

프론트엔드 개발자

김영진

다수의 UI 구현 경험으로 인해 사용자의 다양한 요구에 알맞게 화면 개발이 가능하며 제한된 리소스에서 기획단계부터 참여하여 효율적인 업무 진행과 명확한 목표 설정을 통해 프로젝트를 마무리한 경험을 가지고 있습니다. 다양한 웹 및 모바일 프로젝트 경험을 보유하고 있으며, 최신기술에 대해 능숙한 이해가 가능합니다



블로그 <https://yjym33.tistory.com/>

깃허브 <https://github.com/yjym33>

이메일 yjym33@naver.com

경력&프로젝트

폴라리스 AI , 기업부설연구소 AI팀, 프론트엔드 개발자 (주임)

2024년 07월 - 2025년 3월 / 8개월

• 경동나비엔 홈페이지 AI 챗봇 구축 프로젝트 (2024.08 ~ 2024.12)

- Tech : Front : React, Typescript, NextJS,
Back : Python, FastAPI

• 회사 내부 규정 AI 챗봇 구축 프로젝트 (2024.12 ~ 2025.03)

- Tech : Front : React, Typescript, NextJS,
Back :Springboot, JPA, (NestJS -> Spring) 으로 마이그레이션

지피헬스(주), 어플리케이션팀, 연구원(사원)

2021년 09월 - 2024년 3월 / 2년 6개월

• 사내 개발

• 노인배회감지기 웹 & 어플리케이션 서비스 개발 (2023.09 ~ 2024.02)

- Tech : Flutter (Dio, GetX, flutter screenUtil,)

• 반려동물 고양이 모래관리 어플리케이션 개발 (2021.12 ~ 2022.05)

- Tech : React, Typescript, Ionic,

• 구강 언어치료용 데스크톱 어플리케이션 개발 (2021.10 ~ 2022.04)

- Tech : React, Typescript, Electron

현대건설 "통합결제 시스템 구축 프로젝트" 참여 (인턴)

2020년 02월 - 2020년 8월 / 6개월

• 통합결제 시스템 구축 프로젝트

- 솔로몬테크노시스템(주) 라는 회사의 소속으로 프리랜서 형태의 인턴으로 해당 프로젝트에 파견
- 담당업무
 - 프로젝트의 전체적인 테스트(QA) 업무 진행
 - 통합결제 시스템에 필요한 문서양식 개발 (Javascript, JQuery)

학력

한양대학교 공학대학원 전기전자 컴퓨터공학과	컴퓨터 공학 전공 · [2021.03 ~ 2023.08]
한국방송통신대학교	무역학과 전공 · [2017.09 ~ 2020.02]
신구대학교	글로벌경영학과 전공 · [2014.03 ~ 2016.09]

스킬

Language	Framework
FrontEnd : Javascript, Typescript,	React, Next.JS, ReactQuery, Redux, Zustand

교육 수상 및 기타

KOSTA “자바 웹 개발자 양성과정” 교육 수료	2018년 06월 - 2018년 11월 / 6개월
<ul style="list-style-type: none">자바 웹 개발자 양성과정 교육 이수<ul style="list-style-type: none">자바 기반 웹 어플리케이션 개발에 필요한 기초부터 실무 프로젝트까지의 과정을 이수주요 커리큘럼:<ul style="list-style-type: none">Java 기초 및 객체지향 프로그래밍SP/Servlet 기반 웹 개발Spring Framework 기반 MVC 구조 설계 및 구현Oracle DB 연동 및 SQL 활용팀 프로젝트를 통한 웹 서비스 기획, 설계, 개발 및 배포 경험	

자격증

정보처리기사	2019.11.22
--------	------------

경력기술서

폴라리스 AI [기업부설연구소 AI팀] 프론트엔드 개발자

2024년 07월 - 2025년 3월 / 8개월

경동나비엔 홈페이지 AI 챗봇 구축 프로젝트 (2024.08 ~ 2024.12) [5개월]

프로젝트 개요

본 프로젝트에서는 경동나비엔의 기존 고객지원 시스템을 개선하여, 챗봇 기반의 직관적이고 효율적인 A/S 상담 접수가 가능한 AI 챗봇을 구축하는 프로젝트였습니다.

RAG 기반의 챗봇 메인 서비스는 포지큐브에서 담당하여 구성하였으며 폴라리스 AI에서는 프론트 전체 부분을 담당하여 경동나비엔 사용자 환경 UI 맞게 커스터마이징 하여 수정하였습니다.

본 프로젝트에서는 고객과의 대화를 통해 필요한 서비스 (A/S 접수, 조회, 제품 구매 상담)를 파악하고 챗봇 화면에서 별도 브라우저 창을 통해 연동된 고객지원 화면을 제공하는 방식으로 개발되었습니다.

