

## A Design and Implementation of Bus Information Notification Application

A-Yeon Kang\*, Tae-Hyeon Lee\*, Na-Kyung Lee\*, Won-Joo Lee\*

\*Student, Dept. of Computer Science & Engineering, Inha Technical College, Incheon, Korea

\*Student, Dept. of Computer Science & Engineering, Inha Technical College, Incheon, Korea

\*Student, Dept. of Computer Science & Engineering, Inha Technical College, Incheon, Korea

\*Professor, Dept. of Computer Science & Engineering, Inha Technical College, Incheon, Korea

### [Abstract]

In this paper, we design and implement a bus information notification application based on the GPS sensor of a smartphone. This application provides the ability to check bus stops within a radius of 200m based on the current user's location using the smartphone's GPS sensor, Google Map, and open API. In addition, if you click the marker of the desired stop, you can see the name of the stop, and click the view arrival information button to check the detailed bus arrival information of the stop. In addition, it provides a function to check the location information of pharmacies, nonghyups, and post offices that sell public masks, the names of public mask stores, and mask inventory through the public mask store button. Each icon was used differently to make visual differences in order to easily indicate the difference between the times of public mask sales and bus stops. In addition, if you want to know the information of other bus stops and the route of the desired bus, not around the user's location, click the bus stop search button. Finally, after storing the destination stop or location, it implements a function that provides an alarm when it approaches the location.

▶ **Key words:** Public mask, Android Sensor, GPS Sensor, 오픈 API, Bus notification, Android

### [요 약]

본 논문에서는 스마트 폰의 GPS 센서 기반의 버스 정보 알림 애플리케이션을 설계하고 구현한다. 이 애플리케이션은 스마트 폰의 GPS 센서, Google Map, 오픈 API를 이용하여 현재 사용자의 위치를 기준으로 반경 200m 안의 버스 정류장을 확인할 수 있는 기능을 제공한다. 또한, 원하는 정류장의 마커를 클릭하면 해당 정류장 이름을 알 수 있으며 도착 정보 보기 버튼을 클릭하여 정류장의 상세 버스 도착 정보를 확인할 수 있는 기능을 제공한다. 그리고 공적 마스크 판매장 버튼을 통해 공적 마스크를 판매하는 약국, 농협, 우체국의 위치정보와 공적 마스크 판매장의 이름, 마스크 재고를 확인할 수 있는 기능을 제공한다. 공적 마스크 판매장과 버스정류장의 시각별 차이를 쉽게 표기하기 위해 각 아이콘을 다르게 사용하여 시각적인 차이를 두었다. 또한, 사용자의 위치 주변이 아닌, 다른 버스정류장의 정보와 원하는 버스의 노선을 알고 싶다면 버스 정류장 검색 버튼을 클릭한다. 마지막으로, 목적지 정류장 혹은 위치를 저장한 후에 해당 위치에 근접하게 되면 알람을 제공하는 기능을 구현한다.

▶ **주제어:** 공적 마스크, 안드로이드 센서, GPS 센서, 오픈 API, 버스 정보, 안드로이드

- 
- First Author: A-Yeon Kang, Corresponding Author: Won-Joo Lee
  - \*A-Yeon Kang (rkddkdus8595@naver.com), Dept. of Computer Science & Engineering, Inha Technical College
  - \*Tae-Hyeon Lee (th2610@naver.com), Dept. of Computer Science & Engineering, Inha Technical College
  - \*Na-Kyung Lee (sjmcl912@naver.com), Dept. of Computer Science & Engineering, Inha Technical College
  - \*Won-Joo Lee (wonjoo2@inhac.ac.kr), Dept. of Computer Science & Engineering, Inha Technical College
  - Received: 2021. 01. 25, Revised: 2021. 02. 25, Accepted: 2021. 02. 25.

### I. Introduction

수도권 대중교통 이용객의 교통카드 이용 건수를 살펴 보면은 하루 평균 약 719만 명이 대중교통을 이용하고 있다. 교통카드 이용 건수는 그림 1과 같다[1].

