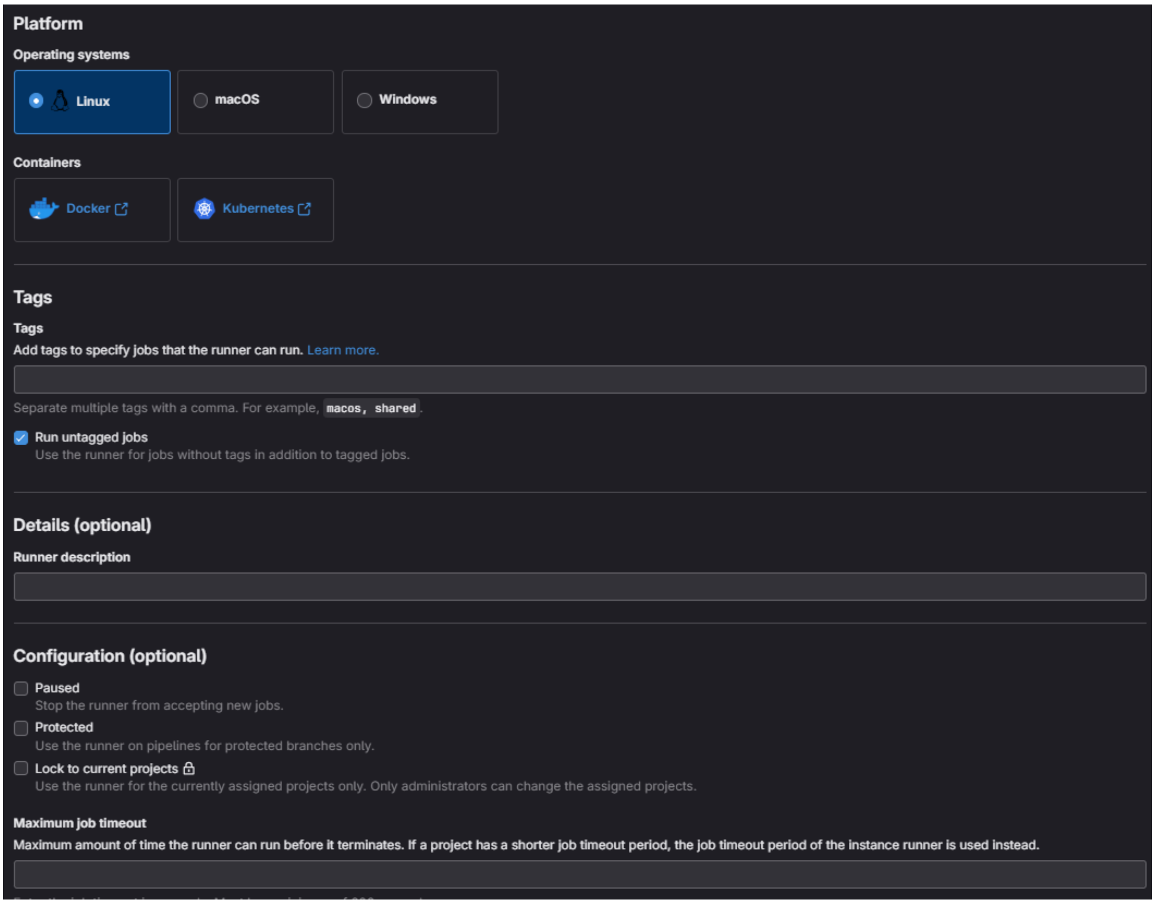
Enable shared runners for this project

This GitLab instance does not provide any shared runners yet. Instance administrators can register shared runners in the admin area.

Group runners

These runners are shared across projects in this group.

    - os는 리눅스 선택, 태그는 따로 지정하지 않음, run untagged jobs를 체크



Platform

Operating systems

● Linux ○ macOS ○ Windows

Containers

Tags

Tags

Add tags to specify jobs that the runner can run. Learn more.

Separate multiple tags with a comma. For example, `macos, shared`.

☑ Run untagged jobs
Use the runner for jobs without tags in addition to tagged jobs.

Details (optional)

Runner description

Configuration (optional)

□ Paused
Stop the runner from accepting new jobs.

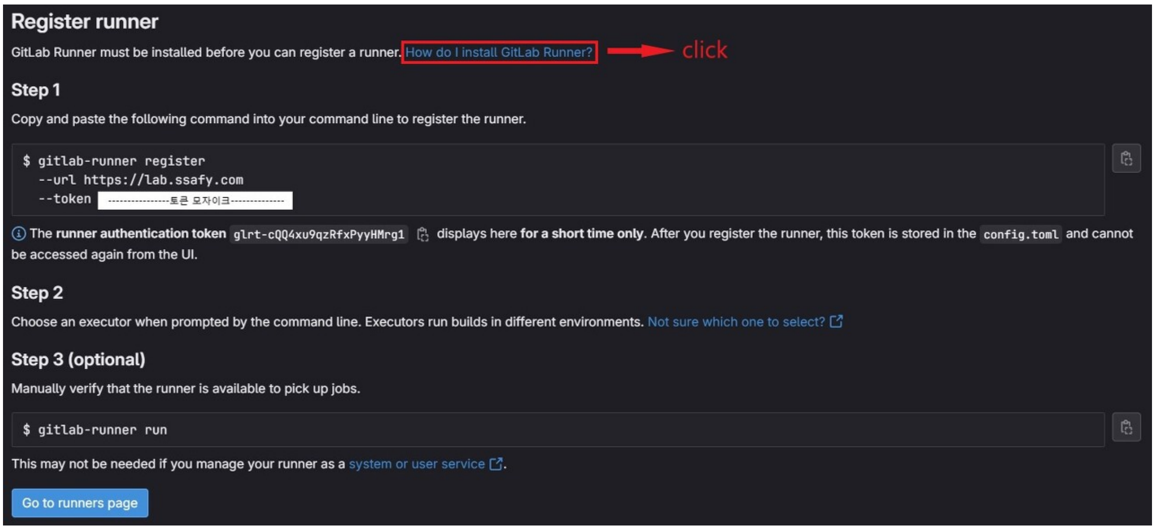
□ Protected
Use the runner on pipelines for protected branches only.

