

말하길

기술 명세서

Team. 쌍둥이 사자, 두 명의 사수, 그리고 물병 안의 물고기
전종헌, 박건후, 박은규, 서지현, 오연주

목차

| | |
|-------------------------------|---|
| I. 개요 | 2 |
| 1. 프로젝트 개요 | 2 |
| 2. 프로젝트 사용 도구 | 2 |
| 3. 개발환경 | 2 |
| 4. 외부 서비스 | 3 |
| 5. .GITIGNORE 처리한 핵심 키들 | 3 |
| II. 빌드 | 3 |
| 1. 환경변수 | 3 |
| 2. 빌드 | 5 |
| 3. 배포 | 5 |

I. 개요

1. 프로젝트 개요

2022 년 통계청이 발표한 자료에 따르면, 뇌졸중 환자의 10 명 중 3 명은 후유증으로 언어 장애를 겪는다. 현재 12 만명이 언어 장애를 겪고있고, 이 수는 계속해서 증가하고 있다. 하지만 언어 치료는 약물 치료와 언어 치료가 병행되어야 하는데, 언어 치료는 보험급여가 적용되지 않아 환자가 느끼는 경제적 부담이 크다. 또한, 장애 특성 상 움직임이 제한되고, 언어 재활치료 시설이 부족하여 접근성이 제한되는 문제가 존재한다.

본 프로젝트 명인 '말하길'은 환자가 잘 말할 수 있기를 바란다는 의미와, 환자가 잘 말할 수 있게 인도해주는 길이라는 두가지를 의미한다. '말하길'은 언어 훈련과 화상 상담 기능을 제공한다. 언어 훈련은 인공지능을 통한 언어 훈련 프로그램을 제공하여 환자들의 언어 능력을 개선을 유도한다. 화상 상담은 환자와 상담사를 매칭하여 화상 상담을 받을 수 있도록 한다.

2. 프로젝트 사용 도구

이슈 관리 : Jira

형상 관리 : Gitlab

커뮤니케이션 : Notion, Mattermost

디자인 : Figma

UCC : Premier pro

CI/CD : Jenkins

Server: Amazon EC2

IDE: IntelliJ, Visual Studio Code

AI: Google Colab

Web: OpenVidu

3. 개발환경

Java: OpenJDK 11

SpringBoot: 2.7.13

Database: MariaDB 10.11.4 & Redis 3.0.504

Node.js: 18.16.1

Python: 3.9

React: 18.2.0

Reduxjs/toolkit: 1.9.5

tensorflow-gpu: 2.12.0

4. 외부 서비스

OpenAPI : Social 로그인을 위한 API, Email 인증을 위한 API

5. .gitignore

React: .env

Spring : application.yml, application.properties

II. 빌드

1. 환경변수

.env:

```
REACT_APP_OPENVIDU=  
REACT_APP_CLIENT_ID=  
REACT_APP_RESPONSE_TYPE=  
REACT_APP_REDIRECT_URI=  
REACT_APP_STATE=
```

.application.yml:

```
spring:  
  security:  
    oauth2:  
      client:  
        registration:  
          naver:  
            client-id:  
            client-secret:  
            redirect-uri:  
        provider:  
          naver:  
            authorization-uri:  
            token-uri:  
            user-info-uri:  
            user-name-attribute:
```

.application.properties

```
spring.datasource.driver-class-name=
spring.datasource.username=
spring.datasource.password=
spring.datasource.url=
spring.jpa.hibernate.ddl-auto=

spring.jpa.database-platform=

spring.mvc.pathmatch.matching-strategy=

jwt.secret=

OPENVIDU_URL:
OPENVIDU_SECRET:

# logger 관련 설정
spring.output.ansi.enabled=
# 로그 메시지가 저장되는 로그 디렉터리
logging.file.path=
# logging.level.{패키지 경로}를 통해 로깅 레벨을 결정할 수 있음
logging.level.gil.toy.medihealth=

# profile 사진 ec2 경로에 맞게 FE 와 매칭해야함!!
file.path.profilePath=

# profile 사진 크기 변경
spring.servlet.multipart.max-file-size=
spring.servlet.multipart.max-request-size=

# email 전송관련 !! 보안 필수
spring.mail.host=
spring.mail.port=
spring.mail.username=
spring.mail.password=
spring.mail.properties.mail.smtp.starttls.enable=
spring.mail.properties.mail.smtp.starttls.required=
spring.mail.properties.mail.smtp.auth=

## redis 설정
spring.redis.host=
spring.redis.port=

spring.security.oauth2.client.registration.naver.client-id=
spring.security.oauth2.client.registration.naver.client-secret
spring.security.oauth2.client.registration.naver.redirect-uri=
```

```
spring.security.oauth2.client.provider.naver.authorization-uri=  
spring.security.oauth2.client.provider.naver.token-uri=  
spring.security.oauth2.client.provider.naver.user-info-uri=  
spring.security.oauth2.client.provider.naver.user-name-attribute=
```

2. 빌드

1) FrontEnd

.env 작성
npm install
npm start

2) BackEnd

application.properties, application.yml 작성
Gradle 실행
Bootjar 실행

3) AI

Model 다운받기
Docker build -t {image-name} .
Docker run -d -p 5000:5000 {image-name}

