**REPORT**

**오픈소스SW기여 최종 보고서**

**과 목 명 : 오픈소스SW기여**

**담당교수 : 송 인 식**

**학 과 : 소프트웨어**

**학 번 : 32144794**

**이 름 : 최 원 제**

**제 출 일 : 06. 13.**

**목차**

1. **프로젝트 배경**
2. **제공 기능**
3. **시스템 구조 및 설계**
4. **구현 과정**
5. **테스트 결과 및 분석**
6. **후기**
7. **참고 문헌**
8. **프로젝트 배경**

ChatGPT의 출시와 함께 세간의 주목을 받게 된 OpenAI에서는 ChatGPT와 함께 다양한 AI 모델 서비스들을 API의 형태로 제공하고 있다.

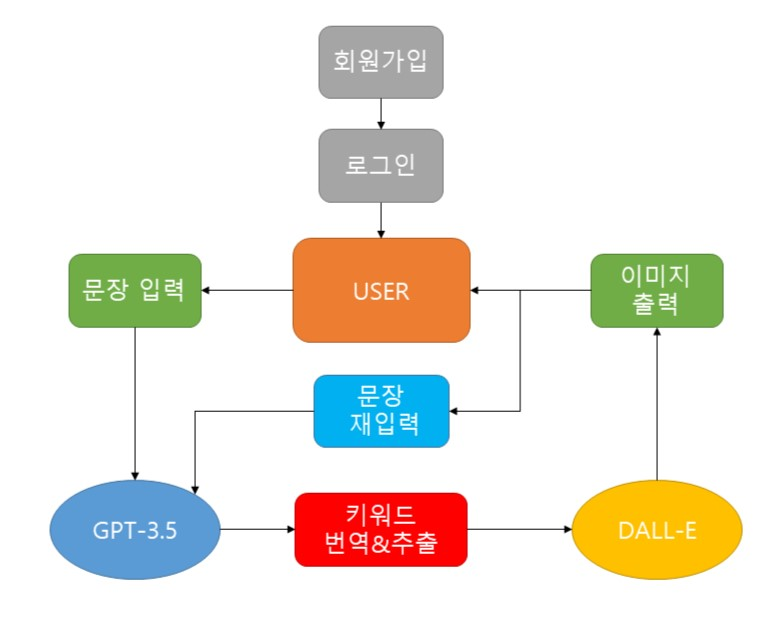
하지만 공식 문서들에서는 이를 활용하는 예시로 python 언어를 제시하고 있으며 Spring boot에 사용되는 Java는 존재하지 않고 구글링을 해 얻을 수 있는 방법들은 다소 복잡하였다.

이는 google이나 다른 유명 AI 모델들의 API서비스 들에서도 비슷하였는데, IntelliJava라는 한 오픈소스를 통해 의존성 주입의 방식으로 python과 같이 간편한 연동이 가능하다는 사실을 알게 되었다.

따라서 본 프로젝트에서는 해당 의존성을 사용해 OpenAI의 GPT-3.5와 DALL-E 2 API를 활용한 이미지 생성 웹 서비스를 구현해 볼 것이다.

이를 통해 Spring boot웹 프레임워크를 사용하는 개발자들에게 간단한 AI 모델 API 연동 레퍼런스를 제시하고, 일반 사용자들에게는 OpenAI에서 제공하는 이미지 생성 서비스를 어렵지 않게 체험해 볼 수 있는 환경을 제공 하려한다.