□ Lock to current projects
Use the runner for the currently assigned projects only. Only administrators can change the assigned projects.

Maximum job timeout

Maximum amount of time the runner can run before it terminates. If a project has a shorter job timeout period, the job timeout period of the instance runner is used instead.

- 하단에 이어지는 화면의 step에 따라 gitlab-runner 설치 및 register 진행



Register runner

GitLab Runner must be installed before you can register a runner. How do I install GitLab Runner? → click

Step 1

Copy and paste the following command into your command line to register the runner.

```
$ gitlab-runner register
  --url https://lab.ssafy.com
  --token ---------------토큰 모자이크--------------
```

ⓘ The **runner authentication token** `glrt-cQQ4xu9qzRfxPyyHMrg1` displays here **for a short time only**. After you register the runner, this token is stored in the `config.toml` and cannot be accessed again from the UI.

Step 2

Choose an executor when prompted by the command line. Executors run builds in different environments. Not sure which one to select?

Step 3 (optional)

Manually verify that the runner is available to pick up jobs.

```
$ gitlab-runner run
```

This may not be needed if you manage your runner as a system or user service.

- 상단에 "————토큰 모자이크————" 복사(이후 레지스터 단계에서 필요)
- 빨간색 체크 해놓은 부분을 누르면 gitlab-runner 설치 스크립트 표시(다음과 같음)

```
# Download the binary for your system
sudo curl -L --output /usr/local/bin/gitlab-runner http

# Give it permission to execute
sudo chmod +x /usr/local/bin/gitlab-runner

# Create a GitLab Runner user
sudo useradd --comment 'GitLab Runner' --create-home gi

# Install and run as a service
sudo gitlab-runner install --user=gitlab-runner --worki
sudo gitlab-runner start
```

2. 도커 컨테이너에 runner 올리기

```
docker run --detach \
--name gitlab-runner \
--restart always \
--volume /srv/gitlab-runner/config:/etc/gitlab-runner \
--volume /var/run/docker.sock:/var/run/docker.sock \
gitlab/gitlab-runner:latest
```

3. gitlab-runner 등록

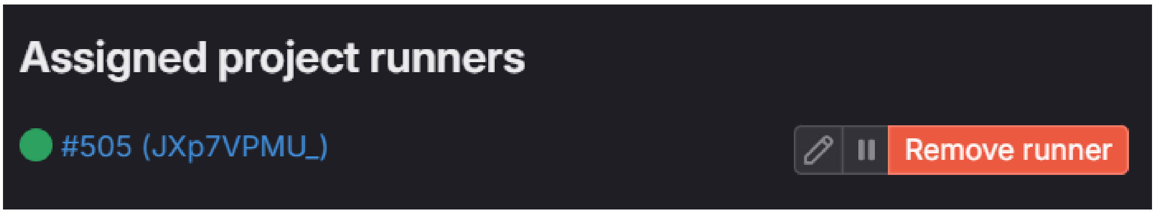
- 컨테이너 접속

```
docker container exec -it gitlab-runner bash
```

- gitlab-runner 설치 시 산출된 토큰으로 컨테이너 내부에서 register

```
gitlab-runner register -n \
--url [gitlab 서버주소] \
--registration-token [복사해뒀던 토큰] \
--description gitlab-runner \
--executor docker \
--docker-image docker:latest \
--docker-volumes /var/run/docker.sock:/var/run/docker.sock
```

4. 연결 확인(Settings - CI/CD - Runners)



Assigned project runners

● #505 (JXp7VPMU_)

Nginx Proxy Manager 세팅

1. 디렉토리 생성

```
# 루트 디렉토리로 이동
cd /

# data 디렉토리 생성
mkdir data

# data 디렉토리로 이동
cd data

# nginx proxy manager(npm) 디렉토리 생성
mkdir npm
```

2. docker-compose.yml 작성

```
# docker-compose.yml vim 모드로 실행(없으면 생성)
vi docker-compose.yml
```

```
# docker-compose.yml 다음과 같이 작성
version: '3'
services:
  app:
    image: 'jc21/nginx-proxy-manager:latest'
    restart: unless-stopped
    ports:
      - '80:80'
      - '81:81'
      - '443:443'
    volumes:
      - ./data:/data
      - ./letsencrypt:/etc/letsencrypt
```

