**프로젝트**

**상세 설계서**



컴퓨터 공학 종합설계 1조

12172133 이종혁

12172133 임유한

12181667 이지율

**목차**

1. **프로젝트 기획 의도**
2. **프로젝트 요구 사항**
3. **프로젝트에 포함될 기능 명세**
4. **주요 키워드 소개**
5. **유저 관련 - 회원가입/로그인**
6. **개요 및 요구사항**
7. **프론트엔드 UI 구현**
8. **백엔드 API 상세 구현**

**C. 일자리**

1. **개요 및 요구사항**
2. **프론트엔드 UI 구현내용**
3. **백엔드 API 구현**
4. **데이터 크롤링 파트**

**D. 의료/복지 관련**

1. **개요 및 요구사항**
2. **프론트엔드 UI 구현**
3. **백엔드 API 구현**

**E. 경제적 지원 관련**

1. **개요 및 개발하고자 하는 기능**
2. **프론트엔드 UI 구현**
3. **백엔드 API 구현**
4. **데이터 크롤링 파트**

**F. 관계 단절 방지**

1. **개요 및 개발하고자 하는 기능**
2. **프론트엔드 UI 구현**
3. **백엔드 구현**
4. **프로젝트 기술 요약/업무 분담/개발 일정**
5. **참고자료**

**1. 프로젝트 기획 의도**

* 2020 인구주택총조사에 따르면 1인 가구는 664만 가구로 전체 일반 가구의 31.7%를 차지하고 있다. 그리고 여기서도 노인 1인 가구의 비율은 현재는 30% 정도를 웃돌며, 20년 안에 45%까지 치솟을 것으로 보인다.
* 또한, 출산율의 하락으로 유례없는 속도로 초고령화 사회로 진입하는 양상을 보이고 있으며, 이는 노년층에 대한 케어 서비스의 중요성을 여실히 보여주고 있다.
* 지금까지의 키워드를 종합해보면 노년층의 1인 가구 증가가 큰 사회적 이슈가 되고 있다고 볼 수 있다. 따라서 디지털, 정보에 취약한 노년층에게 꼭 필요한 통합 맞춤형 서비스를 제공하여 혼자서도 살아갈수 있는 “자립”을 돕는 것을 목표로 하고자 함.

**2. 프로젝트 요구사항**

* UI의 간소화
  + 디지털 기기 사용에 취약한 노년층을 타겟팅하고 있는 서비스이기 때문에, 사용자단에서는 정말 사용하기 쉽도록 UI를 최대한 직관적이고 깔끔하게 만들 예정.
* 데이터의 시각화/필터링
  + 최대한 직관적으로 알아보기 쉬워야 하기 때문에 지도 API, 도표 등을 이용해서 데이터를 시각화하려고 함.
  + 직접 사용자가 설정하고 검색하는 복잡한 기능 보다는 최대한 백엔드/데이터 단에서 조금 추가적인 작업을 더 하더라도 최대한 유저 맞춤형으로 데이터를 필터링해 필요한 정보만을 제공하려고 함.

**3. 프로젝트에 포함될 기능 명세**

1. **주요 키워드 소개**

노년층이 가장 필요로 하는 서비스가 무엇일지를 먼저 토의한 후에, 크게 네 가지 키워드로 분류해서 정의함.

* 일자리
* 의료/건강
* 경제적 지원
* 관계 단절 방지

1. **유저 관련 - 회원가입/로그인**
2. **개요 및 요구사항**

* 노년층을 타겟팅한 어플리케이션 서비스이니만큼, 최대한 직관적으로 회원가입 화면을 구성해야 하며, 사용자 맞춤형으로 데이터를 필터링해 전송하기 위해 최대한 회원가입 시에 유저와 관련된 데이터를 세분화해서 입력하도록 할 예정.

1. **프론트엔드 구현**

* 해당 기능은 공통 부분, 일반 사용자 부분, 기업 회원 부분으로 구분된다. 기능의 설명은 공통 부분, 일반 부분, 기업 회원 부분 순으로 하기로 한다.

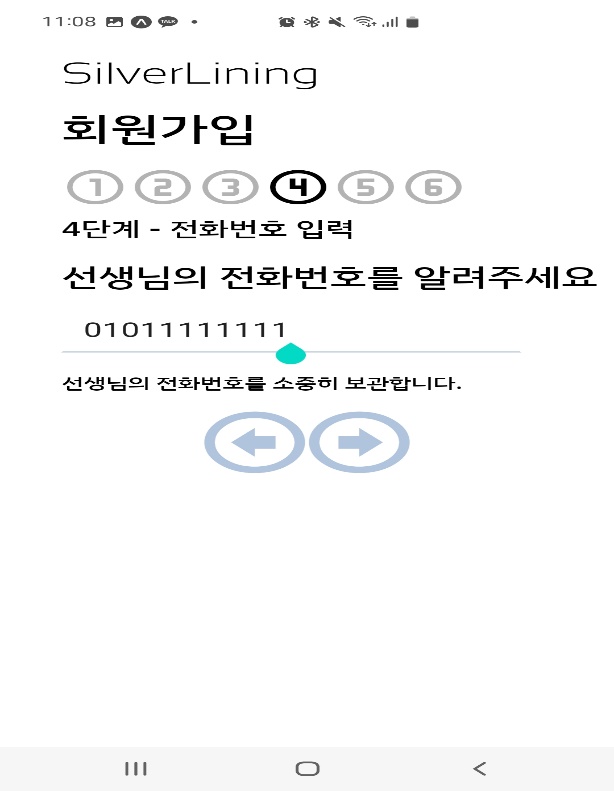
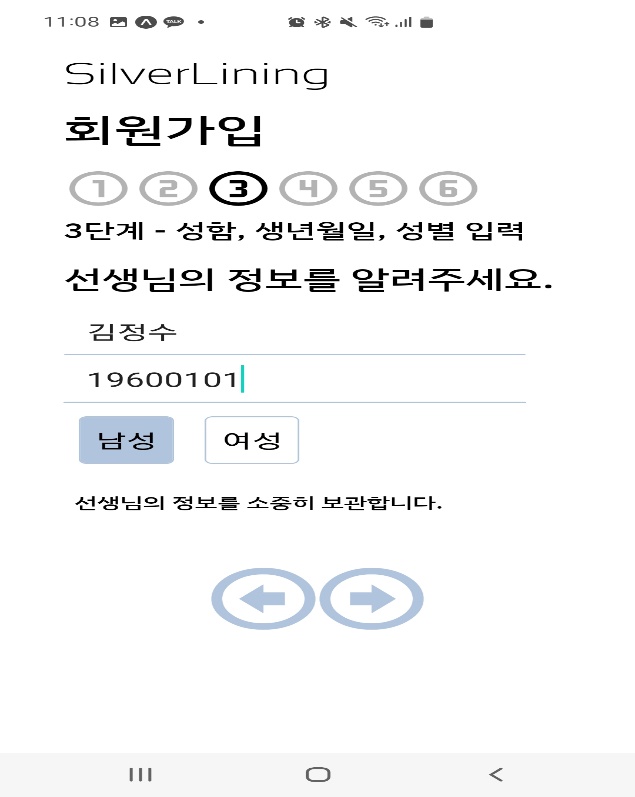
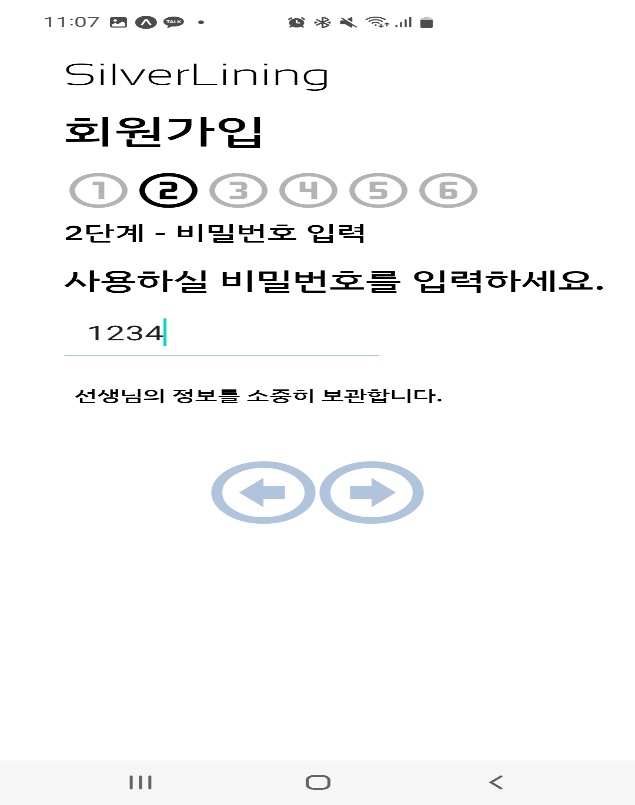
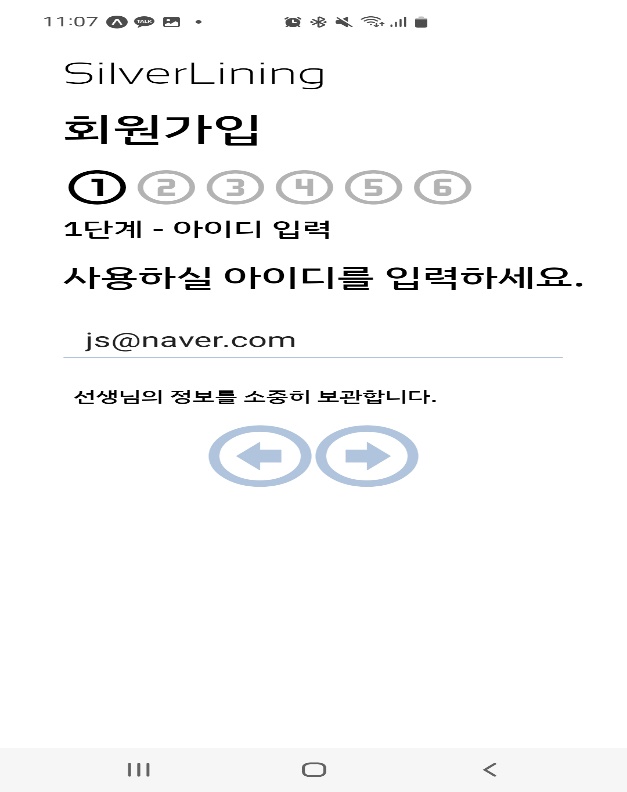
**공통 부분**

