

날씨와 연동한 의상 추천 애플리케이션



부산대학교 정보컴퓨터공학부

팀 이름: ByteStorm

호노카 - 202255574

마디나 - 202155622

신상협 - 201424478

1. 서론: 프로젝트 배경과 필요성

날씨는 옷차림을 결정하는 데 매우 중요한 요소이다. 같은 날씨 조건이라 하더라도 사람마다 최적의 복장은 달라질 수 있다. 여기에 더해 일정(예: 회의, 데이트)이나 소지한 옷의 종류도 영향을 미친다. 기존 앱은 기온별로 일률적인 제안만 가능하여 실용성에 한계가 있다.

본 프로젝트는 '날씨 + 일정 + 보유 옷' 정보를 종합하여 사용자에게 최적의 복장을 제안하는 '날씨 및 일정 연동 개인 맞춤형 패션 추천 애플리케이션' 개발을 목표로 한다.

2. 시스템 구성 및 주요 기능

시스템은 크게 네 가지 핵심 구성 요소로 이루어져 있으며, 각 구성 요소는 다음과 같은 역할을 수행한다:

- **모바일 프론트엔드:**

사용자가 직접 앱을 통해 상호작용하는 영역으로, Flutter 또는 React Native 를 기반으로 구현된다. 사용자는 앱을 통해 자신의 옷을 등록하고, 일정을 입력할 수 있으며, 추천 코드를 확인하고 피드백을 입력할 수 있다. 또한 푸시 알림을 통해 아침마다 추천 코드를 받아볼 수 있으며, 홈 화면 위젯으로 간편한 정보 확인이 가능하다.

- **백엔드 서버:**

전체 시스템의 중심 허브 역할을 하며, Django 또는 Node.js 프레임워크로 구축된다. 서버는 사용자 요청을 처리하고, 외부 날씨 API(OpenWeatherMap, 기상청 등)로부터 실시간 데이터를 수집한다. 또한 Firebase 를 통한 사용자

인증을 수행하고, 수집된 사용자 데이터와 날씨 정보를 AI 추천 엔진에 전달하는 역할을 수행한다.

- **데이터베이스 (DB):**

모든 사용자 관련 데이터를 안전하게 저장하는 저장소이며, PostgreSQL 을 기반으로 한다. 사용자 프로필, 등록된 의류 정보, 일정, 피드백, 추천 이력 등을 구조화된 테이블 형태로 저장하고 관리한다. 이 데이터는 AI 엔진이 코디 추천을 생성할 때 활용된다.

- **AI 추천 엔진:**

전체 시스템의 핵심 요소로, 머신러닝 및 딥러닝 기반의 여러 모델로 구성된다. 먼저, 사용자 특성 분석 모듈은 OpenCV 와 MediaPipe 를 활용하여 사용자의 얼굴 및 전신 이미지를 분석하고 퍼스널 컬러, 체형, 스타일 선호도를 추정한다. 이어서 GPT-4 또는 LLaMA 를 사용해 사용자 특성과 날씨 정보를 바탕으로 자연어 설명을 생성하며, BERT4Rec 모델은 추천 아이템 조합을 구성한다. 마지막으로 XGBoost 를 통해 날씨 적합도를 판단하고 보정하여, 최적의 개인 맞춤형 코디네이션을 완성한다. 이 외에도 사용자의 피드백을 학습하여 다음 추천에 반영하는 피드백 학습 모듈을 포함하고 있다.

3. 데이터 흐름과 사용자 경험 시나리오

아래는 본 애플리케이션이 실제 사용자에게 어떻게 활용되는지를 보여주는 예시이다:

1. 사전 등록

대학생 A 씨는 전날 밤 자신의 보유 옷 사진을 앱에 등록하고, 다음 날 오전 10 시에 예정된 ‘회사 면접’ 일정을 앱의 캘린더에 입력한다. 앱에는 A 씨의 프로필 정보로 ‘가을 웜톤 피부’, ‘포멀 스타일 선호’ 등이 저장되어 있다.

2. 날씨 정보 수신 및 분석

아침 7 시가 되자 A 씨는 앱을 실행한다. 앱은 현재 위치 기반의 날씨 데이터를 자동으로 수신하며, 기온 13 도, 흐림, 약한 비 상태임을 확인한다.

3. AI 추천 코디 생성

백엔드 서버는 수집된 날씨 정보와 함께 A 씨의 일정 및 프로필 정보를 AI 추천 엔진에 전달한다. AI 는 분석을 바탕으로 ‘밝은 베이지 셔츠 + 방수 트렌치코트 + 다크 슬랙스’ 조합을 생성하고, 자연스러운 설명 문장과 함께 이미지 카드 형태로 사용자에게 제공한다.

