🎁

포팅 매뉴얼

버전

API Server

java 11.0.15.1

Gradle

SpringBoot 2.7.2

Database

MariaDB 10.8.3

Client

React - 18.2.0

React - Redux 8.0.2

Getting Started

Installation

1. Install Intellij (IDE)
2. Install VSCode (IDE)
3. Install Docker

Setting Environment Variables (관련 자료 문서 하단에 있습니다.)

application.yml

.env.development / .env.production

AWS

- EC2
- S3 : 버킷 생성, IAM 계정 권한 부여, 버킷 접근 권한 설정
- Cloudfront : S3 버킷 연결, invalidation 설정

MariaDB

- DDL (Create Table)

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/b7bff9fd-3cd9-4d9b-bf9c-bf2233ed7743/a105_db.sql

Run the application

Client

npm i && run dev

Api Server

application.yml 설정 + build gradle -> run

Docker

KMS

```
docker pull kurento/kurento-media-server:latest
docker run --name kms --restart=always -p8888:8888/tcp -p5000-5050:5000-5050/udp -e KMS_MIN_PORT=5000 -e KMS_MAX_PORT=5050 kurento/kurento-media-server:latest
```

MariaDB

```
docker pull mariadb
docker run --name mariadb -d -p 3306:3306 --restart=always -e MYSQL_ROOT_PASSWORD=admin mariadb
docker cp a105_db.sql mariadb:/
docker exec -it mariadb /bin/bash
mysql -u root -p
mysql>source a105_db.sql
```

Coturn

```
sudo apt-get update
sudo apt-get install coturn
sudo vi /etc/turnserver.conf
```

`turnserver.conf`

```
listening-port=3478
tls-listening-port=5349
listening-ip=172.26.15.166
external-ip=3.38.107.146
min-port=49152
max-port=65535
verbose
server-name=meeteat.shop
cert=/etc/letsencrypt/live/meeteat.shop/fullchain.pem
pkey=/etc/letsencrypt/live/meeteat.shop/privkey.pem
dh-file=/etc/letsencrypt/ssl-dhparams.pem
realm=meeteat.shop
fingerprint
listening-ip=0.0.0.0
listening-port=3478
log-file=/var/log/turnserver.log

lt-cred-mech
user=root:meeteat1234
```

Nginx

`/etc/nginx/sites-available/default.conf`

```
upstream server{
    server 172.19.0.4:8443;
}

# 프론트엔드 upstream 설정
upstream client {
    server 172.18.0.2:3000;
}

server {
    server_name meeteat.shop; # managed by Certbot

        location ~ ^/(api) {
            proxy_redirect off;
            rewrite ^/api/(.*)$ /$1 break;
            proxy_pass https://server;
        }
        # / 웹소켓 통신을 위해서
            proxy_set_header Upgrade           $http_upgrade;
            proxy_set_header Connection        "upgrade";
            proxy_http_version 1.1;
#       }

    # / 경로로 오는 요청을 프론트엔드 upstream 의 / 경로로 포워딩
        location / {
           proxy_pass         http://client/;
        }


    listen 443 ssl; # managed by Certbot
    ssl_certificate /etc/letsencrypt/live/meeteat.shop/fullchain.pem; # managed by Certbot
    ssl_certificate_key /etc/letsencrypt/live/meeteat.shop/privkey.pem; # managed by Certbot
    include /etc/letsencrypt/options-ssl-nginx.conf; # managed by Certbot
    ssl_dhparam /etc/letsencrypt/ssl-dhparams.pem; # managed by Certbot

}
server {
    listen 80 ;
    listen [::]:80 ;
    server_name meeteat.shop;
    return 301 https://$host$request_uri;
}
```

