
컴퓨터과학부 종합설계 4조 사용자가 참여하는 **키패드 주차존**

2015920021 박인규
2017920011 김은해
2017920036 양다은
2018920059 허정우

공용 킥보드 주차장 제공 어플리케이션

- 퍼스널 모빌리티 편의성을 해치지 않고 보행자의 통행이나 킥보드 업체에 실질적으로 도움이 되는 주차장
- 사용자 기반 데이터 수집
- 공간적인 데이터 수집 및 분석 진행
- API 형태로 제공



어플리케이션 기능



1) 주차공간 추천

- 사용자가 이용하기 편한 주차공간 추천
- 건물 또는 시설 관계자가 주차금지공간 추천



2) 주차장 평가

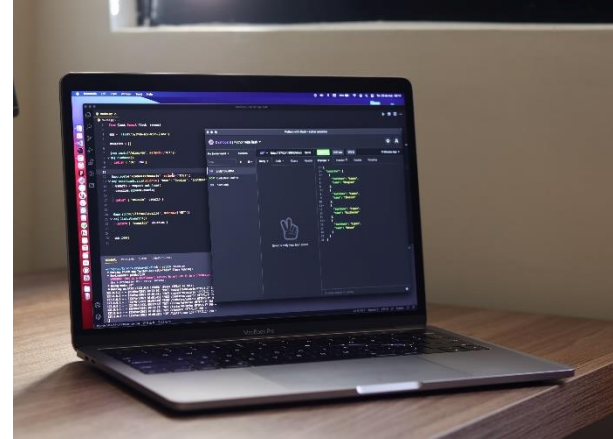
- 제안된 주차장의 사용성 검토
- 주차장과 관련된 의견 수집

어플리케이션 기능



3) 주차장 제안

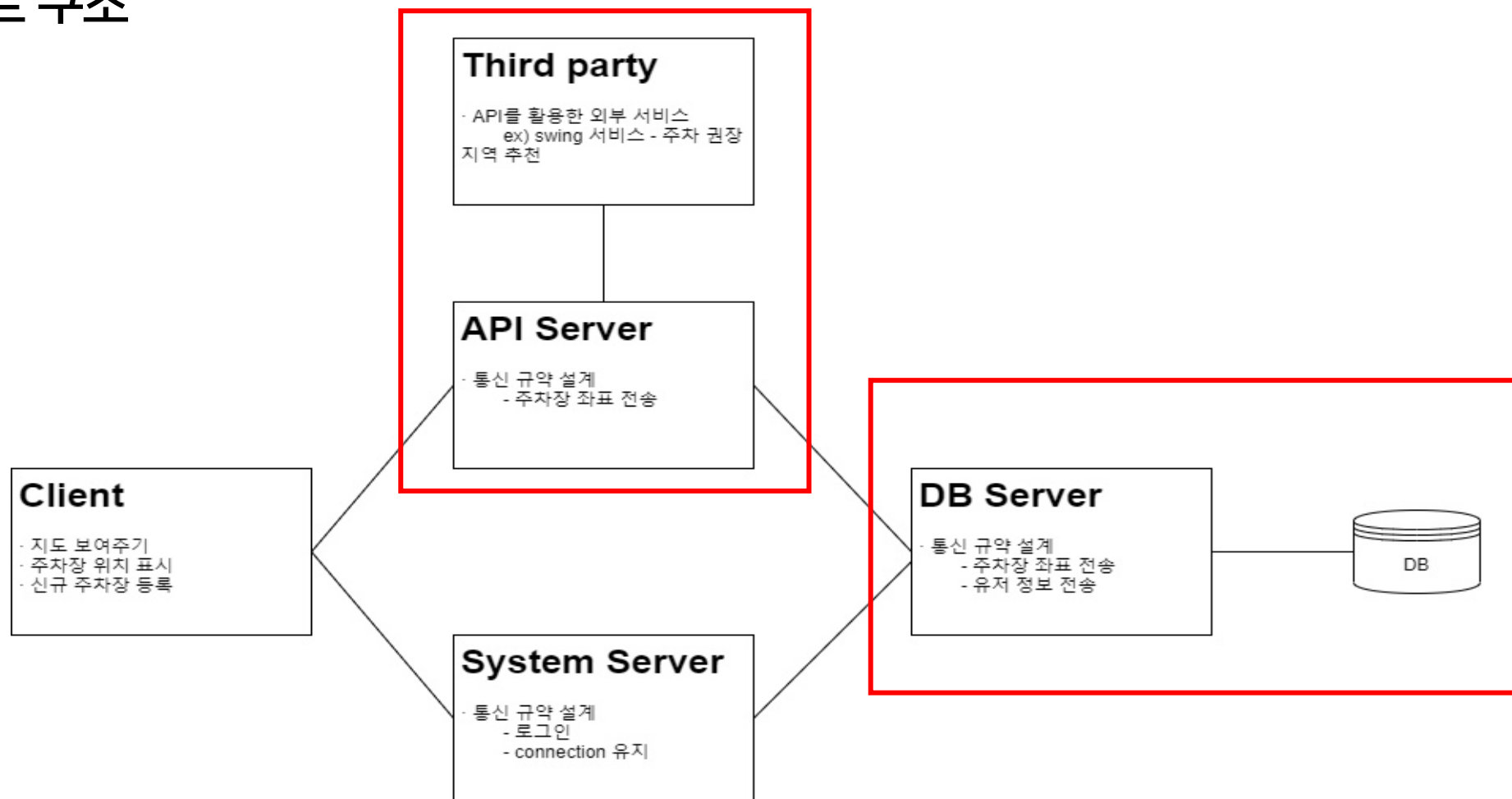
- 데이터를 기반으로 분석한 주차장 제공
- 주차장의 평가, 위치, 형태 등 관련 정보