사용기술

FRONTEND

- 언어 : *Typescript*
- 프레임워크 : *Next.js (React)*
- API 통신 : *Axios*

BACKEND

- 언어 : *Python*
- 프레임워크 : *FastAPI*
- 역할 : 고객지원 화면과 경동나비엔 서버간의 데이터 통신을 위해 중간 통신 역할을 하는 미들웨어 프록시 서버 구축

주요 역할 및 성과

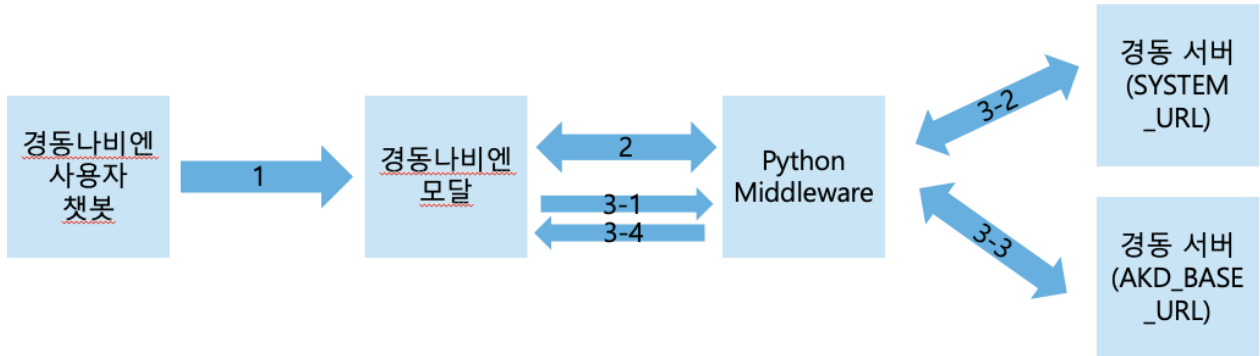
- ROBI-G* 챗봇 메인 UI 및 고객지원 서비스 연동화면 개발
 - Next.js* 기반 챗봇 UI 및 채팅 인터페이스 구성
 - 사용자 인증 및 고객지원 관련 서비스 화면 개발 (A/S 접수, 조회, 제품 구매상담 등)
 - 챗봇 내에서 버튼 클릭 시 새 브라우저 창을 열어 별도 고객지원 화면 제공
- FastAPI* 기반 미들웨어 서버 개발
 - 챗봇에서 전달된 사용자 인증 정보를 통해 별도 팝업창에서도 동일한 사용자로 인증 처리
 - 경동나비엔 *REST API* 명세에 맞춰 고객지원 관련 데이터(*CRUD*) 연동
 - A/S 접수 취소, 구매상담 접수, 엔지니어 알림톡, 가까운 대리점 정보 등 *API* 통신 처리

트러블 슈팅 및 기술적 이슈 해결

1. (A/S 접수 조회, 제품구매 상담 등) 새 브라우저 팝업에서 사용자 인증 유지 문제

- 문제 : 챗봇 내에서는 사용자 정보가 존재하지만 별도의 팝업창에서는 이 정보가 공유되지 않음
- 해결 : 메인 챗봇에서 사용자 정보를 포함한 인증 토큰을 파이썬 서버로 전달하고, 파이썬 서버가 이를 검증하여 별도의 팝업 창에서도 동일한 사용자로 인증 후 서비스를 제공받을수 있도록 진행

구체적으로는 다음과 같은 구조를 통해 경동나비엔 메인 챗봇과 인증을 유지하도록 했음



1. 챗봇에서 Modal 호출 : {modal_url}?user_id=test@&onetime=326224&epoch=1727209897
2. 경동나비엔 모달에서 credential API 호출 middleware에서 "token" 발급 (사용자 챗봇에서 넘어온 사용자가 맞는지 확인하는 용도)
3. 경동나비엔 모달에서 API 호출(품목 조회, 증상 조회 등 정보 처리를 위한 API)
 - 3-1. API를 호출할 때, (2)에서 발급받은 "token"을 헤더에 담아 전달 -> middleware에서 token 유효성 검증
 - 3-2. 경동서버(SYSTEM_URL)의 initialauth API를 호출하여 "id_token"을 발급 (정상적인 middleware가 맞는지 확인하는 용도)
 - 3-3. 경동서버(AKD_BASE_URL)의 API를 호출할 때, (3-2)에서 발급받은 "id_token"을 헤더에 담아 전달하여 데이터 수신
 - 3-4. (3-3)에서 수신한 데이터 전달

```

@app.post("/api/v1/kdwin/credential")
def credential(request: Credential_Request):
    try:
        # Convert request to dict and pass to controller
        is_social_user = controller.credential_social_login_user(request.dict())

        if not is_social_user:
            raise HTTPException(
                status_code=401,
                detail={
                    "error": "unauthorized user"
                }
            )

        # Update auth object
        auth.user_id = request.user_id
        auth.onetime_code = int(request.onetime_code)
        auth.epoch_time = int(request.current_time)

        print(f"{auth.user_id}--{auth.onetime_code}--{auth.epoch_time}")

        if not auth.verify_caller():
            raise HTTPException(
                status_code=401,
                detail={
                    "error": "Parameters are not accepted"
                }
            )

        jwt_token = auth.create_jwt(EXP_TIME)
        refresh_token = auth.create_jwt(REFRESH_TIME)
        return JSONResponse(
            content={
                "code": 200,
                "message": "SUCCESS",
                "data": {
                    "jwt_token": jwt_token,
                    "refresh_token": refresh_token
                }
            },
            status_code=200
        )
    except:
        pass
  
```

좌측의 코드는 위 이미지의 내용중 credential API를 호출 후 이를 검증하는 부분의 코드입니다.

전달된 user_id, onetime_code, current_time(epoch)을 기반으로 유효성을 검증합니다.