1. **타이틀 UI**

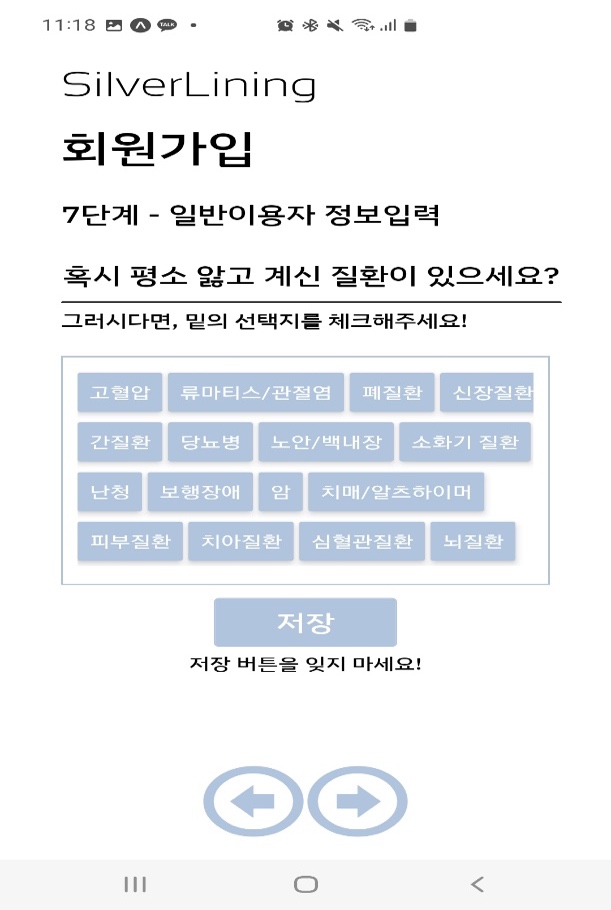
****

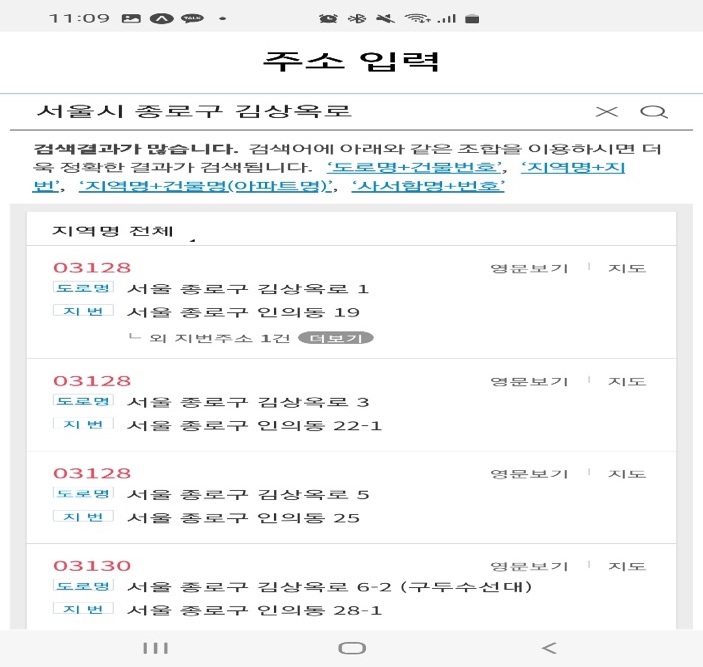
* 공통 부분은 회원가입을 위해 기본적으로 필요한 데이터를 입력 받는 역할을 한다.
* 위 페이지는 타이틀 페이지로, 로그인, 회원가입, 메인 페이지로 이동하는 navigation 역할의 버튼이다.

1. **회원가입 UI**

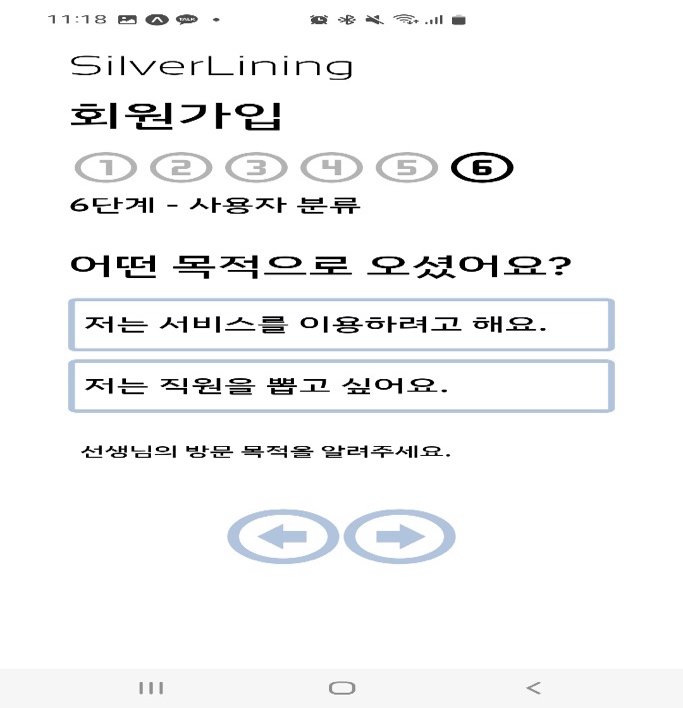


* 해당 페이지는 위와 같이 아이디, 비밀번호, 전화번호, 유저의 개인 정보 등을 입력 받는다. 이중 전화번호 데이터를 이용하여 인증 절차를 진행할 것이다.

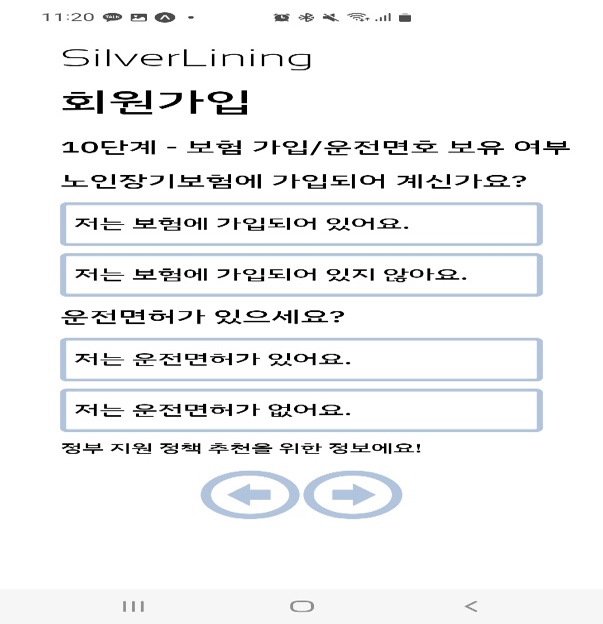


****

* 사용자의 주소를 입력 받는 부분은 카카오 api를 이용한다. 해당 api의 response data를 추출한 후, 유저 인터페이스를 통해 데이터를 프로젝트의 데이터베이스에 입력한다.



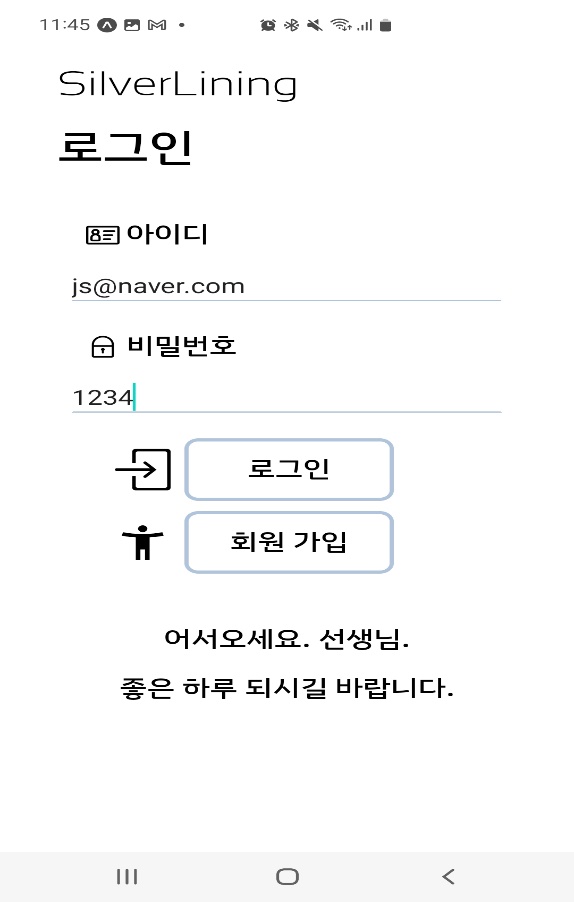
* 회원 가입의 공통 부분 마지막 단계에서는 사용자가 일반 사용자인지 기업 사용자인지를 결정할 수 있으며, 어떤 선택지를 고르는 지에 따라 다른 페이지로 이동하게 된다.



* 유저가 기업사용자인지, 일반 사용자에 따라 입력을 받는 부분이 다르다. 기업 사용자는 추가적으로 데이터를 많이 입력 받지 않으나, 일반 사용자의 경우 위의 이미지와 같이 질병, 보험, 운전 면허 등에 **텍스트이(가) 표시된 사진

  자동 생성된 설명**대하여 입력 받는다.
* 가입이 완료된 경우, 고유한 json web token과 user\_idx를 입력 받는다. 해당 데이터는 백엔드 api를 요청할 때 parameter로 사용된다.

1. **로그인 UI**

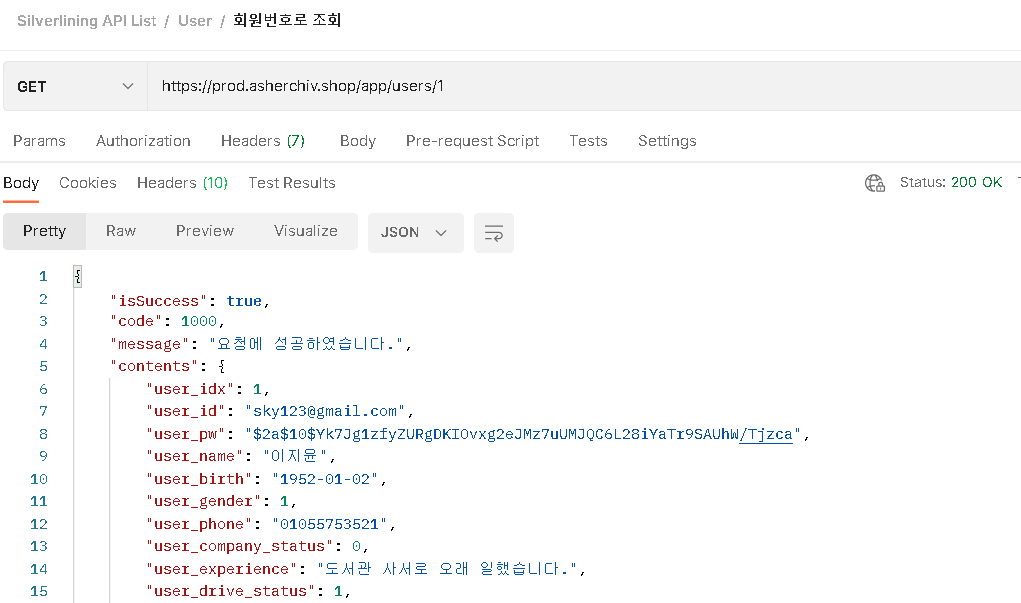
****

* 로그인은 아이디와 비밀번호를 입력 받아 진행되며, 데이터베이스에 존재하는 계정의 정보와 비교하여 일치할 경우 로그인이 수행되게 된다.