4) 주차장 제공

- 추천한 주차장과 관련 정보를
이용하기 쉽게 가공

프로젝트 구조



+ 클러스터링 알고리즘

10주차 회의 내용

[클러스터링 방식 조사 및 논의]

1. DBSCAN(밀도기반 클러스터링)
2. CLARANS : PAM과 CLARA를 결합한 알고리즘
3. 격자기반 클러스터링
4. Fuzzy clustering

[공인 IP를 할당하여 서버간 연동 준비]

- DB서버에서 접속충돌 발생 -> 오류 검토중

[담당 파트 개발 진행 각 서버 진행상황 보고]

[관리자 페이지 구축]

- 추가적 구현 페이지(논의중)

11주차 회의 내용

[적용할 클러스터링 알고리즘 결정 : DBSCAN(밀도기반 클러스터링)]

- 영역정보를 격자형 점 정보로 변환한 뒤에 일반적인 point 클러스터링 알고리즘을 적용해서 군집을 만들고 그걸 다시 영역 정보로 적용
- 작은 데이터셋 뿐만 아니라, 대용량 공간데이터에 대해서도 효율적으로 군집화 연산이 가능

- 알고리즘 과정

- A. 임의의 점 p 선택 (어떤 값이든 동일한 결과 출력)

- B. 해당 p 에서 ϵ 만큼 떨어진 공간이 $MinPts$ 개수를 만족시키는지 확인

$$N_{\epsilon}(p) = \{ q \in D \mid dist(p, q) \leq \epsilon \}$$

- i. 만족시킨다면 $N_{\epsilon}(p)$ 가 다시 ϵ 만큼 떨어진 공간에 $MinPts$ 개수를 만족시키는지 확인

- ii. 만족시키지 못한다면 다시 임의의 점 p 선택

- C. $N_{\epsilon}(p)$ 를 확장시켜 나가면서 $N_{\epsilon}(p)$ 를 만족시키지 못하는 경계 포인트(border point)가 얻어질 때까지 진행