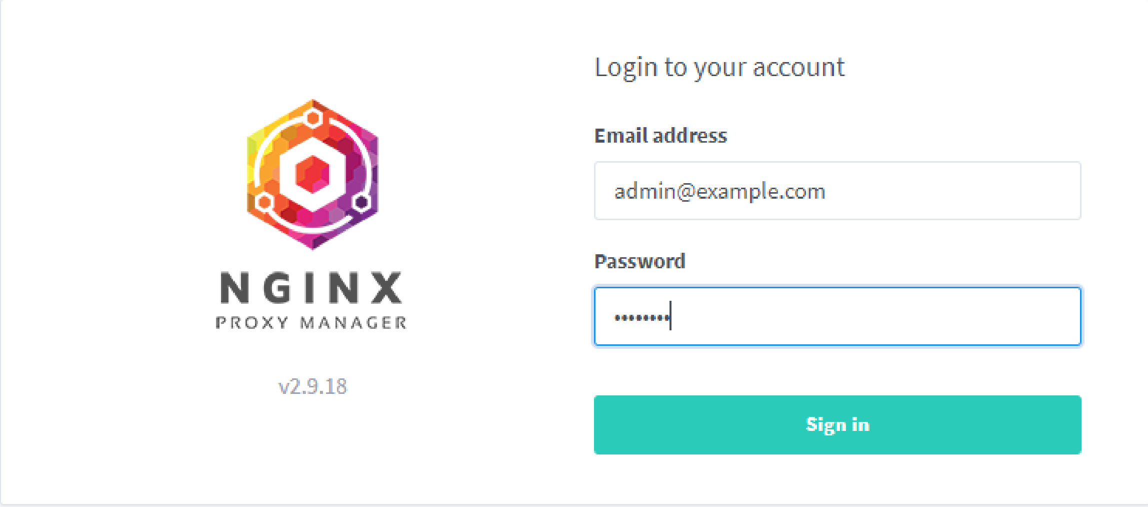
3. 실행

```
# 실행
docker-compose up -d
```

4. 81포트로 접속(예: i10a401.p.ssafy.io:81)

   초기 이메일과 패스워드는 다음과 같다

   - email: admin@example.com
   - password: changme



Login to your account

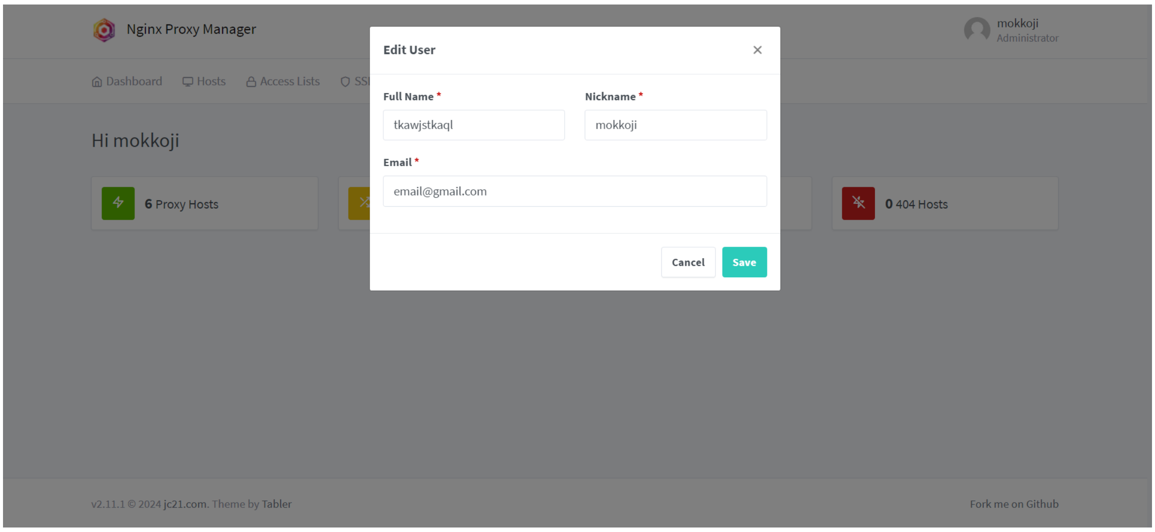
Email address

admin@example.com

Password

••••••••

5. 계정명, pw 변경



Edit User

Full Name * tkawjstkaql

Nickname * mokkoji

Email * email@gmail.com

MySQL 세팅

1. 디렉토리 생성

```
# 루트 디렉토리로 이동
cd /

# data 디렉토리로 이동
cd data

# mysql 디렉토리 생성
mkdir mysql
```

2. docker-compose.yml 작성

```
# docker-compose.yml vim 모드로 실행(없으면 생성)
vi docker-compose.yml

# docker-compose.yml 다음과 같이 작성
version: '3'
services:
    mysql:
        image: mysql:8.0.35
        restart: always
        container_name: mysql
        ports:
          - "3306:3306"
        environment:
            MYSQL_ROOT_PASSWORD: tkawjstkaql
            TZ: Asia/Seoul
        command:
            - --character-set-server=utf8mb4
            - --collation-server=utf8mb4_unicode_ci
        volumes:
            - ./data/mysql/:/var/lib/mysql
```

3. 실행

```
# 실행
docker-compose up -d
```

Redis 세팅

1. 디렉토리 생성

```
# 루트 디렉토리로 이동
cd /
```

```
# data 디렉토리로 이동
cd data

# redis 디렉토리 생성
mkdir redis
```

2. docker-compose.yml 작성

```
# docker-compose.yml vim 모드로 실행(없으면 생성)
vi docker-compose.yml

# docker-compose.yml 다음과 같이 작성
version: '3'
services:
  redis:
    image: redis:7.2
    container_name: redis
    ports:
      - "6379:6379"
    volumes:
      - /usr/local/etc/redis/redis.conf:/usr/local/etc/red
      - /usr/local/etc/redis/users.acl:/usr/local/etc/redi
    restart: always
```

3. 실행

```
# 실행
docker-compose up -d
```

Portainer 세팅

1. 디렉토리 생성

```
# 루트 디렉토리로 이동
cd /

# data 디렉토리로 이동
cd data

# portainer 디렉토리 생성
mkdir portainer
```