프로젝트 구현 내용 및 이미지 정리

챗봇 메인화면

[illegible]

챗봇 채팅화면

三
한국 고객 지원

+ 새 대화 시작하기

검색어 입력

대화 이력
☆ 즐겨찾기

- 2024-11-19 (목) 18:14
가스보일러에서 물이 세는 문제
- 2024-11-18 (월) 10:22
A/S 접수 내역 조회 도와드리겠습니다.
- 2024-11-18 (토) 10:18
문제 발생. 다시 질문 부탁
- 2024-11-18 (토) 10:15
문제 발생. 다시 질문 부탁
- 2024-11-18 (토) 10:14
고객님의 질문 분석 하슈
- 2024-11-13 (수) 11:49
A/S 내역 조회 도와드리겠습니다.
- 2024-11-11 (금) 13:55
1 대화가 1문 2
- 2024-11-11 (금) 13:48
경동나비엔 제품 종류
- 2024-11-11 (금) 13:46
제품 문제 발생. 접수 요청

사용자

가스보일러에서 물이 세

AI 가이드 애벳

가스보일러에서 물이 세는 문제에 대한 자가진단 가이드는 다음과 같습니다:

예상 원인

- 배관, 배관 접속부 및 연통(연도)의 노후 또는 체결 불량
- 보일러 본체에서 누수

조치 방법

- 보일러 본체 또는 배관이나 연통에서 물이 세는지 확인해 주세요.
 - 보일러 본체 이외의 배관이나 배관 접속부 또는 연통에서 물이 세는 경우에는 보일러의 설치업체 또는 배관 전문업체를 통해 조치를 받으세요.
 - 누수량이 많은 경우, 보일러와 연결된 배관 중 지수배관에 밸브를 잠금 추가 누수가 되지 않도록 조치해 주세요.

열림

열림

- 보일러 본체에서 누수가 되는 경우 맨지너어의 점검이 필요합니다.

!

유상 상담

궁금하신 내용을 질문해 주세요.

팝업창 A/S 서비스 화면

✕

A/S 접수를 위해 아래 정보를 입력해 주세요. (*필수 입력사항)

* 품 목	E001(OI) ▼
* 증 상	난방수 과열 잠지 ▼

증상 설명

모델명과 증상을 입력해 주시면 신속한 서비스 가능합니다.
 (최근 NCB300 모델을 사용하고 있는데 물론도 전보다 차가워져서 이상이여 A/S접수 진행함.)

* 고객 명 고객센터

* 연락처 010-1234-5678 재전송

인증을 완료 하려 확인

* 주 소 우편번호 우편번호 검색

우편번호 검색 시 주소가 자동 입력됩니다.

상세주소

* 방문일자 회합 방문일자 선택 방문시간 선택 ▼

신체하신 날짜를 바탕으로 오전(자)미하의 시간을 통해서 방문이 이루어 집니다.

☐ 개인정보 수집 - 이용 동의 [전문하기]

✕

고객님께서 A/S 접수하신 정보는 아래와 같습니다.

접수 이력을 최대 32건 조회 가능하며, 추가 조회한 필요할 때만 접수번호로 조회 하거나 고객상담센터로 문의해 주세요.

2024-03-19	2024031900673	^
구분	정보	
접수번호	2024031900673	
고 객 명	홍길동	
주 소	강원특별자치도 동계읍 양지길 6 (초가음)	
품 목	가스보일러	
종 상	E004, 미시화형	
예약일자	2024-03-19 16시	
진행상태	서비스 예약	
최소사용	<div style="border: 1px solid #ccc; padding: 5px; min-height: 40px;"> 최소사용을 입력해 주세요. </div>	

A/S 접수 취소
엔지니어 면허 요청

2024-03-18	2024031900672	v
2024-03-17	2024031900671	v

고객센터 센터 전화

✕

제품 구입 상단을 위해 아래 정보를 입력해 주세요. (*필수 입력사항)

*고객명	<input type="text"/>
*연락처	<input type="text"/> SMS 인증
*품목	<p>품목 선택</p> <p>제품을 설치하실 주소를 입력해 주시기 바랍니다.</p>
*주소	<p>우편번호</p> <p>우편번호 검색</p> <p>우편번호 검색 시 주소가 자동 입력됩니다.</p> <p>상세주소</p>
대리점	설치 주소에 따른 담당대리점 자동 연동
*희망연락일자	<p>희망 연락일자 선택</p> <p></p>

☐ 개인정보 수집 · 이용 동의

[전보내기]

구매상담 등록

✕

☒ 전체 ☐ 청정원가 특화점 ☐ NCB900 특화점

광역시/도 ▼

시/군/구 ▼

 현재 위치로 찾기

검색

대리점명	주소
(종로점)(주)유림에너지2(영등포중앙대리점)	서울특별시 영등포구 도신로 129, 1층 지중(도림동)
주식회사 경동신에너지(마포대리점)	서울 마포구 동교로 70, 1층(합정동)
경동나비엔 용산대리점(용산대리점)	서울특별시 용산구 새창로 117(용문동)

▼ 더보기

제품구매 상담

대리점 조회

A/S 접수c

A/S 조회

경력기술서

폴라리스 AI [기업부설연구소 AI팀] 프론트엔드 개발자

2024년 07월 - 2025년 3월 / 8개월

사내규정 AI 챗봇 구축 프로젝트 (2024.12 ~ 2025.3) [3개월]

프로젝트 개요

- 사내 규정 문서를 학습시켜 사내 직원들이 관련 정보를 보다 효율적으로 조회할수 있도록 RAG 기반 AI 챗봇 서비스를 구축한 프로젝트입니다.
협력사인 네이버 클라우드 플랫폼에서 제공한 HyperCLOVA X 모델을 기반으로, 로컬 테스트 환경에서 프론트엔드 및 백엔드 환경을 구성하였고, 사내에서 직접 배포 및 활용이 가능하도록 개발을 진행하였습니다. 모델은 내부 문서 임베딩된 벡터를 검색하여 답변을 생성하며, 향후 전사 임직원 대상 확장을 고려하여 구조를 설계하였습니다.

