

# 지도 API 자체 제작을 통한 사용자 경험 개선: 건국대학교 캠퍼스 맵 애플리케이션

Improve user experience through building  
geo-APIs: Campus Map application for  
Konkuk University

김현재(Hyeonjae Kim), 박정환(Jeonghwan Park), 배정재(Seongjae Bae)

## 요약

현재 많은 지도 서비스가 발달되어 있지만, 단체의 고유 시설이나 샛길 등의 정보가 반영되지 않는다. 이에 더해 전국을 대상으로 하는 서비스에서는 시설의 변화를 적절히 반영하는 것이 어렵다. 이런 불편함을 해소하고자 UX 방법론을 도입하여 UI 디자인, 지리데이터 고도화, 라우팅과 상태관리 등을 진행한다. 최종적으로 이를 적용하여 사용자 경험을 최적화한 캠퍼스 맵 애플리케이션을 제안한다.

**키워드:** 지도, 오픈스트리트맵, 사용자 경험, 모바일 애플리케이션

## Abstract

Although various map services are currently available, they do not contain information about organizations' original facilities or trails. Also, it is difficult to properly apply changes in facilities in a nationwide service. To resolve these problems, we adopted UX methodologies for UI design, geodata enhancement, routing, and status management. Finally, we propose a campus map application that optimizes user experience by applying this solution.

**Keyword:** Map, OpenStreetMap, User Experience, Mobile application

## 1. 서론

건국대학교 캠퍼스는 학부생, 대학원생, 교직원 등 약 20,000 여명의 사람들이 이용하며, 매년 새로운 신입생과 임직원들이 캠퍼스를 사용한다. 비교적 넓은 면적과 다양한 건물들로 이루어져 있는 건국대학교 캠퍼스는 이용자에게 넓은 교류의 기회와 다양성을 제공하지만, 이는 또 다른 불편함을 야기하고 있다.

비단 기존 구성원만의 문제가 아니라 매년 새로운 신입생과 임직원이 들어올 때마다 넓은 캠퍼스로 인해 실질적 캠퍼스 활용에 어려움을 겪고 있다. 특정 서비스를 제공하는 시설을 찾기가 어려울 뿐더러, 학교 홈페이지에서 제공되는 캠퍼스맵은 몇 년간 유지보수가 되고 있지 않아 실질적 도움을 받지 못 하고 있다. 기존 캠퍼스맵은 단순한 평면도와 정적인 정보만을 제공하며, 'K-CUBE, 편의점, 카페' 등 새로운 편의시설 데이터 반영이 어려워 사용자들의 요구에 부합하지 않는다.

상용 지도 어플리케이션은 건물과 경로 정보를 제공하고 있지만 교내 캠퍼스 내부 데이터 반영이 어려워 실제와 다른 비효율적인 경로와 정보를 안내하고 있다. 캠퍼스 내의 길 안내 기능 정확도와 효율성 측면의 개선이 필요한 현황이다.

본 연구에서는 지도 API 자체 제작을 통해 사용자 경험을 개선한 건국대학교 맵을 위한 캠퍼스 맵 애플리케이션을 개발해 위와 같은 문제점을 해결하고자 한다. 사용자의 요구에 맞는 실시간 정보 제공과 직관적인 인터페이스(UI), 최적화된 길찾기 기능 등 사용자 경험을 개선하기 위한 연구에 대해 설명할 것이다.

## 2. 배경 지식

### 2.1 크로스플랫폼

크로스 플랫폼은 소프트웨어나 어플리케이션이 Windows, macOS, IOS, Android 등 여러 운영체제나 플랫폼에서 작동할 수 있는 능력을 말한다. 이를 통해 사용자는 다른 기기에서도 동일한 소프트웨어를 원활하게 이용할 수 있고, 호환성과 편의성을 가진다.

#### 2.1.1 Flutter

Flutter 는 Google 에서 개발한 오픈 소스 UI 소프트웨어 개발 킷이다. Flutter 는 모바일 앱, 웹 앱 및 데스크톱 앱을 빌드하기 위한 크로스 플랫폼 프레임워크로, 하나의 코드 베이스에서 IOS, Android, 웹 및 기타 플랫폼용 애플리케이션을 개발할 수 있다.

