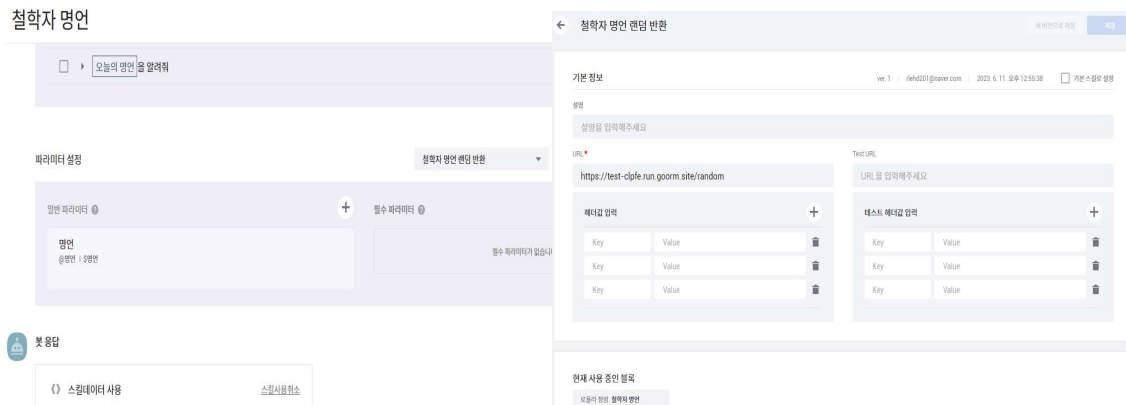


비록 이번 프로젝트를 통해 개발하고자 하는 것이 철학적 챗봇이며 철학적 논의를 전개하는 것이 주된 목적이라 할지라도 사용자와 대화를 나누는 챗봇이라는 특성상 가볍게 사용자들이 요구할 수 있을 기능을 추가하는 것이 필요하다고 생각했다. 이에 따라 철학자의 명언을 랜덤하게 이야기해줄 수 있는 기능을 구현하기 위하여 챗봇의 '스킬'을 활용하였다.



기존 시나리오에서 “철학자 명언을 추천해줘”, “철학자 글귀를 들려줘”, “명언 알려줘” 등과 같은 입력이 주어질 때 철학자 명언을 반환하는 시나리오가 인식되기 원활하도록 ‘명언’을 엔티티로 설정하였다. 그리고 이러한 입력이 주어질 때 파라미터 설정 부분의 우측에 있는 곳에 기존에 설계한 스킬 데이터를 할당하고 봇 응답을 ‘스킬 데이터 사용’으로 설정하면 설정한 스킬 데이터를 이용한 답변이 가능해진다.

스킬 데이터에서는 기존에 프로그래밍한 코드를 바탕으로 작동하는 서버의 URL을 연결시키는 방식으로 설계하였다. 이를 통해 챗봇에 사용자가 관련된 입력을 하게 된다면 등록된 URL을 타고 이동하여 구축되어 있는 서버에서 해당 입력을 인식하고 기존에 설정한 12개의 명언 중 랜덤하게 1개의 명언을 챗봇이 반환할 수 있게 된다. 12개의 명언에는 다음과 같은 항목들이 포함되어 있다.

1. 칸트: 네 의지의 준칙이 언제나 동시에 보편적 입법의 원리가 될 수 있도록 행위하라
2. 스피노자: 내일 지구가 멸망하더라도 오늘 한 그루 사과나무를 심겠다
3. 소크라테스: 너 자신을 알라
4. 니체: 이 인생을 기뻐하라. 즐겁게 살아가라
5. 사르트르: 실존은 본질에 앞선다
6. 하이데거: 언어는 존재의 집이다
7. 마르크스: 하나의 유령이 유럽을 떠돌고 있다, 공산주의라는 유령이

