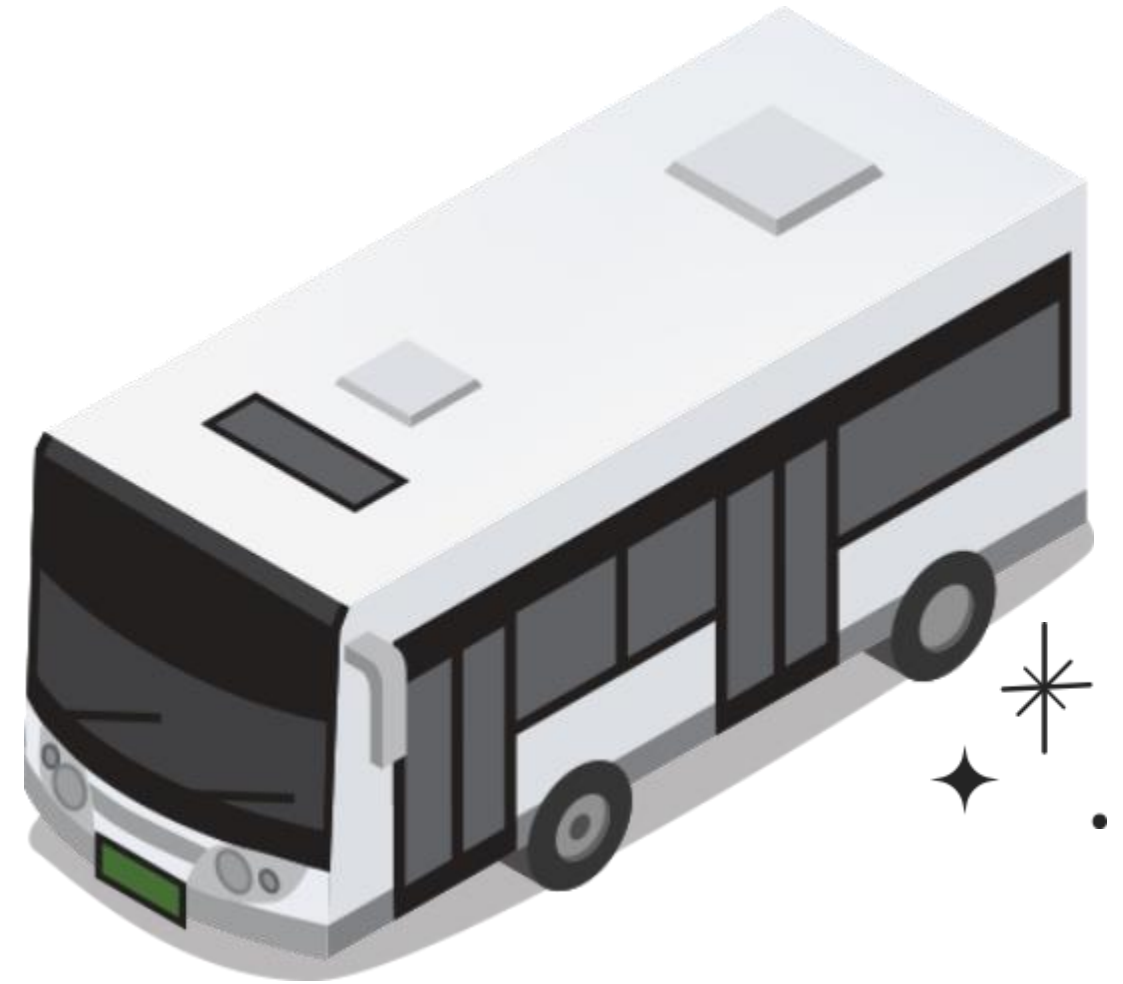




버스 하차 알리미

AI융합학부
20241966 박수빈





목차

개발 동기 01

핵심 코드 설명 03

개발환경 05

느낀점 07

구현 방법 02

UI 04

기대효과 06

1

개발 동기



대학 입학 후 처음으로 장시간 버스통학을 하게 됨

-> 긴 이동시간으로 인해 깊은 잠을 자게 됨

-> 자느라 내려야 할 정류장을 지나치게 됨

-> 많은 사람들이 비슷한 불편을 겪는다는 것을 알게 됨

-> 직접 프로그램을 만들어 불편함을 해소하고자 함

2

구현 방법



1. Flask 서버 설정

-클라이언트에서 요청을 받아 처리하는 API 엔드포인트를 설정합니다.