사용 기술

FRONTEND

- 언어 : Typescript
- 프레임워크 : Next.js (React)
- API 통신 : Axios

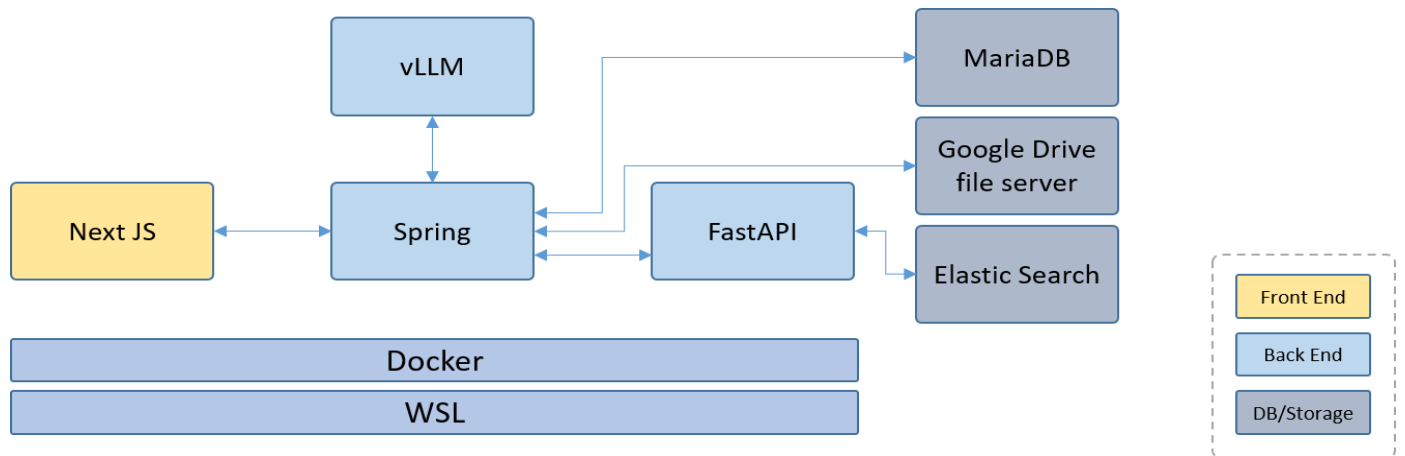
BACKEND

- 언어 : Java, Python
- 프레임워크 : Spring (API 서버), FastAPI(벡터 검색 서버)
- 벡터DB : ElasticSearch
- 데이터 저장소 : Google Drive 기반 파일 서버
- 기타 : Docker, vLLM(HCX-DASH-002), MariaDB, WSL(Ubuntu)

주요 역할 및 성과

- 사내 챗봇 프론트 화면 개발
 - Next.JS 기반 챗봇 UI 및 채팅 인터페이스 구성
 - 사용자 입력 및 AI 응답 출력 UI 구현
 - 챗봇 내부에서 사내 규정 관련 질문 입력 및 응답 기능 제공
 - 추천 질문 기능 구현
- FastAPI 기반 벡터 검색 서버 구성
 - 사내 규정 문서 전처리 및 임베딩을 통해 VectorDB (ElasticSearch) 구성
 - 사용자 질문에 대해 벡터 유사도 검색을 수행하여 적합한 문서 기반으로 응답 생성
- Spring 기반 API 서버 및 프록시 역할 구현
 - LLM 응답과 프론트 간의 통신을 담당하는 API 서버 구성
 - 사용자 인증, 문서 관리 등의 요청 흐름 중개 처리

프로젝트 구현 이전 설계



챗봇 전체 아키텍처 설계



번호	명칭	설명
1	새 대화 생성	챗봇과 대화하기 위한 새로운 대화창을 만드는 버튼
2	대화창	사용자 별 저장된 대화 이력을 확인할 수 있는 대화창
3	사용자 질문	사용자가 입력한 질문
4	LLM 답변	LLM이 생성한 답변
5	질문 입력	사용자가 입력하는 질문
6	답변 출처	LLM 답변 생성에 제공된 사용자 질문 관련 원본 문서

챗봇 채팅화면 설계

트러블 슈팅 및 기술적 이슈 해결

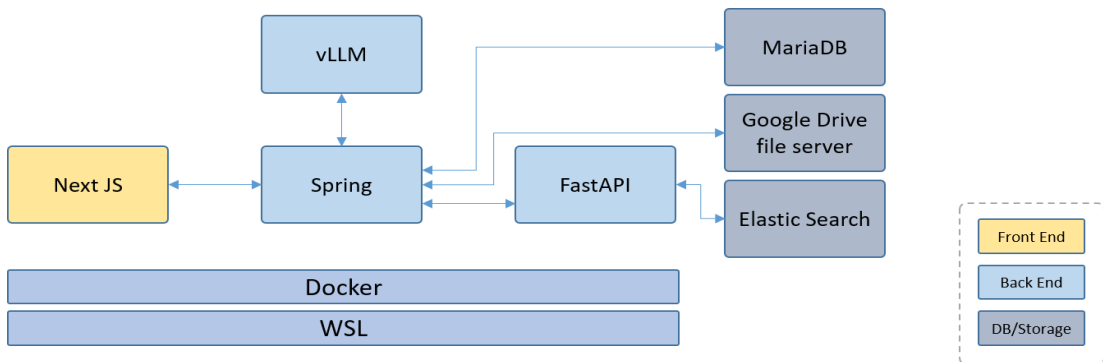
1. 내부망 환경에서 LLM API 호출 불가 문제

원인

- 프로젝트 초기에는 외부 API (예: HyperCLOVA X) 호출을 기반으로 챗봇 답변을 구성했으나, 사내 보안 정책으로 인해 내부망에서는 외부 API 호출이 차단되어 챗봇 기능이 정상 작동하지 않음
- 테스트 시 API 호출 실패 및 Timeout 등의 오류 로그가 다수 발생

