**캡스톤 디자인**

**프로젝트 보고서**



**10조 #김김양**

**20143583 김기환**

**20132862 김현빈**

**20142812 양종만**

**목차**

1. **프로젝트 개요**
   1. **프로젝트 제안 배경**
   2. **프로젝트 정의**
   3. **프로젝트 개발 환경**
2. **프로젝트 구현사항**
   1. **ARCore**
   2. **길 찾기 알고리즘**
   3. **GPS를 통한 현재 위치 파악**
   4. **지자기센서를 통한 현재 상태 파악**
   5. **다양한 정보열람 접근 방법**
3. **개발 이슈** 
   1. **안드로이드 쓰레드의 사용으로 인한 Toast 버그**
   2. **현재 상태를 알기 위해 사용한 센서의 변화**
   3. **ARActivity 진입 시 카메라가 기동 되지 않는 현상**
   4. **ARCore의 광원 인식 기능으로 인한 어두운 곳에서의 사용 제한**
4. **느낀 점**

# **프로젝트 개요 – CaU NAVI**

* 1. **프로젝트 제안 배경**

1. 아이디어 제안 배경

중앙대학교에 처음 방문한 사람들을 위한 AR 캠퍼스 가이드 어플리케이션을 개발한다. 처음 학교에 오는 신입생들은 캠퍼스에 어떤 건물이 어디에 있는지 모르기 때문에 학교를 헤매게 된다. 신입생 뿐만 아니라 수험 시즌에 면접, 논술 시험 등을 위해 학교에 찾아온 학생들 또한 학교 지리에 낯선 것은 마찬가지이다. 그 외에 목적을 가지고 학교를 찾아온 사람들 또한 있을 것이다. 그런 이들을 위해 학교 지리에 익숙해지는데 용이한 어플리케이션을 개발하는 것에 초점을 둔다.

1. 안드로이드 어플리케이션의 개발 배경
   1. 우리나라의 스마트폰 보급률

현재 한국의 스마트폰 보급률은 놀라울 정도로 높으며, 그 중 많은 수를 안드로이드 스마트폰이 차지하고 있다. 이러한 점을 생각했을 때, 안드로이드 어플리케이션을 제작한다면 보다 많은 사람들이 이용할 수 있을 것이라 생각된다.

* 1. 안드로이드 어플리케이션 개발 경험

안드로이드는 현재 스마트폰 시장의 강자로써, 당분간은 이 상황이 유지될 것이다. 그러한 안드로이드 어플리케이션의 개발 경험은 앞으로의 진로에 도움이 될 것이라 생각된다.

* 1. **프로젝트 정의**

AR이란 Augmented Reality의 준말로 한글로 번역하면 증강현실이다. 인간이 인지할 수 있는 현실(Reality)에 컴퓨터가 받아들인 정보를 토대로 이미지를 덧씌우는(Augment) 것이 증강현실이다.

캠퍼스 가이드에 필수적인 것은 길안내이다. 사용자가 현재 장소에서 원하는 장소로 갈 수 있도록 길을 안내해주어야 한다. 사용자가 자신의 현재 위치를 입력하고, 목적지를 설정하면 경로를 안내하도록 한다.

캠퍼스 가이드는 건물, 구조물에 대한 정보를 알려줄 수 있어야한다. 학교에 입학하는 신입생은 이 건물이 어떤 건물인지 궁금해할 것이고, 그 궁금증을 어느정도 해소할 수 있는 수준의 정보를 어플리케이션을 통해서 제공할 수 있게 한다.

* **AR 캠퍼스 경로 안내**

캠퍼스 가이드에 필수적인 것은 길안내이다. 사용자가 현재 장소에서 원하는 장소로 갈 수 있도록 길을 안내해주어야 한다. 사용자가 목적지를 설정하면 GPS기술을 이용하여 현재 위치에서 목적지까지의 경로를 안내하도록 한다.

1. GPS를 통해 사용자의 위치를 입력 받는다.
2. 사용자가 원하는 목적지를 선택한다.
3. 사용자의 카메라를 이용해서 목적지의 방향을 AR로 표시한다. 하단의 알고리즘 이용
   1. 경로가 두가지 이상일 경우, 경로의 특징을 간략히 알려준다. (ex 빠르지만 계단, 에스컬레이터 존재)
4. 목적지에 도착하면 안내를 종료한다.