4. 코디 적용 및 피드백 입력

A 씨는 추천 코디를 착용한 후 면접 장소로 이동한다. 면접을 마친 후, 그는 앱을 통해 “상의가 다소 추웠다”는 피드백을 남긴다.

5. 추천 엔진 개선

해당 피드백은 데이터베이스에 저장되고, AI 는 이를 학습하여 추후 유사한 날씨 조건에서 더 따뜻한 상의가 포함된 코디를 우선 추천하도록 조정한다.

이 시나리오는 사용자가 특별한 조작 없이도 일정과 날씨에 맞는 코디를 효과적으로 추천받을 수 있으며, 반복적인 피드백을 통해 AI 가 점점 더 정교해지는 과정을 보여준다.

4. 주요 기능별 기술 스택 요약

본 프로젝트에서는 다양한 최신 기술들을 적재적소에 활용하여 앞서 설명한 기능들을 구현한다. 아래 표는 주요 기능 또는 모듈에 대해 사용되는 기술 스택을 요약한 것이다:

기능 또는 모듈	사용 기술 및 스택
프론트엔드 개발	Flutter (Dart), React Native (TypeScript)
백엔드 개발	Firebase (인증 및 실시간 DB), PostgreSQL (관계형 DB 로 사용자 정보, 선호도, 기록 저장)
데이터베이스 및 인증	Firebase Authentication, PostgreSQL
이미지 및 신체 분석	OpenCV, MediaPipe, ResNet
추천 및 자연어 처리	GPT-4, LLaMA, BERT4Rec, XGBoost
클라우드 및 배포 인프라	AWS Lambda, AWS EC2, Docker
날씨 API	OpenWeatherMap, 기상청 Open API

5. 개발 일정 및 팀 역할 분담

본 프로젝트 *ByteStorm* 팀은 3 명의 개발 구성원으로 이루어져 있으며, 역할을 분담하여 약 **4 개월 간의 개발 일정**으로 진행할 예정이다. 아래에는 계획된 일정의 주요 단계와 각 단계별 담당을 정리하였다.

5.1 개발 일정 (Timeline)

단계	기간	주요 작업 내용	담당
1. 요구사항 분석 및 설계	1 주차	사용 사례 정의, 기능 확정, 아키텍처 설계, DB 모델링, 기술 스택 결정	팀 전원 (브레인스토밍)
2. 데이터 수집 및 베이스 구축	2~4 주차	의류 이미지 및 날씨 데이터 수집, 퍼스널 컬러 분류 기준 정의	마디나(주도), 신상협(보조)
3. AI 모델 개발	5~8 주차	사용자 특성 분석 모델, 코디 추천 알고리즘 구현 및 테스트	신상협(주도), 마디나(보조)
4. 프론트엔드 및 백엔드 개발	5~8 주차	모바일 UI 제작, REST API 구현, 프론트-백엔드 연동 테스트	호노카(주도), 마디나(협업)
5. 통합 및 시스템 테스트	9~10 주차	전체 시스템 통합, 실날씨 기반 테스트, 버그 수정 및 최적화	팀 전원
6. 베타 출시 및 피드백 수집	11~12 주차	알파테스터 배포, 피드백 수렴 및 개선	호노카(총괄), 신상협/마디나
7. 최종 발표 및 배포	13 주차	보고서 작성, 시연 준비, 앱스토어 배포	팀 전원

5.2 역할 분담 (Team Roles)

- **호노카** – 팀장 역할 및 **프론트엔드 개발** 및 UX 디자인 담당. Flutter 기반 모바일 앱 개발을 주도하고, 사용자 경험 개선, 화면 설계, 아이콘/이미지 등의 시각 디자인 요소를 챙긴다. 또한 알파테스터와의 커뮤니케이션을 통해 요구사항을 수렴한다.
- **신상협** – **AI/ML 리드** 개발자. 퍼스널컬러 분류, 추천 알고리즘 및 모델 학습 파이프라인 구축을 총괄하며, 프로젝트 전반 일정 관리 및 조율을 담당.
- **마디나** – **백엔드 개발** 및 데이터베이스 관리자. Django 서버 및 RESTful API 를 구현하고, Firebase 인증과 PostgreSQL 데이터 모델을 관리한다. 또한 서버와 AI 엔진 통신 모듈을 작성하여 통합을 도모하고, 클라우드 배포 환경 세팅(AWS 등)을 맡는다.