



교내 셔틀버스 안내 어플리케이션

제주대학교 순환버스정보시스템 결과 보고서

2019208013 김규리
2019208014 김난화
2019208015 김소연



Contents



- JNU BUS 배경 및 필요성
- 기존 사례 조사
- 프로젝트의 목표
- 개발시스템의 범위
- 개발시스템의 운영환경
- 프로젝트 팀 구성표
- 개발일정
- 제약사항 및 해결방안
- 시스템 구조도
- 모듈설명
- 구현
- 기대효과 및 활용방안
- 결론
- 미구현된부분
- 프로젝트 의견조사




JNU BUS




기존 교내순환버스의 출도착예정시간 그리고 버스의 노선도를 한눈에 확인할 수 있는 방법이 없어 순환버스의 이용에 어려움이 있다.

모바일 어플리케이션을 이용해 교내순환버스의 다양한 정보를 손쉽게 파악할수 있게 해 순환버스의 이용에 도움을 주게 한다.

JNU BUS 앱의 필요성

- 교내 순환버스의 전체노선 및 시간표를 간편하게 확인할 수 있다.
- 각 노선 정문 출발 시간을 표기해 버스가 언제 출발했는지 알 수 있다.
- 정류장별로 도착예정 시간을 표기해 사용자가 시간을 확인하고 버스가 언제 도착하는지 확인할 수 있다.
- GPS시스템을 이용해 사용자와 가까운 정류장의 위치파악과 도착예정시간을 파악할 수 있다.
- 순환버스를 탔을 때 사용자의 위치를 파악해 다음 도착할 정류장을 알려주어 사용자의 하차를 돕는다.

	개요	주요기능	제공 콘텐츠 (데이터)	유료/무료	개발자	크기	개발날짜
 제대로 통한다	제대로 오는 버스의 출도착시간이 주요서비스 부가적으로 교내 순환버스의 출도착시간을 표기	1.제대에서 출발하는버스시간표기 2.제대에 도착하는 버스표기 3.교내순환버스 출발시간을 표기한후 출발시간을 기준으로 도착예상시간표기	텍스트 이미지	무료	Hong Seong HoS	25MB	2017.11.29
 호서버스	호서대학교 캠퍼스를 연결하는 순환버스 출도착예정 알림	1.각 캠퍼스간 버스 출발시간표기 2. 버스 출발시간을 기준으로 도착예정시간표기 3.슬라이드로 그 다음 도착예정시간 표기 4.웹페이지로 연결해 전체시간표 표기	텍스트 이미지	무료	Leomaja	14.02MB	2018.1.10
 한성대버스	한성대학교 셔틀버스 출도착 시간알림 서비스와 셔틀버사이외 2대의 버스 출도착시간 알림	1.셔틀버스 출도착 시간 표기 2.셔틀버스의 위치 시각적으로 표기 3.그외 일반버스2대의 시간표표기 4.일반버스2대의 위치시각적으로 표기 5.간단한 전체시간표 표기	텍스트 이미지	무료	한성대학교	2.56MB	2019.3.29

	개요	주요기능	제공 콘텐츠 (데이터)	유료/무료	개발자	크기	개발날짜
 서울대학교 셔틀버스	순환버스 및 서울대로 오는 셔틀버스의 출도착 시간을 알려주는 앱	1.교내버스정보및 역까지의 버스정보표시 2.출발지와 목적지에따른 노선과 배차간격 3.GPS기능과 정류장검색기능 4.운행시간 알리기능	텍스트 이미지	무료	서울대학교	573.4KB	2014.1.28
 연세 Y버스	연세대 교내 순환버스 노선정보알림 앱	1.캠퍼스 별로 교내 버스 노선 확인 2.노선에 따라 현재 버스 위치 확인가능 3.버스의 출도착 예정시간 및 셔틀버스 시간표확인 은 외부 웹페이지 링크기능으로 이용	텍스트 이미지	무료	Yonsei University	1.23MB	2015.5.5
 제주버스	제주 시내버스 노선 및 정류장 정보를 알려주는 어플리케이션	1.정류장 및 노선별로 자세한 버스 노선확인 2. 버스 노선에 따라 현재 버스 위치 확인 3.지도를 통해 버스의 대략적 위치와 전체적인 버 스 노선을 파악	텍스트 이미지 지도	무료	Hi App's	3.68MB	2014.5.13

초기목표기능

- 각 노선의 전체노선도 확인 및 현재버스위치 확인
- 정문출발 시간 표기 및 각 정류장별 도착예정시간 표기
- 가까운 정류장 확인
- 전체시간표 확인 및 검색

세부목표

기존 순환 버스앱의 사례들과 다르게 JNU BUS 앱은

- 구글 혹은 다음 지도 앱을 이용해 각 노선도를 자세히 확인할 수 있다.
- 각 정류장별 도착예정시간을 확인할 수 있다.
- 가까운 정류장을 탭 한번으로 확인이 가능하다.
- 각 정류장별로 정리된 전체시간표를 확인할 수 있다.
- 정류장 시간표를 검색할 수 있다.

변경된 목표 기능

- 각 노선의 전체노선도 확인 현재버스위치 확인
- 전체시간표 확인
- 나의 위치를 GPS로 받아 나와 가까운 버스정류장의 위치 확인

변경된 세부목표 기능

- 지도 이미지를 이용한 전체노선도 확인 및
- 현재시간에 따른 버스위치 애니메이션 구현으로 대체
- 각 정류장별 버튼을 이용한 전체시간표 확인

초기목표범위

- 각 노선의 전체노선도 확인 및 현재버스위치 확인
- 전체시간표 확인 및 검색
- 나의 위치를 GPS로 받아 나와 가까운 버스정류장의 위치 알림