해결 방법

- LLM API 의존도를 제거한 대체 아키텍처를 설계. 프로젝트 로컬 환경에서 프롬프트 튜닝을 마친 후 LLM 없이도 답변 생성이 가능하도록 RAG 기반 사내 문서 기반 검색-응답 체계를 구축.
- 이를 위해 Spring 서버를 구성 후 FastAPI를 임베딩 서버로 두어 VectorDB (Elastic Search)와 통신하게끔 하여 저장된 사내규정 정보를 가져올수 있도록 진행
- 구체적으로는 다음과 같은 구조를 가짐



성과 및 배운 점

- 네트워크 제약이 있는 환경에서도 RAG 기반 AI 응답이 가능하도록 구조를 개선함으로써, LLM 의존도를 낮추는 설계 역량과, 사내 정책을 고려한 유연한 대응 방식을 익힘.

2. 사내 문서 기반 검색 정확도 저하 이슈

원인

- ElasticSearch 기반 검색 결과에서 키워드 기반 단순 매칭으로 인해, 사용자의 질문과 관련성이 낮은 문서가 상위에 노출되는 문제가 발생

해결 방법

- 문서 임베딩 벡터를 활용한 벡터 검색(Vector Search)을 도입하여, 질문-문서 간 의미적 유사도 기반 검색이 가능하도록 개선
- Sentence-transformers 기반 사전 학습된 모델을 사용하여 문서와 질문을 벡터화한뒤, ElasticSearch의 벡터 검색 기능으로 유사도 기반 결과 제공

성과 및 배운 점

- 의미 기반 검색 도입으로 챗봇의 답변 정확도가 크게 향상되었으며, 사용자 만족도도 개선됨
- 키워드 매칭 한계를 보완하는 벡터 검색 기술을 실무에 적용해보며 검색 품질 향상에 대한 인사이트 확보

프로젝트 구현 내용 및 이미지 정리

회원가입

이름

이메일

비밀번호

보기

비밀번호 확인

회원가입

이미 계정이 있으신가요? 로그인

로그인

이메일

비밀번호

로그인

계정이 없으신가요? 회원가입

로그인

회원가입

Chat Mode

LLM

대화 목록

새 대화 시작

새로운 대화

무엇을 도와드릴까요?

글쓰기 도우미

각종 문서와 메시지 작성을 도와드립니다

여행 추천

맞춤형 여행지와 코스를 추천해드립니다

계획 수립

목표 달성을 위한 구체적인 계획을 제시합니다

브레인스토밍

창의적인 아이디어 발상을 도와드립니다

메시지를 입력하세요

채팅화면

경력기술서

지피헬스(주), [어플리케이션팀], 연구원(사원)

2021년 09월 - 2024년 3월 / 2년 6개월

치매노인 배회감지 어플리케이션 개발 (지금어디) (2023.09 ~ 2024.02) [6개월]

프로젝트 개요

- 치매노인의 실종 및 미아상황을 예방하고, 실시간 위치 파악을 통해 빠르게 대응할 수 있도록 설계된 배회감지 어플리케이션
- SigFox + GPS + WPS(WIFI) 기반의 위치추적 장치를 통해 실내외 어디서든 위치 확인이 가능하며 NFC(손톱 부착) 기술을 통해 배회감지기 분실 시에도 위치 추적이 가능한 이중 대응 구조
- 보호자와 복지사가 앱을 통해 노인의 현재 위치 확인, 발견신고, 기기 등록/관리 가능

사용 기술

FRONTEND

- 언어 : Dart
- 프레임워크 : Flutter
- API 통신 : Dio
- 상태관리 :GetX

Backend

- 언어 PHP (서버와의 연동을 위한 API 응답 처리용, 프레임워크 / 라이브러리는 별도의 백엔드 팀이 담당하여 상세 미확인)

주요 역할 및 성과

- 프론트엔드 개발
 - Flutter 기반 앱 전체 UI 및 핵심 기능 개발
 - 사용자 인증(로그인/로그아웃), 보호자 등록, 보호대상자 등록, 배회감지기 기기 등록 등 핵심 흐름 구현
 - 기능별 페이지를 명확히 분리하여 재사용 가능한 구조로 설계
 - Flutter의 Dio 라이브러리를 활용한 REST API 통신 모듈 설계 및 구현
 - 백엔드 서버와의 통신을 위한 Dio 기반 HTTP 모듈 구성
 - 데이터 전송 및 수신 과정에서 오류 처리를 위한 예외 처리 로직 적용
 - GetX 상태관리 패턴 적용을 통한 전역 상태 동기화
 - 보호자 정보, 보호대상자 정보, 기기 등록 정보 등 다수의 상태를 GetX 컨트롤러로 관리
 - 데이터 변경 시 UI에 실시간 반영되도록 구성하여 사용자 경험 향상