**C) 백엔드 API 구현 내역**

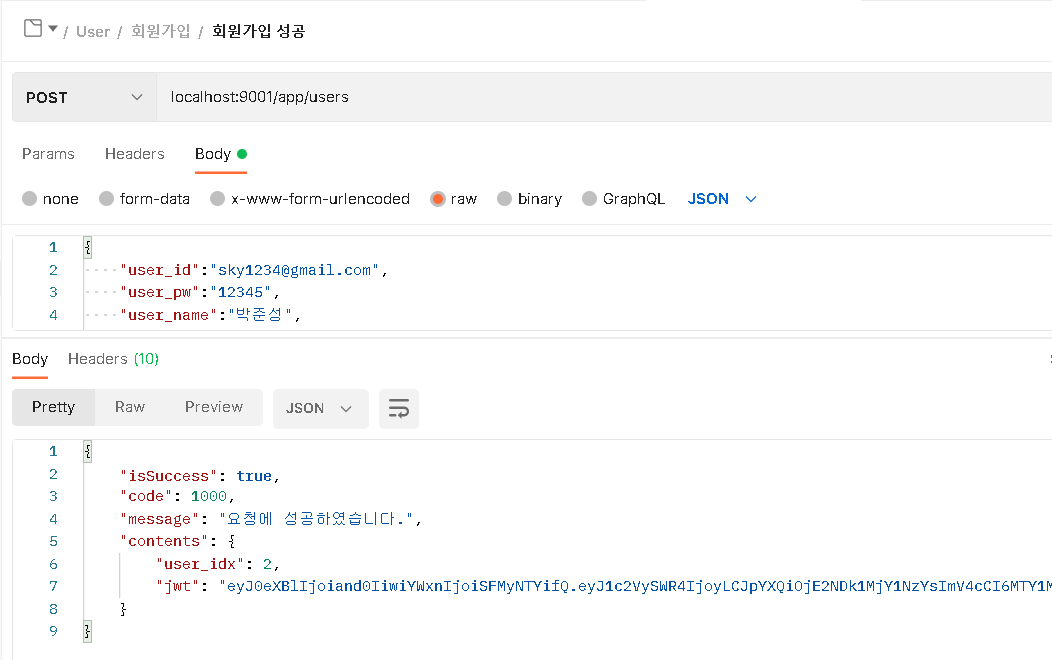
1) 회원 번호로 조회 API (/app/users/{userNum})

: 회원 번호를 request에 포함시키면 DB에 저장된 회원의 모든 정보를 조회하는 기능을 구현함.



2) 회원가입 API (/app/users)

: 어플리케이션에서 입력받은 회원의 정보를 DB에 일련의 과정을 거쳐 삽입한 뒤, 생성된 회원번호와 jwt를 결과값으로 반환한다.

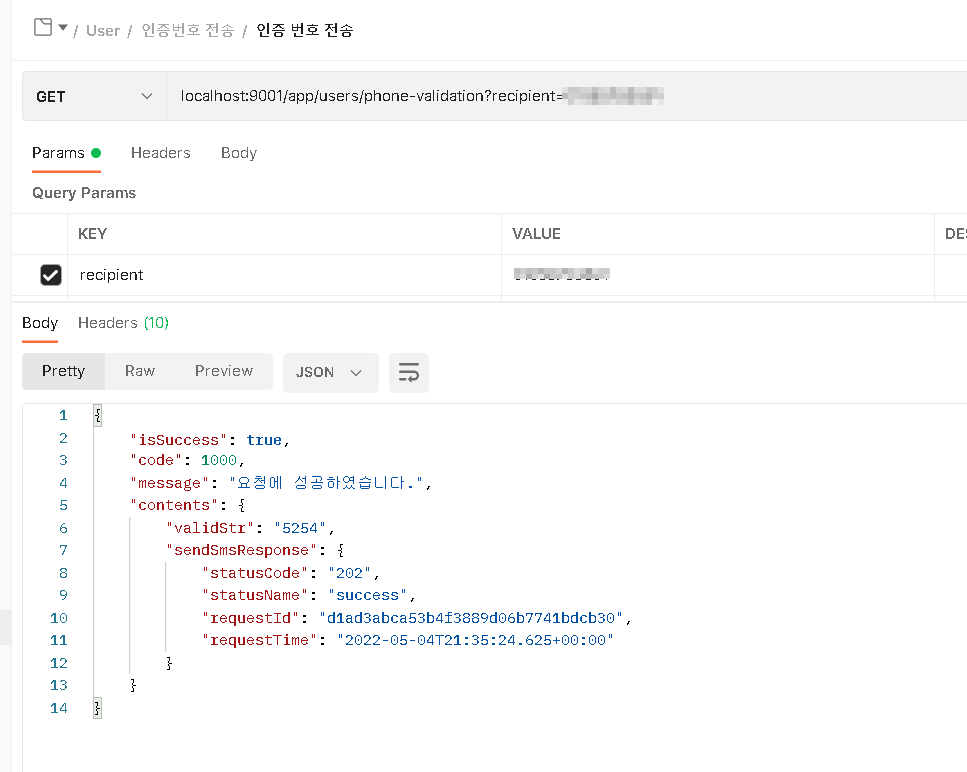


3) 로그인 API (/app/users/sign-in)

: DB에 저장된 id와 pw를 입력하면, 일련의 검증 절차를 거쳐 유저 인덱스와 jwt를 반환하고, 이때 발급된 jwt는 추후 인가가 필요한 API에서 인증시에 사용된다.

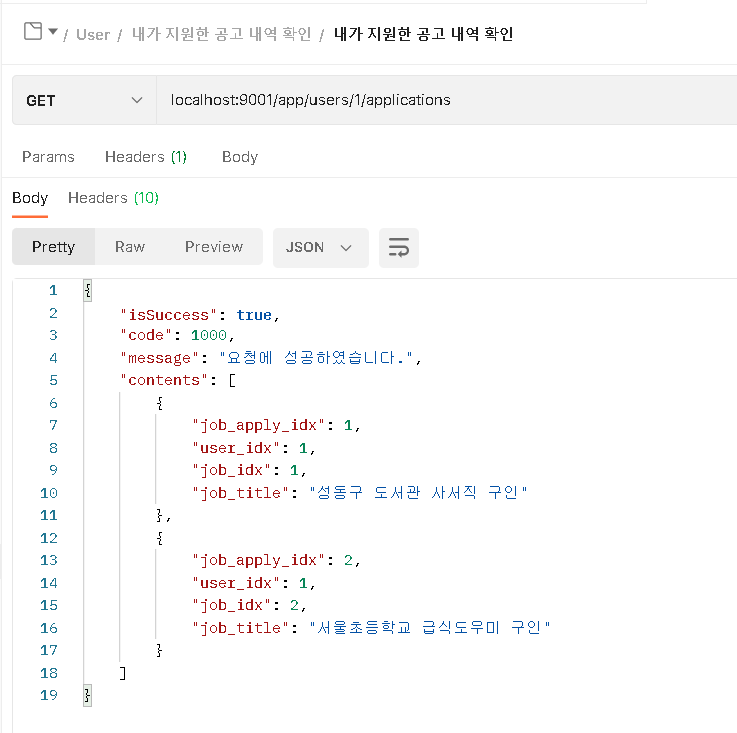
4) 회원가입 시 문자로 본인인증 API (/app/users/phone-validation?recipient=)

: 네이버 SENS 문자 전송 API를 활용하여 서버단에서 네자리 인증 번호를 랜덤으로 생성한뒤, 문자로 비밀번호를 전송하고 이를 어플리케이션에 입력하여 일치하면 나머지 회원가입 절차를 진행할 수 있도록 구현함.



5) 내가 지원한 공고 내역 확인 API (/app/users/{userNum}/applications)

:내가 지원한 구인 공고 내역을 확인할 수 있도록 리스트를 조회하는 API



1. **일자리- 노년층 맞춤형 구인구직 플랫폼**
2. **개요 및 요구사항**

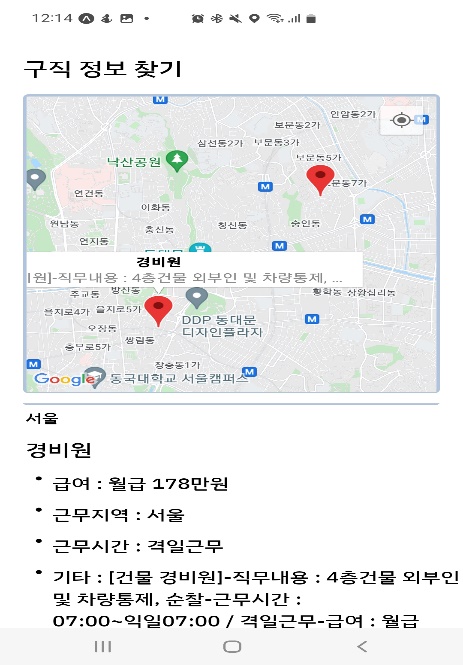
* “노인일자리”라 함은 노인이 일을 하는 주체가 아닌, 주로 일의 대상이 되어오기만 함. (사회복지사 등) 초고령화 사회로 진입한 만큼 인구 피라미드가 역삼각형으로 형성될 것이 확실시된 지금, 노동을 원하는 노인 수의 증가 및 노인의 노동력을 활용해야 할 필요성이 증가되고 있다.
* 그러나 현재 대한민구의 일자리 플랫폼등은 대부분 청년층에 치우쳐저 노인의 일자리에 관한 플랫폼은 매우 부족한 상태이다. 그 예로, 노인 일자리 센터 등이 존재하나 내부 일자리 정보는 대개 공고 안내의 기능에서 그치며, 데이터 필터링 정도 또한 낮은 수준으로 노인 개인의 관심 분야를 살리기 어렵기에 노인이 쉽게 접근할 수 있는 일자리 플랫폼이 필요함.
* 사용자 측면
  + 사용자의 나이, 거주지역, 희망직종, 보유 기능 등을 상세하게 입력받고, 유저의 정보를 토대로 데이터를 필터링해 맞춤형 일자리를 제공함.
  + 직관적인 이해가 쉽도록 지도 API를 이용해 거주지역 과 가까운 일자리 정보를 시각화하며, 관심분야에 맞는 일자리 목록을 렌더링
  + 미리 받아둔 정보를 토대로 지원서 양식을 어플리케이션 상에서 생성해주며, 버튼 하나로 빠른 구직 지원이 가능하도록 함.
* 기업 측면
  + 공고를 양식에 맞춰서 등록하면, 지원자 리스트와 각 지원자의 이력서를 한눈에 확인할 수 있도록 함.