2. docker-compose.yml 작성

```
# docker-compose.yml vim 모드로 실행(없으면 생성)
vi docker-compose.yml

# docker-compose.yml 다음과 같이 작성
version: '3'

services:
  portainer:
    image: portainer/portainer-ce:alpine
    container_name: portainer
    restart: always
    ports:
      - 9000:9000
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock
      - ./portainer_data:/data

volumes:
  portainer_data:
```

3. 실행

```
# 실행
```

```
docker-compose up -d
```

Openvidu 세팅

참고: on-premises 방식으로 배포할 것임

1. 경로 이동

```
# openvidu 설치 위치로 이동
cd /opt
```

2. Openvidu 설치

```
# openvidu 설치
curl https://s3-eu-west-1.amazonaws.com/aws.openvidu.io/in
```

3. Openvidu 디렉토리로 이동

```
# openvidu 디렉토리로 이동
cd /opt/openvidu
```

4. .env 파일 설정

```
# OpenVidu configuration
# ----------------------
# Documentation: https://docs.openvidu.io/en/stable/refere

# NOTE: This file doesn't need to quote assignment values,
# All values are stored as-is, even if they contain spaces
# Domain name. If you do not have one, the public IP of th
# For example: 198.51.100.1, or openvidu.example.com
DOMAIN_OR_PUBLIC_IP=mokkoji-openvidu.duckdns.org

# OpenVidu SECRET used for apps to connect to OpenVidu ser
OPENVIDU_SECRET=MOKKOJI_SECRET

# Certificate type:
# - selfsigned:  Self signed certificate. Not recommended 
#                Users will see an ERROR when connected to
# - owncert:     Valid certificate purchased in a Internet
#                Please put the certificates files inside 
#                with names certificate.key and certificat
# - letsencrypt: Generate a new certificate using letsencr
#                required contact email for Let's Encrypt 
#                variable.
CERTIFICATE_TYPE=letsencrypt

# If CERTIFICATE_TYPE=letsencrypt, you need to configure a
LETSENCRYPT_EMAIL=dkfnak6789@gmail.com

# Proxy configuration
# If you want to change the ports on which openvidu listen

# Allows any request to http://DOMAIN_OR_PUBLIC_IP:HTTP_PO
# redirected to https://DOMAIN_OR_PUBLIC_IP:HTTPS_PORT/.
# WARNING: the default port 80 cannot be changed during th
# if you have chosen to deploy with the option CERTIFICATE
HTTP_PORT=80

# Changes the port of all services exposed by OpenVidu.
# SDKs, REST clients and browsers will have to connect to 
HTTPS_PORT=443

... 이하 생략
```

2. Openvidu 실행

```
# openvidu 디렉토리로 이동
cd /opt/openvidu

# openvidu 실행
./openvidu start
```

파이프라인 세팅

배포 파이프라인의 스테이지 구성음 다음과 같음

1. 환경변수 파일 생성
2. 빌드
3. 도커 빌드
4. 도커 pull

백엔드

1. docker-compose 작성
    - 프로젝트 최상위 디렉토리에 docker-compose.yml 생성
    - 포트는 8080:8080으로 바인딩

```
version: '3'
services:
  app:
    image: ${DOCKER_USERNAME}/${DOCKER_REPO}:backend-lates
    ports:
      - "8080:8080"
    env_file:
      - ./.env
```

2. Dockerfile 작성

- backend 디렉토리 내부에 생성

```
FROM openjdk:17-alpine
WORKDIR /usr/src/app
ARG JAR_FILE=./build/libs/jansorry-0.0.1-SNAPSHOT.jar
COPY ${JAR_FILE} /usr/src/app/app.jar
EXPOSE 8080
ENV TZ Asia/Seoul
ENTRYPOINT ["java", "-jar", "./app.jar"]
```

3. 스크립트 작성

- 프로젝트 최상위 디렉토리에 .gitlab-ci.yml 파일 생성

```
stages:
  - create-env
  - backend-build
  - docker-build
  - docker-pull

variables:
  DOCKER_COMPOSE_FILE: docker-compose.yml

create-env:
  stage: create-env
  script:
    - echo "DB_URL=${DB_URL}" >> .env
    - echo "DB_NAME=${DB_NAME}" >> .env
    - echo "DB_USERNAME=${DB_USERNAME}" >> .env
    - echo "DB_PASSWORD=${DB_PASSWORD}" >> .env
    - echo "REDIS_HOST = ${REDIS_HOST}" >> .env
    - echo "REDIS_PORT = ${REDIS_PORT}" >> .env
    - echo "REDIS_PASSWORD = ${REDIS_PASSWORD}" >> .env
    - echo "JWT_SECRET = ${JWT_SECRET}" >> .env
    - echo "OAUTH2_SECRET_GOOGLE = ${OAUTH2_SECRET_GOOGLE}
    - echo "OAUTH2_SECRET_NAVER = ${OAUTH2_SECRET_NAVER}"
    - echo "OAUTH2_SECRET_KAKAO = ${OAUTH2_SECRET_KAKAO}"
    - echo "S3_ACCESS_KEY = ${S3_ACCESS_KEY}" >> .env
    - echo "S3_BUCKET = ${S3_BUCKET}" >> .env
    - echo "S3_REGION = ${S3_REGION}" >> .env
    - echo "S3_SECRET_KEY = ${S3_SECRET_KEY}" >> .env
    - echo "DOCKER_USERNAME = ${DOCKER_USERNAME}" >> .env
    - echo "DOCKER_REPO = ${DOCKER_REPO}" >> .env
    - echo "OPENVIDU_URL = ${OPENVIDU_URL}" >> .env
    - echo "OPENVIDU_SECRET = ${OPENVIDU_SECRET}" >> .env
  artifacts:
    paths:
      - .env
  only:
    - backend-deploy

backend-build:
  stage: backend-build
  image: gradle:jdk17
  script:
    - cd backend
    - cd mokkoji
    - chmod +x gradlew
    - ./gradlew clean build
  cache:
    paths:
      - .gradle/wrapper
      - .gradle/caches
  artifacts:
    paths:
      - backend/mokkoji/build/libs/mokkoji-0.0.1-SNAPSHOT.
  only:
    - backend-deploy

docker-build:
  stage: docker-build
  dependencies:
    - create-env
    - backend-build
  script:
    # image removal
    - docker rmi ${DOCKER_USERNAME}/${DOCKER_REPO}:backend
    # login
    - docker login -u ${DOCKER_USERNAME} -p ${DOCKER_PASSW
    # backend push
    - cd backend
    - cd mokkoji
    - docker build -t ${DOCKER_USERNAME}/${DOCKER_REPO}:ba
    - docker push ${DOCKER_USERNAME}/${DOCKER_REPO}:backen
  only:
    - backend-deploy

docker-pull:
  stage: docker-pull
  script:
    # login again
    - docker login -u ${DOCKER_USERNAME} -p ${DOCKER_PASSW
    # container removal
    - docker stop $(docker ps -q --filter ancestor=${DOCKE
    - docker rm $(docker ps -aq --filter ancestor=${DOCKER_
    - docker-compose -f $DOCKER_COMPOSE_FILE down || true
    - docker rmi $(docker images -q ${DOCKER_USERNAME}/${D
    # deploy
    - docker-compose -f $DOCKER_COMPOSE_FILE pull
    - docker-compose -f $DOCKER_COMPOSE_FILE up -d
    - docker image prune -f
  only:
    - backend-deploy
```