Flutter는 Dart라는 프로그래밍 언어로 작성되며, 기본적으로 네이티브 코드가 아닌 UI를 렌더링하기 위해 Skia 그래픽 엔진을 사용한다. 이는 앱의 성능을 향상시켜 네이티브 앱과 유사한 사용자 인터페이스를 제공한다.

### 2.1.2 Provider

Provider는 Flutter에서 애플리케이션의 다른 부분 간에 상태(state)를 관리하고 공유할 수 있는 상태 관리 라이브러리이다. Provider는 Flutter에서 제공하는 InheritedWidget 개념을 따르며, 상태의 관리와 상태가 변경될 때 위젯을 업데이트하는 과정을 간소화한다.

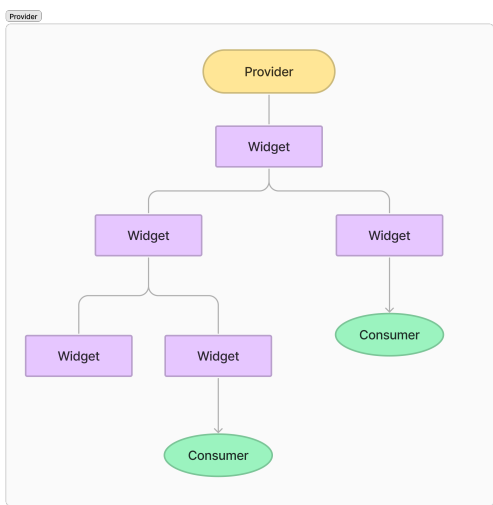


그림 1. Provider 개념 설명

Provider 패키지는 프로바이더(Provider)라는 개념을 핵심으로 하는데, 프로바이더는 상태를 보유하고 의존하는 위젯이 해당 상태에 접근하고 사용할 수 있도록 한다. 즉, 프로바이더는 상태나 데이터를 저장하고 제공하는 컨테이너로 생각할 수 있다. 본 패키지를 사용함으로써 비즈니스 로직과 상태 관리를 UI 구성 요소와 분리할 수 있어 모듈화되고 유지 관리가 용이한 코드베이스를 구축할 수 있다. Provider는 단방향 데이터 플로우 개념을 장려하며, 확장 가능하고 성능이 우수한 Flutter 애플리케이션 개발에 도움을 준다.

## 2.2 공간 정보화

공간정보는 건물, 산, 도로 등 지리적인 정보를 수집하고, 관리, 분석하는 등의 과정을 통해 데이터화한 정보자원이다. 지리 정보 시스템(GIS)을 기반으로 이루어지며, 다양한 지리적 특성과 위치정보를 중심으로 데이터를 조작한다.

### 2.2.1 오픈스트리트맵(OpenStreetMap)

오픈스트리트맵은 누구나 자유롭게 지도를 만들고, 제약없이 이용하도록 하기 위해 2004년에 시작된 오픈소스 지리공간 데이터베이스이다. 도로, 건물, 식당 뿐 아니라 극장의 수용인원 등 세세한 정보를 올리고 공유할 수 있을 뿐더러, 사용자가 새로운 태그를 생성할 수도 있다.

개방형 자료로서 누구나 수정 요청을 제출하거나 기여할 수 있으며, 기본측량성과를 활용하지 않기 때문에 국내법령(공간정보관리법)상 제약에서 상대적으로 자유롭다. ODbL 오픈 라이선스를 통해 개인이 독립적인 데이터베이스를 자유롭게 구축하는 등 다양한 연구와 서비스에 활용되기도 한다.

### 2.2.2 지오코딩(Geo-Coding)

지오코딩은 주소나 건물의 이름 등 고유명칭을 이용해 좌표값을 얻어내는 과정을 의미한다. 흔히 말하는 지도 검색을 위해 주로 사용되며 자연어 처리, 데이터베이스 쿼리 등을 활용해 지리정보의 고유 값과 좌표정보를 서로 매칭하게 된다.

## 2.3 도커

도커(Docker)는 Linux 컨테이너를 만들고 사용할 수 있도록 하는 컨테이너화 기술이다. 컨테이너 기술로서 운영체제를 가상화하지 않고 레이어를 모듈처럼 조합한 방식으로 가상머신에 비해 가벼우며 호스트의 자원을 그대로 활용할 수 있다는 장점이 있다. 또한, 컨테이너를 복사하여 다른 환경에서도 동일한 환경을 빠르게 구축할 수 있고, 호스트와 컨테이너가 분리되어 보안상 안전성을 보장할 수 있다.