참고자료

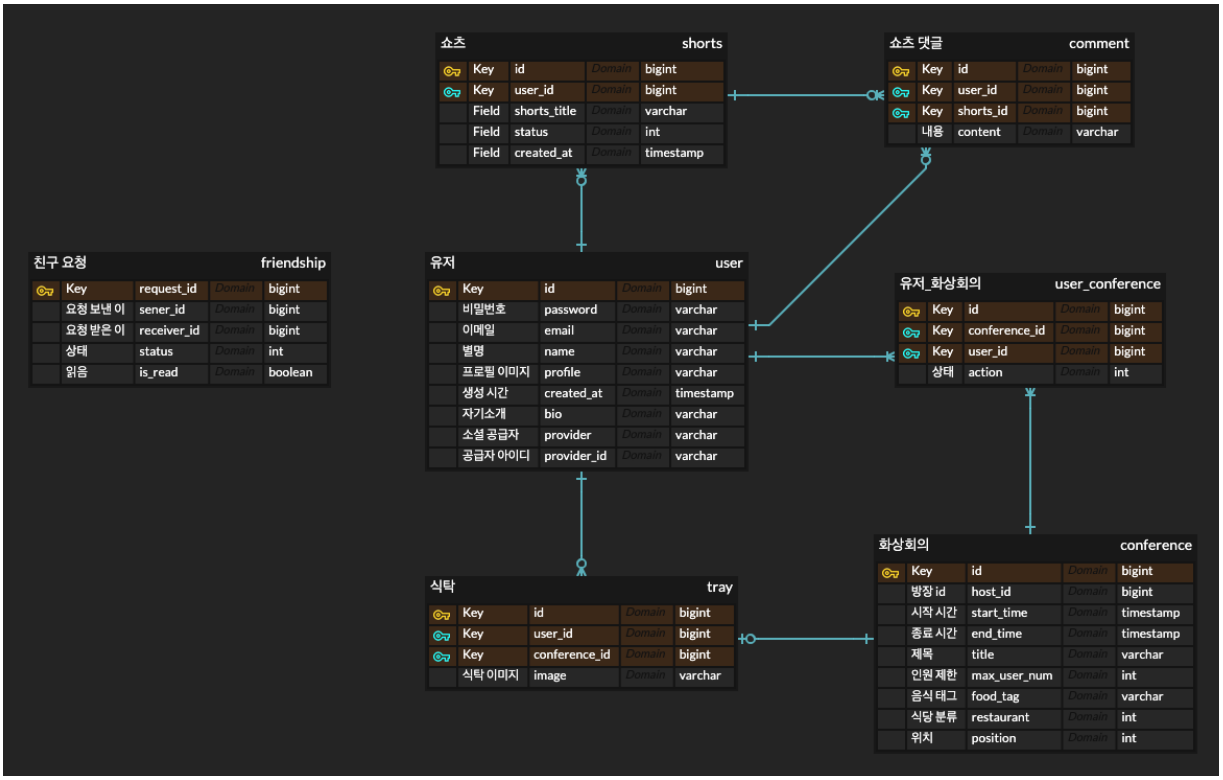
API Server

`application.yml`

```
jpa:
  hibernate:
    ddl-auto: update
  properties:
    hibernate:
      format_sql: true #To beautify or pretty print the SQL
      show_sql: true #show sql

logging:
  level:
    org.hibernate:
      type.descriptor.sql: trace #show parameter binding
      SQL: DEBUG
    com:
      a105:
        kurento: DEBUG

oauth2:
  provider:
    google:
      client-id: 347493483528-h0eb923lqrqe12a39b1a044c6r0rjcik.apps.googleusercontent.com
      client-secret: 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
      token-uri: https://oauth2.googleapis.com/token
      user-info-uri: https://www.googleapis.com/oauth2/v1/userinfo
    kakao:
      client-id: b045f7299f24b09cfad7a21c8cd91ef6
      client-secret: AU3XmTowoE1VBoCqFK8YDWxtq4ctnTrO
      token-uri: https://kauth.kakao.com/oauth/token
      user-info-uri: https://kapi.kakao.com/v2/user/me
    naver:
      client-id: LEj86kXb7NOQ3MK3z7x0
      client-secret: ua1tiYD5Tr
      token-uri: https://nid.naver.com/oauth2.0/token
      user-info-uri: https://openapi.naver.com/v1/nid/me

jwt:
  secret-key: A1$0$5MeetEat-hayunA1$0$5MeetEat-hayunA1$0$5MeetEat-hayun
  access-token:
    expiry: 864000000
  refresh-token:
    expiry: 864000000

app:
  auth:
    tokenSecret: 04ca023b39512e46d0c2cf4b48d5aac61d34302994c87ed4eff225dcf3b0a218739f3897051a057f9b846a69ea2927a587044164b7bae5e1306219
    tokenExpirationMsec: 864000000
  cors:
    allowedOrigins: http://localhost:3000,http://localhost:8080, http://meeteat.shop, https://meeteat.shop
  oauth2:
    authorizedRedirectUris:
      - http://localhost:3000/oauth2/redirect

cloud:
  aws:
    instance:
      data:
        enabled:
          - false
    credentials:
      access-key: AKIA2NWJP6PSU63KCJPL
      secret-key: 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
    region:
      static: ap-northeast-2 # 리전 정보
    stack:
      auto: false
    s3:
      bucket: meeteat-bucket
    cloudfront:
      domain: https://d2ju23e8zeqnl1.cloudfront.net/



#local환경
---
spring:
  config:
    activate:
      on-profile: dev
  datasource:
    driver-class-name: org.mariadb.jdbc.Driver
    url: jdbc:mariadb://localhost:3306/a105?characterEncoding=UTF-8&serverTimezone=UTC
    username: root
    password: admin

#aws환경
---
server:
  port: 8443
  ssl:
    key-store: classpath:keystore.p12
    key-store-type: PKCS12
    key-store-password: 13579

spring:
  config:
    activate:
      on-profile: prod
  datasource:
    driver-class-name: org.mariadb.jdbc.Driver
    url: jdbc:mariadb://mariadb:3306/a105?characterEncoding=UTF-8&serverTimezone=UTC
    username: root
    password: meeteat1234
```