실제 구현한 범위

- 각 노선의 전체노선도 및 정류장 확인
- 현재 버스위치 애니메이션으로 구현화
- 각 정류장별 전체시간표 로딩

하드웨어

사용기종	갤럭시노트FE
운영체제	안드로이드9.0
사용버전	API 28 PIE

소프트웨어

프로그램	안드로이드스튜디오
사용언어	JAVA

초기 인력 역할

기획 및 설계

김소연, 김난화

개발

김소연, 김규리, 김난화

변경된 역할

기획 및 설계

김소연, 김난화

메인메뉴

김난화

전체시간표

김규리

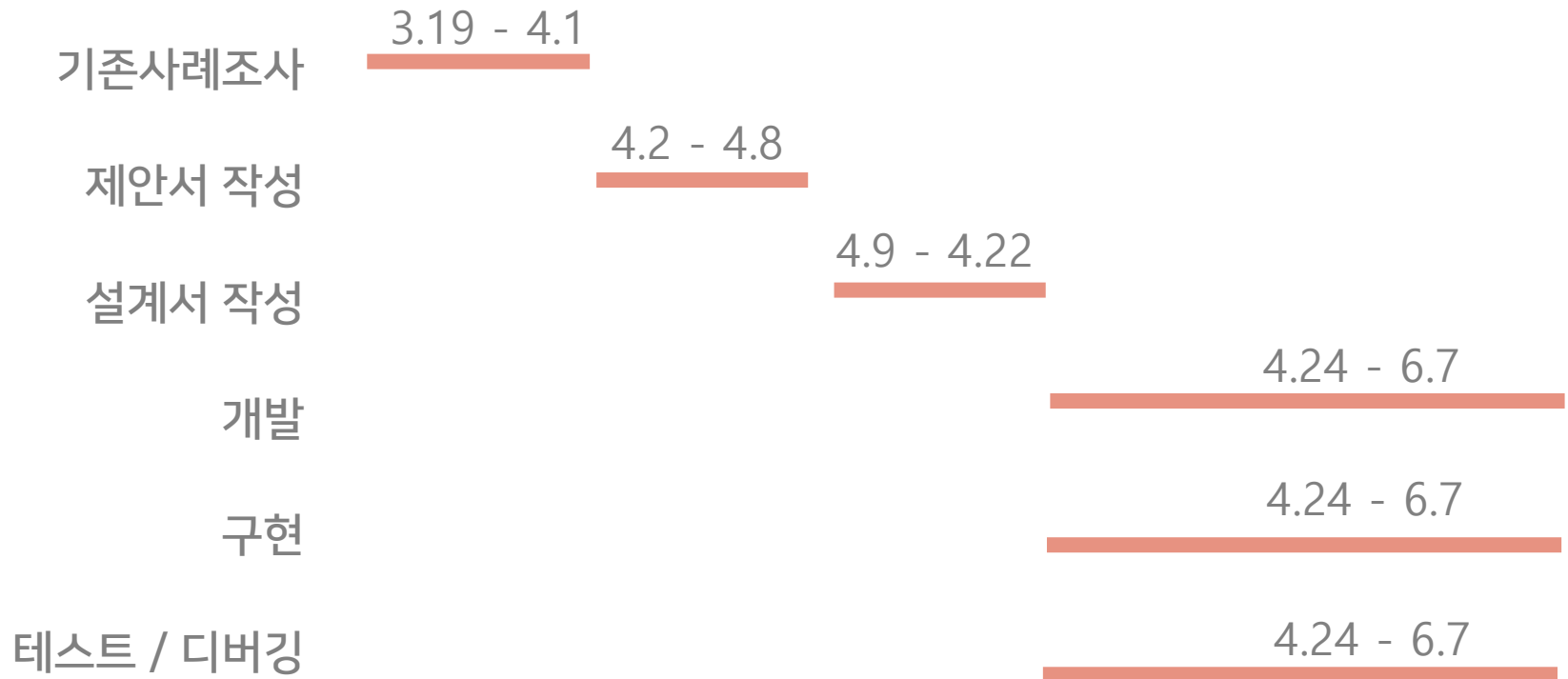
GPS

김소연

계획서 상의 일정

	3월 29일 ~ 4월 8일	4월 9일 ~ 4월 30일	5월 1일 ~ 5월 21일	5월 22일 ~ 6월 15일	담당자
기존 사례 조사	분석작업 3/29-4/1				김소연
제안서 작성	분석작업 4/2-4/8				김소연
설계서 작성		설계서 작성 4/9-4/22			김난화
데이터 정리		설계서 작성 4/22-4/30			김난화
메인 화면 제작			화면 구성 5/1-5/7		김소연
UI 제작			화면 구성 5/8-5/15		김소연
GPS API 로딩			화면 구성 5/15-5/17		김소연
노선도 제작			화면 구성 5/18-5/19		김소연
각 정류장 표기			화면 구성 5/19-5/21		김소연
정류장별 시간 표기 기능				개발 5/22-6/1	김난화
전체 시간표 기능				개발 6/2-6/5	김난화
사용자 GPS 허용기능				개발	김난화
테스트				개발 5/22-6/15	김난화
수정 및 백업				개발 5/25-6/15	김난화

실제 개발상의 일정



01 제약사항



안드로이드 앱 개발 환경으로 제한한다.

노선도 및 전체시간표는 한눈에 확인할수 있어야 한다.

오래된 버전의 안드로이드 폰에서도 구동이 가능해야 한다.

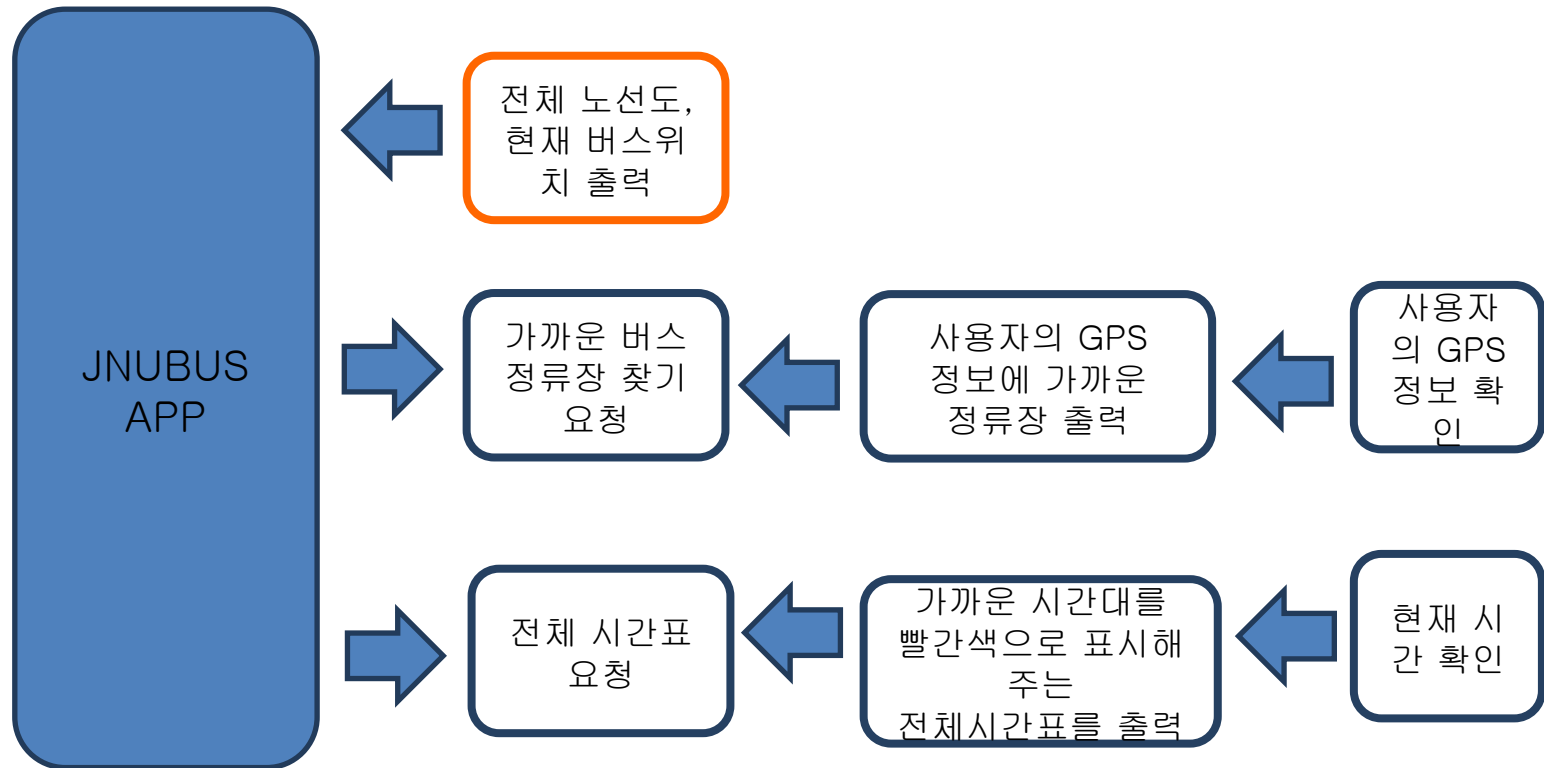
02 해결방안



안드로이드 스튜디오를 이용한 안드로이드 앱 개발 환경

한눈에 보기 쉬운 화면구성으로 디자인

최소 구동버전을 아이스크림버전으로 설정



화면	기능
1.메인화면	노선도 및 버스현재 위치 표기
2.하단바	노선도/전체시간표/근처버스정류장/설정
>노선도	A,B 각 노선의 전체노선도 및 버스현재위치
>전체시간표	A,B 각 노선의 정류장별 전체시간표 표기
>근처버스정류장	현재나의 위치 확인후 가까운 정류장 확인
>설정	GPS 동의 여부설정