## 3. 사용자 경험 개선(UI·UX)

### 3.1 디자인 시스템

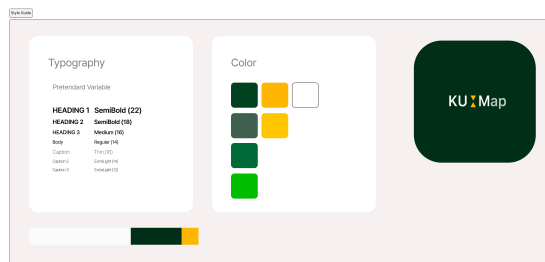


그림 2. KU:MAP Design Style Guide

브랜드 컬러는 ■“#1B4426”로 건국대학교 학생증의 색깔을 참고하여 외부 사용자 접근성을 높인다. 색의 비율은 인테리어에서 사용되는 60-30-10 법칙을 활용하여 60%에는 □“#FEFEFE”를 배경색으로 30%에는 브랜드 컬러를 배분하고, 강조하고자 하는 요소에는 브랜드 컬러에 색상 대비를 가진 ■“#FFBC00”를 10% 사용하여 UI 가시성을 확보한다.

사용자 눈의 피로감을 줄이기 위해 UI 컬러들을 색조를 조절하여 밝기가 낮은 컬러를 사용하고, 흰색(#FFFFFF)이 아닌 밝기가 적은 하얀 흰색(#FEFEFE)을 사용하여 사용자 눈의 피로감을 개선한다.

### 3.2 페르소나 & IA

사용자의 목적을 명시하기 위해 페르소나 UX 방법론을 사용한다. 사용자 페르소나는 대상 고객을 나타내는 가상의 인물을 의미하며, 실제 사람이 아니지만 실제 인터뷰, 설문 조사 및 기타 형태의 사용자 연구를 통해 얻은 데이터와 사실을 기반으로 만든다. 이를 통해 사용자의 한계, 힘든 점, 성공 및 목표를 연관시켜 고객의 요구 사항에 최적화된 맞춤형 사용자 환경을 구축할 수 있다.

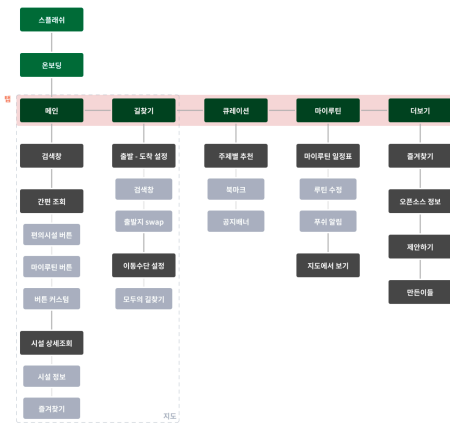


그림 3. IA 중 일부

페르소나를 통해 여러 Feature(제품이나 서비스에서 하나의 기능 단위)들을 나열하였고, Feasibility(실현가능성)를 파악한 후 최종 기능을 뽑아 화면 구조도를 작성한다. Key Feature는 자체 제작 지도 API를 통한 검색과 길찾기 화면, 사용자에게 다양한 장소를 제공하는 큐레이션, 앱의 지속 사용성을 제공하는 마이루틴 화면이다.

<sup>1</sup> 사용자에게 주어진 선택지 수나 복잡성이 늘어나면, 결정에 걸리는 시간이 많아진다.