8. 헤겔: 미네르바의 부엉이는 황혼이 질 때만 날개를 편다
9. 공자: 군자는 의(義)를 밝히고 소인은 이(利)를 밝힌다
10. 노자: 도를 도라고 하면 이미 도가 아니다
11. 비트겐슈타인: 말할 수 있는 것에 대해 분명히 말하고 말할 수 없는 것에 대해 침묵하라
12. 아리스토텔레스: 아는 것에 의해서가 아니라 아는 것을 실천할 때 비로소 지혜로운 사람이 될 수 있다

```
1 from flask import Flask, jsonify
2 import sys
3 import random
4 application = Flask(__name__)
5
6
7 @application.route("/random", methods=["POST"])
8 def random_function():
9     rand = random.randint(1, 12)
10    if rand == 1:
11        message = "네 의지의 준칙이 언제나 동시에 보편적 입법의 원리가 될 수 있도록 행위하라 - 칸트"
12    elif rand == 2:
13        message = "내일 지구가 멸망하더라도 오늘 한 그루 사과나무를 심겠다 - 스피노자"
14    elif rand == 3:
15        message = "너 자신을 알라 - 소크라테스"
16    elif rand == 4:
17        message = "이 인생을 기뻐하라. 즐겁게 살아가라 - 니체"
18    elif rand == 5:
19        message = "실존은 본질에 앞선다 - 사르트르"
20    elif rand == 6:
21        message = "언어는 존재의 집이다 - 하이데거"
22    elif rand == 7:
23        message = "하나의 유령이 유럽을 떠돌고 있다, 공산주의라는 유령이 - 마르크스"
24    elif rand == 8:
25        message = "미네르바의 부엉이는 황혼이 질 때만 날개를 편다 - 헤겔"
26    elif rand == 9:
27        message = "군자는 의(義)를 밝히고 소인은 이(利)를 밝힌다 - 공자"
28    elif rand == 10:
29        message = "도를 도라고 하면 이미 도가 아니다 - 노자"
30    elif rand == 11:
31        message = "말할 수 있는 것에 대해 분명히 말하고 말할 수 없는 것에 대해 침묵하라 - 비트겐슈타인"
32    elif rand == 12:
33        message = "아는 것에 의해서가 아니라 아는 것을 실천할 때 비로소 지혜로운 사람이 될 수 있다 - 아리스토텔레스"
34    else:
35        message = "오늘은 추천해드릴 명언이 떠오르지 않네요"
```

서버를 구현하는 코드는 Flask를 이용하여 작성되었다. Flask는 Python 웹 프레임워크로, 웹 애플리케이션과 API를 개발하는 데 사용됩니다. 간결하고 유연한 설계를 가지고 있는 것이 특징이며, URL 라우팅을 지원하여 요청된 URL에 대해 특정한 함수를 실행할 수 있도록 한다. 이를 통해 요청과 응답을 처리하고 필요한 데이터를 보여줄 수 있다. URL 라우팅이 되는 것이 장점이기 때문에 이번 카카오톡 챗봇의 스킬 데이터를 사용하는 부분에서 URL을 통한 라우팅이 기본적인 작동 구조이기 때문에 이에 부합하는 Flask를 사용하였다.

```
디버그  터미널  🔁  검색  리소스 모니터  린트
root@goorm:/workspace/test# python application.py 80
* Serving Flask app "application" (lazy loading)
* Environment: production
  WARNING: This is a development server. Do not use it in a production deployment.
  Use a production WSGI server instead.
* Debug mode: on
* Running on http://0.0.0.0:80/ (Press CTRL+C to quit)
* Restarting with stat
* Debugger is active!
* Debugger PIN: 573-959-735 172.17.0.1 - - [11/Jun/2023 03:58:52] "POST /random HTTP/1.1" 200 -
172.17.0.1 - - [11/Jun/2023 03:59:00] "POST /random HTTP/1.1" 200 -
172.17.0.1 - - [11/Jun/2023 03:59:04] "POST /random HTTP/1.1" 200 -
172.17.0.1 - - [11/Jun/2023 03:59:17] "POST /random HTTP/1.1" 200 -
```



위와 같이 서버를 작동시켜서 입력을 받는 중에는 챗봇에서 “철학자 명언”과 관련된 키워드가 입력되었을 때 기존에 미리 설정한 12개의 명언 중 랜덤하게 명언을 대답해주고 있는 것을 확인할 수 있다.

하지만 이러한 구조의 문제점은 서버를 구축하고 서버가 작동되고 있는 동안만 챗봇에 주어진 입력을 서버가 받아들이고 적절한 결과를 다시 챗봇에 보내줄 수 있다는 것이다. 이는 즉 서버가 작동하지 않는다면 명언에 관련된 부분은 제대로 작동하지 않는다는 것을 의미한다. 이번 프로젝트에서 서버를 구축하는데 사용한 사이트는 GoormIDE로 추가적인 결제를 통해 서버를 항상 켜두는 기능을 사용할 수 있어 실제로 챗봇을 배포하고 이 기능을 유지하는 것에는 문제가 없을 것이다. 하지만 실제로 챗봇을 상용화하지 않은 상황에서는 수동으로 서버를 작동시키고 있는 상황에서만 해당 기능이 작동한다.