1. **제공 기능**



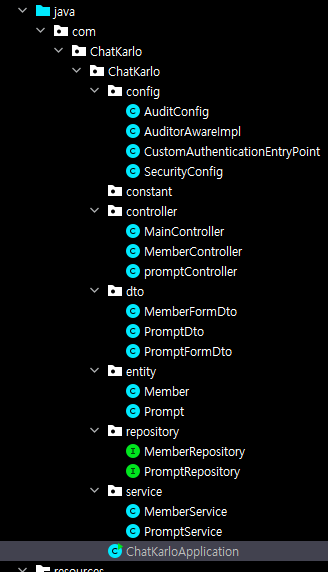
해당 웹 서비스의 사용자는 회원가입과 로그인 과정을 거쳐 이미지 생성 서비스를 사용할 수 있다. 이는 GPT-3.5와 DALL-E 서비스가 요금제로 이루어지기 때문에 만든 최소한의 제한이다.

이후 사용자가 이미지 생성을 위한 정리되지 않은 한국어 자연어 문장을 입력하게 되면 GPT-3.5에서 이를 DALL-E가 사용할 수 있는 간단한 영어 문장 형태의 지시어로 번역 및 추출한다.

이렇게 생성된 지시어를 DALL-E에 보내 1024x1024 사이즈의 이미지 1개를 생성한 뒤 출력하게 된다.

해당 이미지와 입력한 문장은 계정마다 데이터화 되어 저장되기 때문에 사이트를 나간 뒤 다시 들어와도 확인할 수 있으며 다른 자연어를 재입력하게 되면 GPT-3.5와 DALL-E를 거쳐 새로운 이미지를 생성하게 된다.

1. **시스템 구조 및 설계**



해당 웹 서비스는 Spring boot 웹 프레임워크를 사용해 만들어졌다. 따라서 프로그래밍 언어로 JAVA를 사용하며 디자인 패턴으로 MVC 2 패턴을 사용하고 있다.

우선 DB의 경우 JPA를 사용하여 회원 정보와 지시어 및 이미지 데이터를 관리하고 있으며 각각의 테이블의 구조는 아래의 표와 같다.

|  |  |
| --- | --- |
| Member | 회원 |
| NO | NAME | DATA TYPE | PK | FK | AI | NULLABLE |
| 1 | member\_id | Long | O |  | O |  |
| 2 | email | String |  |  |  |  |
| 3 | password | String |  |  |  |  |
| 4 | nickname | String |  |  |  |  |

|  |  |
| --- | --- |
| Prompt | 제시어 |
| NO | NAME | DATA TYPE | PK | FK | AI | NULLABLE |
| 1 | prompt\_id | Long | O |  | O |  |
| 2 | beforeWords | String |  |  |  | O |
| 3 | afterWords | String |  |  |  | O |
| 4 | url | String |  |  |  | O |
| 5 | member\_id | Long |  | O |  |  |

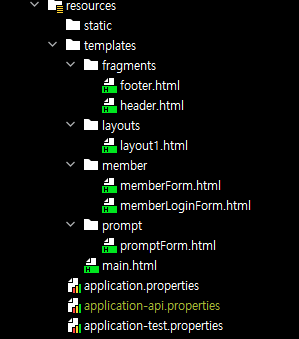
(PK: 기본 키, FK: 외래 키, AI: 자동 증가(Auto Increase), NULLABLE: NULL값 가능)

회원의 이메일, 비밀번호, 닉네임을 담고있는 Member와 지시어 및 이미지 정보를 가지고 있는 Prompt 테이블로 구성이 되어있으며

Prompt의 경우 사용자가 입력한 자연어 문장은 beforeWords, GPT-3.5가 이를 번역 및 추출한 지시어는 afterWords, 해당 지시어를 통해 DALL-E가 생성한 이미지의 url을 url에 저장하여 관리하게 된다.

또한 Member와 Prompt 간에는 1:1 매핑 관계가 있어 한 명의 Member가 하나의 Prompt 데이터를 가지게 되며 이를 통해 자신이 생성한 이미지와 문장 입력을 관리할 수 있게 된다.