트러블 슈팅 및 기술적 이슈 해결

1. 보호자/기기/보호대상자 등록 화면 간 데이터 불일치 문제

원인

- 각 등록 화면이 독립적으로 동작하며, 상태값을 전역으로 공유하지 않음
- 기존에는 **Stateful Widget 중심의 로컬 상태관리**만 사용하고 있었음

해결방법

- Flutter의 **GetX 상태관리 라이브러리**를 도입하여, 보호자/대상자/기기 등의 등록 데이터를 전역 컨트롤러에서 관리하도록 구조 개편
- 기존 로직을 **ViewModel 형태로 분리**하여 재사용성과 유지보수성 향상

성과 및 배운점

- 등록된 정보가 **모든 화면에서 실시간으로 동기화**되어 UX 개선
- 전역 상태 관리 도입 시 고려사항 및 구조 분리에 대한 **실무 경험**을 습득

2. 다양한 기기에서 UI 깨짐 및 레이아웃 비정상 출력 현상

원인 :

- 해상도가 낮거나 화면 비율이 다른 기기에서 UI 요소(버튼, 텍스트 등)가 잘림
- 픽셀 기준 **고정 레이아웃 사용**으로 인해 다양한 해상도와 화면 밀도에 제대로 대응하지 못함

해결방법

- **flutter_screenutil 라이브러리**를 도입하여 기기 해상도에 따라 자동으로 비율을 조정하도록 UI 요소 크기 전면 수정
- 각 화면 초기화 시점에 ScreenUtilinit() 호출을 통해 일관된 반응형 설계 구조 수립

성과 및 배운점

- 다양한 디바이스 환경에서도 **UI가 일관되고 반응형으로 동작**하여 사용자 경험 향상
 - 반응형 설계에 필요한 기술적 적용 방법 및 **크로스디바이스 대응 전략에 대한 이해도 향상**
-

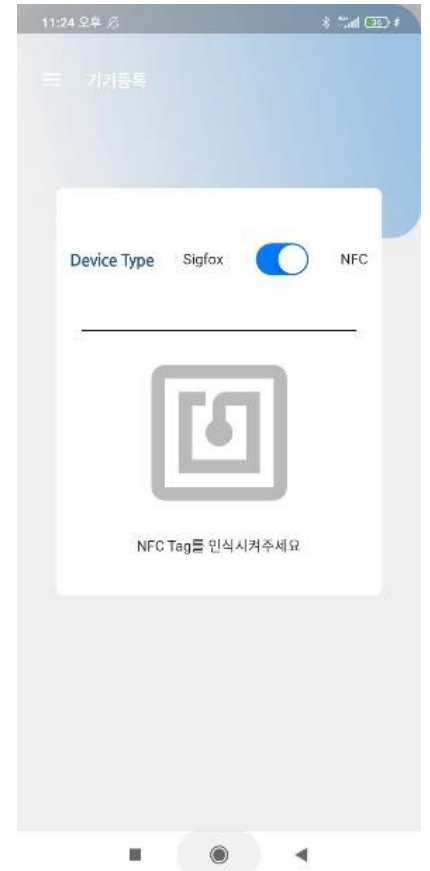
프로젝트 구현 내용 및 이미지 정리



로그인 화면



메뉴화면



NFC 인식 화면



발견신고 화면



위치확인 화면

경력기술서

지피헬스(주), [어플리케이션팀], 연구원(사원)

2021년 09월 - 2024년 3월 / 2년 6개월

반려동물 고양이 모래관리 어플리케이션 개발 (2021.12 ~ 2022.05) [6개월]

(<https://play.google.com/store/apps/details?id=com.gfyhealth.medisand&hl=ko>)

프로젝트 개요

- 고객사 '리틀캣'에 제공하기 위한 **고양이 전용 모래 건강 관리 앱**으로, 고양이의 건강 상태를 모래의 변화나 활동 패턴을 통해 간접적으로 추적할 수 있도록 사용자 중심의 UI/UX로 구현하였습니다.
- 해당 앱은 반려묘 보호자들이 건강 이상 징후를 빠르게 감지하고 주변 동물 병원 정보까지 확인할 수 있도록 **위치기반 기능**과 **다국어 대응**, **이미지 업로드 기능**을 포함한 통합 서비스로 설계되었습니다.

사용기술

FRONTEND

- 언어 : Typescript
- 프레임워크 : React
- 상태관리 / 국제화 : React-i18next (다국어 처리), useReducer, Context API
- API 통신 : Axios (지도 API 연동 등)
- 기타 : LocalStorage(로컬 데이터 저장)

BACKEND

- 구성 방식** : 별도의 백엔드 서버 없이 프론트엔드에서 직접 외부 API와 통신
- 도입이유** :
 - 본 프로젝트는 하드웨어(센서, 기기 등)와 직접 연결되는 기능을 중심으로 개발되었으며, 앱 내에서 실시간으로 디바이스를 제어해야 했기 때문에 **서버를 거치지 않고 로컬에서 처리**할 수 있도록 구성
 - 예: 설문 결과값에 따른 상태 변화, 기기 응답 등을 **LocalStorage** 기반으로 직접 처리
- API 연동 방식** :
 - 지도 기능은 Axios를 이용해 외부 API(카카오, 네이버 등)와 직접 통신
 - CORS 우회를 위해 회사 서버를 **프록시 서버(cor-anywhere 기반)**로 설정하여 API 호출 가능하게 처리