### 3.3 User Flow

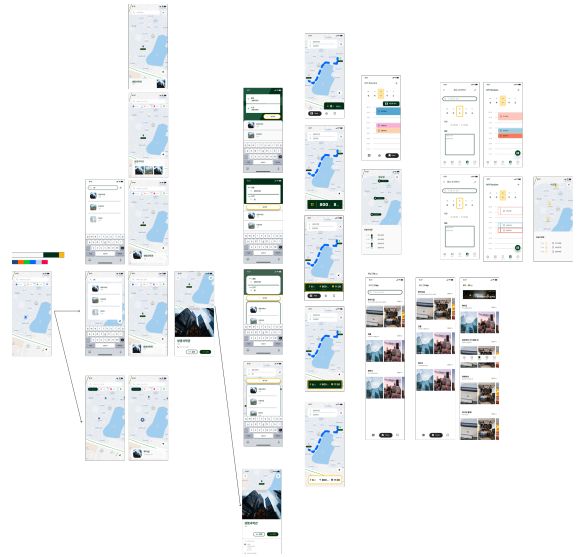


그림 4. User Flow

Figma를 통해 단순성, 가시성, 접근성, 명확성, 일관성있게 다양한 UI를 디자인하고, A/B Test를 통해 UI를 선택하였다.

화면설계의 큰 틀로 제이콥의 법칙을 적용해 상용 서비스를 참고하여 사용자가 새 모델을 익힐 필요 없이 바로 이해하고 사용할 수 있도록 한다. 하지만 지도는 자체적으로 많은 정보를 가지고 있기 때문에 헉스의 법칙<sup>1</sup>을 따라 UI 디자인할 때 복잡성을 단순화했다.

애플리케이션은 도허티 임계법칙<sup>2</sup>으로 상태관리와 앱의 안정성에 초점을 두어 개발한다.

### 4. 시스템 구성 및 구현

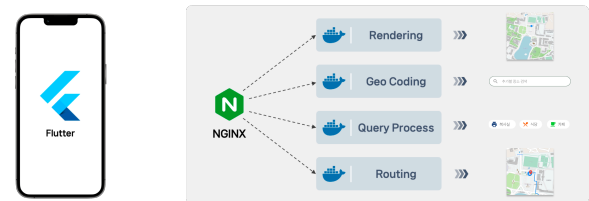


그림 5. 시스템 구성도

본 서비스는 Flutter를 활용해 안드로이드와 IOS 애플리케이션으로 빌드된다. 이때, 학교의 고유 시설(K-Cube, K-Hub, 민원발급기 등)과 상세한 도보

<sup>2</sup> 컴퓨터와 사용자가 서로를 기다리지 않아도 되는 속도로 인터랙션 하면 생산성은 급격하게 높아진다.

정보를 반영하기 위해 자체적인 지리 데이터베이스와 API 서버를 구축한다.

타일 렌더링, 지오코딩, DB 쿼리, 라우팅 각 기능을 도커 컨테이너로 구성하여 단일서버에서도 상호 간섭없이 독립적으로 실행하도록 설계하고, NGINX 를 통해 로드밸런싱을 적용하였다.

#### 4.1 프론트엔드(Front-end)

##### 4.1.1 캠퍼스맵 시스템

본 애플리케이션은 홈 화면에 건국대학교 지도를 배치함으로써 사용자가 교내의 시설을 빠르게 확인하고자 하는 사용자 요구사항을 만족했다.

처음 앱 실행 시 사용자로부터 위치정보에 대한 Permission 여부를 받아와 serviceStatus 에 따라 기능 제공을 달리한다. 그림 6 화면에서 사용자는 위치정보 제공 여부를 선택한다.

사용자의 현 위치는 실시간으로 트래킹되며 지도 위의 오른쪽 하단 플로팅 버튼을 클릭해 현재 위치 시점으로 화면이 전환된다.

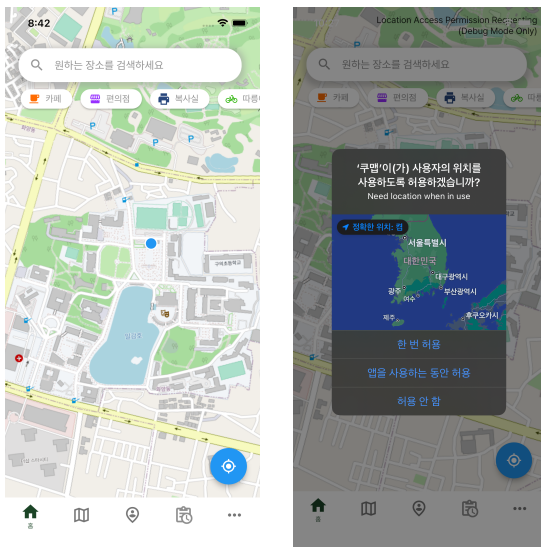


그림 6. 홈 화면(좌) / 위치정보 동의(우)