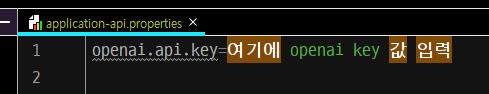
이를 바탕으로 view에서 회원정보와 지시어 관련 데이터를 요청하거나 보내게 되면 이 데이터를 Dto의 형태로 담아 Controller를 통해 view 와 DB를 오가는 것이다.



여기서 view의 경우 위의 사진과 같이 header, content, footer로 구성된 thymeleaf layout을 통해 구성하였으며 header 항목에 있는 메뉴를 클릭하거나 각각의 기능에 따라 content 항목에 들어갈 html 파일들이 전환되는 구조이다..

또한 해당 웹 서비스의 구동에 필요한 설정 관련 정보들은 application.propties에 저장되어 사용되는데 이 파일 또한 GitHub에 업로드가 되기 때문에 API Key값과 같이 공유가 불가능한 정보는 application-api.properties 파일에 따로 저장한 뒤 GitHub에 업로드하지 않았다.

따라서 해당 웹 서비스를 GitHub에서 clone해 구동하기 위해서는 직접 application-api.properties 파일을 작성해야 하며 아래의 예시처럼 작성하여 저장하면 된다.



1. **구현 과정**

|  |  |  |  |
| --- | --- | --- | --- |
| 분류 | | | 세부사항 |
| 1depth | 2depth | 3depth |
| 로그인 | 로그인 성공 |  |  |
| 로그인 실패 | 잘못된  이메일 입력 | 해당 이메일을 가진 회원이 존재하지 않는다는 텍스트를 출력할 것 |
| 잘못된  비밀번호 입력 | 해당 이메일을 가진 계정의 비밀번호가 아니라는 정보를 텍스트로 출력 할 것 |
| 회원가입 | 회원가입  성공 |  | 이메일, 비밀번호, 닉네임의 정보를 입력 받아 회원가입을 진행 할 것 |
| 회원가입  실패 | 중복된  이메일 입력 | 중복된 이메일이라는 알림 텍스트 출력할 것 |
| 규칙에 벗어난  비밀번호 입력 | 비밀번호의 길이는 8글자 이상이 될 것 |
| 중복된  닉네임 입력 | 중복된 닉네임을 입력 시 “해당 닉네임+랜덤 2자리 숫자”의 닉네임을 추천할 것 |
| GPT-3.5 | 문장 입력 |  | 특정 회원의 자연어 문장 데이터에 입력될 것 |
| 문장 재입력 | 문장을 재 입력 받아 특정 회원의 자연어 문장 데이터를 수정할 것 |
| 지시어  번역 및 추출 |  | 입력 받은 자연어 문장을 GPT-3.5를 통해 DALL-E 가 요구하는 간결한 영어 문장 형태의 지시어로 번역 및 추출할 것 |
| DALL-E | 이미지 생성 |  | 위의 추출된 지시어를 바탕으로 1024X1024 사이즈의 이미지 1개를 생성한 뒤 url 값을 보낼 것 |
| 기타 | 오류 안내 |  | 위의 GPT-3.5와 DALL-E의 지시어 추출 및 이미지 생성 과정에서 오류가 발생하였을 시 안내 창을 띄운 뒤 해당 과정을 중지할 것 |