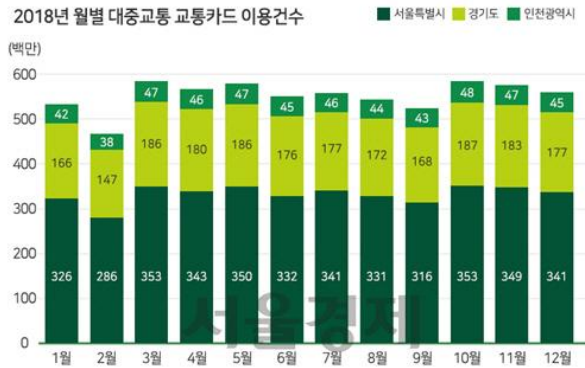


Fig. 1. Number of public transportation card usage per month

대중교통은 많은 사용자로 인해 실내의 이산화탄소 농도가 높아지고, 이로 인해 졸음이 쏟아지는 환경이 될 수 있다. 서울 지하철 9호선 전동차량의 실내 공기 질을 측정한 결과 요일에 상관없이 혼잡시간 대에 일반열차와 급행열차 모두 이산화탄소 농도가 2,868~4,033ppm으로 권고 기준 2,000ppm을 초과한다[2]. 또한 지하철의 진동수 2Hz는 사람들에게 졸음이 쏟아지는 환경을 제공한다[3]. 이러한 환경으로 인해 대중교통 사용자들은 졸음으로 인해 목적지에서 하차하지 못하는 일이 많이 발생한다. 따라서 대중교통을 이용하는 사용자들에게 이러한 불편을 줄일 수 있는 스마트 폰 앱이 필요하다.

현재 안드로이드 시장 점유율이 가장 높다[4]. 안드로이드 이용자 수는 안드로이드 폰의 시장 점유율이 증가할수록 지속적으로 증가하고 있다[5]. 안드로이드 이용자 수가 증가하는 만큼 안드로이드 앱(App)의 사용 빈도수도 함께 증가한다. 안드로이드는 기존의 모바일 플랫폼들과는 다르게 오픈 소스를 이용해서 누구나 쉽게 안드로이드 기반의 앱을 만들 수 있는 개방형 플랫폼이다[6-7]. 안드로이드 기반의 앱은 테스트, 유지보수 등과 같은 개발 소요 시간을 줄일 수 있다는 장점이 있다. 개발자들이 이용자 수를 늘리기 위해 어떤 종류의 앱(App)이 인기가 많은지를 분석하여, 개발하면 기존 시장에서 살아남을 수 있을 뿐만 아니라 앱 기능 자체의 발전이 계속될 것이다[8]. 따라서 본 논문에서는 조도 센서, 오픈 API, Google Map을 활용하여 대중교통 사용자들이 목적지를 설정할 수 있는 안드로이드 기반의 버스 정보 알림 앱을 설계하고 구현한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존 앱의 장·단점을 분석하고 3장과 4장에서는 각각 버스 정보 알림 앱을 설계하고 구현한다. 그리고 5장에서 결론을 맺는다.

### II. Preliminaries

#### 1. 버스 정보 시스템

버스 정보 시스템은 기존 버스 교통에 첨단 IT 기술을 접목하여 버스 위치를 파악하고 버스 이용객 혹은 관리자에게 유용한 정보를 제공하는 시스템이다[9]. 대부분의 버스 정류장에는 버스 정보 시스템이 존재하기 때문에 승객이 정류장에 도착했을 때 그림 2와 같은 버스 정보 시스템을 통하여 버스의 도착 여부를 알 수 있다[10].