사용자인터페이스 설계



1. 로딩화면(GPS 동의)

JNU BUS 로고 표시 및
GPS 동의 여부 팝업 표시



2-1. 메인화면

A, B 노선에 따른 버스 정류장 위치와 현재 버스 위치를 표시

사용자인터페이스 설계



2 - 2. 전체시간표 표시 및 검색

A노선 / B노선에 따른
각 노선별 시간표 및 검색

사용자인터페이스 설계



2 - 3. 사용자 근처 버스정류장 표시

사용자의 GPS를 통해
현재 위치에서 가장 가까운
버스 정류장 표시

사용자인터페이스 설계



2 - 4. GPS 설정 , 앱 설정

GPS 동의 여부 설정 및
앱의 현재 버전과 앱 정보 표시

현재구현된 부분

1. 로딩화면

JNU BUS 앱의 실행을
알리는 메인로고화면



```
<?xml version="1.0" encoding="utf-8"?>
<vector xmlns:android="http://schemas.android.com/apk/res/android"
    android:width="141.85dp"
    android:height="279.48dp"
    android:viewportWidth="141.85"
    android:viewportHeight="279.48">

    <path
        android:fillColor="#282828"
        android:pathData="M 0 0 H 141.85 V 279.48 H 0 V 0 Z"
        android:fillType="nonZero" />

    <path
        android:fillColor="#fff"
        android:pathData="M50.84,149l0.45-1.78c0.57 0.05 ,1.09 0.07 ,1.56 0.07 a3.8,3.8,0,0,0,2.38-0.52
        android:fillType="nonZero"/>

    <path
        android:fillColor="#fff"
        android:pathData="M73.19,148.92H71L65.6,140l-2,8.93H61.37L64,137H66.9l4.88,8.15,1.8-8.15h2.24Z"
        android:fillType="nonZero"/>

    <path
        android:fillColor="#fff" android:fillType="nonZero"
        android:pathData="M90.22,137l-1.7,7.4a7.49,7.49,0,0,1-0.85,2.26,4.28,4.28,0,0,1-1.87,1.43,7.71,
    <path
        android:fillColor="#fff" android:fillType="nonZero"
        android:pathData="M71,120V102.3l18.5 0.8 L92,120Z" />

    <path
        android:fillColor="#fff" android:fillType="nonZero"
        android:pathData="M49.91,115c0-1.32 0.05 -2.59,0-3.87,0-0.74 0.2 -1.09 0.94 -1.4,5.73-2.36,11.4
    <path
        android:fillColor="#fff" android:fillType="nonZero"
        android:pathData="M60.41,126.09a13.66,13.66,0,0,1,0.57-4,3.32,3.32,0,0,1,0.36-0.76c0.35-0.53 0.
```

현재구현된 부분

2-1. 메인화면

A, B 노선에 따른 버스 정류장
위치와 현재 버스 위치를 표시



```
package com.example.jnubus_19_06_02;
import android.Manifest;
import android.content.Context;
import android.content.Intent;
import android.location.LocationManager;
import android.support.annotation.NonNull;
import android.support.design.widget.BottomNavigationView;
import android.support.v4.app.Fragment;
import android.support.v4.app.FragmentManager;
import android.support.v4.app.FragmentTransaction;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.LayoutInflater;
import android.view.Menuitem;
import android.view.View;
import android.view.animation.Animation;
import android.view.animation.AnimationUtils;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.Toast;
```

```
import com.gun0912.tedpermission.PermissionListener;
import com.gun0912.tedpermission.TedPermission;
```

```
import java.util.ArrayList;
```

```
public class MainActivity extends AppCompatActivity {
```

```
// FrameLayout에 각 메뉴의 Fragment를 바꿔 줌
final FragmentManager FM = getSupportFragmentManager();
// 4개의 메뉴에 들어갈 Fragment들
```

```
final RoutemapActivity Routemap = new RoutemapActivity();
```


현재구현된 부분



	A노선	B노선
정문	08:05	14:30
제2도서관	08:25	15:05
해양대1호관	08:45	15:25
본관	09:30	15:45
학생회관	10:05	16:30
인문대 서쪽	10:25	17:00
학생생활관	10:45	17:20
인문대 동쪽	11:20	17:40
중앙도서관	12:40	18:00
의전원	13:05	18:20
공대4호관	13:25	18:40
교양동	13:45	

2-2. 전체시간표 화면

A노선 / B노선에 따른
각 노선별 시간표 및 검색

```
package com.example.jnubus_19_06_02;
```

```
import android.os.Bundle;
import android.support.annotation.NonNull;
import android.support.annotation.Nullable;
import android.support.v4.app.Fragment;
import android.util.Log;
import android.view.LayoutInflater;
import android.view.View;
import android.view.ViewGroup;
import android.graphics.Color;
import android.support.v7.app.AppCompatActivity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.TextView;
```

```
public class TimeTableActivity extends Fragment {
    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable View
```

```
View view = inflater.inflate(R.layout.time_table, container, false);
```

```
final TextView text1 = (TextView)view.findViewById(R.id.textView1);
final TextView text2 = (TextView)view.findViewById(R.id.textView2);
final TextView text3 = (TextView)view.findViewById(R.id.textView3);
final TextView text4 = (TextView)view.findViewById(R.id.textView4);
final TextView text5 = (TextView)view.findViewById(R.id.textView5);
final TextView text6 = (TextView)view.findViewById(R.id.textView6);
final TextView text7 = (TextView)view.findViewById(R.id.textView7);
```