주요 역할 및 성과

- 홈/프로필 화면 구현 및 다국어 대응**
 - 사용자 정보 등록 및 고양이 정보 관리를 위한 프로필 화면 구성
 - React-i18next를 활용한 다국어 처리로 글로벌 서비스 대응 준비
- 위치 기반 동물병원 찾기 기능 개발**
 - Axios를 통한 카카오/네이버/구글 지도 API 연동
 - 가장 가까운 동물병원 자동 탐색 기능 구현
- 앱 마켓 배포 및 디버깅 대응**
 - Android/iOS 양 플랫폼 대응을 위한 빌드 및 오류 대응 완료

트러블 슈팅 및 기술적 이슈 해결

1. 지도 API 호출 실패 (CORS 정책 위반 이슈)

원인

- 병원찾기 기능 구현 시, 외부 지도 API(카카오/네이버 등)를 호출하는 과정에서 브라우저의 보안 정책(CORS)에 의해 요청이 차단됨
- 프론트엔드(React)와 외부 API 간 출처(origin)가 달라 발생한 CORS 에러

해결방법

- cor-anywhere를 사용하여 회사 서버를 프록시 서버로 설정
- 서버 측에서 Access-Control-Allow-Origin 헤더를 명시하여 프론트에서의 API 요청 허용
- 기존 React 앱에서 Axios 요청 시 해당 프록시 서버를 경유하여 API 호출 가능하도록 구조 변경

성과 및 배운점

- CORS 정책에 대한 이해를 바탕으로 프록시 서버를 활용한 유연한 구조 대응 경험
- 지도 기능 정상 작동 및 병원 위치 확인 기능 안정화
- 프론트 단에서 발생 가능한 보안 관련 이슈 대응 역량 강화

2. 다국어 처리 시 라우팅 및 텍스트 누락 이슈

원인

- React-i18next로 다국어 처리를 구현하던 중, 언어 변경 시 라우팅된 화면에서 일부 텍스트가 누락되거나 초기 언어로 되돌아가는 현상 발생
- i18n.changeLanguage() 적용 타이밍과 라우팅 리렌더링 충돌로 인한 동기화 문제

해결방법

- 언어 변경 시 전역 상태를 통해 즉시 반영되도록 useEffect 내에 i18n.language 추적 적용
- 각 라우터 진입 시 t() 함수 재호출되도록 구조 개선

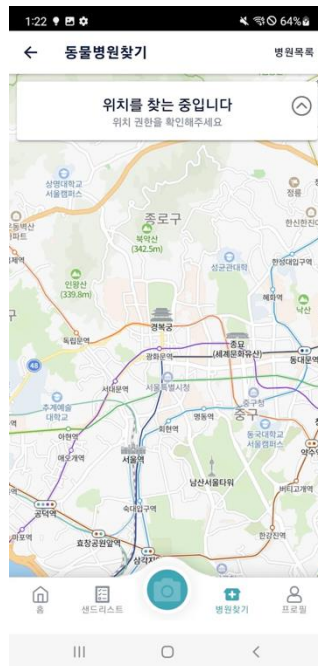
성과 및 배운점

- 동적으로 로딩되는 컴포넌트에도 Suspense와 fallback 처리 추가
 - React-i18next의 동작 메커니즘을 실무에 맞춰 조정하는 경험 축적
 - 사용자 경험(UX) 개선 및 UI 일관성 확보
-

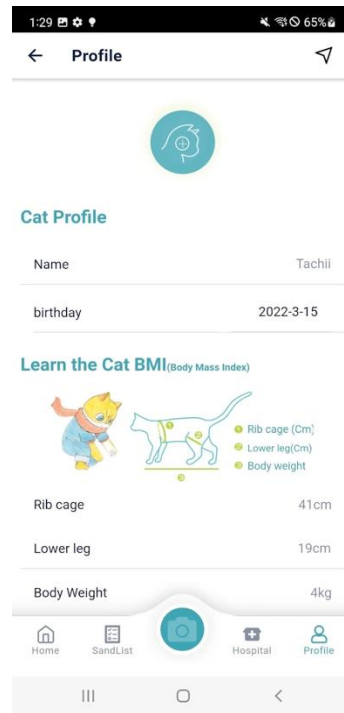
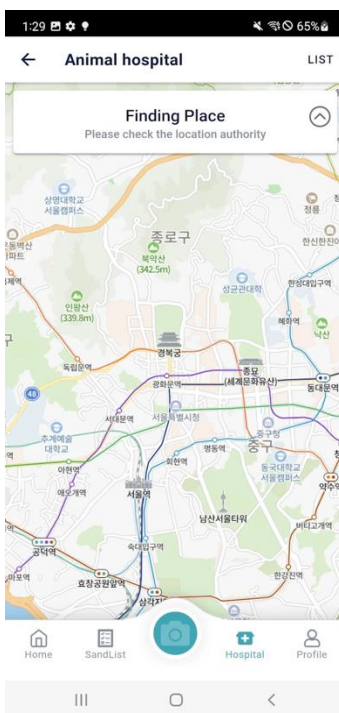
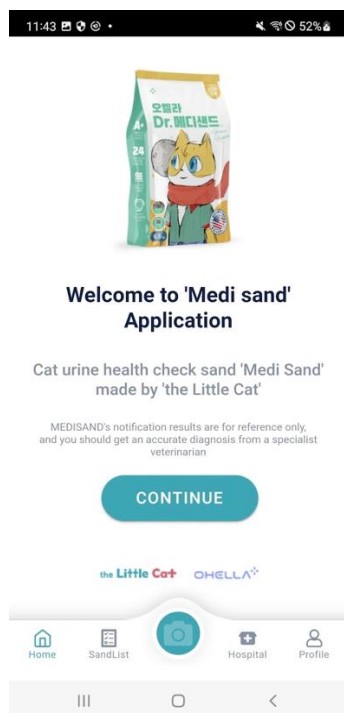
프로젝트 구현 내용 및 이미지 정리