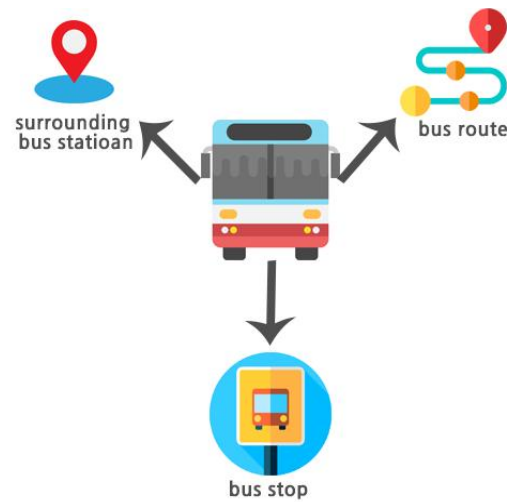
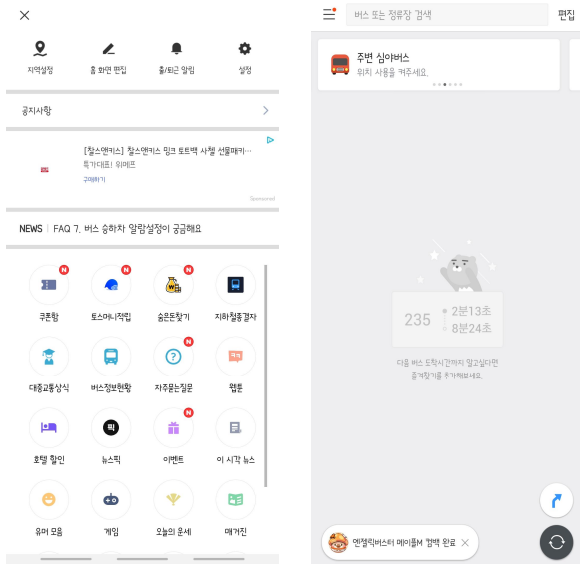


Fig. 2. Bus Provision Information.

승객들은 지도 앱을 통해 버스 정보를 확인하거나 버스 전용 앱을 설치하여 사용하기도 한다. 지도 앱의 경우 여러 가지 정보를 제공하고 있지만, 버스 전용 앱의 정보는 제한적이다. 버스 전용 앱에도 상황을 반영한 유용한 정보가 추가되어야 한다.

#### 2. 기존 버스 정보 제공 시스템

대표적으로 많이 사용하는 버스 정보 앱은 그림 3과 같이 카카오 버스와 스마트버스가 있다. 그림 3 (a)의 스마트버스 앱은 기본적인 버스 정보 외에도 출/퇴근 알림, 승하차 알림, 운세 보기 등 다양한 메뉴를 제공한다. 다양한 정보를 즐기는 사용자들이 선호하게 되는 앱이다. 그림 3 (b)의 카카오 버스 앱은 주변 심야버스, 승하차 알림 등의 다양한 메뉴를 제공한다.



(a) Smart bus (b) Kakao bus

Fig. 3. Existing bus app

스마트버스를 포함한 대부분의 스마트 폰 기반의 버스 정보 앱들은 너무 많은 정보를 포함하고 있어서 사용자들이 단순한 정보를 얻기가 어렵다는 것이다. 과다한 정보 제공은 사용자의 정보 선택을 어렵게 한다[11]. 또한 코로나(COVID-19)가 유행하는 현재 상황을 반영한 공적 마스크 판매점 위치와 종류, 마스크 재고 등의 정보를 제공하는 기능이 필요하다. 공적 마스크 판매점의 정보는 오픈 API로 제공되며 이를 활용하여 애플리케이션에서 제공할 수 있도록 구현할 수 있다. 그림 4는 공적 마스크 판매 정보에 대한 서비스 구현 흐름이다.



Fig. 4. Public Mask Sales Status Service Implementation Flow [12]

코로나가 유행하는 현 상황을 앱에 반영하여 버스 정보 앱의 활용도를 높일 수 있다. 버스 정보 앱에서 제공하는 기능은 사용자 위치 주변의 버스 정류장, 공적 마스크 판매점의 위치와 버스 정류장 검색, 하차할 정류장 알람과 같은 다양한 오픈 API와 센서, 스마트 폰의 GPS를 이용하여 정보를 제공한다. 본 논문에서는 기존의 버스 정보 시스템을 개선하여 코로나 정보 등을 추가한 버스 정보 알림 앱을 설계하고 구현한다.

### III. Design of Bus Information Notification Application

본 논문에서는 GPS를 활용한 사용자 현재 위치 주변의 버스 정류장과 도착 정보, 조도 센서를 이용하여 앱의 화면 밝기를 조절해주는 등의 다양한 정보를 사용자에게 제공하는 애플리케이션을 설계한다.

#### 1. Android Sensor

대부분 안드로이드 기기에는 다양한 조건을 측정하는 센서가 내장되어있다. 센서는 기기의 움직임 또는 위치를 모니터링하거나 기기 근처 주변 환경의 변화를 모니터링 하려는 경우에 유용하다. 스마트 폰에는 표 1의 센서들이 존재하며 해당 센서들을 앱에서 이용한다.