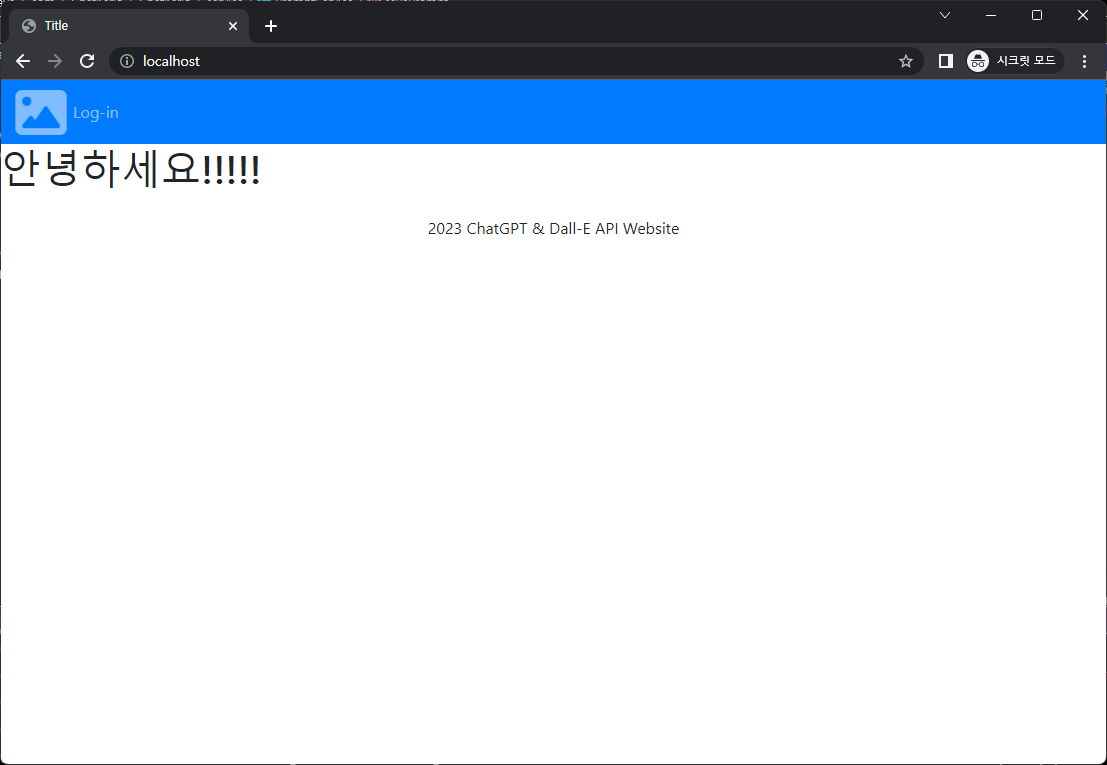
이번 프로젝트의 웹 서비스에서 구현한 기능들의 항목을 위의 표로 정리해보았다.

여기서 GPT-3.5의 지시어 번역 및 추출과 DALL-E의 이미지 생성 기능의 경우 IntelliJava의 의존성 모듈을 통해 간단한 함수의 형태로 구현할 수 있었다.