현재구현된 부분



2-3. 사용자 근처 버스정류장 표시

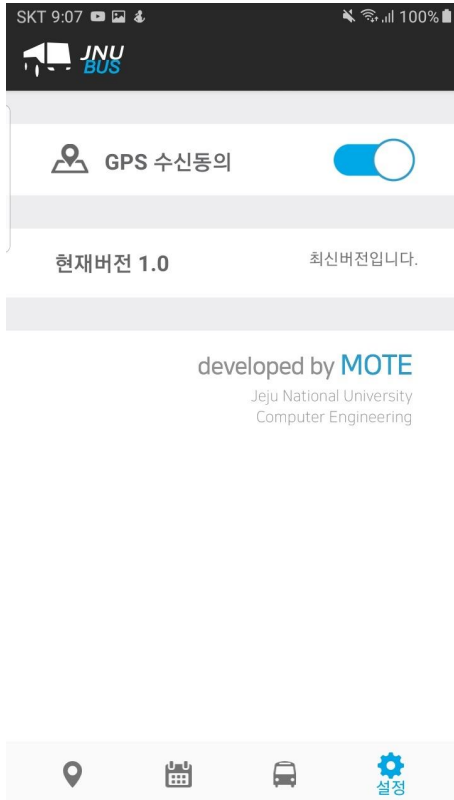
사용자의 GPS를 통해
현재 위치에서 가장 가까운
버스 정류장 표시

```
package com.example.jnubus_19_06_02;

import com.google.android.gms.common.ConnectionResult;
import com.google.android.gms.common.api.GoogleApiClient;
import com.google.android.gms.common.api.PendingResult;
import com.google.android.gms.common.api.ResultCallback;
import com.google.android.gms.common.api.Status;
import com.google.android.gms.location.LocationListener;
import com.google.android.gms.location.LocationRequest;
import com.google.android.gms.location.LocationServices;
import com.google.android.gms.location.places.Place;
import com.google.android.gms.location.places.PlaceLikelihood;
import com.google.android.gms.location.places.PlaceLikelihoodBuffer;
import com.google.android.gms.location.places.Places;
import com.google.android.gms.location.places.ui.PlaceAutocompleteFragment;
import com.google.android.gms.location.places.ui.PlaceSelectionListener;
import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.MapView;
import com.google.android.gms.maps.MapInitializer;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.model.BitmapDescriptorFactory;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.Marker;
import com.google.android.gms.maps.model.MarkerOptions;

public class NearBusstopActivity extends Fragment implements OnMapReadyCallback,
    GoogleApiClient.ConnectionCallbacks,
    GoogleApiClient.OnConnectionFailedListener,
    LocationListener
{
    private static final LatLng DEFAULT_LOCATION = new LatLng(33.454758, 126.565427);
    private static final String TAG = "googlemap_example";
    private static final int GPS_ENABLE_REQUEST_CODE = 2001;
```

현재구현된 부분



2-4.GPS 설정 화면

GPS 동의 여부 설정 및
앱의 현재 버전과 앱 정보 표시

```
package com.example.jnubus_19_06_02;
```

```
import android.annotation.TargetApi;
import android.app.AlertDialog;
import android.app.Service;
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Location;
import android.location.LocationListener;
import android.location.LocationManager;
import android.os.Build;
import android.os.Bundle;
import android.os.IBinder;
import android.provider.Settings;
import android.support.v4.content.ContextCompat;
```

```
public class GpsInfo extends Service implements LocationListener {
    private final Context mContext;
```

```
// 현재 GPS 사용유무
boolean isGPSEnabled = false;
```

```
// 네트워크 사용유무
boolean isNetworkEnabled = false;
```

```
// GPS 상태값
boolean isGetLocation = false;
```

```
Location location;
double lat; // 위도
double lon; // 경도
```

```

public class MainActivity extends AppCompatActivity {

    // Franelayout에 각 메뉴의 Fragment를 바꿔 줌
    final FragmentManager FM = getSupportFragmentManager();
    // 4개의 메뉴에 들어갈 Fragment들

    final RoutemapActivity Routemap = new RoutemapActivity();
    final TimeTableActivity Timetable = new TimeTableActivity();
    final NearBusstopActivity NearBusstop = new NearBusstopActivity();
    final NavigationSettingActivity NavigationSetting = new NavigationSettingActivity();
    Fragment active = Routemap;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        BottomNavigationView bottomNavigationView = findViewById(R.id.bottom_navigation_view);
        // 첫 화면 지정
        final FragmentTransaction transaction = FM.beginTransaction();
        FM.beginTransaction().add(R.id.frame_layout, NavigationSetting).hide(NavigationSetting).commit();
        FM.beginTransaction().add(R.id.frame_layout, NearBusstop).hide(NearBusstop).commit();
        FM.beginTransaction().add(R.id.frame_layout, Timetable).hide(Timetable).commit();
        FM.beginTransaction().add(R.id.frame_layout, Routemap).commit();

        // bottomNavigationView의 아이템이 선택될 때 호출될 리스너 등록
        bottomNavigationView.setOnNavigationItemSelectedListener((item) -> {
            switch (item.getItemId()) {

```

```

package com.example.jnubus_19_06_02;

import ...

public class RoutemapActivity extends Fragment {

    @Nullable
    @Override
    public View onCreateView(@NonNull LayoutInflater inflater, @Nullable ViewGroup container, @N
        setActivityChange();
        View view = inflater.inflate(R.layout.route_map, container, attachToRoot: false);
        final ImageView bus = (ImageView) view.findViewById(R.id.bus);
        final ImageButton aline = (ImageButton) view.findViewById(R.id.aline);
        final ImageButton bline = (ImageButton) view.findViewById(R.id.bline);

        final Animation anima = AnimationUtils.loadAnimation(getContext(), R.anim.seta_anim);
        final Animation animb = AnimationUtils.loadAnimation(getContext(), R.anim.setb_anim);
        bus.startAnimation(anim);
        // a노선/b노선 버튼
        aline.setOnClickListener((v) -> {
            aline.setImageResource(R.drawable.aline);
            bline.setImageResource(R.drawable.bline_not);
            bus.startAnimation(anim);
        });
        bline.setOnClickListener((v) -> {
            aline.setImageResource(R.drawable.aline_not);
            bline.setImageResource(R.drawable.bline);
            bus.startAnimation(anim);
        });

        return view;
    }
}

```

```

@Override
protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    BottomNavigationView bottomNavigationView = findViewById(R.id.bottom_navigation_view);
    // 첫 화면 지정
    final FragmentTransaction transaction = FM.beginTransaction();
    FM.beginTransaction().add(R.id.frame_layout, NavigationSetting).hide(NavigationSetting).commit();
    FM.beginTransaction().add(R.id.frame_layout, NearBusstop).hide(NearBusstop).commit();
    FM.beginTransaction().add(R.id.frame_layout, Timetable).hide(Timetable).commit();
    FM.beginTransaction().add(R.id.frame_layout, Routemap).commit();

    // bottomNavigationView의 아이템이 선택될 때 호출될 리스너 등록
    bottomNavigationView.setOnNavigationItemSelectedListener((item) -> {
        switch (item.getItemId()) {
            case R.id.navigation_routemap: {
                FM.beginTransaction().hide(active).show(Routemap).commit();
                active = Routemap;
                return true;
            }
            case R.id.navigation_timetable: {
                FM.beginTransaction().hide(active).show(Timetable).commit();
                active = Timetable;
                return true;
            }
            case R.id.navigation_nearbusstop: {
                FM.beginTransaction().hide(active).show(NearBusstop).commit();
                active = NearBusstop;
                return true;
            }

```

```
import ...

public class NearBusstopActivity extends Fragment implements OnMapReadyCallback,
    GoogleApiClient.ConnectionCallbacks,
    GoogleApiClient.OnConnectionFailedListener,
    LocationListener
{
    private static final LatLng DEFAULT_LOCATION = new LatLng( v: 33.454758, vt: 126.565427);
    private static final String TAG = "googlemap_example";
    private static final int GPS_ENABLE_REQUEST_CODE = 2001;
    private static final int PERMISSIONS_REQUEST_ACCESS_FINE_LOCATION = 2002;
    private static final int UPDATE_INTERVAL_MS = 15000;
    private static final int FASTEST_UPDATE_INTERVAL_MS = 15000;

    private GoogleMap googleMap = null;
    private MapView mapView = null;
    private GoogleApiClient googleApiClient = null;
    private Marker currentMarker = null;

    private final static int MAXENTRIES = 5;
    private String[] LikelyPlaceNames = null;
    private String[] LikelyAddresses = null;
    private String[] LikelyAttributions = null;
    private LatLng[] LikelyLatLngs = null;

    public NearBusstopActivity()
    {
        // required
    }

    public void setCurrentLocation(Location location, String markerTitle, String markerSnippet) {
```

```
@TargetApi(23)
public Location getLocation() {
    if ( Build.VERSION.SDK_INT >= 23 &&
        ContextCompat.checkSelfPermission(
            mContext, android.Manifest.permission.ACCESS_FINE_LOCATION )
            != PackageManager.PERMISSION_GRANTED &&
        ContextCompat.checkSelfPermission(
            mContext, android.Manifest.permission.ACCESS_COARSE_LOCATION )
            != PackageManager.PERMISSION_GRANTED) {

        return null;
    }
    try {
        locationManager = (LocationManager) mContext
            .getSystemService(LOCATION_SERVICE);

        // GPS 정보 가져오기
        isGPSEnabled = locationManager
            .isProviderEnabled(LocationManager.GPS_PROVIDER);

        // 현재 네트워크 상태 값 알아오기
        isNetworkEnabled = locationManager
            .isProviderEnabled(LocationManager.NETWORK_PROVIDER);

        if (!isGPSEnabled && !isNetworkEnabled) {
            // GPS 와 네트워크사용이 가능하지 않을때 소스 구현
        } else {
            this.isGetLocation = true;
            // 네트워크 정보로 부터 위치값 가져오기
            if (isNetworkEnabled) {
                locationManager.requestLocationUpdates(
```

