**디지털 핵심 실무인재 양성사업(K-Digital Training)**

**융복합 프로젝트 기획안 V.1.3 updated 23.02.12**

**2023년 02월 12 일**

| 프로젝트 조 | 4강의장 3조 | | |
| --- | --- | --- | --- |
| 프로젝트 팀원 | 팀명: 피존(PZONE)  팀장: 조정희(클)  팀원: (빅) 박민정, 서대훈, 이준희 (IoT) 김양호 (클) 김효빈, 강위, 윤일선 | | |
| 프로젝트 주제 | 개인형 이동장치 주차 시스템 개선 | | |

| 프로젝트 목적 | 현 상황   * 개인형 이동장치 증가: 공유 개인형 이동장치는 지속적으로 증가하고 있음 * PM 주차장 늘어나고는 있지만, 그 속도가 개인형 이동장치 증가에 미치지 못함.   + 목적지 근처에 없는 경우도 있고 (‘22년 기준 마포구 32개소, 강남구 6개소 등)   + 주차장을 운영하지 않는 지역도 있음. * 그렇기 때문에 주차 공간이 아닌 곳에 주차를 할 수 밖에 없다? 하려 한다?   문제 상황   * =========== 다른 사람의 불편함 ===========, 사고 건수, 견인 건수, 민원 건수 등등 혹은 뉴스 헤드라인 * 이를 해결하기 위해 개인형 이동장치 업체에서 다양한 것을 시도하고 있음   기존 서비스 분석   * 지쿠터: 지도에 금지 구역 적색 표시 및 금지 구역 주차 시 추가금 부여, 주차 시 이동장치선가 포함된 사진을 찍어야 반납( 견인 시 증명용 ) , 주차 결과 확인 * 스윙: 지도에 금지 구역 마커로 표기 * 킥고잉: 주차 관련 없는 것 같음 * 빔: 지도에 적색 표시, 페널티 및 반납 금지 * (일본) 반납 시 주차장의 라인을 인식해야만 반납할 수 있는 시스템 * ⇒ 현 사용자가 주차를 잘 하는지 관리를 잘 안 함 – 민원이 들어오거나 견인되면 페널티 → 즉각적인 피드백 * ⇒ 실제로 사람들이 불편을 겪는 점을 판단하지 못함.   + 1) 지상 소화전, 지하철 역 입구, 버스정거장 등을 포함되지 않음 → GPS   + 2) 보도블럭 중앙, 점자블럭 등 주차된 공간 노면 → 이미지   + 3) 주차 가능 구역이라 하더라도 킥보드가 누워져 있는 것도 통행에 방해가 됨 → 센서 * ⇒ 인근에 주차 가능 구역 위치를 알기 어려움(반납 확인 필요)   서비스 소개   * 개인형 이동장치의 즉각적인 해결을 위해 사진, 위경도, 기울기 데이터를 통해 주차 가능 점수로 적합도 알림   + 이용자의 주차 매너 점수 기준을 두어 제재를 통해 교통질서 개선 * 현재 위치로부터 가장 근처의 개인형 이동장치 주차장 안내   기대 효과   | 기존 서비스 한계점 혹은 문제 등등 | 우리 서비스에서 어떻게 하려고 하는지 | 기대 효과 | | --- | --- | --- | | ⇒ 현 사용자가 주차를 잘 하는지 관리를 잘 안 함 – 민원이 들어오거나 견인되면 페널티 → 즉각적인 피드백 | 점수제 도입  금지구역에 따라 가중치를 부여했다 | 즉각적으로 사용자의 주차 상태 파악 가능 | | 1) 지상 소화전, 지하철 역 입구, 버스정거장 등을 포함되지 않음 → GPS | GPS 분석: 주차금지구역을 다양하게 추가했다 | 실제로 사람들이 불편을 겪는 점을 추가로 개선할 수 있움 | | 2) 보도블럭 중앙, 점자블럭 등 위치 정보로만 파악하기 어려운 구역이 있음 | 이미지 분석: 노면 상태 분석 | | 3) 주차 가능 구역이라 하더라도 킥보드가 누워져 있는 것도 통행에 방해가 됨 -> 킥보드 누워져 있는 거 파악 못 함 | 기울기 센서 분석: 킥보드가 누워져 있는지 파악 | | 인근에 주차 가능 구역 위치를 알기 어려움  (요 부분은 실제로 반납할 때 어떻게 되는지 확인 필요) | 인근 주차 가능 구역 위치와 경로를 안내함 | 주차를 하기 전에는 목적지 근처 주차장 확인할 수 있음  주차할 때에는 확실하게 주차가 가능한 구역을 확인할 수 있음 |   ==== 마무리 ====  확장 가능성   * 사용자의 주차 점수를 통해 납부 보험료 측정 * 모니터링 시스템 기준으로 사용 가능   + 특히 평균 점수가 낮은 회원의 경우   보완점   * 안내가 필요함 : 점수가 어떤 의미가 있느냐 왜 마커가 마커로 설정되었느냐 * 사진 데이터의 사각지대 이용   + 주차 금지 구역을 등지고 찍음: 주차 가능 구역을 연석으로 한정 + timestamp   + 저장된 이미지 사용 못 하게 * 주차금지구역 관련해서 뉴스, 리뷰 등 다양한 데이터 분석 ← 무슨 근거로 주차금지구역 선정함? * 기울기 센서의 연장선으로 주차 위치의 경사로 각도? 