프론트

1. docker-compose 작성
    - 프로젝트 최상위 디렉토리에 docker-compose.yml 생성
    - 포트는 5173:5173으로 바인딩

```
version: '3'
services:
  frontend:
    image: ${DOCKER_USERNAME}/${DOCKER_REPO}:frontend-late
    ports:
      - '5173:5173'
    env_file:
      - ./.env
```

2. Dockerfile 작성

- frontend 디렉토리 내부에 생성

```
# 기반 이미지
FROM node:18-alpine

# 작업 디렉토리 설정
WORKDIR /usr/src/app

# 빌드된 파일들을 이미지로 복사
COPY package*.json .

#ADD src .
ADD . .

RUN npm install

# 애플리케이션에 할당할 포트
EXPOSE 5173

# 애플리케이션 실행 명령어
CMD ["npm", "run", "dev"]
```

3. 스크립트 작성

- 프로젝트 최상위 디렉토리에 .gitlab-ci.yml 파일 생성

```
stages:
  - create-env
  - frontend-build
  - docker-build
  - docker-pull
```

```
variables:
  DOCKER_COMPOSE_FILE: docker-compose.yml

create-env:
  stage: create-env
  script:
    - echo "DOCKER_USERNAME = ${DOCKER_USERNAME}" >> .env
    - echo "DOCKER_REPO = ${DOCKER_REPO}" >> .env
  artifacts:
    paths:
      - .env
  only:
    - frontend-deploy

frontend-build:
  stage: frontend-build
  image: node:18-alpine
  script:
    - cd frontend
    - npm install
    - npm run build
  artifacts:
    paths:
      - frontend/dist
  only:
    - frontend-deploy

docker-build:
  stage: docker-build
  dependencies:
    - create-env
    - frontend-build
  script:
    # image removal
    - docker rmi ${DOCKER_USERNAME}/${DOCKER_REPO}:fronten
    # login
    - docker login -u ${DOCKER_USERNAME} -p ${DOCKER_PASSW
    # frontend push
    - cd frontend
    - docker build -t ${DOCKER_USERNAME}/${DOCKER_REPO}:fr
    - docker push ${DOCKER_USERNAME}/${DOCKER_REPO}:fronte
  only:
    - frontend-deploy

docker-pull:
  stage: docker-pull
  script:
    # login again
    - docker login -u ${DOCKER_USERNAME} -p ${DOCKER_PASSW
    # container removal
    - docker stop $(docker ps -q --filter ancestor=${DOCKE
    - docker rm $(docker ps -aq --filter ancestor=${DOCKER_
    - docker-compose -f $DOCKER_COMPOSE_FILE down || true
    - docker rmi $(docker images -q ${DOCKER_USERNAME}/${D
    # deploy
    - docker-compose -f $DOCKER_COMPOSE_FILE pull
    - docker-compose -f $DOCKER_COMPOSE_FILE up -d
    - docker image prune -f
  only:
    - frontend-deploy
```