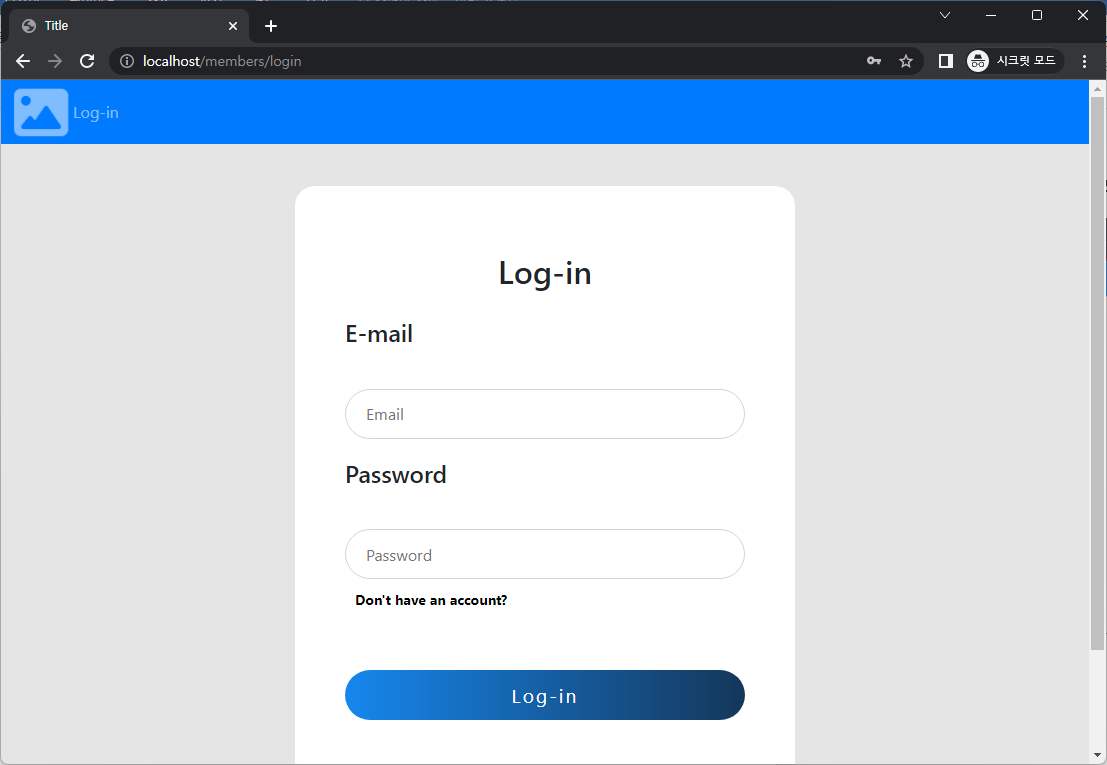
기능들을 테스트 및 구현하며 중점을 두었던 부분은 우선적으로 기능이 작동하는 것도 중요하지만 여러가지 이유로 인해 정상적인 작동이 불가능할 때 이를 안내하거나 적절한 처리를 하는 과정을 만드는 것 이였다.

이를 통해 사용자가 기능을 사용하던 도중 오류가 발생하더라도 이를 인지하고 적절한 대응을 하는 데 불편함이 없도록 하는 것이 목표였다.