2. HTML 페이지 생성 & JavaScript 코드 작성

-카카오 지도를 사용하여 사용자가 정류장을 선택하고, 현재 위치를 서버에 전송하여 거리를 계산합니다.



3

핵심 코드 설명 1. Flask 서버 측 코드

Flask 서버 설정 및 라우트 정의

Flask 서버를 설정하고, 클라이언트에게 HTML 페이지를 제공하는 라우트를 정의

```
from flask import Flask, render_template, request, jsonify
from haversine import haversine, Unit

app = Flask(__name__)

selected_station = None

@app.route('/')
def index():
    return render_template('index.html')
```

```
data = request.json
current_location = (data['lat'], data['lng'])

print(f"Current location: {current_location}")
print(f"Selected station: {selected_station}")

try:
    distance = haversine(current_location, selected_station, unit=Unit.KILOMETERS)
    print(f"Current distance: {distance} km")

    if distance <= 5:
        print("하차 준비를 시작하세요!")
        return jsonify({"status": "success", "message": "하차 준비를 시작하세요!"})
    else:
        return jsonify({"status": "success", "distance": distance})
except Exception as e:
    print(f"Error calculating distance: {e}")
    return jsonify({"status": "error", "message": "Error calculating distance"})

if __name__ == '__main__':
    app.run(debug=True)
```

현재 위치 업데이트 및 거리 계산

- 클라이언트에서 POST 요청으로 전송된 현재 위치를 받아와서 selected_station과의 거리를 계산
- 거리가 5km 이내인 경우 "하차 준비를 시작하세요!" 메시지를 클라이언트에 전송



3

핵심 코드 설명 2. 클라이언트 측 코드

```
<script>
  var map;
  var marker;
  var selectedPosition = null;

  function initMap(position) {
    var lat = position.coords.latitude;
    var lng = position.coords.longitude;

    var mapContainer = document.getElementById('map'),
        mapOption = {
          center: new kakao.maps.LatLng(lat, lng),
          level: 3
        };

    map = new kakao.maps.Map(mapContainer, mapOption);

    var currentMarker = new kakao.maps.Marker({
      position: new kakao.maps.LatLng(lat, lng)
    });
    currentMarker.setMap(map);
  }
</script>
```

지도 초기화 및 현재 위치 설정

- 사용자의 현재 위치를 가져와 지도의 중심으로 설정
- 사용자의 현재 위치에 마커를 표시합니다.

4

비

버스 승하차 알리미



이 페이지 내용:

