1. 프로젝트 개요

2. 프로젝트 사용 도구

이슈 관리 : JIRA

형상 관리 : Gitlab

커뮤니케이션 : Notion, Mattermost

디자인 : Figma

UCC : Movavi

CI/CD : Jenkins

3. 개발 환경

Server : Ubuntu 20.04.4 LTS (GNU/Linux 5.15.0-1017-aws x86_64), Amazon S3
JVM : 11
Build Tool: Gradle
DB : mysql, redis
Node : v16.17.0

4. 외부 서비스

Google Mail SMTP

5. Gitignore 처리한 키들

```
*.yml
.gradle
build/
```

빌드

1. 환경변수 형태

application-jwt.yml

```
mail.smtp.auth=true
mail.smtp.starttls.required=true
mail.smtp.starttls.enable=true
mail.smtp.socketFactory.class=javax.net.ssl.SSLSocketFactory
mail.smtp.socketFactory.fallback=false
mail.smtp.port=465
mail.smtp.socketFactory.port=465

#admin ?? ??
AdminMail.id = 이메일
AdminMail.password = PWD
```

application-aws.yml

```
cloud:
  aws:
    credentials:
      access-key: S3액세스키
      secret-key: S3
    stack:
      auto: false
```

email.properties

```
mail.smtp.auth=true
mail.smtp.starttls.required=true
mail.smtp.starttls.enable=true
mail.smtp.socketFactory.class=javax.net.ssl.SSLSocketFactory
mail.smtp.socketFactory.fallback=false
mail.smtp.port=465
mail.smtp.socketFactory.port=465

#admin ?? ??
AdminMail.id = 구글이메일
AdminMail.password = 구글2단계인증PWD
```

2. EC2 세팅

A. Nginx

EC2에 Nginx 설치

```bash
sudo apt install nginx
```

Nginx 설치 확인

```bash
nginx -v
sudo service nginx start
```

도메인 입력해서 Nginx 웹페이지 확인

B. Docker

백엔드

```
docker build -t ygpark96/backend .
docker stop backend
docker rm backend
docker run -d -p 8080:8080 --name backend ygpark96/backend
```

프론트

```
docker build -t ygpark96/frontend .
docker run -d -p 8081:80 --name frontend ygpark96/frontend
```

C. Mysql

1. 빌드하기

1) Front
   npm run build
2) Back
   Gradle 실행

2. 배포하기

Nginx 설정

```
server{
        if ($host = i7a506.p.ssafy.io) {
                return 301 https://$host$request_uri;
        } # managed by Certbot


                listen 80 default_server;
                listen [::]:80 default_server;
                server_name i7a506.p.ssafy.io;
                return 404; # managed by Certbot


}
server {
                listen 443 ssl; # managed by Certbot
                listen [::]:443 ssl; # managed by Certbot
                ssl_certificate /etc/letsencrypt/live/i7a506.p.ssafy.io/fullchain.pem; # managed by
Certbot
                ssl_certificate_key /etc/letsencrypt/live/i7a506.p.ssafy.io/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_name i7a506.p.ssafy.io;

                location / {
                                proxy_pass http://127.0.0.1:8081;
                }

                location /api {
                                # First attempt to serve request as file, then
                                # as directory, then fall back to displaying a 404.
                                #try_files $uri $uri/ =404;
                                error_page 405 = $uri;
                                proxy_redirect off;
                                charset utf-8;

                                proxy_pass http://i7a506.p.ssafy.io:8080;
                                proxy_set_header X-Real-IP $remote_addr;
                                proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
                                proxy_set_header X-Forwarded-Proto $scheme;
                                proxy_set_header X-NginX-Proxy true;
                                proxy_set_header Host $http_host;
                }
}
user www-data;
worker_processes auto;
pid /run/nginx.pid;
include /etc/nginx/modules-enabled/*.conf;

events {
	worker_connections 768;
	# multi_accept on;
}

http {
	client_max_body_size 50M;

	sendfile on;
	tcp_nopush on;
	tcp_nodelay on;
	keepalive_timeout 65;
	types_hash_max_size 2048;

	include /etc/nginx/mime.types;
	default_type application/octet-stream;

	ssl_protocols TLSv1 TLSv1.1 TLSv1.2 TLSv1.3; # Dropping SSLv3, ref: POODLE
	ssl_prefer_server_ciphers on;

	access_log /var/log/nginx/access.log;
	error_log /var/log/nginx/error.log;

	gzip on;

	include /etc/nginx/conf.d/*.conf;
	include /etc/nginx/sites-enabled/*;
}
```

3.