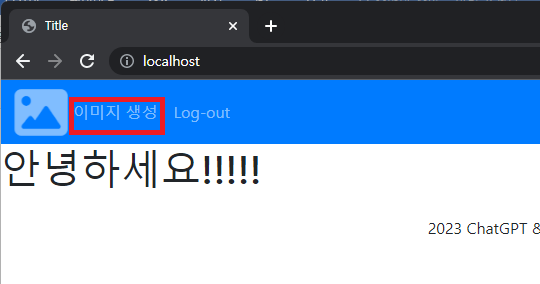
1. **테스트 결과 및 분석**



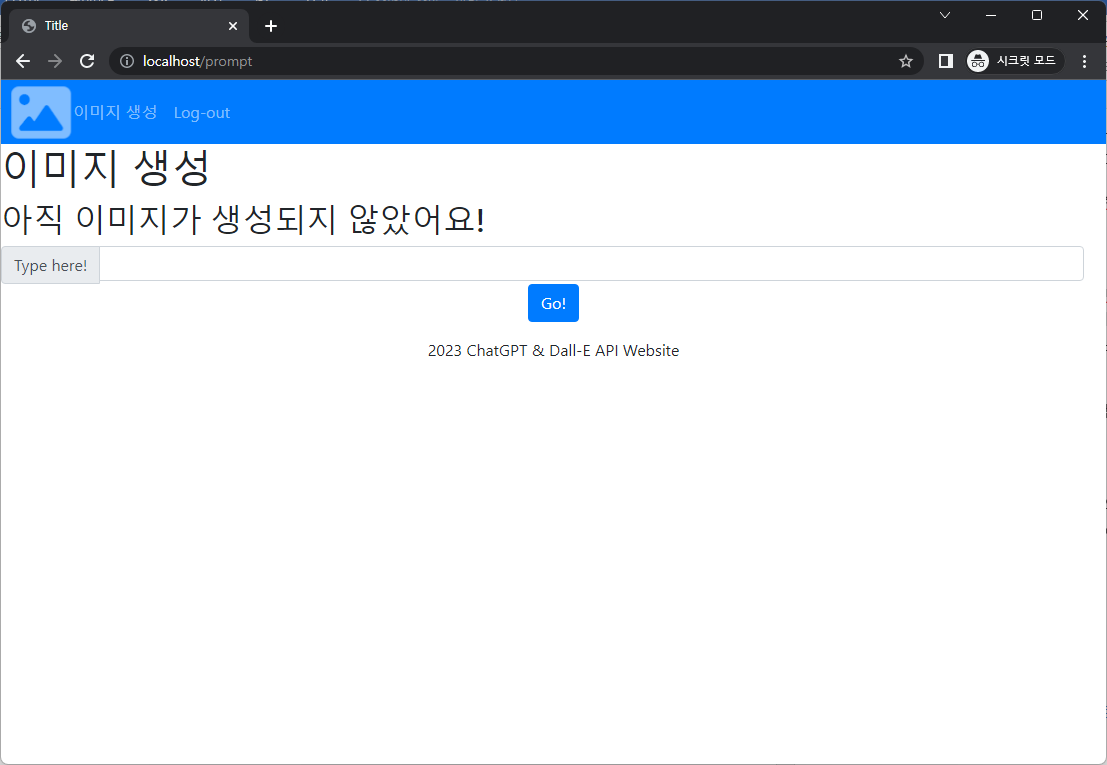
해당 웹 서비스에 접속하게 되면 간단한 메인 화면과 함께 상단 메뉴바에 로그인 메뉴가 뜨게 된다.



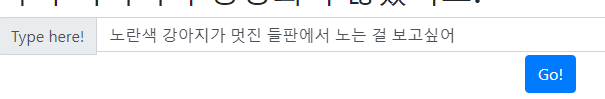
해당 메뉴로 들어가게 되면 로그인 화면이 뜨게 되며 하단의 Don’t have an account 글귀를 클릭하게 되면 회원가입 화면이 나와 이곳에서 계정을 생성할 수 있다.



이후 생성한 계정을 통해 로그인하게 되면 상단의 메뉴바가 이미지 생성과 로그아웃 메뉴로 변경된 것을 볼 수 있다.