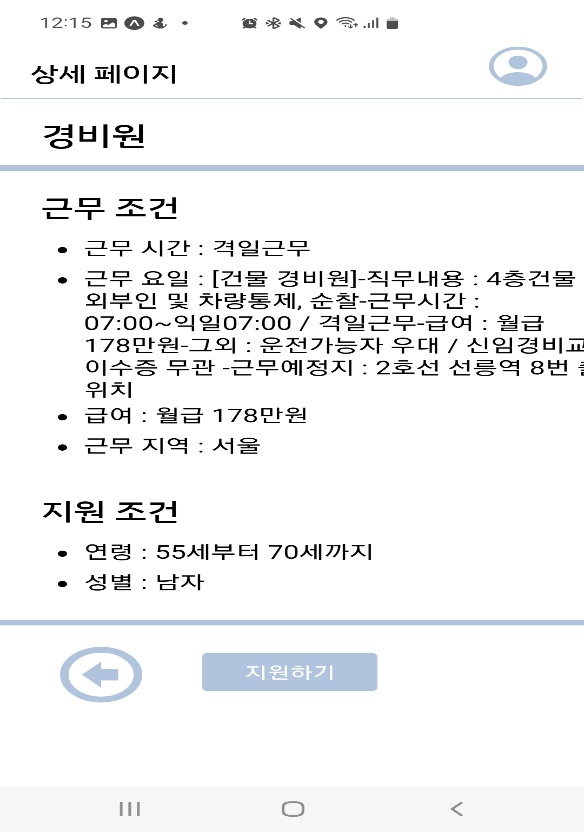
1. **프론트엔드 구현**

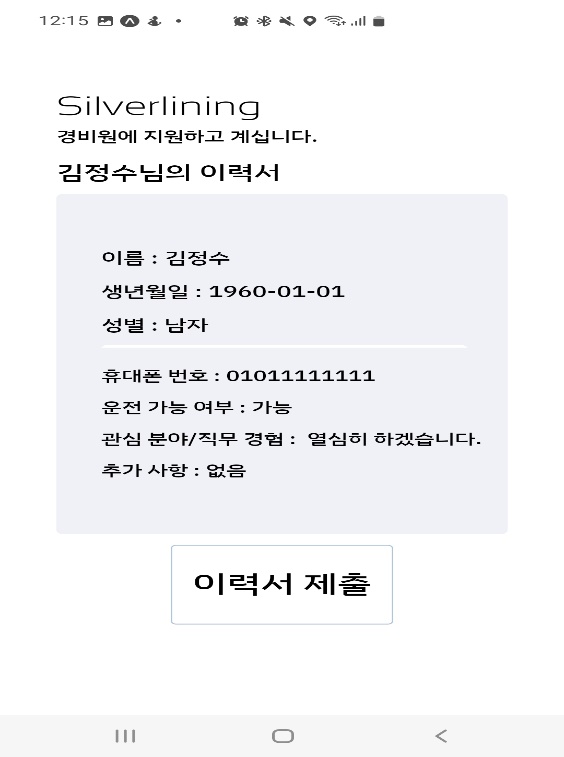


* 구인구직 플랫폼은 일반 회원, 기업 회원에 따라 접근할 수 있는 기능에 차이가 있다.
* 일반 회원은 구직정보 찾기 기능들을 이용할 수 있으며, 기업 회원은 구직정보 등록 및 지원자 조회 기능 등을 이용할 수 있다.

1. **일반 사용자의 경우**



* 일반 사용자의 경우, 지도 api 기반으로 유저 근방 1km 이내의 노인 공고를 불러온다. 노인 공고는 유저가 회원가입시에 입력한 정보를 기반으로 불러와진다.지도에서 마커를 클릭할 경우, 공고의 정보가 리스트 뷰에 불러와진다. 해당 리스트 뷰를 클릭할 경우 아래의 페이지로 넘어간다.
* 해당 공고의 상세 페이지를 조회할 수 있으며, 지원하기 버튼을 클릭해 해당 공고를 올린 기업 사용자가 지원 정보를 확인할 수 있게 된다.

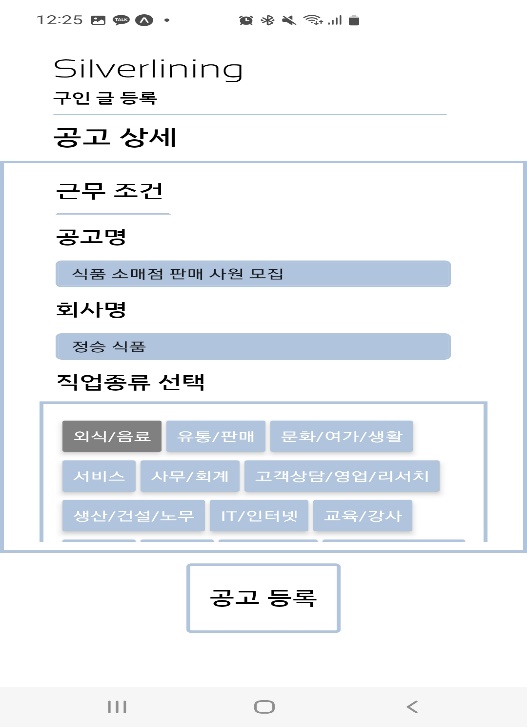


* 지원하기 버튼을 클릭할 경우, 위의 페이지로 이동하게 된다. 자동으로 작성된 이력서가 사용자에게 표시되며, 여기에 추가적인 정보를 부여한 채로 공고에 지원할 수 있다.

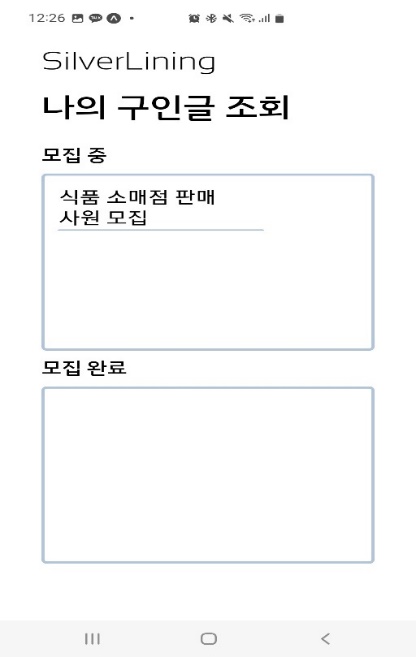
1. **기업 사용자의 경우**

****

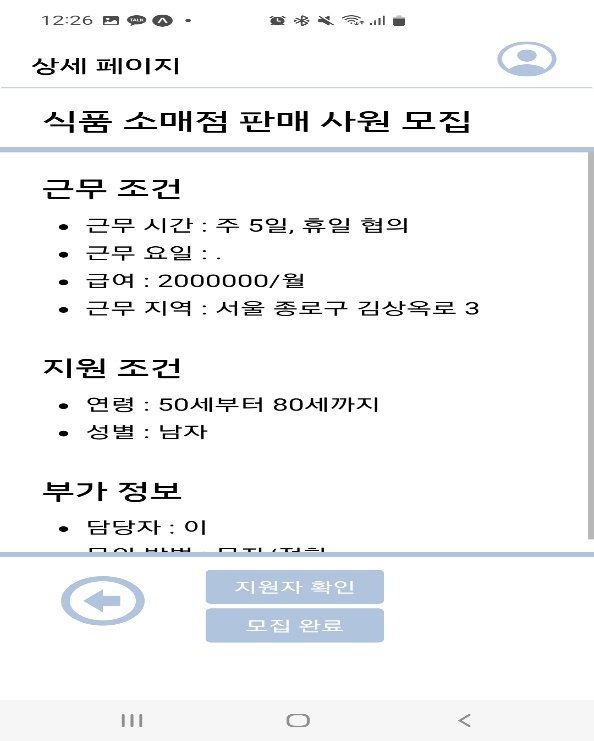
* 기업 사용자는 해당 플랫폼에서 두 개의 기능을 이용할 수 있다.
* 하나는 구인 글 등록이며, 다른 하나는 작성한 구인 글의 조회 기능이다.



* 또한 기업 사용자는 공고를 위처럼 작성할 수 있으며, 공고명, 회사명, 직업 종류, 연령, 성별 등의 정보를 기입하여 요건에 맞는 일반 사용자에게 노출되도록 한다.
* 작성한 정보는 공고등록 버튼을 통해 데이터베이스에 등록할 수 있다.



* 나의 구인 글 조회 기능을 들어가면, 위처럼 자신이 작성한 글을 조회할 수 있는 페이지로 넘어갈 수 있다.
* 또한 자신이 작성한 구인 글을 누르면, 아래의 페이지로 넘어가게 된다.

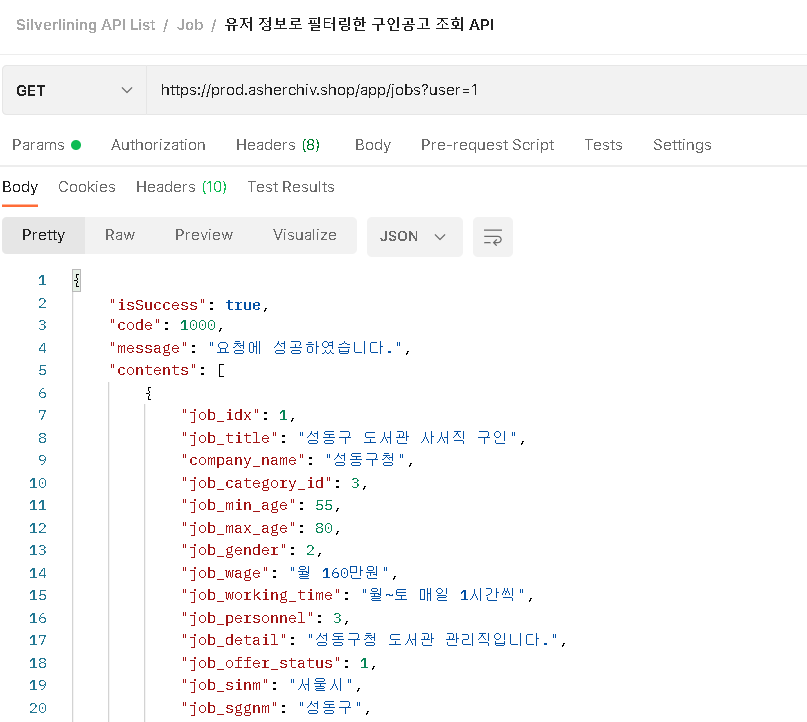


* 해당 페이지에서는 해당 공고의 내용을 확인할 수 있으며, 지원자를 확인하거나 모집완료 처리를 할 수 있다.