Database

ERD



쇼츠				shorts
🔑	Key	id	Domain	bigint
🔑	Key	user_id	Domain	bigint
	Field	shorts_title	Domain	varchar
	Field	status	Domain	int
	Field	created_at	Domain	timestamp

쇼츠 댓글				comment
🔑	Key	id	Domain	bigint
🔑	Key	user_id	Domain	bigint
🔑	Key	shorts_id	Domain	bigint
	내용	content	Domain	varchar

친구 요청				friendship
🔑	Key	request_id	Domain	bigint
	요청 보낸 이	sener_id	Domain	bigint
	요청 받은 이	receiver_id	Domain	bigint
	상태	status	Domain	int
	읽음	is_read	Domain	boolean

유저				user
🔑	Key	id	Domain	bigint
	비밀번호	password	Domain	varchar
	이메일	email	Domain	varchar
	별명	name	Domain	varchar
	프로필 이미지	profile	Domain	varchar
	생성 시간	created_at	Domain	timestamp
	자기소개	bio	Domain	varchar
	소셜 공급자	provider	Domain	varchar
	공급자 아이디	provider_id	Domain	varchar

유저_화상회의				user_conference
🔑	Key	id	Domain	bigint
🔑	Key	conference_id	Domain	bigint
🔑	Key	user_id	Domain	bigint
	상태	action	Domain	int

식탁				tray
🔑	Key	id	Domain	bigint
🔑	Key	user_id	Domain	bigint
🔑	Key	conference_id	Domain	bigint
	식탁 이미지	image	Domain	varchar

화상회의				conference
🔑	Key	id	Domain	bigint
	방장 id	host_id	Domain	bigint
	시작 시간	start_time	Domain	timestamp
	종료 시간	end_time	Domain	timestamp
	제목	title	Domain	varchar
	인원 제한	max_user_num	Domain	int
	음식 태그	food_tag	Domain	varchar
	식당 분류	restaurant	Domain	int
	위치	position	Domain	int

Client

```
REACT_APP_CLIENT_PROTOCOL = "https"
REACT_APP_CLIENT_URL = "meeteat.shop"
REACT_APP_API_URL = "meeteat.shop/api"
REACT_APP_API_PROTOCOL = "https"
REACT_APP_WS_PROTOCOL = "wss"
REACT_APP_NAVER_CLIENT_ID = "LEj86kXb7NOQ3MK3z7x0"
REACT_APP_KAKAO_CLIENT_ID = "b045f7299f24b09cfad7a21c8cd91ef6"
REACT_APP_GOOGLE_CLIENT_ID = "347493483528-h0eb923lqrqe12a39b1a044c6r0rjcik.apps.googleusercontent.com"
```

DB 덤프

DB 덤프 파일 없어도 정상 동작합니다

밋잇 실행화면 설명서

https://s3-us-west-2.amazonaws.com/secure.notion-static.com/a5f847d5-d79a-4f6e-8beb-9fed4ecf8147/%EB%B0%8B%EC%9E%87_%EC%8B%A4%ED%96%89%ED%99%94%EB%A9%B4_%EC%BA%A1%EC%B2%98.pdf