11주차 회의 내용

[클러스터링 후 지정된 주차장 데이터의 정확성에 대한 논의]

= 클러스터링 알고리즘이 추가적인 구현이 필요할 경우

- “오류제보” 버튼 추가 구현
- 더미데이터 추가
- 더 효율적인 딥러닝 알고리즘 앙상블 진행

[공인 IP 할당하여 개인 IP 배분 -> 접속충돌 문제 해결]

- DB서버에서 접속충돌 -> 오류: 이미 사용중인 포트 -> 서버포트 재할당 -> 문제 해결

[담당 파트 개발 진행 보고]

- DBserver - APIserver 연동을 위한 담당자 오프라인 논의

[관리자 페이지 구축을 위한 담당 업무 추가]

- 페이지 구현에 필요한 데이터 정리 및 CRUD 작성
- 관리자 페이지 UI 제작

11주차 회의 내용 [DBserver - APIserver 연동을 위한 담당자 소규모 논의]

Schema	Term	Used for & Event
users	유저 정보	<ul style="list-style-type: none"> sign up/sign in한 유저에 대한 데이터 저장 유저별 comments(댓글) 배열 정보 저장
zones	주차공간 정보	<ul style="list-style-type: none"> 유저가 추천한 주차공간 혹은 주차금지공간 (위도, 경도) 좌표형태 데이터 저장
parklots	주차장 정보	<ul style="list-style-type: none"> 유저기반 공간데이터를 기반으로 클러스터링을 진행한 후 생성된 주차장 (위도, 경도) 좌표형태 데이터 저장 UI를 위한 주차장 이름 데이터 저장 섹션(section)별 주차장들을 구분하여 sections 스키마에 저장
rates	주차장 평가 정보	<ul style="list-style-type: none"> 제안한 주차장에 대한 유저 평가 ⭐/👍 데이터 저장 주차장 사용성 검토를 위한 스키마로, 해당 데이터도 추후 업데이트 클러스터링 알고리즘의 데이터로 사용 Boolean 아닌 Number 타입
comments	주차장 댓글 정보	<ul style="list-style-type: none"> 제안한 주차장에 대한 유저에 대한 텍스트 평가 저장 스키마 검증(validate) 옵션으로 텍스트 수 제한 작성한 유저와 주차장 정보 저장

11주차 회의 내용 [DBserver - APIserver 연동을 위한 담당자 소규모 논의]

- Zone Information

Verb	Action	Path	Used for
GET	read All	/zones/	주차포인트 목록 조회
POST	create	/zones/	신규 주차포인트 생성
GET	read	/zones/no/:no	point_no가 no인 주차포인트 조회
PUT	update	/zones/no/:no	point_no가 no인 주차포인트 갱신
DELETE	delete	/zones/no/:no	point_no가 no인 주차포인트 삭제
GET	read	/zones/id/:id	_id가 id인 주차포인트 조회
PUT	update	/zones/id/:id	_id가 id인 주차포인트 갱신
DELETE	delete	/zones/id/:id	_id가 id인 주차포인트 삭제

[요청 URI]

no : 검색, 갱신, 삭제할 zoneid의 값

id : 검색할 _id의 값

-> 여기 수정할 예정 : no와 id 동작이 동일하여
통합할 것!