Table 1. Sensor Specification [13]

상수	타입	설명	단위
TYPE_ALL		모든 센서	
TYPE_ACCELEROMETER	HW	가속도 센서	m/sec <sup>2</sup>
TYPE_GRAVITY	SW or HW	중력 센서	m/sec <sup>2</sup>
TYPE_GYROSCOPE	HW	자이로스코프	
TYPE_ORIENTATION	SW	방향 센서	도 (degree)
TYPE_LIGHT	HW	조도 센서	Lux
TYPE_LINEAR_ACCELERATION	SW or HW	선형 가속도 센서	
TYPE_MAGNETIC_FIELD	HW	지자기 센서	마이크로테슬라 (uT)
TYPE_PRESSURE	HW	압력 센서	
TYPE_PROXIMITY	HW	근접 센서	미터
TYPE_ROTATION_VECTOR	SW or HW	회전 벡터 센서	
TYPE_AMBIENT_TEMPERATURE	HW	온도 센서	

이러한 스마트 폰 센서들을 사용하기 위해 권한이 필요한 경우 AndroidManifest.xml 파일에 사용 권한을 추가한다.

#### 2. GPS Sensor

대부분의 스마트 폰은 GPS 센서를 내장하고 있다. GPS 센서를 통해 사용자의 위치를 파악하고, 위치를 이용하여 길 찾기, 아이 위치 추적 등과 같은 위치 기반 서비스 기능을 구현할 수 있다. 구글 지도에서 스마트 폰의 GPS 센서를 이용하여 파란색 원으로 사용자의 위치를 표시한다. 사용자 위치의 위도와 경도 값으로 사용자 주변의 버스 정류장 정보를 이용할 수 있다. 사용자의 위도와 경도 값은 GPS 센서를 이용하여 얻을 수 있다. GPS 센서를 사용하기 위해서는 사용 권한을 얻어야 하며 GPS 센서 사용 권한이 없는 경우 AndroidManifest.xml에서 권한을 얻을 수 있도록 사용 권한을 추가한다. GPS 권한을 받을 수 있는 조건은 아래 표 2를 참고하여 권한을 추가한다.

Table 2. Permission

Provider	Permission
GPS	android.permission.ACCESS_FINE_LOCATION
Network	android.permission.ACCESS_COARSE_LOCATION android.permission.ACCESS_FINE_LOCATION
Passive	android.permission.ACCESS_FINE_LOCATION

### 3. 오픈 API

오픈 API와 Google Map을 이용하면 그림 5의 구조를 가진 안드로이드 앱을 구현할 수 있다.



Fig. 5. App Architecture

앱을 구현할 때 제공되는 오픈 API를 이용하면 개발시간과 비용을 절약하여 빠르게 앱을 구현할 수 있다. 오픈 API를 이용할 수 있는 곳은 다양하며 본 논문에서는 Google Map API, 공공데이터 포털, 공적 마스크 API를 이용하여 앱을 설계하고 구현한다. 공공데이터 포털에서는 서비스를 신청한 후 활용 신청에 대한 허가가 나면 API를 사용할 수 있다. 버스 관련 API 중 위도, 경도를 통해 찾을 수 있는 버스 정류장 위치, 버스의 노선, 정류장의 정보와 같은 API를 사용한다. 공적 마스크 판매점을 나타내기 위한 API는 건강보험심사평가원에서 제공하는 마스크 정보를 바탕으로 공공데이터 활용 지원센터에서 제공한다. 공적 마스크 판매점의 위치는 사용자의 위치 반경 5km 이내 까지 확인할 수 있으며, 해당 API가 제공하는 정보는 재고 상태, 입고 시간, 주소, 위도, 경도, 이름, 판매처 유형 등이다. 공적 마스크 재고 상태를 통해 마스크의 수량을 파악하며 판매처 유형으로 공적 마스크 판매점의 정보를 파악한다. GPS 센서를 이용하여 얻은 사용자의 현재 위치 위도, 경도를 통해 사용자의 위치 주변의 판매점 정보를 표시할 수 있다.