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.design.widget.CoordinatorLayout xmlns:android="http://schemas.android.com/apk/res-auto"
    android:id="@+id/main_container"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <!-- 각각의 Fragment를 넣을 Container -->
    <FrameLayout
        android:id="@+id/frame_layout"
        android:layout_width="match_parent"
        android:layout_height="674dp" />

    <android.support.design.widget.BottomNavigationView
        android:id="@+id/bottom_navigation_view"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_gravity="bottom"
        android:background="?android:attr/windowBackground"
        app:menu="@menu/bottom_nav_menu"/>

</android.support.design.widget.CoordinatorLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorBackground">

    <ImageView
        android:layout_width="match_parent"
        android:layout_height="30dp"
        android:background="@color/colorLine" />

    <TextView
        android:id="@+id/textView2"
        android:layout_width="130dp"
        android:layout_height="60dp"
        android:layout_alignParentStart="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:layout_marginStart="82dp"
        android:layout_marginLeft="82dp"
        android:layout_marginTop="34dp"
        android:gravity="center"
        android:text="GPS 수신동의"
        android:textColor="@color/colorText"
        android:textSize="20dp"
        android:textStyle="bold" />

    <ImageView
        android:id="@+id/imageView2"
```

```
<TextView
    android:id="@+id/textView3"
    android:layout_width="188dp"
    android:layout_height="60dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="2dp"
    android:layout_marginLeft="2dp"
    android:layout_marginTop="127dp"
    android:gravity="center"
    android:text="현재버전 1.0"
    android:textColor="@color/colorText"
    android:textSize="20dp"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="154dp"
    android:layout_height="60dp"
    android:layout_alignParentStart="true"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:layout_marginStart="248dp"
    android:layout_marginLeft="248dp"
    android:layout_marginTop="123dp"
    android:gravity="center"
    android:text="최신버전입니다." />
```

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#000000">

    <!-- Activity에서는 fragment로 구현해야되지만
    Fragment로 구현하기위해서는 MapView를 사용해야함-->
    <com.google.android.gms.maps.MapView
        android:id="@+id/map"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:name="com.google.android.gms.maps.MapFragment"
    />

    <fragment
        android:id="@+id/place_autocomplete_fragment"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:name="com.google.android.gms.location.places.ui.PlaceAutocompleteFragment"
    />
</RelativeLayout>
```

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <ImageView
        android:id="@+id/imageView"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:contentDescription="JNUBUS"
        android:translationY="-50dp"
        app:srcCompat="@drawable/map1" />

    <ImageView
        android:id="@+id/bus"
        android:layout_width="37dp"
        android:layout_height="40dp"
        android:translationY="122dp"
        android:translationX="192dp"
        app:srcCompat="@drawable/bus" />

    <View
        android:id="@+id/view2"
        android:layout_width="match_parent"
        android:layout_height="80dp"
        android:alpha="0.5"
        android:background="#ffff" />

    <ImageButton
        android:id="@+id/align"
        android:layout_width="60dp"
```

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="700dp"
    android:background="@color/colorBackground"
    tools:context=".TimeTableActivity">

    <ImageView
        android:id="@+id/imageView7"
        android:layout_width="133dp"
        android:layout_height="match_parent"
        android:layout_marginStart="0dp"
        android:layout_marginLeft="0dp"
        android:layout_marginTop="0dp"
        app:srcCompat="@color/colorLine" />

    <ImageView
        android:id="@+id/imageView8"
        android:layout_width="242dp"
        android:layout_height="70dp"
        android:layout_marginStart="0dp"
        android:layout_marginLeft="0dp"
        android:layout_marginTop="0dp"
        app:srcCompat="@color/colorLine" />

    <Button
        android:id="@+id/button1"
        style="?android:attr/borderlessButtonStyle"
```

SKT 9:03 100%

JNU BUS

🔍	A노선	B노선
정문	08:05	14:30
제2도서관	08:25	15:05
해양대1호관	08:45	15:25
본관	09:30	15:45
학생회관	10:05	16:30
인문대 서쪽	10:25	17:00
학생생활관	10:45	17:20
인문대 동쪽	11:20	17:40
중앙도서관	12:40	18:00
의전원	13:05	18:20
공대4호관	13:25	18:40
교양동	13:45	

📍 시간표 🚌 ⚙️

SKT 9:03 100%

JNU BUS

🔍	A노선	B노선
정문	08:12	14:37
제2도서관	08:32	15:12
해양대1호관	08:52	15:32
본관	09:37	15:52
학생회관	10:12	16:37
인문대 서쪽	10:32	17:07
학생생활관	10:52	17:27
인문대 동쪽	11:27	17:47
중앙도서관	12:47	18:07
의전원	13:12	18:27
공대4호관	13:32	18:47
교양동	13:52	