상단에 자주 찾는 '카페 / 편의점 / 복사실 / 파릉이 / 민원발급기' 등 편의시설에 대한 토글 버튼을 둥글게 하여 사용자의 편의성을 높인다. 토글 버튼을 클릭하면 그림 7 지도 위에 마커가 생성되고, 마커를 클릭 시 장소에 대한 상세정보가 모달 바 형식으로 나타난다. 모달 구현 시 DraggableScrollableSheet 를 사용해, 사용자는 상세정보를 드래그해서 볼 수 있다.

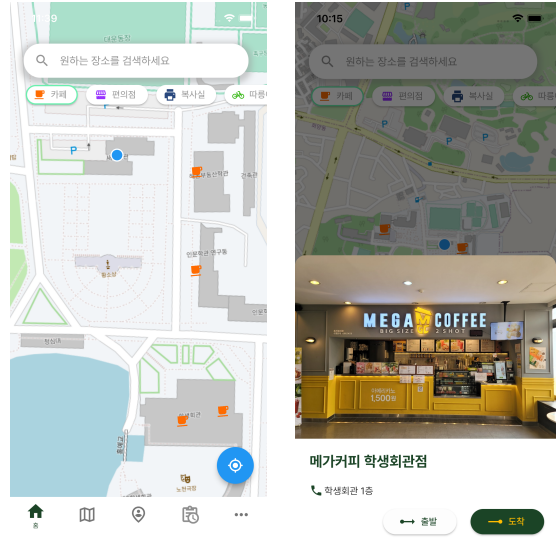


그림 7. 지도 위 마커(좌) / 마커 클릭 시 모달(우)

사용자는 출발 지점과 도착 지점을 설정해 최적의 경로로 길찾기 기능을 사용할 수 있다. 상용 지도 애플리케이션에 반영되지 않은 길들을 직접 GPS 기반으로 수집한 지도 데이터를 API 에 적용함으로써 최적화된 길찾기 안내가 가능하다.

사용자는 현재 위치로부터 길찾기를 수행할 수 있으며, 하단 플로팅 버튼 클릭 시 휴대용 기기에 내장된 나침반을 기반으로 사용자가 바라보고 있는 화면 방향으로 전환이 된다. 사용자는 본 애플리케이션의 길찾기 기능을 이용함으로써 개선된 경험을 할 수 있다.

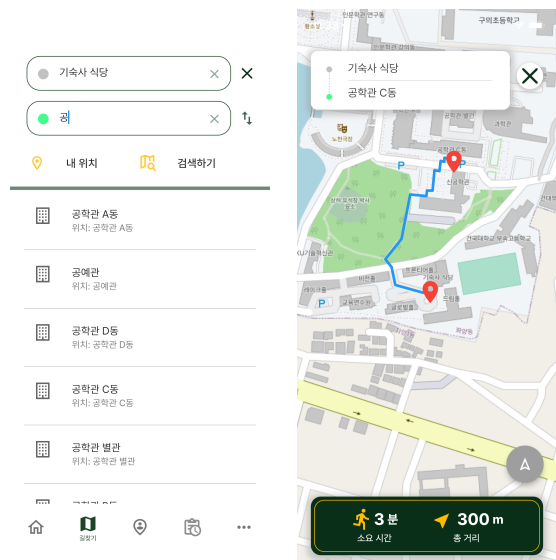


그림 8. 길 찾기 검색(좌) / 길찾기 경로 안내(우)

#### 4.1.2 마이루틴

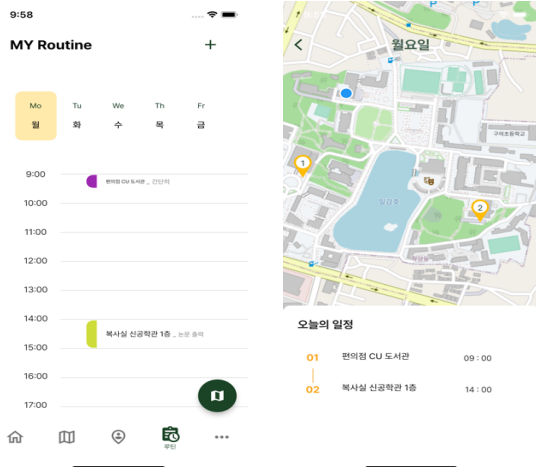


그림 9. 마이루틴 화면(좌) / 마이루틴 지도 화면(우)