## IV. Implementation of Bus Information Notification Application

본 논문에서는 GPS 센서와 조도 센서, 오픈 API를 이용한 안드로이드 기반의 버스 정보 알림 앱을 구현한다. 이 앱은 자바 언어 기반으로 Android Studio를 이용하여 구

현한다. 버스 정보 알림 앱의 액티비티와 메소드는 표 3과 같다.

Table 3. Function

Activity	Method	
bus_view	onCreate()	
bussub	onCreate()	DownloadWebpageTask()
inform	getRouteName()	setRouteName()
	getLowPlate()	setLowPlate()
	getPredictTime()	setPredictTime()
	getLocationNo()	setLocationNo()
	getRemainSeat()	setRemainSeat()
MainActivity	getFlag()	setFlag()
	onCreate()	onItemClick()
	SSearch()	BSearch()
	getBusName()	getBusArrive()
MapsActivity	DownloadWebpageTask()	
	onCreate()	getLight()
	onResume()	onPause()
	onSensorChanged()	onAccuracyChanged()
	onMapReady()	onMyLocationButtonClicked()
	onMarkerClick()	getCorona()
	getBusArrive()	getBusName()
DownloadWebpageTask()		
Recycler Adapter	RecyclerViewAdapter()	ViewHolder()
	onBindViewHolder()	getItemCount()
	ViewHolder onCreateViewHolder()	
Splash Activity	onCreate()	

표 3에서 bus\_view는 리사이클러 뷰를 통해 버스 정보를 제공하는 액티비티이다. bussub는 버스 선택 시 정류장 정보 화면을 출력하는 액티비티이다. inform은 리사이클러 뷰를 이용하기 위한 set, get 메소드 모음이다. MainActivity는 API를 연결하고 버스 정보에 필요한 각종 변수 정보를 저장하는 액티비티이다. MapsActivity는 지도에 표현할 정보와 API를 연결해서 찾고, 지도 관련 정보를 제어하는 액티비티이다. RecyclerViewAdapter는 리사이클러 뷰를 이용하기 위한 어댑터이다. SplashActivity는 스플래시를 실행하기 위한 액티비티이다.

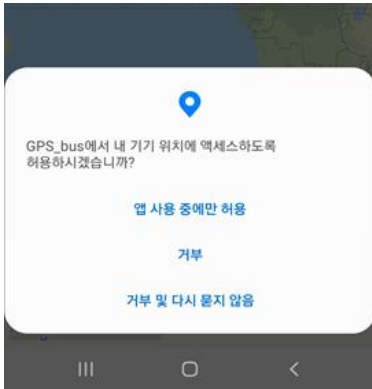
```
public void onMapReady(GoogleMap googleMap)
```

```
if (ContextCompat.checkSelfPermission(
```

(a) necessary function

```
ActivityCompat.requestPermissions(
```

(b) necessary function2



(c) location permission request image

Fig. 6. Request Permission

MapsActivity.java에 포함된 지도를 출력하기 위한 선제 조건은 그림 6 (a) onMapReady 메소드에 권한을 체크해 주는 코드를 구현하고, 권한이 없다면 (b)를 통해 권한을 요청하는 방식을 사용한다. 해당 코드를 통해 권한을 요청하면 (c)의 동작 이미지를 얻는다.

그림 7의 사용자 위치를 기반으로 반경 200m 내의 버스 정류장들의 마커를 표시한다. 버스 정류장 마커를 클릭하면 marker.getSnippet() 메서드를 사용하여 정류장 번호를 저장한다. 저장된 정보를 오픈 API를 이용하여 버스 정류장의 정보를 얻는다.



Fig. 7. User Surrounding Bus Stop

정류장 마커를 클릭한 후 도착 정보보기 버튼을 클릭하면 해당 정류장의 그림 8의 버스 도착 정보를 확인할 수 있다.



Fig. 8. Bus Arrival Information

그림 8의 버스 도착 정보는 RecyclerView를 이용하여 그림 9의 (a) RecyclerView Frame을 생성한 후, RecyclerViewAdapter.java 파일에 있는 그림 9의 (b) RecyclerView Adapter의 코드를 포함하여 정보를 출력한다.