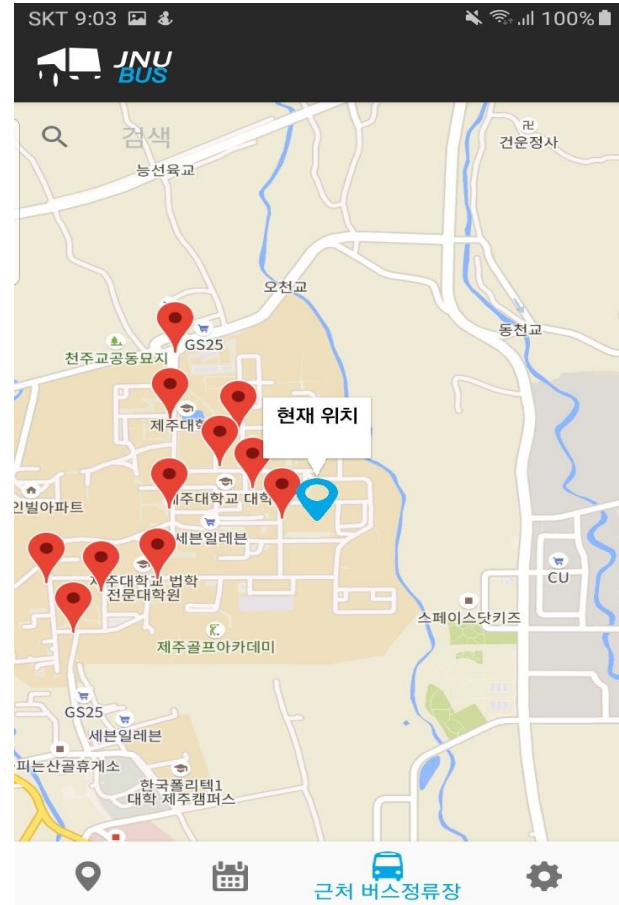
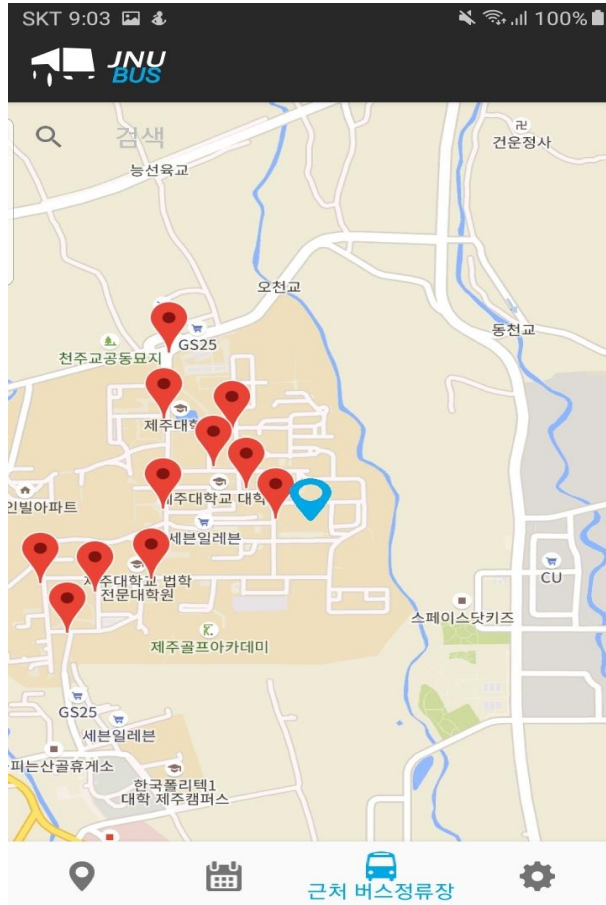
📍 시간표 🚌 ⚙️

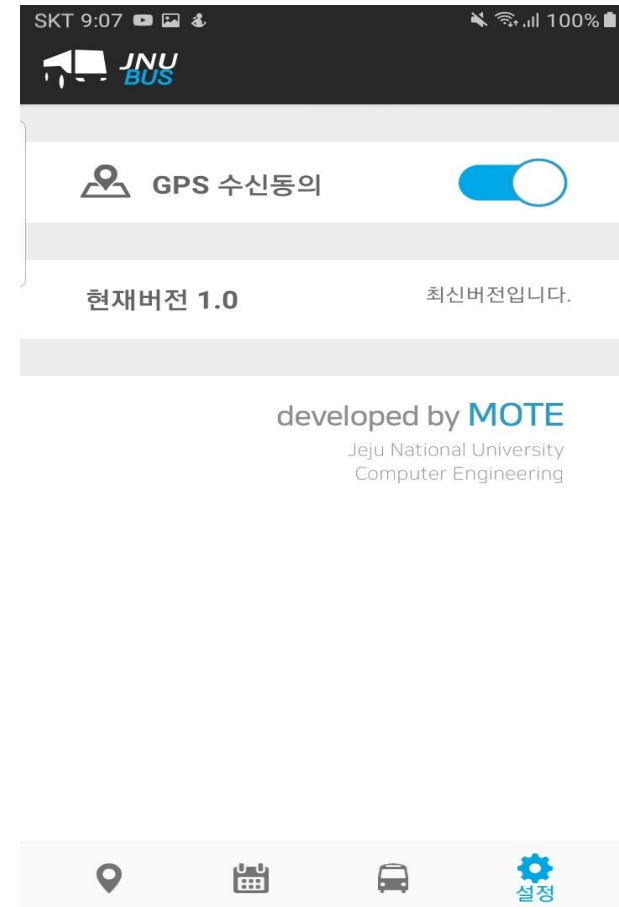
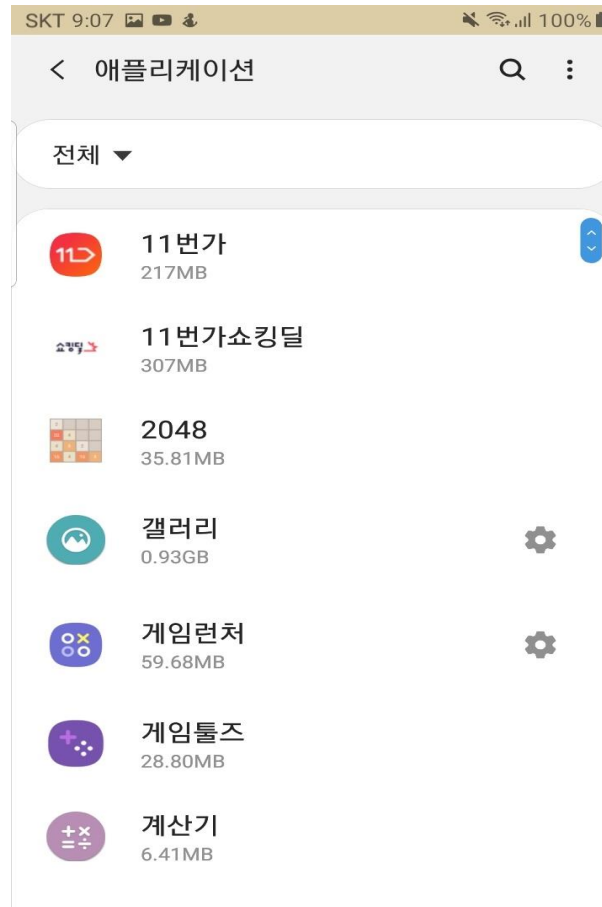
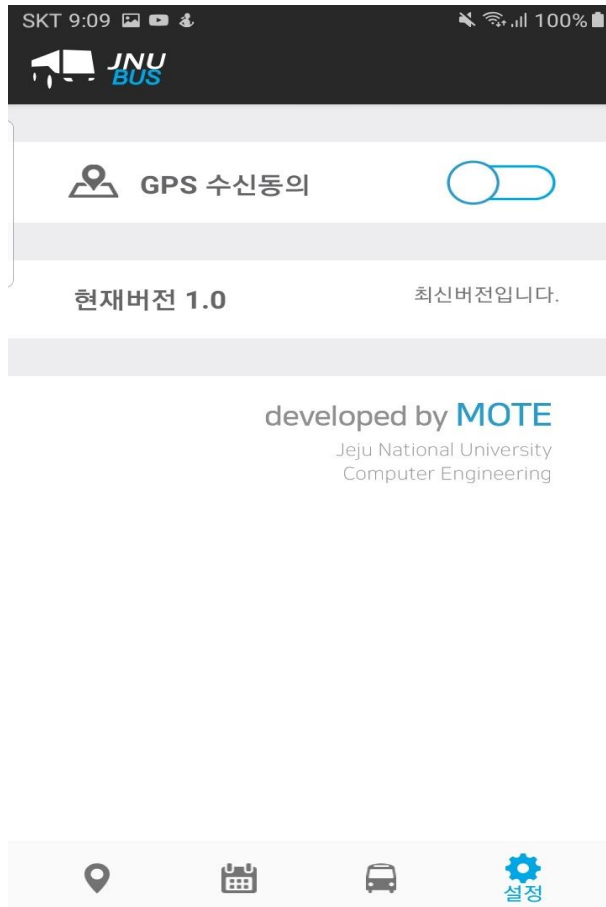
SKT 9:03 100%