3. 배포

Jenkins 설정

```
# Required for Jenkins websocket agents  
map $http_upgrade $connection_upgrade {  
    default upgrade;  
    " close;  
}
```

```
server {  
    listen 8081 ssl;  
    listen [::]:8081 ssl;  
    server_name i9c109.p.ssafy.io;  
  
    # SSL Config  
    ssl_certificate      /etc/letsencrypt/live/i9c109.p.ssafy.io/fullchain.pem;  
    ssl_certificate_key  /etc/letsencrypt/live/i9c109.p.ssafy.io/privkey.pem;
```

```
ssl_trusted_certificate /etc/letsencrypt/live/i9c109.p.ssafy.io/fullchain.pem;

ssl_session_cache shared:SSL:50m;
ssl_session_timeout 5m;
ssl_stapling on;
ssl_stapling_verify on;

ssl_protocols TLSv1.2 TLSv1.3;
ssl_ciphers "ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-
SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-
ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-
SHA256:DHE-RSA-AES256-GCM-SHA384";
ssl_prefer_server_ciphers off;

add_header Strict-Transport-Security "max-age=63072000" always;

# pass through headers from Jenkins that Nginx considers invalid
ignore_invalid_headers off;

location ~ "^/static/[0-9a-fA-F]{8}\W/(.*)$" {
    # rewrite all static files into requests to the root
    # E.g /static/12345678/css/something.css will become /css/something.css
    rewrite "^/static/[0-9a-fA-F]{8}\W/(.*)" /$1 last;
}

location /userContent {
    # have nginx handle all the static requests to userContent folder
    # note : This is the $JENKINS_HOME dir
    root /var/lib/jenkins/;
    if (!-f $request_filename){
        # this file does not exist, might be a directory or a /**view** url
        rewrite (.*?) /$1 last;
        break;
    }
    sendfile on;
```

```

}

location / {
    sendfile off;
    proxy_pass      http://localhost:8080;
    proxy_redirect  default;
    proxy_http_version 1.1;

    # Required for Jenkins websocket agents
    proxy_set_header Connection      $connection_upgrade;
    proxy_set_header Upgrade         $http_upgrade;

    proxy_set_header Host            $http_host;
    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_max_temp_file_size 0;

    #this is the maximum upload size
    client_max_body_size      10m;
    client_body_buffer_size   128k;

    proxy_connect_timeout     90;
    proxy_send_timeout        90;
    proxy_read_timeout        90;
    proxy_buffering            off;
    proxy_request_buffering    off; # Required for HTTP CLI commands
    proxy_set_header Connection ""; # Clear for keepalive
}
}

```

AI 서버(Flask)

```

# Required for Jenkins websocket agents
map $http_upgrade $connection_upgrade {
    default upgrade;
    "" close;
}

```



```
}

server {
    listen 5001 ssl;
    listen [::]:5001 ssl;
    server_name i9c109.p.ssafy.io;

    # SSL Config
    ssl_certificate      /etc/letsencrypt/live/i9c109.p.ssafy.io/fullchain.pem;
    ssl_certificate_key  /etc/letsencrypt/live/i9c109.p.ssafy.io/privkey.pem;
    ssl_trusted_certificate /etc/letsencrypt/live/i9c109.p.ssafy.io/fullchain.pem;

    ssl_session_cache shared:SSL:50m;
    ssl_session_timeout 5m;
    ssl_stapling on;
    ssl_stapling_verify on;

    ssl_protocols TLSv1.2 TLSv1.3;
    ssl_ciphers "ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384";
    ssl_prefer_server_ciphers off;

    add_header Strict-Transport-Security "max-age=63072000" always;

    # pass through headers from Jenkins that Nginx considers invalid
    ignore_invalid_headers off;

    location ~ "^/static/[0-9a-fA-F]{8}W/(.*)$" {
        # rewrite all static files into requests to the root
        # E.g /static/12345678/css/something.css will become /css/something.css
        rewrite "^/static/[0-9a-fA-F]{8}W/(.*)" /$1 last;
    }
}
```

```
location /userContent {
    # have nginx handle all the static requests to userContent folder
    # note : This is the $JENKINS_HOME dir
    root /var/lib/jenkins/;
    if (!-f $request_filename){
        # this file does not exist, might be a directory or a /**view** url
        rewrite (.*?) /$1 last;
        break;
    }
    sendfile on;
}

location / {
    sendfile off;
    proxy_pass      http://localhost:5000;
    proxy_redirect   default;
    proxy_http_version 1.1;

    # Required for Jenkins websocket agents
    proxy_set_header Connection      $connection_upgrade;
    proxy_set_header Upgrade         $http_upgrade;

    proxy_set_header Host            $http_host;
    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_max_temp_file_size 0;

    #this is the maximum upload size
    client_max_body_size      10m;
    client_body_buffer_size   128k;

    proxy_connect_timeout     90;
    proxy_send_timeout        90;
    proxy_read_timeout        90;
```

```
proxy_buffering      off;  
proxy_request_buffering  off; # Required for HTTP CLI commands  
proxy_set_header Connection ""; # Clear for keepalive  
}  
}
```