(a) RecyclerView Frame

```

@Override
public void onBindViewHolder(@NonNull ItemViewHolder holder,
                             int position) {
    // Item을 하나, 하나 보여주는(bind 되는) 함수입니다.
    holder.busName1.setText(list.get(position).routeName);
    holder.low1.setText(list.get(position).lowPlate1);
    holder.low3.setText(list.get(position).lowPlate2);
    holder.predictTime1.setText(list.get(position).predictTime1);
    holder.predictTime3.setText(list.get(position).predictTime2);
    holder.location1.setText(list.get(position).locationNo1);
    holder.location3.setText(list.get(position).locationNo2);
    holder.remainSeat1.setText(list.get(position).remainSeat1);
    holder.remainSeat3.setText(list.get(position).remainSeat2);
    holder.busPass1.setText(list.get(position).flag);
}
    
```

(b) RecyclerView Adapter

Fig. 9. RecyclerView

MapsActivity.java 파일에 있는 그림 10의 getCorona() 메서드로 오픈 API에 연결하여 반경 3km 내의 정보를 얻는다.

```
public void getCorona(){
    String url3="";
    serviceUrl="https://8oi9s0nnth.apigw.ntruss.com/c
    url3=serviceUrl+"lat="+d1+"&lng="+d2+"&m=3000";
```

Fig. 10. Function

앱에서 공적 마스크 판매장 버튼을 클릭하면 그림 11의 (b) public mask store image의 마커를 통해 공적 마스크 판매장의 위치를 알 수 있다. 또한 해당 판매장의 이름과 종류, 남은 개수를 알 수 있다. 해당 정보는 사용자가 보기 어려운 숫자 형식으로 되어있기 때문에 편의성을 위해 그림 11의 (a) conversion code를 통해 한글로 변환한다.

```
if(arrType.get(idx).equals("01")){
    type123="약국";
}else if(arrType.get(idx).equals("02")){
    type123="우체국";
}else if(arrType.get(idx).equals("03")){
    type123="농협";
}
```

(a) Conversion Code



(b) Public Mask Store Image

Fig. 11. Public Mask Store



(a) Station

(b) Bus Information

Fig. 12. Station Search Image

사용자는 주변의 버스 정류장이 아닌 원하는 버스와 정류장도 검색을 통해 그림 12의 (a) station을 확인할 수 있으며 클릭 시 (b) bus information를 얻을 수 있다.



Fig. 13. Bus Search Image

그림 13에서 사용자는 버스 입력란에 원하는 버스의 번호를 입력하여 해당 버스의 노선을 확인할 수 있다. 그림 13의 왼쪽 이미지에서 버스를 선택하면 해당 아이템의 포지션이 저장되며, 저장된 포지션으로 다시 오픈 API를 이용하여 정보를 얻는다.

사용자는 버스 정류장을 저장하여 하차하기 위한 알람을 설정할 수 있다. 알람 기능을 이용하는 방법은 지도에서 원하는 버스 정류장 마커를 클릭하면 자동으로 저장되며, 그림 14의 코드를 이용하여 일정 수치만큼 저장된 정류장에 가까워지면 그림 15의 메시지와 함께 진동으로 알람이 울린다. 확인을 누르면 알림창 표시와 함께 알람을 끌 수 있다.

```

if ((d1+0.0025 > markerClick1 || d1-0.0025 > markerClick1) &&
(d2+0.0025 > markerClick2 || d2-0.0025 > markerClick2) && marked == true){
    new AlertDialog.Builder( context: MapsActivity.this)
        .setMessage("다와갑니다! 일어나세요")
}

```

Fig. 14. Distance Proximity Code



Fig. 15. Arrival Notification

## V. Conclusions

본 논문에서는 Android Studio를 이용하여 위치 기반의 버스 정보 알림 앱을 설계하고 구현하였다. 각 메뉴는 도착 정보보기, 공적 마스크 판매장, 버스 정보 검색, 정류장 정보검색으로 구성하였다. 이 앱은 사용자가 앱을 실행하면 스플래시를 이용하여 어떤 앱인지 이미지로 표시해주고, 위치를 실시간으로 받으며 Google Map에 현재 위치를 표시한다. 또한 사용자의 현재 위치에서 반경 200m까지 거리의 주변 정류장 조회를 이용할 수 있다. 주변 정류장 조회뿐만 아니라 사용자의 위치 근처 3km 반경의 공적 마스크 판매장 위치를 확인할 수 있다. 그리고 사용자의 현재 위치 주변만 조회하는 것이 아닌 검색을 통해 다른 정류장의 버스 정보, 버스를 검색하여 버스 노선을 볼 수 있으며 사용자가 내릴 장소를 잊지 않도록 알람을 받을 수 있다. 이 앱을 통해 사용자는 좀 더 단순한 UI 환경에서 편리하게 버스 정보를 이용할 수 있다.