User 스키마는 System서버와 통신 및
CRUD 작성 - 보안문제

11주차 회의 내용 [DBserver - APIserver 연동을 위한 담당자 소규모 논의]

- Parklot Information

Verb	Action	Path	Used for
GET	read All	/parklots/	주차장 목록 조회
POST	create	/parklots/	신규 주차장 생성
GET	read	/parklots/no/:no	park_no가 no인 주차장 조회
PUT	update	/parklots/no/:no	park_no가 no인 주차장 갱신
DELETE	delete	/parklots/no/:no	park_no가 no인 주차장 삭제
GET	read	/parklots/id/:id	_id가 id인 주차장 조회
PUT	update	/parklots/id/:id	_id가 id인 주차장 갱신
DELETE	delete	/parklots/id/:id	_id가 id인 주차장 삭제

[요청 URI]

no : 검색, 갱신, 삭제할 lotid의 값

id : 검색할 _id의 값

11주차 회의 내용 [DBserver - APIserver 연동을 위한 담당자 소규모 논의]

- Rate Information

Verb	Action	Path	Used for
GET	read All	/rates/	평가 목록 조회
POST	create	/rates/	신규 평가 생성
GET	read	/rates/no/:no	lotid가 no인 평가 조회
PUT	update	/rates/no/:no	lotid가 no인 평가 갱신
DELETE	delete	/rates/no/:no	lotid가 no인 평가 삭제
GET	read	/rates/id/:id	_id가 id인 평가 조회
PUT	update	/rates/id/:id	_id가 id인 평가 갱신
DELETE	delete	/rates/id/:id	_id가 id인 평가 삭제

[요청 URI]

no : 검색, 갱신, 삭제할 lotid의 값

id : 검색할 _id의 값

11주차 회의 내용 [DBserver - APIserver 연동을 위한 담당자 소규모 논의]

- Comment Information

Verb	Action	Path	Used for
GET	read All	/comments/	댓글 목록 조회
POST	create	/comments/	신규 댓글 생성
GET	read	/comments/id/:id	_id가 id인 댓글 조회
PUT	update	/comments/id/:id	_id가 id인 댓글 갱신
DELETE	delete	/comments/id/:id	_id가 id인 댓글 삭제

[요청 URI]

id : 검색할 _id의 값

11주차 회의 내용 [DBserver - APIserver 연동을 위한 담당자 소규모 논의]

- Comment Information

Verb	Action	Path	Used for
GET	read All	/comments/	댓글 목록 조회
POST	create	/comments/	신규 댓글 생성
GET	read	/comments/id/:id	_id가 id인 댓글 조회
PUT	update	/comments/id/:id	_id가 id인 댓글 갱신
DELETE	delete	/comments/id/:id	_id가 id인 댓글 삭제

[요청 URI]

id : 검색할 _id의 값

11주차 회의 내용 [DBserver - APIserver 연동을 위한 담당자 소규모 논의]

```
{
  "_id": "6191f728c9de61f83e1b0968",
  "lotid": 1,
  "latitude": "127.5461353",
  "longitude": "37.51354",
  "comments": [
    {
      "user": {
        "_id": "6191f6cec9de61f83e1b0964",
        "nickname": "이루매"
      },
      "comment": {
        "_id": "6191f76dc9de61f83e1b096e",
        "comment": "이루매 많이많이 사랑해주세요!"
      },
      "_id": "6191f76dc9de61f83e1b096f"
    }
  ],
  "rate": {
    "_id": "6191f728c9de61f83e1b0969",
    "like": 0,
    "dislike": 0
  },
  "createdAt": "2021-11-15T05:59:04.219Z",
  "updatedAt": "2021-11-15T06:00:13.766Z",
  "__v": 1
}
```

EX)

- GET방식
- 경로 : /parklots/no/:no
- lotid가 no인 주차장 조회(Read)

DB서버와 API서버 연동되어 동작 확인

향후 계획

[연동 및 통신을 위한 소규모 논의 진행]

- DB서버 - System서버 / System서버 - Client/Client - API서버 각 담당자끼리 매끄러운 연동 및 통신을 위한 소규모 논의 진행

[오프라인 미팅 : 11월 말 예정]

- 시스템 활성화(동작)
- 스트레스 테스트 프로그램, 서버 모니터링 프로그램 진행
- 어플리케이션 검토
- 추가적인 업무 배분
- 담당지도교수님과 최종미팅 일정을 조율할 예정