**c. 백엔드 API 구현**

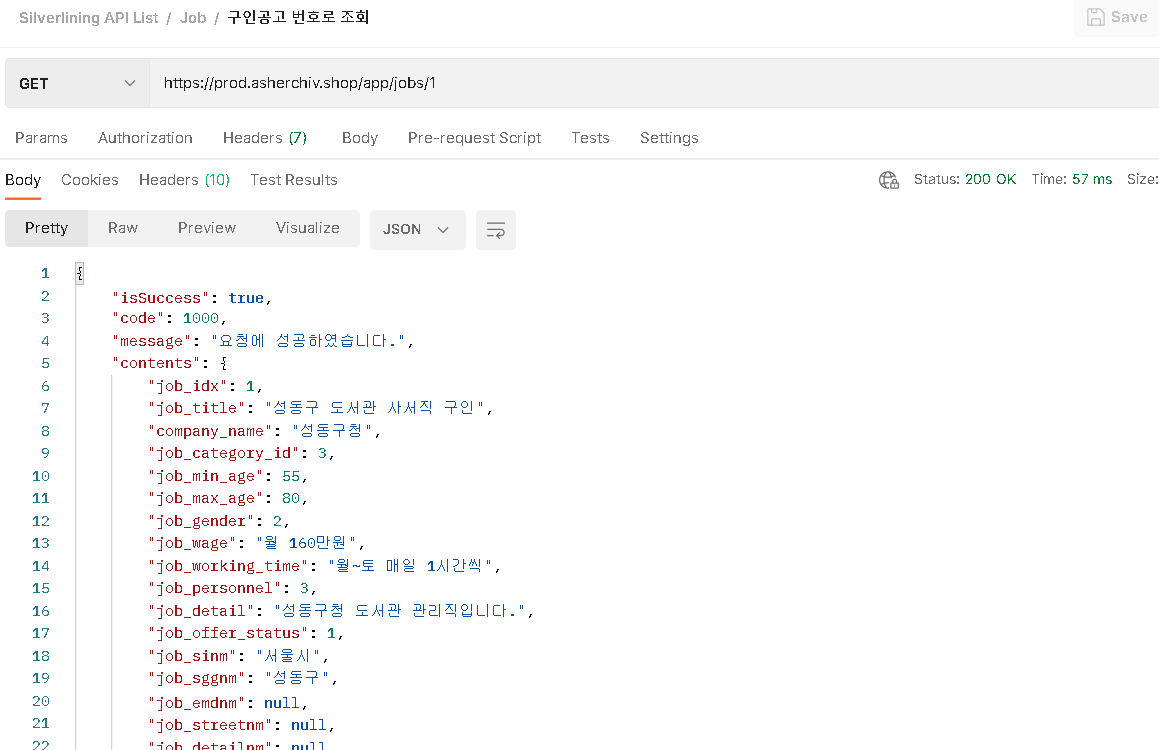
1) 유저 정보로 필터링한 구인 정보 조회 API (/app/jobs?user=)

: 기존에 회원가입시에 입력된 데이터를 통해 구인공고를 필터링하며, 필터링의 기준은 나이, 성별, 직장과 유저간의 거리등을 고려하였다. 조회 시에는 구인 공고의 모든 정보를 response로 출력한다.



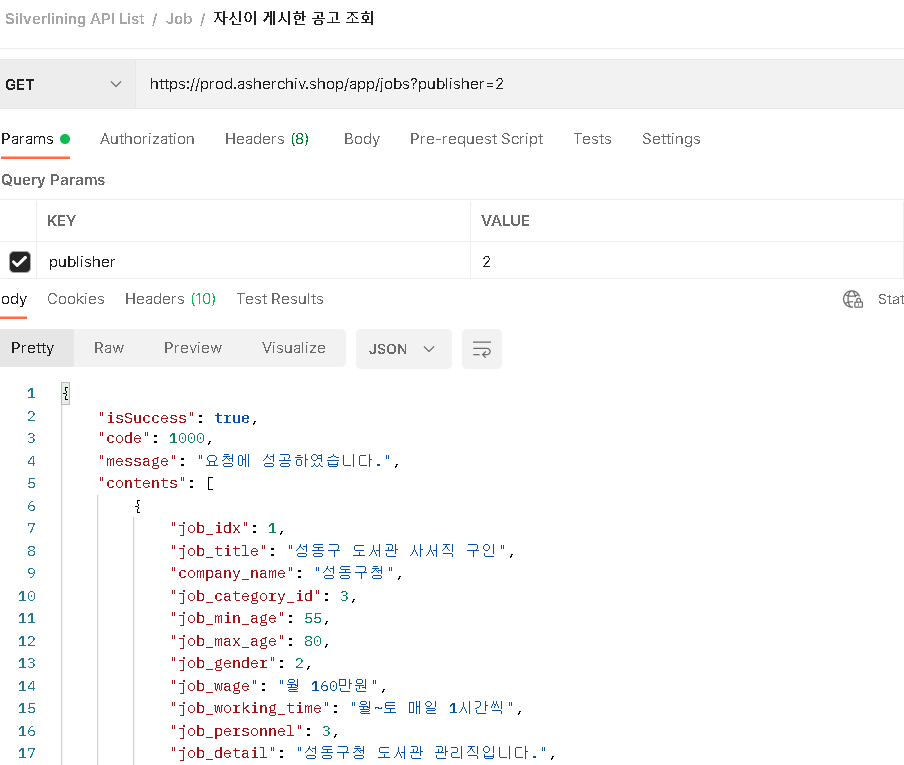
2) 구인 공고 번호로 조회 API (app/jobs/:jobIdx)

: 구인 공고에 부여된 번호를 통해 조회하며, 구인 공고 게시 시에 입력받은 모든 정보를 조회한다.



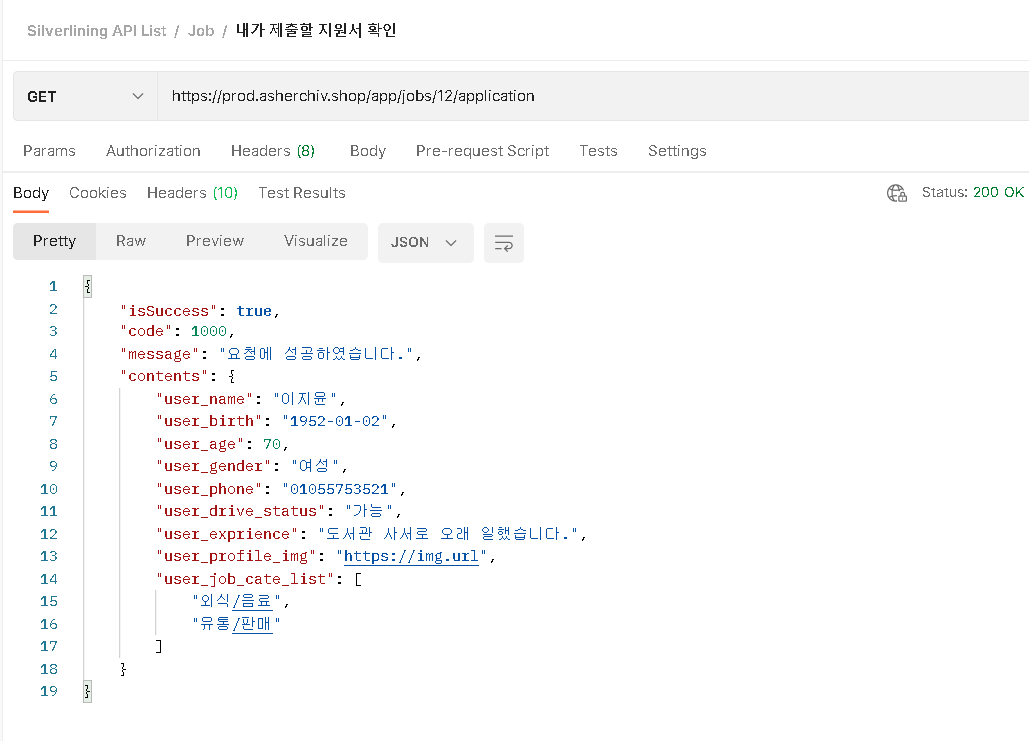
3) 자신이 게시한 공고 조회 API (/app/jobs?publisher=)

: 기업이 게시한 공고에는 회원 번호를 저장하는데, 이를 통해 기업에서는 직접 게시한 공고만을 조회할 수 있다.



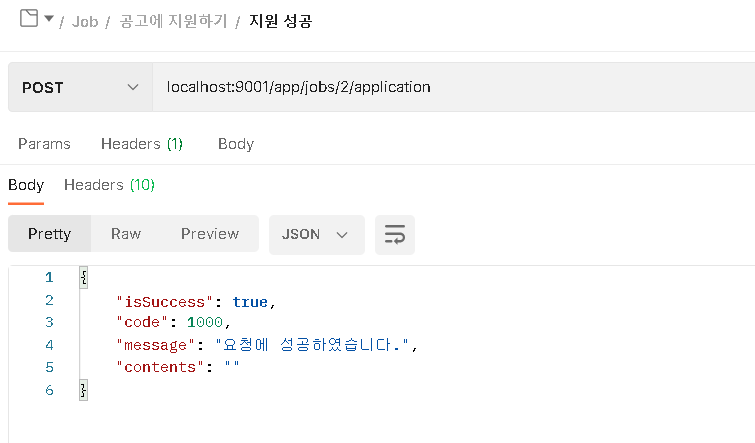
4) 내가 제출할 지원서 확인 API (/app/jobs/:jobIdx/application)

: 유저가 공고에 지원하면, 먼저 유저가 제출할 이력서를 검토할 수 있도록 유저의 관련된 데이터를 조회한다.



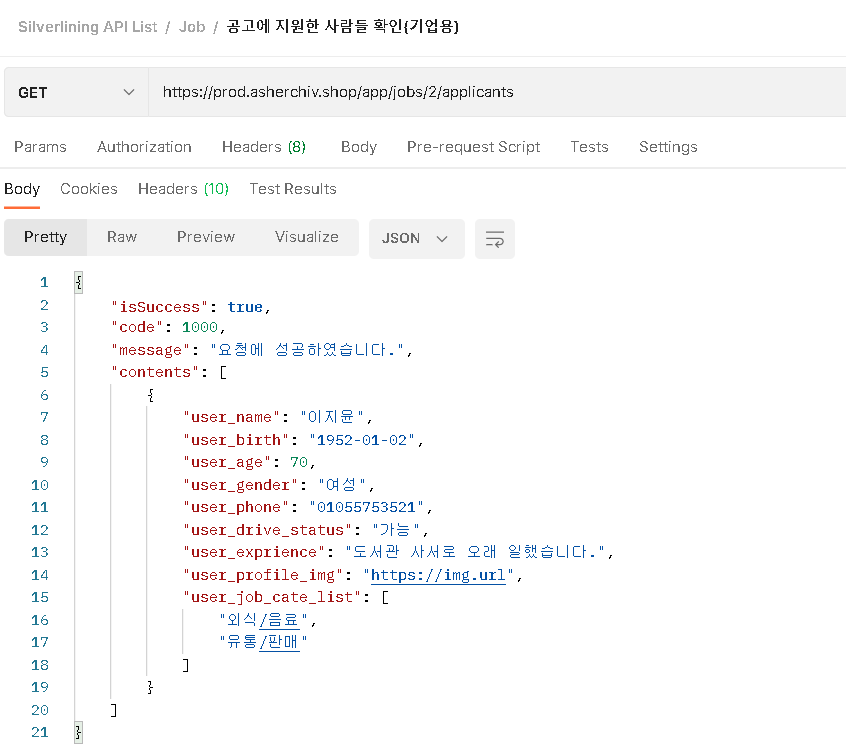
5) 공고 지원 API (/app/jobs/:jobIdx/application)

: 게시된 구인공고에 지원할 수 있도록 하는 기능이다.



6) 공고에 지원한 사람들 확인 API (/app/jobs/:jobIdx/applicants)

: 기업 차원에서 자신의 공고에 지원한 사람들의 정보를 확인할 수 있는 API이다.



7) 구인 공고 게시 API (/app/jobs)

: 기업 측에서 직접 구인 공고를 게시할 수 있도록 정보를 입력받고, DB에 삽입하는 기능이다.