경사 파악해서 점수에 반영 * client가 보낸 이미지 데이터로 지속적인 모델 개선 architecture 만들기 * WAF 사용 * SECRET MANAGER 사용 * GITHUB ACTION 사용 * 백업 로그 사용 * s3 thumbnail 사용 | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 필수 기능 | 1. 개인형 이동장치 주차 가능 점수 안내  * 사용자가 개인형 이동장치가 주차된 모습을 촬영한 사진으로 개인형 이동장치가 놓인 곳이 자전거도로, 점자블럭, 횡단보도 근처인지 파악 * 사용자의 핸드폰 GPS로 버스정류장, 지하철역 근처에 개인형 이동장치가 주차되었는지 파악 * 기울기 센서 데이터로 현재 개인형 이동장치가 누워있는 상태인지 파악 * 위 데이터들을 분석해 점수 환산 및 알림  1. 개인형 이동장치 주차장 안내  * 개인형 이동장치 전용 주차 구역 위치 안내 * 사용자의 핸드폰 GPS로 현재 위치에서 제일 가까운 개인형 이동장치 전용 주차 구역의 위치를 안내  1. 계정 평균 점수 관리:  * 계정의 평균 점수를 통해 등급 분류 * 등급에 따른 서비스 이용 안내 | | |
| **빅데이터** | **IoT** | **클라우드** |
| 전송받은 사진으로 주차금지구역 판별  주차금지구역인 경우, 근처의 주차 가능 지역 안내  현재 위치 정보와 DB의 주차장 정보를 비교해 가까운 주차장 위치  데이터분석 모델을 통해 받은 데이터들을 분석 및 점수환산 | IoT 기울기 센서 -> NODE MCU -> IOT 서버 까지의 기울기 데이터를 획득 및 서버로 전송 | -모델 서버와 백엔드서버 웹서버를 통하여 사진 위경도 데이터 전송 점수 환산 및 전달  -RDS 서버-백엔드서버-웹서버를 통하여 현재 위치와 가까운 주차장 비교, 알림 서비스 배포 및 가까운 주차금지 구역 안내 및 점수 환산  -센서 ->IOT 서버 -> 백엔드 서버->모델서버를 통한 데이터 전송 및 분석을 통해 점수 환산 및 전달 |
| UI/UX 구성도 | 홈 - 앱 안내 , 주차 금지구역 안내,주차 적합도 확인 버튼, 근처 주차장 안내 버튼  주차 적합도 확인 버튼 -모바일 카메라 사진 촬영 및 위치정보보 전송 후  주차 점수 안내 및 점수에 따른 주차 적합도 안내  현 위치 근처 주차 금지구역 지도 표기 및 사진 내 주차 금지구역 표기  현 사진 내 금지구역 안내 및 현 위치 내 근처 금지구역 안내  주차장 안내 버튼 - 현 위치를 기반으로 가장 가까운 주차장 까지 최단 거리 표기  위치 정보 및 도착지까지 거리 표기 | | |
| 인프라 구성도 | 주요 툴: AWS, kubernetes  EKS cluster 생성( VPC , worker NODE, 가용영역, NAT gateway , Internet gateway 등등 )  Frontend service, Backend service ,model service 로 금지구역 안내 서비스  Frontend service, Backen service , RDS server연결로 근처 주차장 안내 서비스  ALB(INGRESS)로 서비스 path 지정 및 인증서 처리  HPA로 pod들에 일정 traffic 이상 걸릴 시 autoscaling  CA로 t3.medium node에 허용 pod 개수 넘길시 node autoscaling | | |
| 포함기술 | **빅데이터** | **IoT** | **클라우드** |
| **현황 분석**  Pandas  Matplotlib  **주차금지구역 판별**  YOLO darknet  **주차장 위치 데이터 변환**  Pandas  KakaoMap API | Mjpeg stream(라즈베리파이) -> 메   |  | | --- |   시지 보내기 | -WEB SERVER-  Nginx  -WEB FRAMEWORK-  Django  fastapi  -CONTAINER-  Docker(Container),  Kubernetes(Container Oche)  -AWS CLOUD-  EFS(File system)  IOT(이미지 데이터 전송및 이벤트 발생)  EKS(kubernetes)  EC2 |

## ㅁWBS첨부

[WBS\_3조.xlsx](https://docs.google.com/spreadsheets/d/1v7AYUUgMotmvAEZxlxQ8DLHqR3kLiNAy/edit?usp=sharing&ouid=116444438415932754699&rtpof=true&sd=true)