DB접속 프로퍼티

DB Dump 파일

mokkoji.sql

```
-- MySQL dump 10.13  Distrib 8.0.35, for Linux (x86_64)
--
-- Host: localhost    Database: mokkoji
-- ------------------------------------------------------
-- Server version	8.0.35

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESUL
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION
/*!50503 SET NAMES utf8mb4 */;
/*!40103 SET @OLD_TIME_ZONE=@@TIME_ZONE */;
/*!40103 SET TIME_ZONE='+00:00' */;
/*!40014 SET @OLD_UNIQUE_CHECKS=@@UNIQUE_CHECKS, UNIQUE_CHECK
/*!40014 SET @OLD_FOREIGN_KEY_CHECKS=@@FOREIGN_KEY_CHECKS, FO
/*!40101 SET @OLD_SQL_MODE=@@SQL_MODE, SQL_MODE='NO_AUTO_VALU
/*!40111 SET @OLD_SQL_NOTES=@@SQL_NOTES, SQL_NOTES=0 */;

--
-- Table structure for table `account`
--

DROP TABLE IF EXISTS `account`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `account` (
  `account_id` bigint NOT NULL AUTO_INCREMENT,
  `user_id` bigint DEFAULT NULL,
  `bank` varchar(10) COLLATE utf8mb4_unicode_ci NOT NULL,
  `number` varchar(20) COLLATE utf8mb4_unicode_ci NOT NULL,
  PRIMARY KEY (`account_id`),
  UNIQUE KEY `UK_h6dr47em6vg85yuwt4e2roca4` (`user_id`),
  CONSTRAINT `FK7m8ru44m93ukyb61dfxw0apf6` FOREIGN KEY (`user
) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=utf8mb4 COL
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `account`
--
```

```
LOCK TABLES `account` WRITE;
/*!40000 ALTER TABLE `account` DISABLE KEYS */;
INSERT INTO `account` VALUES (1,1,'카카오뱅크','2222111122222'),
/*!40000 ALTER TABLE `account` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `account_seq`
--

DROP TABLE IF EXISTS `account_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `account_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `account_seq`
--

LOCK TABLES `account_seq` WRITE;
/*!40000 ALTER TABLE `account_seq` DISABLE KEYS */;
INSERT INTO `account_seq` VALUES (1);
/*!40000 ALTER TABLE `account_seq` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `background_template`
--

DROP TABLE IF EXISTS `background_template`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `background_template` (
  `background_id` int NOT NULL,
  `background_path` varchar(255) COLLATE utf8mb4_unicode_ci D
  `background_name` enum('BASIC','WEDDING','SCHOOL','LUNAR','
  PRIMARY KEY (`background_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `background_template`
--

LOCK TABLES `background_template` WRITE;
/*!40000 ALTER TABLE `background_template` DISABLE KEYS */;
INSERT INTO `background_template` VALUES (1,'https://mokkoji-
/*!40000 ALTER TABLE `background_template` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `background_template_seq`
--

DROP TABLE IF EXISTS `background_template_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `background_template_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `background_template_seq`
--

LOCK TABLES `background_template_seq` WRITE;
/*!40000 ALTER TABLE `background_template_seq` DISABLE KEYS *
INSERT INTO `background_template_seq` VALUES (1);
/*!40000 ALTER TABLE `background_template_seq` ENABLE KEYS */
UNLOCK TABLES;
```

```
--
-- Table structure for table `event`
--

DROP TABLE IF EXISTS `event`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `event` (
  `participant_count` int DEFAULT '0',
  `end_time` datetime(6) DEFAULT NULL,
  `event_id` bigint NOT NULL AUTO_INCREMENT,
  `start_time` datetime(6) DEFAULT NULL,
  `user_id` bigint DEFAULT NULL,
  `session_id` varchar(100) COLLATE utf8mb4_unicode_ci DEFAUL
  `status` enum('ACTIVE','CLOSED') COLLATE utf8mb4_unicode_ci
  PRIMARY KEY (`event_id`),
  KEY `FKi8bsvlthqr8lngsyshiqsodak` (`user_id`),
  CONSTRAINT `FKi8bsvlthqr8lngsyshiqsodak` FOREIGN KEY (`user_
) ENGINE=InnoDB AUTO_INCREMENT=115 DEFAULT CHARSET=utf8mb4 CO
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `event`
--

LOCK TABLES `event` WRITE;
/*!40000 ALTER TABLE `event` DISABLE KEYS */;
INSERT INTO `event` VALUES (2,'2024-02-15 15:02:23.185563',2,
/*!40000 ALTER TABLE `event` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `event_seq`
--

DROP TABLE IF EXISTS `event_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
```

```
CREATE TABLE `event_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `event_seq`
--

LOCK TABLES `event_seq` WRITE;
/*!40000 ALTER TABLE `event_seq` DISABLE KEYS */;
INSERT INTO `event_seq` VALUES (801);
/*!40000 ALTER TABLE `event_seq` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `message`
--

DROP TABLE IF EXISTS `message`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `message` (
  `writer` varchar(6) COLLATE utf8mb4_unicode_ci NOT NULL,
  `message_id` bigint NOT NULL AUTO_INCREMENT,
  `rollingpaper_id` bigint DEFAULT NULL,
  `text` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NULL
  `video_path` varchar(255) COLLATE utf8mb4_unicode_ci DEFAUL
  `voice_path` varchar(255) COLLATE utf8mb4_unicode_ci DEFAUL
  PRIMARY KEY (`message_id`),
  KEY `FKg8eesqg0by9msqj7wy8i9h0hp` (`rollingpaper_id`),
  CONSTRAINT `FKg8eesqg0by9msqj7wy8i9h0hp` FOREIGN KEY (`roll
) ENGINE=InnoDB AUTO_INCREMENT=95 DEFAULT CHARSET=utf8mb4 COL