**d. 데이터 크롤링 파트.**

**1) 데이터 크롤링을 진행한 사이트는** [**https://goldenjob.or.kr/job/find-person.asp**](https://goldenjob.or.kr/job/find-person.asp)

**서울시 어르신 취업 진행 센터이다. 해당 페이지에서 구직중에 해당하는 페이지만**

**데이터를 크롤링 하였다.**

**지도이(가) 표시된 사진

자동 생성된 설명**

**2) 크롤링은 주피터 노트북을 사용하여 BeautifulSoup 라이브러리를 활용하여 페이지 정보를 엑셀에 정리하였다.**

**텍스트이(가) 표시된 사진

자동 생성된 설명**

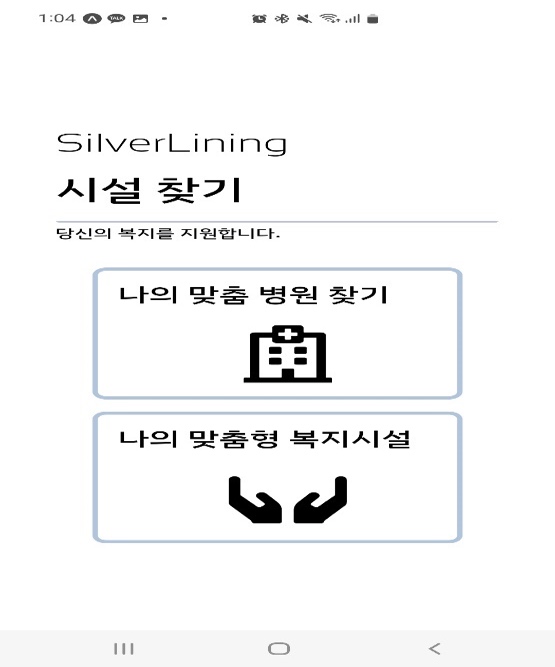
**3) 엑셀로 정리된 정보들에서 추가적으로 필요되는 정보에 한하여 구글링을 통해 엑셀에 추가하여 요구되는 데이터들을 api 명세서에 맞게 정리하였다.**

테이블이(가) 표시된 사진

자동 생성된 설명

1. **의료/복지 관련 - 맞춤형 주변 병원/복지시설 조회 기능**
   1. **개요 및 요구사항**
      1. 노년층은 각자가 갖고 있는 질환이나 건강 관리에 대한 관심도가 높을 수 밖에 없음.
      2. 따라서 미리 입력받은 정보를 토대로 주변에 위치한 병원 또는 복지시설을 추천해주고, 위치를 지도로 시각화하여 안내해주는 기능을 개발하고자 함.
      3. 국가 공공데이터포털에서 제공하는 병원, 노인 복지/여가/요양 시설 위치 API와 지도 관련 외부 API를 적극 활용해서 개발을 진행할 예정.
   2. **프론트엔드 구현**

* 해당 부분은 유저의 정보를 기반으로 하여, 위도, 경도 데이터를 통해 주변 병원 및 복지시설을 렌더링하는 기능이다. 지도 api 상에서 마커로써 렌더링된 ui를 통해, 병원 및 복지시설 정보를 받아올 수 있다.



* 사용자는 위의 두 가지를 사용할 수 있으며, 두 기능이 매우 흡사하므로 위의 기능을 설명하여 아래 기능의 설명을 포함하도록 한다.



* 자신의 위치를 기반으로 지도의 초기 중심점이 설정되며, 반경 1km내의 병원, 5km 이내의 노인 관련 의료복지 시설 정보를 가져온다. 마커를 클릭할 경우, 하단 리스트 뷰에 간략한 정보가 표시된다. 해당 리스트 뷰를 클릭할 경우, 아래 페이지로 이동한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

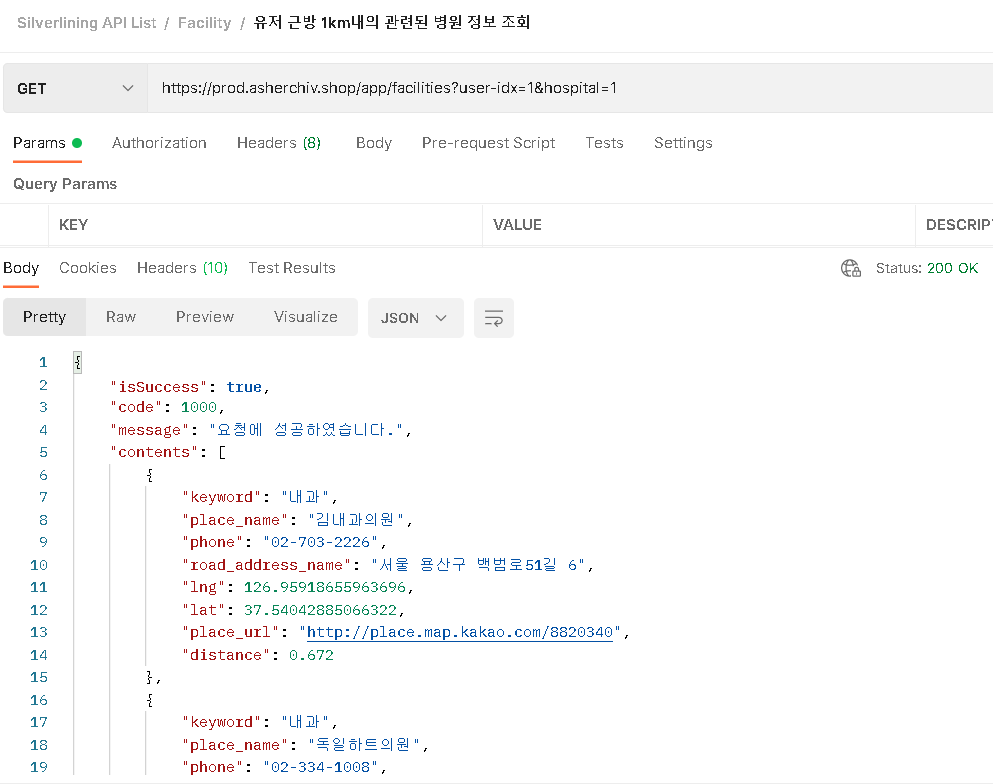
* 위와 같이 상세한 페이지로 이동하게 되며, 표시되는 url을 클릭하면 웹 뷰로 넘어가게 된다. 웹 뷰 페이지는 아래와 같다.



**c. 백엔드 API 구현**

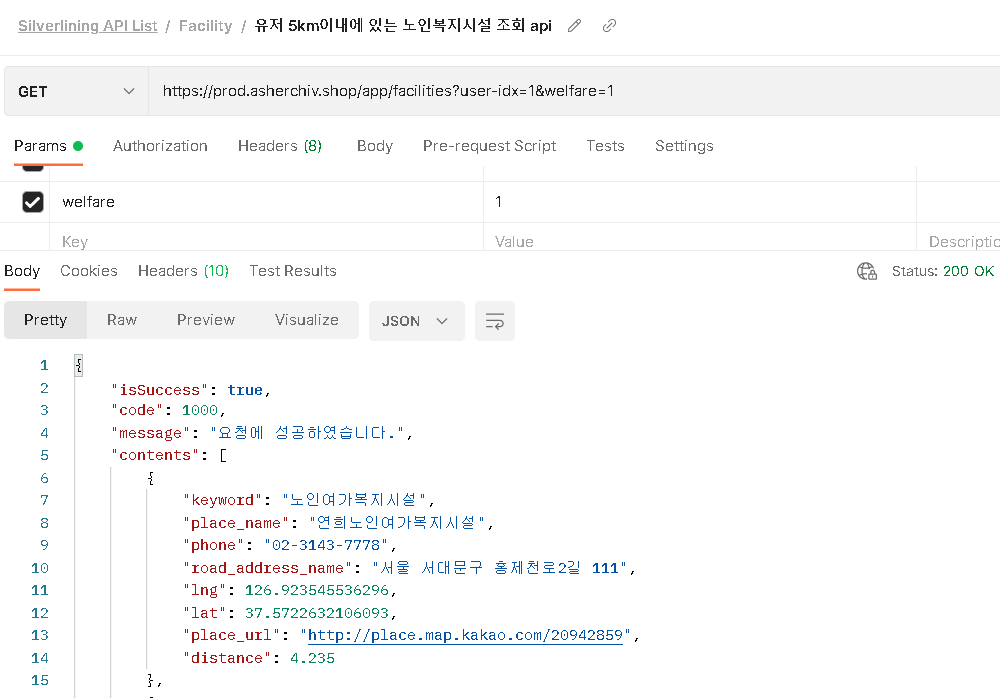
1) 유저의 기저질환과 관련된 반경 1km 내의 병원 정보 조회API (/app/faclilities?user-idx=1&hospital=1)

: 일전에 회원가입시에 유저가 갖고 있는 기저질환과 거주지 정보를 조사받았다. 이를 DB에 저장해두고, 이 질환과 관련된 병원의 진료과 리스트를 추출하였고, 이 리스트를 검색 키워드로 하여 카카오 로컬 API를 통해 유저 근방 1km내의 관련된 병원 정보를 출력했고, 이 결과에서 필요한 정보만을 추출해 가공한 뒤 해당 키워드 별로 분류해 response로 보내게 된다.

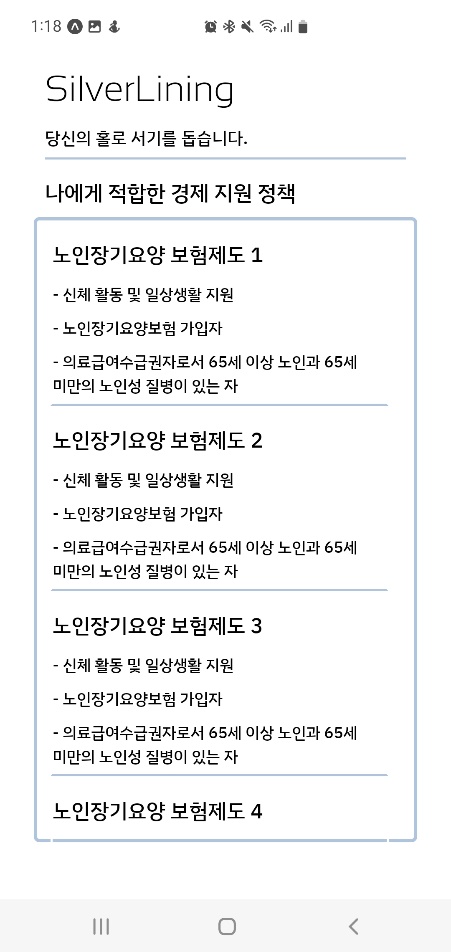


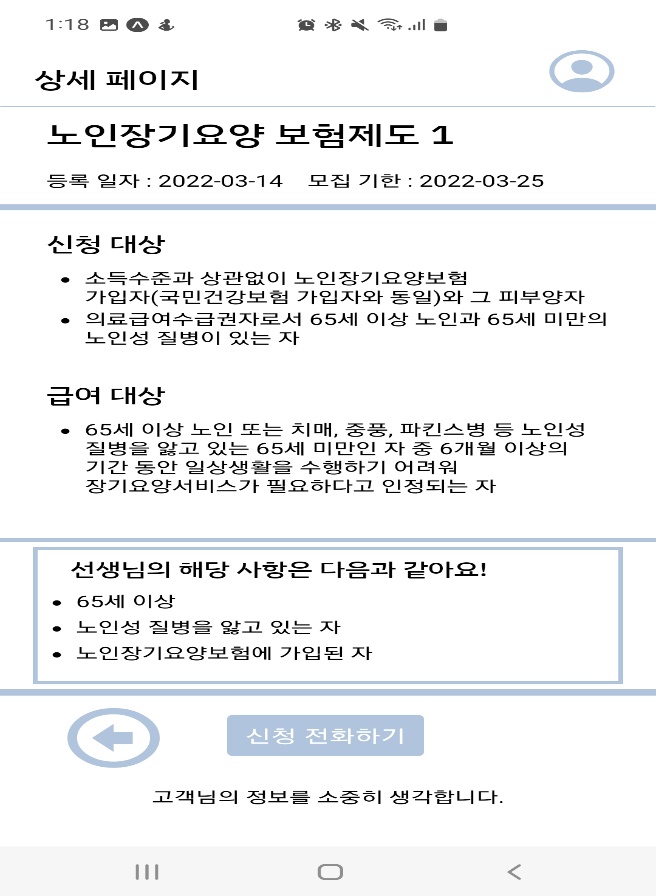
2) 유저 반경 5km 내의 노인 복지 시설 조회 API (/app/faclilities?user-idx=1&welfare=1)