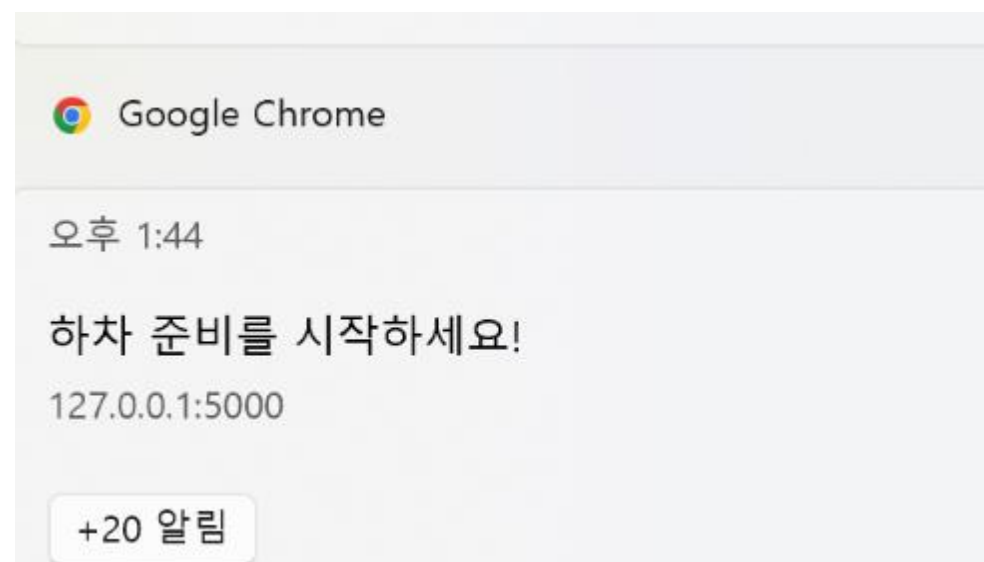
클릭한 위치의 위도는 37.18673932386062 이고, 경도는 127.07049500316957 입니다

확인

4

미

버스 승하차 알리미



5

개발환경

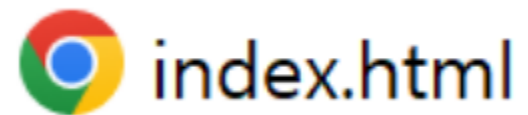
Visual Studio Code



Flask: 웹 애플리케이션 프레임워크

Haversine: 두 좌표 사이의 거리를 계산하기 위한 라이브러리

HTML



Visual Studio Code를 통해 편집한 코드를
Google Chrome를 통해 디버깅 함



카카오 지도 API

kakao developers

카카오 디벨로퍼스에 가입하여 지도 API 키를 발급받습니다.

6

기대효과

1. 사용자 편의성 향상

학생 및 직장인

통학이나 출퇴근 시 버스 이용이 잦은 사용자에게 매우 유용할 수 있다.

관광객

낯선 도시를 여행하는 관광객이 버스를 보다 효율적으로 이용할 수 있도록 도와준다.

2. 확장 가능성

다양한 기능 추가 가능

향후 버스 노선 정보, 정류장 정보, 대중교통 연계 정보 등을 추가하여 더욱 유용한 애플리케이션으로 발전시킬 수 있다.

3. 데이터 수집

앱을 통해 사용자에게 대한 다양한 데이터를 수집하게 되고 이는 더욱 효율적인 버스 운영 및 노선 조정이 가능해지게 하며, 이는 교통 시스템 전반의 품질을 향상시킬 수 있습니다.

7

느낌점

사용자의 관점에서 문제 해결을 위해 생각하고 또 이를 구현하는 프로젝트를 완성해나가는 과정에서 Flask를 사용한 백엔드 서버와 JavaScript를 사용한 프론트엔드가 어떻게 상호작용하는지 배울 수 있었습니다. 특히, API를 통해 데이터를 주고받는 과정에서 많은 것을 배웠습니다. 처음 API를 다루게 되어 많은 책과 자료들을 참고했으나 완벽하게 이해하는 데는 꽤나 오랜 시간이 걸렸습니다. 카카오 지도 API를 사용하여 외부 데이터를 활용하는 방법을 익힐 수 있었습니다. 이는 다른 프로젝트에서도 유사한 API를 활용하는 데 큰 도움이 될 것이라고 생각했습니다. 또 프로젝트 진행 중간에 원하는 방향으로 프로젝트가 흘러가지 않아 처음부터 다시 시작해야하는 상황에 처했었습니다. 하지만 긴 시간에 걸쳐 원하는 결과물을 얻어낼 수 있었고 이런 경험을 통해 나중에도 프로젝트를 진행하는 과정에서의 좋은 밑바탕이 될 것이라고 생각했습니다.



THANK YOU!

경청해주셔서 감사합니다.