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `message`
--
```

```
LOCK TABLES `message` WRITE;
/*!40000 ALTER TABLE `message` DISABLE KEYS */;
INSERT INTO `message` VALUES ('이정민',2,2,'ㅋㅋㅋㅋㅋㅋㅋㅋ생일 축
/*!40000 ALTER TABLE `message` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `message_seq`
--

DROP TABLE IF EXISTS `message_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `message_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `message_seq`
--

LOCK TABLES `message_seq` WRITE;
/*!40000 ALTER TABLE `message_seq` DISABLE KEYS */;
INSERT INTO `message_seq` VALUES (1);
/*!40000 ALTER TABLE `message_seq` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `photo`
--

DROP TABLE IF EXISTS `photo`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `photo` (
  `photo_id` bigint NOT NULL AUTO_INCREMENT,
  `result_id` bigint DEFAULT NULL,
  `photo_path` varchar(255) COLLATE utf8mb4_unicode_ci NOT NU
  PRIMARY KEY (`photo_id`),
  KEY `FKl3a4d5y45xa4k8iqtuafpbjx5` (`result_id`),
  CONSTRAINT `FKl3a4d5y45xa4k8iqtuafpbjx5` FOREIGN KEY (`resu
) ENGINE=InnoDB AUTO_INCREMENT=527 DEFAULT CHARSET=utf8mb4 CO
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `photo`
--

LOCK TABLES `photo` WRITE;
/*!40000 ALTER TABLE `photo` DISABLE KEYS */;
INSERT INTO `photo` VALUES (1,2,'https://mokkoji-bucket.s3.ap
/*!40000 ALTER TABLE `photo` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `photo_seq`
--

DROP TABLE IF EXISTS `photo_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `photo_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `photo_seq`
--

LOCK TABLES `photo_seq` WRITE;
/*!40000 ALTER TABLE `photo_seq` DISABLE KEYS */;
INSERT INTO `photo_seq` VALUES (1351);
/*!40000 ALTER TABLE `photo_seq` ENABLE KEYS */;
```

```
UNLOCK TABLES;

--
-- Table structure for table `photomosaic`
--

DROP TABLE IF EXISTS `photomosaic`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `photomosaic` (
  `photomosaic_id` bigint NOT NULL AUTO_INCREMENT,
  `result_id` bigint DEFAULT NULL,
  `path` varchar(100) COLLATE utf8mb4_unicode_ci DEFAULT NULL
  PRIMARY KEY (`photomosaic_id`),
  UNIQUE KEY `UK_3ji5athsqy9ieso8b3hnm56p4` (`result_id`),
  CONSTRAINT `FKq76aqq8vsn5m8ygt6980u4xcg` FOREIGN KEY (`resu
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `photomosaic`
--

LOCK TABLES `photomosaic` WRITE;
/*!40000 ALTER TABLE `photomosaic` DISABLE KEYS */;
/*!40000 ALTER TABLE `photomosaic` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `photomosaic_seq`
--

DROP TABLE IF EXISTS `photomosaic_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `photomosaic_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
```

```
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `photomosaic_seq`
--

LOCK TABLES `photomosaic_seq` WRITE;
/*!40000 ALTER TABLE `photomosaic_seq` DISABLE KEYS */;
INSERT INTO `photomosaic_seq` VALUES (1);
/*!40000 ALTER TABLE `photomosaic_seq` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `postit_template`
--

DROP TABLE IF EXISTS `postit_template`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `postit_template` (
  `postit_id` int NOT NULL,
  `postit_path` varchar(255) COLLATE utf8mb4_unicode_ci DEFAU
  `postit_name` enum('RAINBOW','GREEN','BLUE','PINK','YELLOW'
  PRIMARY KEY (`postit_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `postit_template`
--

LOCK TABLES `postit_template` WRITE;
/*!40000 ALTER TABLE `postit_template` DISABLE KEYS */;
INSERT INTO `postit_template` VALUES (1,'https://mokkoji-buck
/*!40000 ALTER TABLE `postit_template` ENABLE KEYS */;
UNLOCK TABLES;

--
```

```
-- Table structure for table `postit_template_seq`
--

DROP TABLE IF EXISTS `postit_template_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `postit_template_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `postit_template_seq`
--

LOCK TABLES `postit_template_seq` WRITE;
/*!40000 ALTER TABLE `postit_template_seq` DISABLE KEYS */;
INSERT INTO `postit_template_seq` VALUES (1);
/*!40000 ALTER TABLE `postit_template_seq` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `record`
--

DROP TABLE IF EXISTS `record`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `record` (
  `event_count` int NOT NULL DEFAULT '0',
  `total_message` int NOT NULL DEFAULT '0',
  `total_participant` int NOT NULL DEFAULT '0',
  `total_time` int NOT NULL DEFAULT '0',
  `record_id` bigint NOT NULL AUTO_INCREMENT,
  `user_id` bigint NOT NULL,
  PRIMARY KEY (`record_id`),
  UNIQUE KEY `UK_np4qx16bnj5i2i34ak9miooqm` (`user_id`),
  CONSTRAINT `FKeny3549xar8rnrcmdw3hl0la1` FOREIGN KEY (`user
) ENGINE=InnoDB AUTO_INCREMENT=12 DEFAULT CHARSET=utf8mb4 COL
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `record`
--

LOCK TABLES `record` WRITE;
/*!40000 ALTER TABLE `record` DISABLE KEYS */;
INSERT INTO `record` VALUES (10,11,19,10,1,1),(9,34,10,4,2,2)
/*!40000 ALTER TABLE `record` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `record_seq`
--

DROP TABLE IF EXISTS `record_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `record_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `record_seq`