마이루틴 화면은 사용자가 직접 갈 장소를 Sqflite 를 통해 로컬 DB 에 저장 후 시간표처럼 볼 수 있도록 하는 화면이다. Adaptive\_dialog 패키지를 사용하여 IOS, Android 플랫폼별로 반응형 다이얼로그를 개발하였고, flutter\_datetime\_picker 패키지를 직접 커스텀 후 UI 디자인에 적합한 time picker widget 을 개발하였다.

중복되지 않는 시간으로 로컬 데이터베이스에 저장하는 알고리즘을 개발하여 앱의 사용성을 향상시켰다. 더불어, 모듈화와 유지 관리를 고려하여 Provider 를 활용하여 개발하였다.

### 4.2 백엔드

#### 4.2.1 데이터베이스 고도화

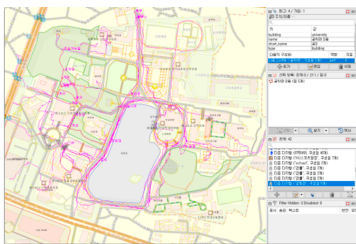


그림 10. 데이터베이스 편집기 화면(분홍선: GPS)

OSM 데이터 편집기인 JOSM 을 이용하여 단축어(공학관 C 동 → 공 C 등)와 대체어(예술문화관 → 예디대 등)를 지정한다. 추가로 연구동과 강의동, 종합강의동과 법학전문도서관과 같이 건물의 상세 용도가 나뉠 경우 part 를 구분하여 독립적인 객체를 추가로 인덱싱하였다. 셋길, 도보의 추가 구축 및 잘못된 도로 수정을 통해 라우팅 기반을 다졌으며, 객체별로 이미지 정보, 운영주체, 홈페이지 등

가능한 정보를 추가로 담고, 학교 고유의 시설에 추가로 태그를 부여하였다.

단축어, 대체어, 건물의 상세정보 구축에 더해 최종적으로 도보를 통해 측정된 GPS 기록을 통해 건물의 위치를 보정함으로써 데이터베이스의 고도화를 이루었다.

#### 4.2.2 편의시설 추출(DB 쿼리)

특정 태그를 가진 객체의 추출(예. 편의점, 식당 등)을 위해 데이터베이스를 처리하는 쿼리 프로세싱 API 를 독립적으로 구축한다. Overpass 시스템을 기반으로 쿼리를 작성하여 편의시설 간편검색 기능 등에 활용한다.

#### 4.2.3 라우팅

OSM 데이터에 특화된 OSRM 라우팅 엔진을 활용하여 구축된 모든 길에 대한 전처리를 진행한다. MLD 알고리즘에 따라 계산된 경로를 빠르게 생성할 수 있으며, google polyline 알고리즘을 통해 문자열로 인코딩하여 데이터 전송의 부하를 줄인다.

기본적으로 제공되는 profile 에 더해 계단이나 숲길 등을 이용하기 어려운 장애 학우를 위해 Lua 스크립트로 라우팅 계산 가중치, barrier 에 대한 blacklist 를 부여하여 barrier-free profile 을 구축하였다.

### 5. 결론 및 향후 연구 방향

기존의 상용 지도 API 를 사용하는 대신, 직접 지리 데이터베이스를 구축하고 사용자에게 제공함으로써 사용자 경험을 개선할 수 있었다. 본 서비스는 출시 이후 사용자 행동 데이터와 사용자 피드백을 반영한 연구를 진행할 예정이다.

#### 참고 문헌

[1] Hussain, Azham, et al. "A user experience evaluation of Amazon Kindle mobile application." *AIP conference proceedings*. Vol. 1891. No. 1. AIP Publishing LLC, 2017.

[2] Huber, Stephan, and Christoph Rust. "Calculate travel time and distance with OpenStreetMap data using the Open Source Routing Machine (OSRM)." *The Stata Journal* 16.2 (2016): 416-423.