## REFERENCES

- [1] <https://www.sedaily.com/NewsView/1VHPZLZO5L>
- [2] [http://news.khan.co.kr/kh\\_news/khan\\_art\\_view.html?art\\_id=201604261734001](http://news.khan.co.kr/kh_news/khan_art_view.html?art_id=201604261734001)
- [3] [http://health.chosun.com/site/data/html\\_dir/2019/05/09/2019050901801.html](http://health.chosun.com/site/data/html_dir/2019/05/09/2019050901801.html)
- [4] <https://gs.statcounter.com/os-market-share>
- [5] <http://www.itworld.co.kr/news/104206>
- [6] Ko, Seok-Hoon, "A Trend of Android Platform", korean contents journal, Vol.8 Issue 2, Pages.45-49, June 2010
- [7] Jang, Yun-Jeong, Kim, Cheol-Woo "스마트폰 시장의 진화와 안

- 드로이드의 영향", Korean Institute of Information Scientists and Engineers, Vol 28 Issue 5, Pages.48-56, May 2010
- [8] Pil Sang Ju, Young Hwan Pan, "Study on the development and changes of smart-phone application market", Korean Society of Basic Design & Art, Vol.17 no.2, Pages.495-507, 2016
- [9] Jo, Jeong-Hyeong, O, Yeong-Tae, "The Effectiveness Analysis for Bus Information Systems : The case of Bucheon City", Intelligent Transportation Systems, Pages.288-293, Nov 2004
- [10] Seungcheon Kim, "Bus Information System based on Smart-Phone Apps using GPS Information", The Institute of Internet, Broadcasting and Communication, Vol.11 Issue 3, Pages.169-174, June 2011
- [11] Park, Jae-Heung, Kang, Sun-Hee, Seo, Yeong-Geon, "Advanced Bus Information System Using Smart Phone GPS", Korean Society of Computer Information, Vol.19 Issue 12, Pages.247-255, Dec 2014
- [12] <http://www.hitnews.co.kr/news/articleView.html?idxno=15545>
- [13] <https://onepinetwopine.tistory.com/110>

## Authors



A-Yeon Kang received the A.S. degree in 2021 from the Department of Computer Science Inha Technical College, Incheon Korea. She is currently a student in Inha Technical College. She will have received

the B.S. degree from the Department of Computer Science and Engineering Inha Technical College, Incheon Korea in 2021. Her research interests include Mobile Computing, Data Analysis and Machine Learning.



Tae-Hyeon Lee received A.S. degree in 2021 from the Department of Computer Science Inha Technical College, Incheon Korea. He is currently a student in Inha Technical College. He will have received the B.S.

degree from the Department of Computer Science and Engineering Inha Technical College, Incheon Korea in 2021. His research interests include Mobile Computing, Data Analysis and Machine Learning.



Na-Kyung Lee received A.S. degree in 2021 from the Department of Computer Science Inha Technical College, Incheon Korea. She is currently a student in Inha Technical College. She will have received the B.S.

degree from the Department of Computer Science and Engineering Inha Technical College, Incheon Korea in 2021. Her research interests include Mobile Computing, Data Analysis and Machine Learning.



Won-Joo Lee received the B.S., M.S. and Ph.D. degrees in Computer Science and Engineering from Hanyang University, Korea, in 1989, 1991 and 2004, respectively. Dr. Lee joined the faculty of the Department of

Computer Science at Inha Technical College, Incheon, Korea, in 2008, where he has served as the Director of the Department of Computer Science. He is currently a Professor in the Department of Computer Science, Inha Technical College. He has also served as the Vice-president of The Korean Society of Computer Information. He is interested in parallel computing, internet and mobile computing, and cloud computing, data science, artificial intelligence.