--

LOCK TABLES `record_seq` WRITE;
/*!40000 ALTER TABLE `record_seq` DISABLE KEYS */;
INSERT INTO `record_seq` VALUES (1);
/*!40000 ALTER TABLE `record_seq` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `result`
--
```

```
DROP TABLE IF EXISTS `result`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `result` (
  `event_id` bigint DEFAULT NULL,
  `result_id` bigint NOT NULL AUTO_INCREMENT,
  `user_id` bigint DEFAULT NULL,
  `name` varchar(15) COLLATE utf8mb4_unicode_ci DEFAULT NULL,
  `content` varchar(40) COLLATE utf8mb4_unicode_ci DEFAULT NU
  `image` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT 'ht
  `status` enum('MEMORY','RECOLLECTION') COLLATE utf8mb4_unic
  PRIMARY KEY (`result_id`),
  UNIQUE KEY `UK_c79qmsofnl7knkprooflyklyy` (`event_id`),
  KEY `FKpjjrrf0483ih2cvyfmx70a16b` (`user_id`),
  CONSTRAINT `FKniu3a6j6tu4wh9l9v4gckbloo` FOREIGN KEY (`even
  CONSTRAINT `FKpjjrrf0483ih2cvyfmx70a16b` FOREIGN KEY (`user_
) ENGINE=InnoDB AUTO_INCREMENT=115 DEFAULT CHARSET=utf8mb4 CO
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `result`
--

LOCK TABLES `result` WRITE;
/*!40000 ALTER TABLE `result` DISABLE KEYS */;
INSERT INTO `result` VALUES (2,2,1,NULL,NULL,'https://mokkoji
/*!40000 ALTER TABLE `result` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `result_seq`
--

DROP TABLE IF EXISTS `result_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `result_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `result_seq`
--

LOCK TABLES `result_seq` WRITE;
/*!40000 ALTER TABLE `result_seq` DISABLE KEYS */;
INSERT INTO `result_seq` VALUES (951);
/*!40000 ALTER TABLE `result_seq` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `rolling_paper`
--

DROP TABLE IF EXISTS `rolling_paper`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `rolling_paper` (
  `background_id` int DEFAULT NULL,
  `is_edited` bit(1) NOT NULL DEFAULT b'0',
  `postit_id` int DEFAULT NULL,
  `result_id` bigint DEFAULT NULL,
  `rollingpaper_id` bigint NOT NULL,
  PRIMARY KEY (`rollingpaper_id`),
  UNIQUE KEY `UK_f9q060wo44d7ev9plmuogg9ag` (`result_id`),
  KEY `FKevalsyuad70kj65s77f64aqqn` (`background_id`),
  KEY `FKtlss8wro6ly2qhfc95vpj1yb0` (`postit_id`),
  CONSTRAINT `FKevalsyuad70kj65s77f64aqqn` FOREIGN KEY (`back
  CONSTRAINT `FKq5xif1n1drxhpsvk9d85rticu` FOREIGN KEY (`resu
  CONSTRAINT `FKtlss8wro6ly2qhfc95vpj1yb0` FOREIGN KEY (`post
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
```

```
-- Dumping data for table `rolling_paper`
--

LOCK TABLES `rolling_paper` WRITE;
/*!40000 ALTER TABLE `rolling_paper` DISABLE KEYS */;
INSERT INTO `rolling_paper` VALUES (1,_binary '',1,2,1),(1,_b
/*!40000 ALTER TABLE `rolling_paper` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `rolling_paper_seq`
--

DROP TABLE IF EXISTS `rolling_paper_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `rolling_paper_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `rolling_paper_seq`
--

LOCK TABLES `rolling_paper_seq` WRITE;
/*!40000 ALTER TABLE `rolling_paper_seq` DISABLE KEYS */;
INSERT INTO `rolling_paper_seq` VALUES (451);
/*!40000 ALTER TABLE `rolling_paper_seq` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `user`
--

DROP TABLE IF EXISTS `user`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
```

```
CREATE TABLE `user` (
  `name` varchar(6) COLLATE utf8mb4_unicode_ci NOT NULL,
  `user_id` bigint NOT NULL AUTO_INCREMENT,
  `email` varchar(30) COLLATE utf8mb4_unicode_ci NOT NULL,
  `image` varchar(255) COLLATE utf8mb4_unicode_ci DEFAULT NUL
  `refresh_token` text COLLATE utf8mb4_unicode_ci,
  `authority` enum('GUEST','USER','ADMIN') COLLATE utf8mb4_un
  `provider` enum('GOOGLE','NAVER','KAKAO') COLLATE utf8mb4_u
  PRIMARY KEY (`user_id`)
) ENGINE=InnoDB AUTO_INCREMENT=13 DEFAULT CHARSET=utf8mb4 COL
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `user`
--

LOCK TABLES `user` WRITE;
/*!40000 ALTER TABLE `user` DISABLE KEYS */;
INSERT INTO `user` VALUES ('김지은',1,'jieun9912@naver.com','ht
/*!40000 ALTER TABLE `user` ENABLE KEYS */;
UNLOCK TABLES;

--
-- Table structure for table `user_seq`
--

DROP TABLE IF EXISTS `user_seq`;
/*!40101 SET @saved_cs_client     = @@character_set_client */
/*!50503 SET character_set_client = utf8mb4 */;
CREATE TABLE `user_seq` (
  `next_val` bigint DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_unico
/*!40101 SET character_set_client = @saved_cs_client */;

--
-- Dumping data for table `user_seq`
--
```

```
LOCK TABLES `user_seq` WRITE;
/*!40000 ALTER TABLE `user_seq` DISABLE KEYS */;
INSERT INTO `user_seq` VALUES (401);
/*!40000 ALTER TABLE `user_seq` ENABLE KEYS */;
UNLOCK TABLES;
/*!40103 SET TIME_ZONE=@OLD_TIME_ZONE */;

/*!40101 SET SQL_MODE=@OLD_SQL_MODE */;
/*!40014 SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS */;
/*!40014 SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS */;
/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT *
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION *
/*!40111 SET SQL_NOTES=@OLD_SQL_NOTES */;

-- Dump completed on 2024-02-16  3:13:20
```