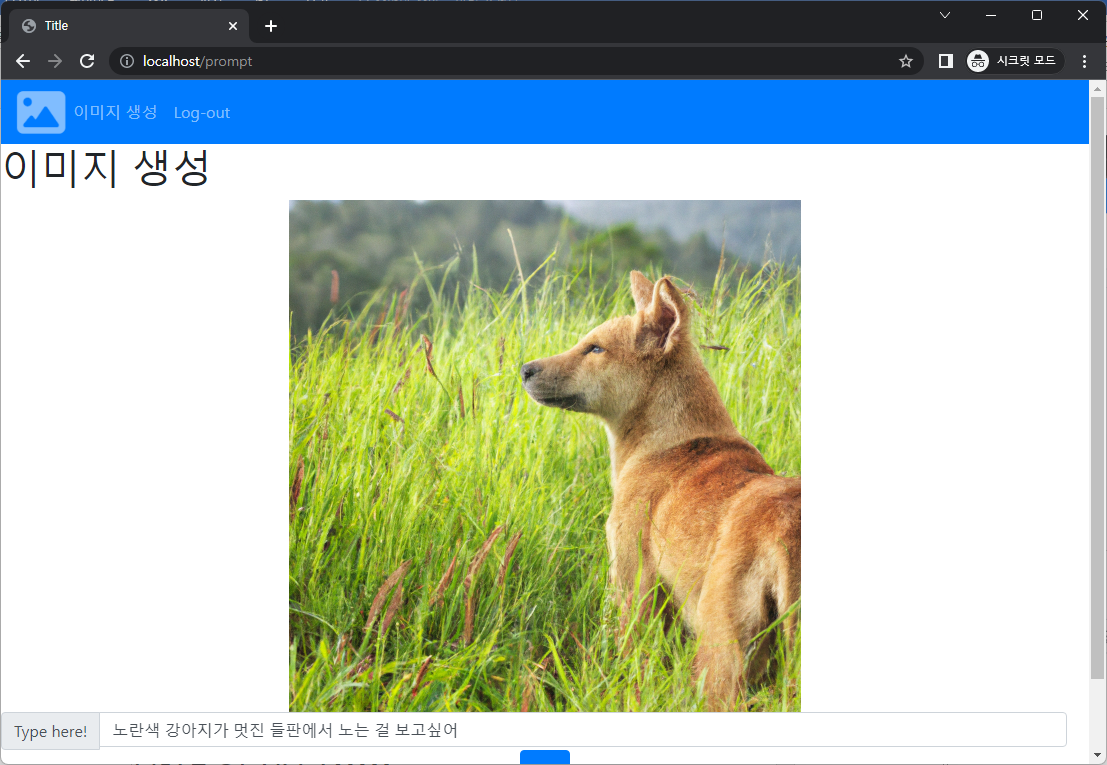


이미지 생성 메뉴를 클릭하게 되면 위 사진과 같이 이미지 생성을 위한 문장을 입력하는 창이 출력된다.



텍스트 입력 창에 원하는 문장을 입력한 뒤 Go! 버튼을 누르게 되면 잠시 로딩이 된 후 메인 화면으로 돌아오게 된다.

이미지 생성 과정에서 오류가 발생하였을 시 이를 알리는 안내창이 뜨고 문장 입력창이 초기화된 상태로 화면이 새로고침 된다.



이미지 생성 과정이 정상적으로 진행되어 메인 화면으로 가게 되었을 시 다시 이미지 생성 메뉴를 누르게 되면 이전에 입력하였던 문장과 해당 문장을 번역 및 추출한 지시어를 통해 생성된 이미지가 화면에 노출된다.

이 상태에서 입력되었던 문장을 수정하거나 다시 작성하여 이미지를 재생성 할 수 있다.

1. **후기**

해당 프로젝트는 본래 GPT-3.5 API와 Kakao 사에서 제공하는 이미지 생성 오픈소스인 Karlo API를 통해 제작될 예정이었다.

하지만 Karlo API의 JAVA 연동 및 활용 과정에 문제가 생겼고 이를 긴 시간 동안 해결하지 못하고 막바지에 다다라 포기하게 되었다.

이를 해결하기 위해 OpenAI에서 제공하는 이미지 생성 AI 모델인 DALL-E를 사용하였으나 GPT-3.5와 DALL-E는 오픈소스가 아닌 요금을 내야하는 서비스였기 때문에 본래 프로젝트의 취지에 벗어나게 되어 고민이 많았다.

하지만 그러던 와중 openAI와 여러 유명 AI 모델 API 서비스들을 Spring boot에서 연동하기 쉽게 만들어주는 오픈소스인 IntelliJava를 알게 되었고 이를 활용하게 되면서 오픈소스SW기여 프로젝트라는 원래의 취지를 되찾을 수 있게 되었다.

하지만 프로젝트 기간 막바지에 다다른 상태에서 생긴 변화였기에, 기존에 구상하였던 기능들 에서 많은 요소들이 빠진 결과물이 나오게 되어 아쉬운 점이 있다.

이후에 추가하고 싶은 요소로는 SNS로그인과 음성 인식 API 서비스의 접목을 통한 접근성 향상, 자신이 생성한 이미지 목록 확인 및 관리 기능, UI 디자인 개선 등이 있다.

1. **참고 문헌**

* “IntelliJava”, Barqawiz GitHub,   
  <https://github.com/Barqawiz/IntelliJava>
* “OpenAI Document”, OpenAI API, <https://platform.openai.com/docs/introduction>