: 해당 API에서는 앞의 API와 같은 방식으로 키워드와 유저 위치를 통해 시설 정보를 조회하였으며, 이번에는 5대 노인 복지시설 (주거복지시설/ 여가복지시설/ 재가노인복지시설/ 노인보호전문기관/ 노인일자리지원기관)을 키워드로 하여 유저 근방에 있는 노인복지시설 정보를 추출하여 응답으로 전송하였다.



1. **경제적 지원 관련 - 노년층 지원사업/정책 알리미**
2. **개요 및 요구사항**
   * 1. 현재 노인관련 정책, 또는 지원 사업을 정부에서 여러가지 추진하고 있으나, 디지털 기기에 친숙하지 않는 노인분들은 이를 직접 찾아보고 지원하거나 활용하는데 어려움이 많음. 실제로 친구도 공무원으로 재직중인데, 어떤 정책이 있는지도 잘 모르셔서 지원을 못 받는 것은 물론이고, 사람이 일일히 안내를 해주고 있다고 함.
     2. 해당 유저의 경제적 상태, 질병, 거주지역등을 가입 시에 입력받아 각자에게 적합한 정책을 맞춤형으로 알려주는 서비스의 필요성 느낌.
     3. 챗봇, 문자메세지, 푸시 알림 등을 활용해 해당 대상에게 정보를 안내하고 쉽게 신청할 수 있도록 유도하는 방식으로 구현해보려고 함.
   1. **프론트엔드 구현**



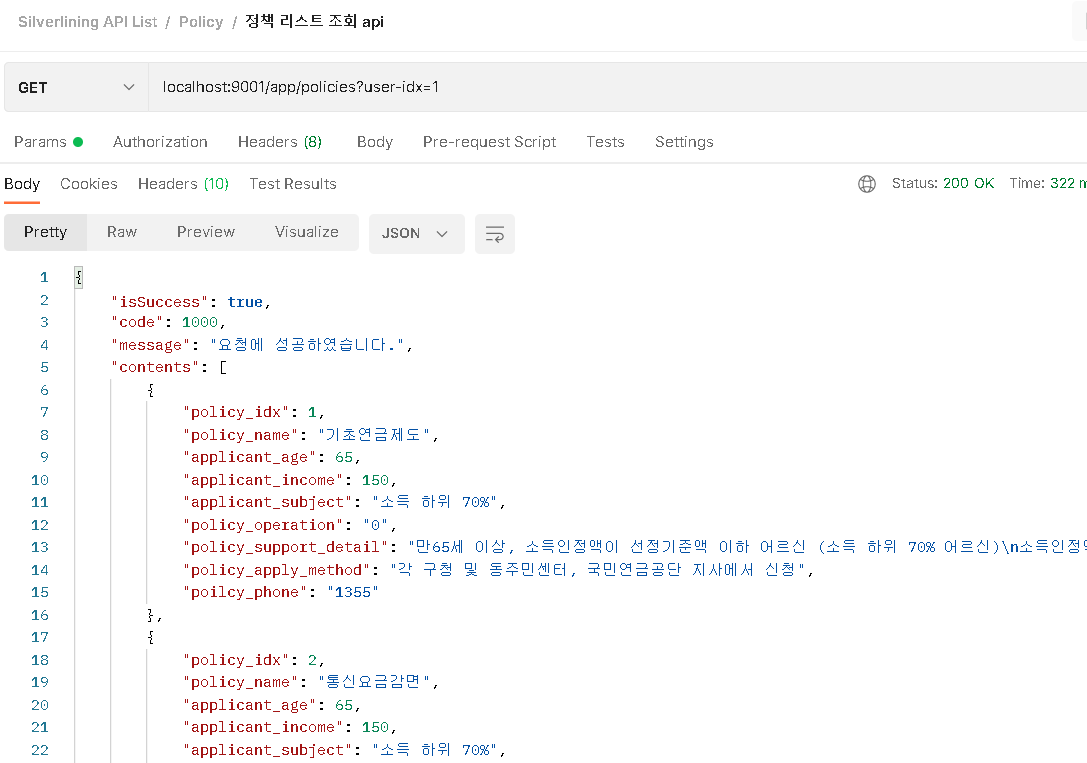


* 해당 부분은 크롤링 데이터를 기반으로, 데이터 베이스에서 유저와 매칭되는 부분이 있는 공고를 렌더링해주는 기능을 한다.
* 위와 같은 형식의 그리드 형식 뷰가 될 것이며, 각 그리드 내부 컨텐츠를 누를 시 우측 이미지와 같이 해당 공고의 정보를 조회할 수 있다.

C. 백엔드 API 구현

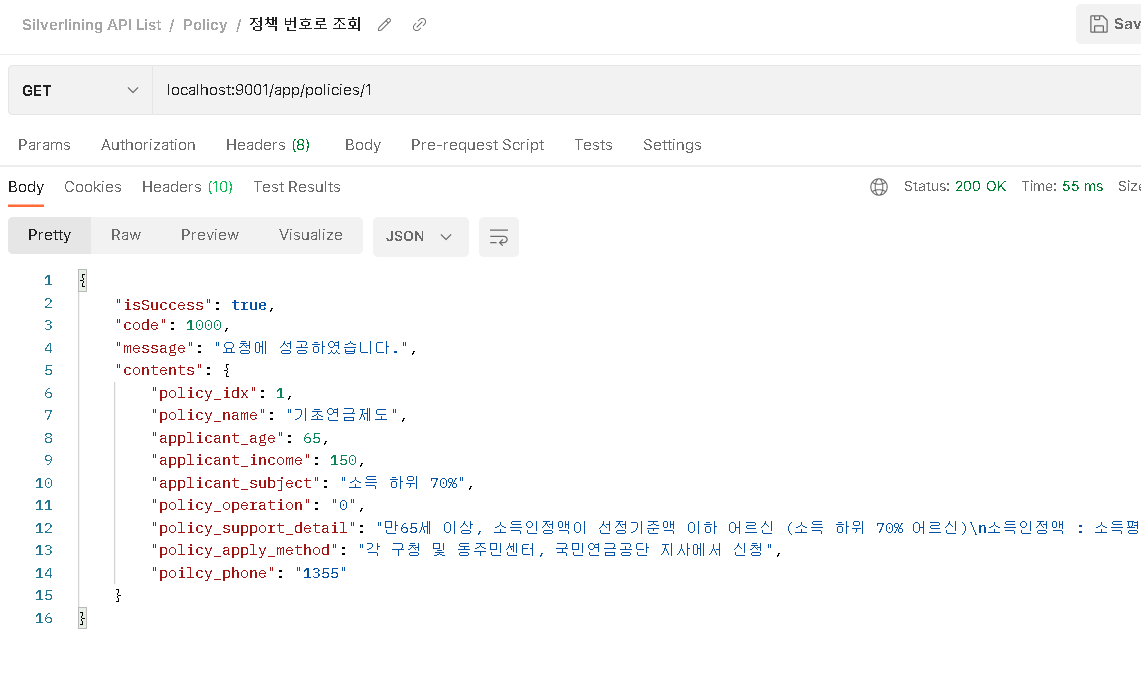
1) 유저 관련 정책 리스트 조회 API (/app/policies?user-idx=1)

: 여러가지 노인관련 복지정책/지원사업에 대한 정보들을 크롤링을 통해 모아왔고, 이를 가공해 DB에 저장해둔 상태이다. 이때 유저와 관련도가 높은 정책을 추천하기 위해서 유저의 나이, 중위소득을 토대로 해당되는 정책의 정보만 조회하도록 하였다.



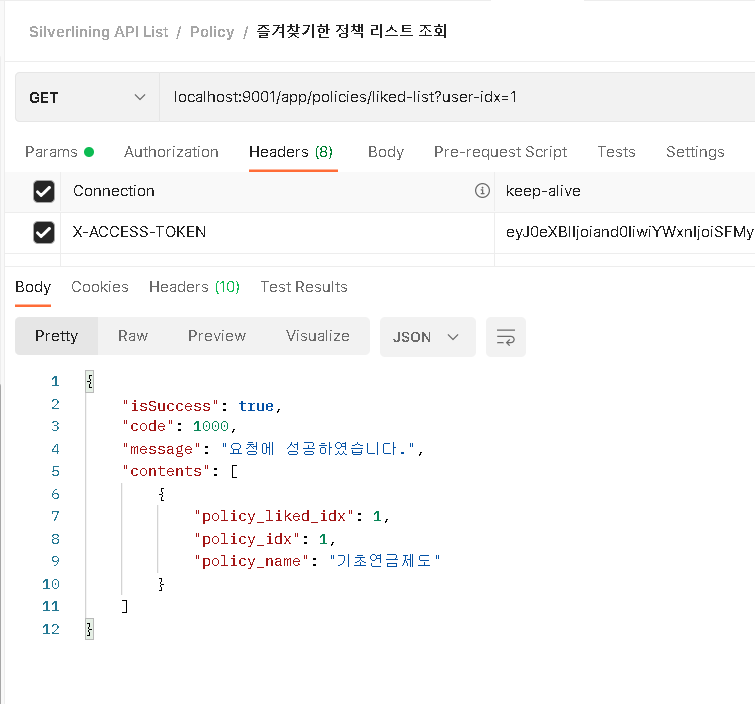
2) 정책 번호로 조회 API (/app/policies/:policyIdx)

: 정책에 모두 인덱스를 부여해놓았는데, 해당 인덱스를 통해 정책의 정보를 조회할 수 있도록 하였다.



