

QA (배포용)

I. 주차별 진행 계획 (예정)

- 1주차 (10/19~10/20)
 - 공통 개발환경 세팅, 프로젝트 이해 (질문)
- 2주차 (10/23~10/27)
 - 프로젝트 기획, DB 구축
- 3주차 (10/30~11/03)
 - 기획 마무리(Deadline)
- 4주차 (11/06~11/10)
 - Demo 제작 (1주 내로 끝내기)
- 5주차 (11/13~11/17)
 - Product 제작
- 6주차 (11/20~11/24)
 - Product 제작
- 7주차 (11/27~12/01)
 - 발표 준비
- 비교
 - 기획안은 최종 보고서가 아니며 아키텍처 설계한 것을 보여줄 예정

II. 프로젝트 배경

- **Semantic Search**
 - 본래 의미는 “문맥을 고려한 탐색”
 - 기존의 복지로는 Keyword base search로 **정해진 키워드에 대해서만** 정확한 상세 검색 가능함. 이번 프로젝트에서 구현할 Semantic Search는 검색어의 문맥상의 의미를 파악하여 **검색어 상관 없이 직관적인 검색 가능**
 - 프로젝트 진행 방향(추후 질문)
 1. Graph DB를 사용하여 온톨로지 기반 Semantic search
 2. Vector DB를 사용한 Semantic search
- **프로젝트 추진 배경**
 - 복지로의 현재 문제점
 - 정해진 keyword를 포함한 검색어일 경우에만 제대로 작동하며

- 세부 항목을 추가로 체크해야 수렴적인 결과가 나옴.
- 따라서 **개인(사용자)에게 맞춤형된 복지 서비스 제공이 어려우며**
복지 혜택 대상자임에도 **정보가 없어서 받지 못하는 문제점**이 있다.

→ 형식에 얽매이지 않으면서, 개인에게 딱힌 상황만 입력해도 원하는 결과를 찾을 수 있으면 좋을 것이다.

III. 질문 내용

- **SSIS가 바라는것이 정확히 무엇인가?**
 - 기존 서비스(=복지로) target층을 위한 성능 개선 → 복지매칭서비스를 가장 자주 사용하는 연령 및 계층 대상
 - 새로운 target층을 유입시키는 것 → 복지매칭서비스를 앞으로 사용할 사람들의 편의성을 위한 개선
 - 복지로 사용자가 받을 수 있는데 '몰라서' 받지 못하는 복지 서비스들을 더 잘 찾아주는것 → 복지사각지대를 개선
- **복지로를 가장 많이 이용하는 사람은 누구인가? (연령대 or 사용 용도) -> 검색어의 형식 고려**
- **기존의 복지로가 채택한 검색방식이 무엇인가?**
 - 검색어에 내포된 Keyword를 매칭하여 해당하는 복지서비스를 찾아주는지?
 - 위에 더하여 세부항목을 추가로 체크함으로써 더 맞춤형 결과를 도출하는지?

※ (추가) VectorDB 활용 테스트

```
import chromadb

client = chromadb.PersistentClient() # 데이터를 파일에 저장

# Collections 생성
# posts = client.create_collection(
#     name="posts"
# )

# Collections 로드
posts = client.get_collection("posts")

# 임시 Document 생성
# post1 = '국민기초생활보장법에 따른 국민기초생활보장제도는 저소득층 수급권자에 대해 최저생활을 보장하고 자립을 유도하는 제도' # 맞춤형 기초생활보장제도
# post2 = '생계유지가 곤란한 저소득층으로 사전안내된 위기사유와 소득, 재산기준을 충족하는 가구에 대해 지원하는 제도' # 긴급복지 지원제도
# post3 = '안정적 주거생활이 필요한 분들에 한해 보다 저렴하게 주택을 공급받을 수 있도록 지원하는 제도' # 주택임대 제도
# post4 = '주택수리, 노후시설교체 등 주거생활 개선이 필요한분들이 보다 쾌적한 주거환경에서 생활할 수 있도록 지원하는 제도' # 주거환경개선지원

# 특정 Collections에 임시 Document 저장
# posts.add(
#     documents=[post1, post2, post3, post4],
#     ids=["1", "2", "3", "4"]
# )

# 테스트 1. '생계지원'과 유사한 Document 상위 3개 추출
result = posts.query(
    query_texts=['생계지원'],
    n_results=3
)

print(result)
```

```
# 테스트 2. '주택공급'과 유사한 Document 상위 3개 추출
result2 = posts.query(
    query_texts=['주택공급'],
    n_results=3
)

print(result2)

# 테스트 3. '주택지원'의 의미를 가진 문장과 유사한 Document 상위 3개 추출
result3 = posts.query(
    query_texts=['배가 너무 고프고, 좁고, 줄리고, 돈도 벌고 싶어요... 그러려면 안락한 보금자리가 필요할 거 같아요...'],
    n_results=3
)

print(result3)
```

```
# Results
# 테스트 1. '생계지원'과 유사한 Document 상위 3개 추출
# query_texts=['생계지원'],
{'ids': [['2', '3', '4']],
'distances': [[1.472671394742208, 1.6688863568594992, 1.7218888285280312]],
'metadatas': [[None, None, None]],
'embeddings': None,
'documents': [
    ['생계유지가 곤란한 저소득층으로 사전안내된 위기사유와 소득, 재산기준을 충족하는 가구에 대해 지원하는 제도',
    '안정적 주거생활이 필요한 분들에 한해 보다 저렴하게 주택을 공급받을 수 있도록 지원하는 제도',
    '주택수리, 노후시설교체 등 주거생활 개선이 필요한분들이 보다 쾌적한 주거환경에서 생활할 수 있도록 지원하는 제도']
]
}
# 테스트 2. '주택공급'과 유사한 Document 상위 3개 추출
# query_texts=['주택공급'],
{'ids': [['3', '1', '2']],
'distances': [[0.7140313884064894, 0.7172854397776197, 0.7454014487835212]],
'metadatas': [[None, None, None]],
'embeddings': None,
'documents': [
    ['안정적 주거생활이 필요한 분들에 한해 보다 저렴하게 주택을 공급받을 수 있도록 지원하는 제도',
    '국민기초생활보장법에 따른 국민기초생활보장제도는 저소득층 수급권자에 대해 최저생활을 보장하고 자립을 유도하는 제도',
    '생계유지가 곤란한 저소득층으로 사전안내된 위기사유와 소득, 재산기준을 충족하는 가구에 대해 지원하는 제도']
]
}
# 테스트 3. '주택지원'의 의미를 가진 문장과 유사한 Document 상위 3개 추출
# query_texts=['배가 너무 고프고, 좁고, 줄리고, 돈도 벌고 싶어요... 그러려면 안락한 보금자리가 필요할 거 같아요...'],
{'ids': [['2', '3', '4']],
'distances': [[0.581908355056316, 0.6484227347544917, 0.7700067381975219]],
'metadatas': [[None, None, None]],
'embeddings': None,
'documents': [
    ['생계유지가 곤란한 저소득층으로 사전안내된 위기사유와 소득, 재산기준을 충족하는 가구에 대해 지원하는 제도',
    '안정적 주거생활이 필요한 분들에 한해 보다 저렴하게 주택을 공급받을 수 있도록 지원하는 제도',
    '주택수리, 노후시설교체 등 주거생활 개선이 필요한분들이 보다 쾌적한 주거환경에서 생활할 수 있도록 지원하는 제도']
]
}
```