JNU BUS

🔍	A노선	B노선
정문	08:17	14:47
제2도서관	08:37	15:17
해양대1호관	08:57	15:37
본관	09:47	15:57
학생회관	10:17	16:47
인문대 서쪽	10:37	17:17
학생생활관	10:57	17:37
인문대 동쪽	11:37	17:5
중앙도서관	12:57	18:17
의전원	13:17	18:37
공대4호관	13:37	18:57
교양동	13:57	

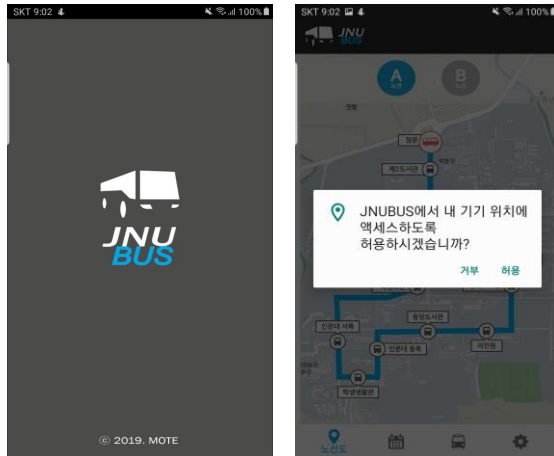
📍 시간표 🚌 ⚙️





현재구현된 부분

세부 기능별 사용자 인터페이스 화면 내용



1.로딩화면 JNU BUS 로고 표시 및 GPS 동의 여부 팝업 표시

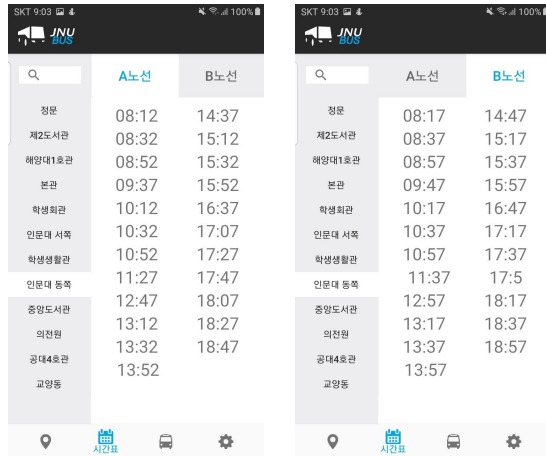
2-1.메인화면

A, B 노선에 따른 버스 정류장
위치와 현재 버스 위치를 표시



현재구현된 부분

세부 기능별 사용자 인터페이스 화면 내용



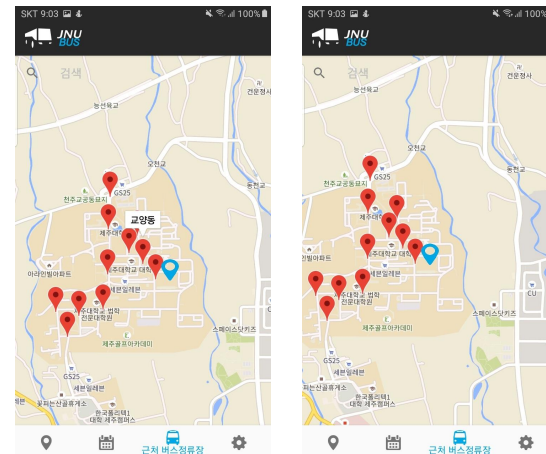
	A노선	B노선
정문	08:12	14:37
제2도서관	08:32	15:12
해양대1호관	08:52	15:32
본관	09:37	15:52
학생회관	10:12	16:37
인문대 서쪽	10:32	17:07
학생생활관	10:52	17:27
인문대 동쪽	11:27	17:47
중앙도서관	12:47	18:07
의전원	13:12	18:27
공대4호관	13:32	18:47
교양동	13:52	

2-2.전체시간표

A노선 / B노선에 따른 각 노선별 시간표 및 검색

2-3.근처 버스 정류장 표시

사용자의 GPS를 통해 현재 위치에서 가장 가까운 버스 정류장 표시

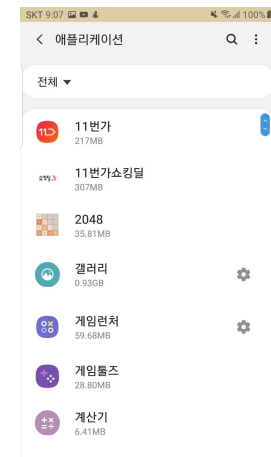
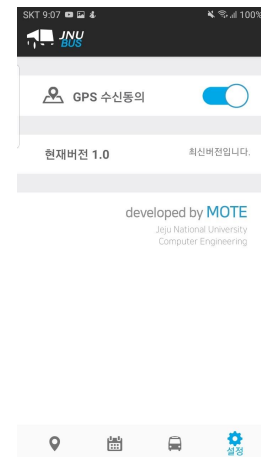
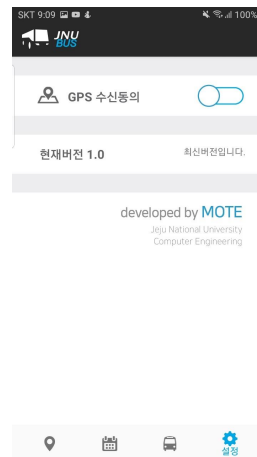