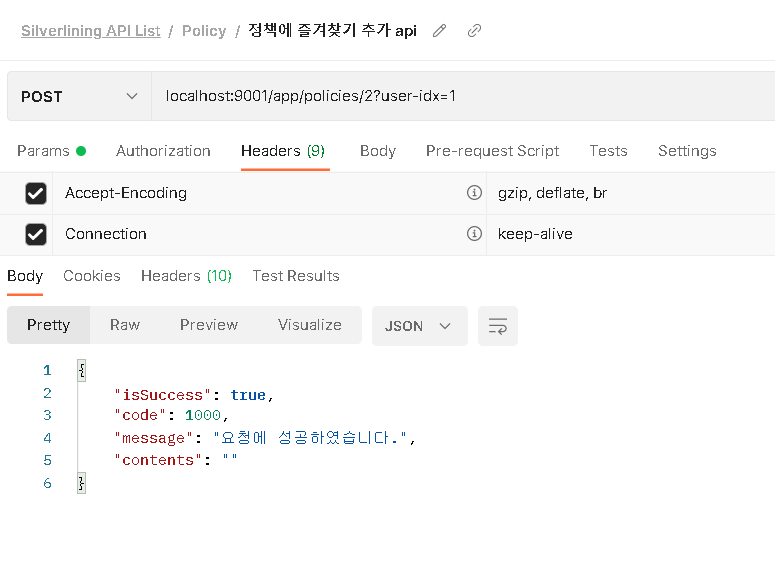
3) 즐겨찾기한 정책 리스트 조회 API (/app/policies/liked-list?user-idx=1)

: 유저가 관심있어하는 정책을 즐겨찾기할 수 있도록 하며, 이를 저장해두고 추후에 유저가 즐겨찾기 해둔 정책의 리스트르 조회할 수 있도록 하였다.



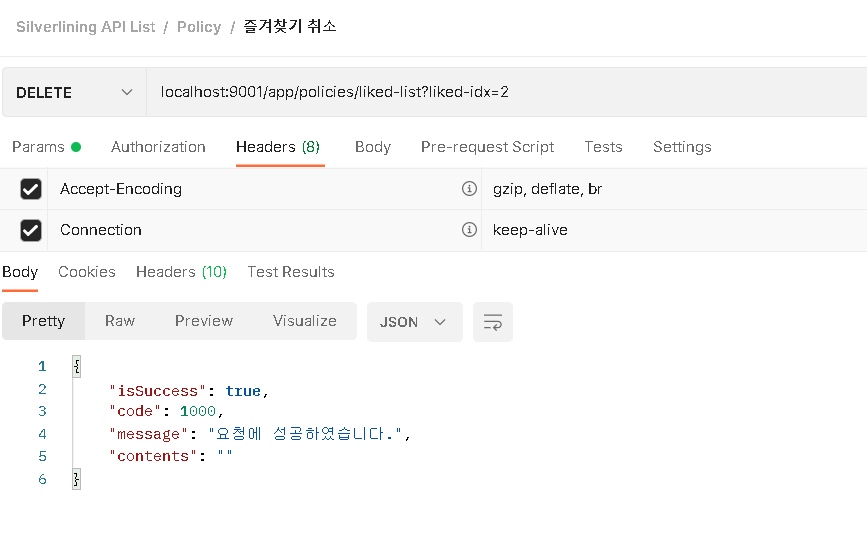
4) 정책에 즐겨찾기 추가 API (/app/policies/:policyIdx?user-idx=1)

: 유저가 관심있어 하는 정책을 즐겨찾기에 추가할 수 있도록 구현하였다.



5) 정책에 즐겨찾기 취소 API (/app/policies/liked-list?liked-idx=2)

: 또한 위에서 즐겨찾기에 추가해둔 목록을 삭제할 수도 있도록 구현하였다.



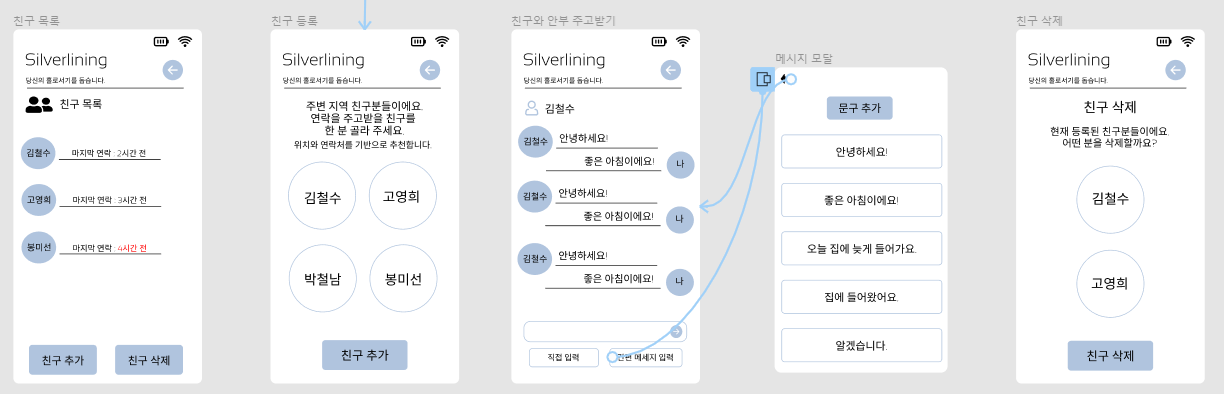
D. 데이터 크롤링

1)경제적 지원에 해당하는 크롤링 데이터는 정리된 웹페이지가 없고, 요구되는 API

에 맞게 정리되어 있지도 않는 특성과, 데이터가 적다는 사항을 고려하여 여러 경제적 지원 내역들을 구글링을 통해 정리하고 엑셀 파일을 API 명세에 맞게 정리하여 바로 CSV 파일 형태로 제공하여 사용하는 방식으로 데이터 크롤링을 진행하였다.



1. **관계 단절 관련 - 주변 친구 맺기**
2. **개요 및 요구사항**
   1. 요즘 갈수록 고독사 비중이 늘어나고, 사고등을 당해도 제때 발견되지 못해 변을 당하는 노인분들의 사례가 많음.
   2. 따라서, 가까운 주변 유저와 친구를 맺어 서로의 안부를 확인할수 있도록 기능을 개발할 예정.
   3. 안부 연락은 채팅 형식으로 개발하며, 버튼 하나로 간단하게 여러 메세지를 전송하거나 직접 메세지를 작성할 수 있도록 한다. 만약 각자 정해진 시간 안에 답이 오지 않으면 주변 친구 or 관공서에 메세지를 보내 행여나 있을 사고 등을 방지하도록 할 예정.
3. **Application 작동 프로세스 - UI/UX 설계도**

C) 프론트엔드 설계 방식 (예정)

 1) HTML5 에서 지원하는 웹 소켓 모듈을 사용하여 채팅 어플을 구현.

Language : JavaScript, View Framework : React.js

State Management : Redux.js , Component Dev Environment : story-book

2) 채팅에 익숙하지 않은 사용자층을 고려하여, 쉽게 활용할 수 있는

문구를 추가하여 채팅을 할 때 이용할 수 있도록 구현할 예정임.

3) 예전 피드백으로 나왔던 피싱에 취약한 연령층임을 고려하여

피싱을 방지하기 위한 금기어를 설정하여 방지할 예정임.

D) 백엔드 설계 방식 (예정)

1) Socket.io 를 이용하여 구현.

2) 최초에 프론트 앱을 서빙한 이후 HTTP 통신 없이 모든 네트워킹을

WS 통신으로 처리, 이후 별도의 AJAX 통신 없이 웹 소켓 간 JSON 을 주고

받는 방식으로 구현 할 예정.

**4. 어플리케이션 기술 요약 / 업무 분담/개발 일정**

1. **Server 파트 - 이지율**

: UX/UI 디자인/ 서버-DB 관리/ ERD 작성 / 백엔드 REST API 개발

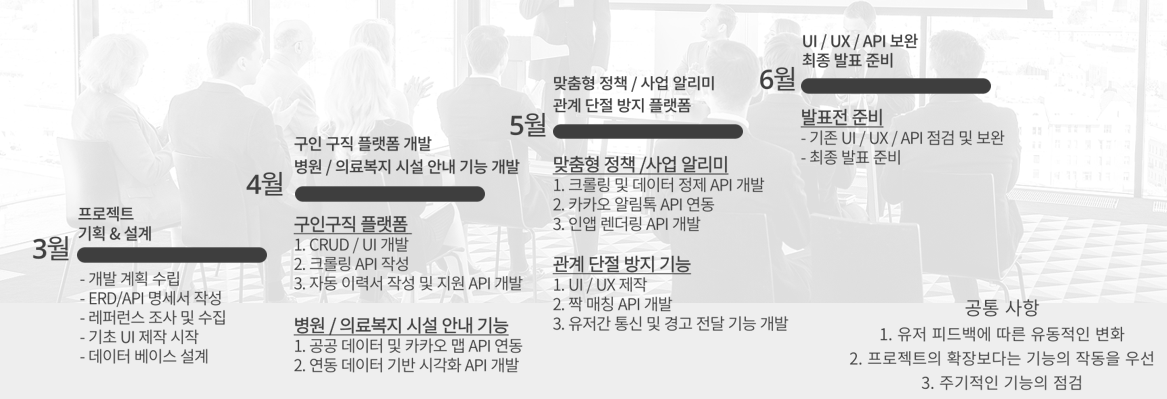
* 1. Cloud Server - AWS ec2
  2. Server Engine- Nginx
  3. DB - AWS RDS Database/MySQL
  4. Backend Language- Spring Boot

1. Front-End 파트 - 이종혁

: UI/UX 개발 및 REST API 방식 로직 개발

1. Framework - React Native
2. Stack - node.js, javascript…
3. UI Library - styled-components, Material-UI, iconfy...
4. 데이터크롤링, 프론트엔드 - 임유한
5. Data crawling - Python (BeautifulSoup)
6. Framework - React Native
7. Stack - node.js , javascript , mongoDB

* **프로젝트 개발 일정**



**5. 참고 사이트/ 레퍼런스**

* 전국 병의원 찾기 API
  + <https://www.data.go.kr/data/15000736/openapi.do>: 병‧의원 정보를 시도/시군구/진료요일/기관별/진료과목별로 조회할 수 있다.
* 서울시 노인 여가시설 데이터
  + <https://www.data.go.kr/data/15046517/fileData.do>
* 서울시 노인 주거복지시설 데이터
  + <https://www.data.go.kr/data/15046494/fileData.do>
* 국토교통부 노인복지시설 지도 API
  + <https://www.data.go.kr/data/15056929/openapi.do>
* 유사사이트 : 굿닥
  + <https://www.goodoc.co.kr/>
  + 질병명을 토대로 가까운 병원을 찾아주는 어플리케이션

# 이모잡 (<https://2mojob.com/>)

* 중장년층을 위한 일자리 매칭 플랫폼. 특히 은퇴 직후인 50대 후반 ~ 60대 초반 연령대를 타겟으로 함. 회원가입시 자신의 직종이나 신상 명세
* 서울시 어르신 취업지원센터

​​​​<https://goldenjob.or.kr/job/find-person.asp?keyfield=&keyword=&p=2>