홈화면

샌드 리스트
화면가까운 동물병원
찾기 화면

프로필 화면



경력기술서

지피헬스(주), [어플리케이션팀], 연구원(사원)

2021년 09월 - 2024년 3월 / 2년 6개월

구강 언어치료용 데스크톱 어플리케이션 개발 (2021.10 ~ 2022.04) [6개월]

프로젝트 개요

- 본 프로젝트는 치과 의사들이 환자 치료 과정에서 사용하는 구강 모형에 부착된 센서를 통해, 환자의 씹는 패턴과 구강 움직임을 인식하고 분석하는 데스크톱 어플리케이션을 개발한 사례입니다.
- 센서를 통해 수집된 데이터를 시각적으로 표현하고, 음성 녹음 및 재생 기능을 통해 환자의 언어 재활 훈련을 지원하는 UI/UX 중심의 데스크톱 기반 치료 보조 시스템입니다.

사용기술

FRONTEND (Electron 기반 데스크톱 앱)

- 언어 : Typescript
- 프레임워크 : React, Electron
- 주요 라이브러리 : Wavesurfer.js (음성 재생) Video.js (음성 녹음)
- 데이터 처리: CSV 파일 저장, 센서 데이터 시각화
- 기타 : LocalStorage (로컬 데이터 저장)

BACKEND

- 구성 방식 : 별도의 백엔드 서버 없이 프론트엔드에서 LocalStorage와 통신 진행
- 도입이유 :
 - 본 프로젝트는 치아 모형 기기에 부착된 센서를 통해 사용자의 구강 운동 데이터를 실시간으로 수집 및 제어해야 하는 구조였기 때문에, 서버를 거치지 않고 직접적으로 데이터를 처리할 수 있는 구조가 필요했습니다.

주요 역할 및 성과

센서 데이터 수집 및 시각화 기능 개발

- 하드웨어팀에서 전달받은 센서값을 실시간으로 시각화하고 CSV 파일로 저장하는 기능 구현
- Threshold 수치 보정 알고리즘 도입: 센서의 min/max 값을 기준으로 1/3, 2/3 지점을 계산하여 센서값의 신뢰도 확보

음성 기능 인터페이스 개발

- Wavesurfer.js 기반 음성 재생 기능, Video.js 기반 녹음 기능 구현
- 환자의 발음 데이터를 실시간으로 시각화하며 치료 보조에 필요한 UI 제공

하드웨어 데이터 연동 보정 알고리즘 구현

- 센서값이 침의 유무 등 외부 요인에 의해 편차가 커지는 문제 해결을 위해 센서별 Threshold 기반 수치 보정 기능 구현
- in_min, in_max 기준으로 사용자 조정 가능한 threshold 비율 계산 로직 작성

트러블 슈팅 및 기술적 이슈 해결

1. 센서값 편차가 심해 데이터 기준선 설정이 어려운 문제

원인

- 구강 모형에 부착된 센서가 사용자의 침 등에 영향을 받아 **센서값의 편차가 크게 발생함**
- 센서값이 실시간으로 변동하면서 **데이터 기준값(threshold)**을 일관되게 설정하기 어려움

해결방법

- 센서값의 안정성을 높이기 위해, **센서별 in_min, in_max 값을 기준으로 Threshold1/Threshold2를 자동 계산**하는 수식을 적용
- 각각의 센서값을 **1/3, 2/3 지점의 비율로 조절**할 수 있도록 구현하여 **데이터 기준선의 유연성 확보**
- Buffer를 활용하여 **센서값의 실시간 변화에도 안정적인 데이터 처리 가능**

성과 및 배운점

- 다양한 환경에서도 **센서 입력값의 일관성과 신뢰도**를 유지할 수 있는 **기준선 산정 구조를 확보**
- 실시간 센서 데이터 처리의 특성과 한계를 이해하고, **Threshold 비율 기반 보정 전략**을 실무에 적용
- 하드웨어와 연동되는 UI 개발 시, **데이터 수집의 정확도와 유저 피드백 사이의 균형 필요성**을 인지

2. 녹음된 음성 데이터 시각화 및 저장 과정에서 재생 지연 이슈

원인

- Video.js와 Wavesurfer.js를 병행 사용하는 구조에서, **음성 파일 처리 타이밍이 맞지 않아 재생 지연 발생**
- 특히 녹음 완료 직후 바로 재생을 시도할 경우, **파일 write 완료 이전에 read가 발생하며 오류 발생 가능**

해결방법

- FileReader 비동기 흐름을 명확히 제어하여, **파일 저장 완료 이후에만 재생 트리거가 활성화**되도록 설계
- Wavesurfer.js의 ready 이벤트를 활용**하여, 시각화 컴포넌트가 준비된 이후 재생 기능을 활성화

성과 및 배운점

- 파일 처리의 비동기 타이밍에 대한 명확한 흐름 이해**를 바탕으로, UI 반응성과 기능 안정성을 동시에 확보
 - 오디오 재생 컴포넌트 간 이벤트 흐름 설계 경험 축적
 - 프론트엔드 단에서도 **미디어 처리 안정성**을 위한 **비동기 로직 설계 역량 강화**
-

프로젝트 구현 내용 및 이미지 정리