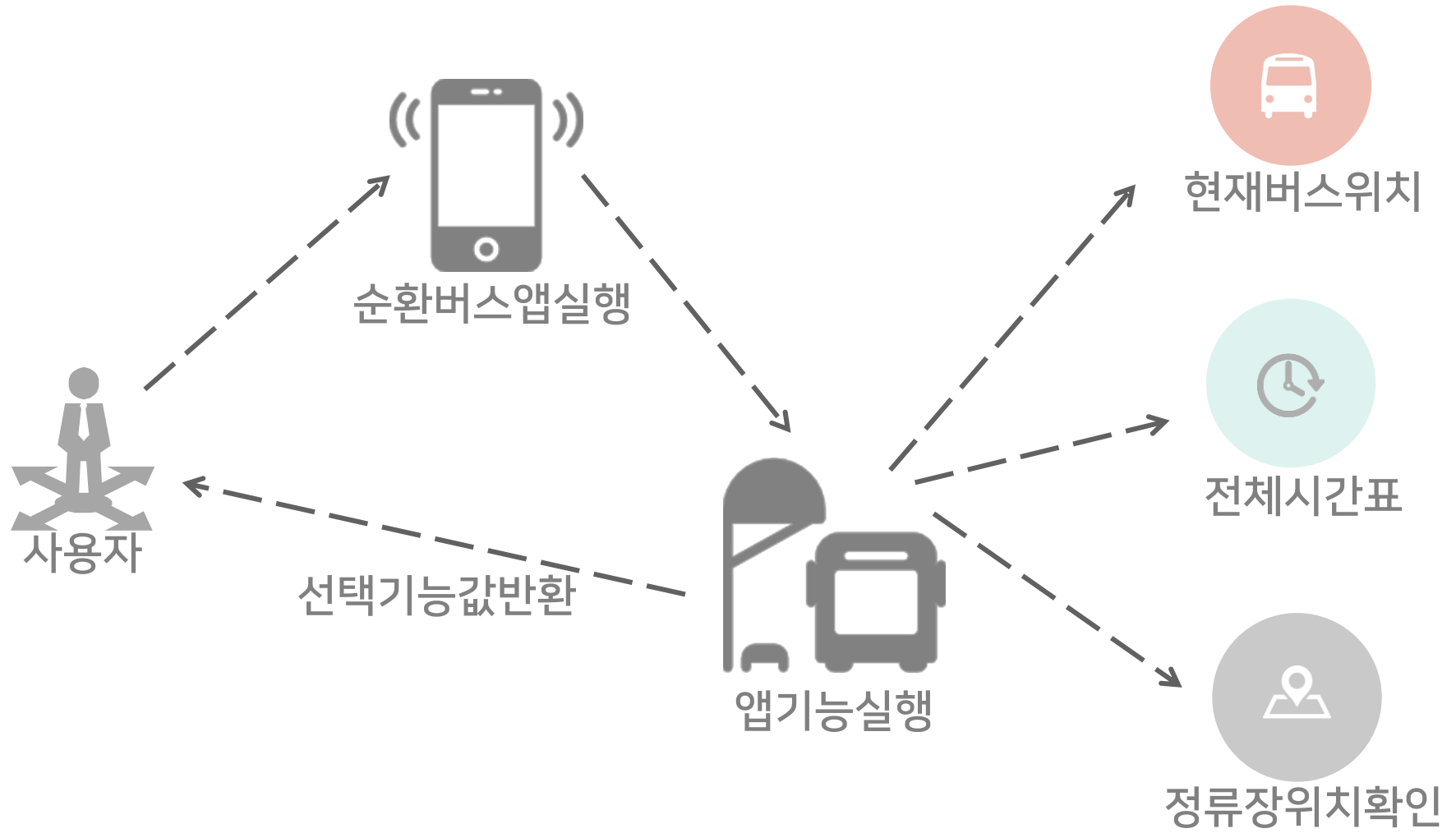
현재구현된 부분




세부 기능별 사용자 인터페이스 화면 내용




2-4.GPS 설정

GPS 동의 여부 설정 및
앱의 현재 버전과 앱 정보 표시

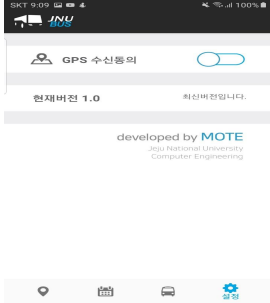
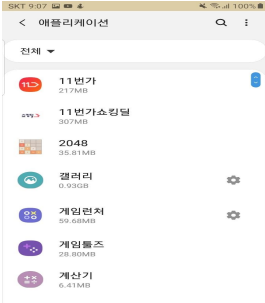
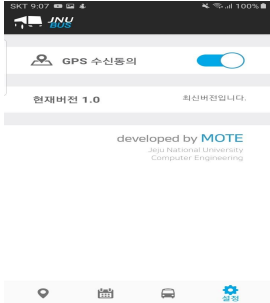




서비스구동화면	서비스구동설명
 <p>A black loading screen with the JNU BUS logo in the center. The logo consists of a white bus icon above the text 'JNU' in white and 'BUS' in blue. At the bottom, it says '© 2019. MOTE'.</p>	<p>1-1. 로딩 화면 앱을 시작하면 처음 나오는 화면</p>
 <p>A map interface with a white dialog box in the center. The dialog box contains the text: 'JNUBUS에서 내 기기 위치에 액세스하도록 허용하시겠습니까?' (Allow JNUBUS to access your device's location?). Below the text are two buttons: '거부' (Deny) and '허용' (Allow). The background shows a map with a blue route and various location markers.</p>	<p>1-2. 로딩 화면(GPS 동의) 앱을 시작하면 처음으로 나오는 화면 GPS설정을 허용하거나 거부 할수 있음</p>
 <p>A map interface showing a blue route. The map includes various location markers and labels in Korean. At the bottom, there is a navigation bar with icons for location, search, and settings.</p>	<p>2-1. 메인화면 앱을 실행했을 때의 메인화면 A노선의 현재 버스 위치를 확인 할수 있음</p>

서비스구동화면	서비스구동설명
	<p>2-1. 메인화면</p> <p>앱을 실행했을 때의 메인화면 B노선의 현재 버스 위치를 확인 할수 있음</p>
	<p>2-2. 전체시간표 화면</p> <p>A노선을 클릭하여 각정류장의 버스 도착 예정 시간을 확인 할 수 있음</p>
	<p>2-2. 전체시간표 화면</p> <p>B노선을 클릭하여 각정류장의 버스 도착 예정 시간을 확인 할 수 있음</p>

서비스구동화면	서비스구동설명
	<p>2-3. 사용자 근처 버스정류장 표시 화면 사용자의 현재 위치를 확인 하고 근처 위치를 찾는 화면</p>
	<p>2-3. 사용자 근처 버스정류장 표시 화면 사용자의 현재위치를 확인하여 위치가 설정된 화면</p>
	<p>2-3. 사용자 근처 버스정류장 표시 화면 사용자의 위치가 설정되어 근처 버스정류장을 나타낸 화면</p>

서비스구동화면	서비스구동설명
	<p>2-4. GPS 설정화면 GPS 수신 동의를 하지 않은 화면</p>
	<p>2-4. GPS 설정화면 GPS 수신동의를 선택할수 있는 화면</p>
	<p>2-4. GPS 설정화면 GPS의 수신동의를 한 화면</p>

- 1.교내 순환버스의 전체노선 및 시간표를 간편하게 확인할 수 있다.
- 2.각 노선 정문 출발 시간을 표기해 버스가 언제 출발했는지 알수 있다
- 3.정류장별로 도착예정 시간을 표기해 사용자가 시간을 확인하고
버스가 언제 도착하는지 확인할 수있다.
- 4.GPS시스템을 이용해 사용자와 가까운 정류장의 위치파악과
도착예정시간을 파악할 수 있다.
5. 순환버스를 탔을 때 사용자의 위치를 파악해 다음 도착할 정류장을
알려주어 사용자의 하차를 돕는다.

1. 교내 순환버스의 각 노선도와 정류장의 정보를 학생들에게 손쉽게 전달할 수 있어 순환버스의 이용에 도움을 준다.
 2. 전체시간표를 확인해 미리 도착정보를 확인할 수 있다.
 3. 나의 위치를 지도상에 표기해 가까운 버스정류장을 확인할 수 있다.
- > 언제든지 JNUBUS 앱을 통해 순환버스이용에 도움을 받을 수 있다.

1. 버스정류장 검색기능
2. 가까운 버스 정류장 알림.
3. 전체시간표 기능에 현재시간과 가까운 시간을 표기하는 기능
4. 각 정류장별 도착예정알림 서비스
5. 현재시간을 받아 애니메이션 시작 설정 기능

이번 프로젝트를 통해 아이디어를 고안해 생각을 정리하고 기획서와 설계서를 작성하는 것은 물론 실제로 개발을 진행하게 되면서 많은 어려움이 있었고, 프로젝트가 끝난 지금도 미흡한 부분이 많지만 많은 것을 배우게 된것 같습니다.

특히 혼자서가 아닌 조별로 과제를 진행하게 되면서, 다른사람에게 나의 의견을 전달하고 다른사람의 의견을 제대로 듣고 이해하는 등의 의사소통의 중요성과 각 역할을 분담해 개발하고 의견을 조율하는 이러한 과정은 굉장히 중요한 경험이었다고 생각합니다.

감사합니다.