* **길안내 알고리즘**

1. 현재 위치를 GPS 좌표화 한다.
2. 목적지까지의 경로를 선정한다.
   1. 캠퍼스 내에서 방향의 전환이 있을 수 있는 지점이 선정되어있다.
   2. 목적지까지의 경로 안내는 지점에서 다음 지점으로 이동하도록 안내한다.
   3. 목적지까지의 경로는 지점들을 vertex로 한 다익스트라 알고리즘을 활용.
   4. 각 지점 간에는 특징 정보가 있다(평지, 계단, 경사).
3. 사용자 스마트폰의 GPS 좌표와 지자기 센서를 이용한 방위각을 구한다.
4. 현재 위치한 지점에서 지점까지의 경로를 카메라 AR을 이용해 표시한다. (방위각을 이용)

* **대학의 역사와 건물에 대한 정보 안내 기능**

캠퍼스 가이드는 건물, 구조물에 대한 정보를 알려줄 수 있어야한다. 학교에 입학하는 신입생들과 대학탐방을 온 학생들은 이 건물이 어떤 건물인지 궁금해할 것이고, 그 궁금증을 어느정도 해소할 수 있는 수준의 정보를 어플리케이션을 통해서 제공한다.

1. 목적지로 설정한 위치에 도착했을 시, 그 목적지의 정보 안내 기능
   1. 팝업창을 이용하여 정보를 안내 여부를 선택할 수 있다.
2. 한번 자세히 본 정보는 읽음표시를 해서 구분을 한다.
3. 사용자가 원한다면 모든 정보를 위치에 상관없이 읽을 수 있다.

* **캠퍼스 투어 모드**

캠퍼스 투어 모드는 대학탐방을 온 중, 고등학생을 위한 기능이다. 중, 고등학생을 위한 학교 정보 안내가 목적이다.

1. 2가지의 기본루트를 설정 (정문 -> 후문, 후문 -> 정문)
   1. 사용자가 직접 루트를 설정할 수 있다.
2. 루트에 따라서 사용자를 안내한다.
3. 목적지로 향하는 경로에 있는 건물에 대한 정보와 역사를 자동으로 간략히 표시
   1. 원한다면 자세히 보거나, 정보 안내를 끌 수 있다.

* **자유 모드**

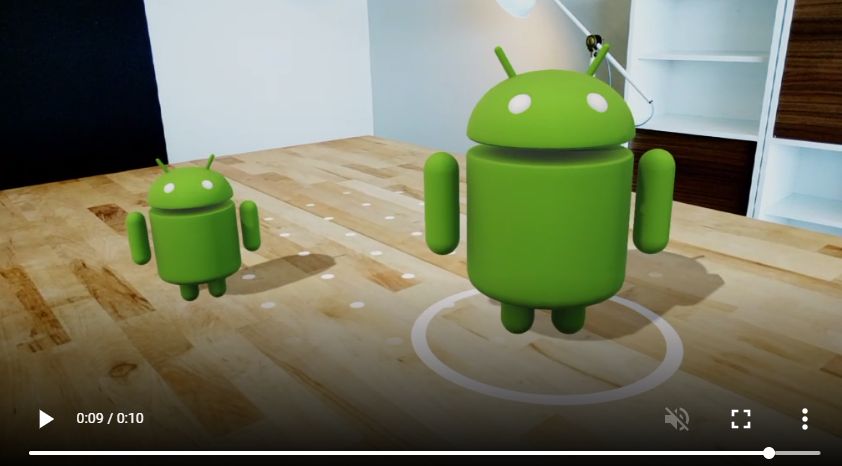
캠퍼스 투어 모드와 달리 정해진 루트 없이 자유롭게 돌아다니며 주위 건물에 대한 정보와 대학 역사에 대한 안내를 받을 수 있는 모드이다.

1. 현재 위치 근처에 있는 건물에 대한 정보와 역사를 자동으로 간략히 표시
   1. 원한다면 자세히 보거나, 정보 안내를 끌 수 있다.
   2. 한번 자세히 본 정보는 자동으로 표시되지 않는다.
2. 자유 모드 도중에도 특정 건물에 대해 길안내를 받고 싶다면 건물에 대한 안내 기능
   1. 캠퍼스 투어 모드와 마찬가지로 경로 도중에 있는 정보 표시
   2. **프로젝트 개발 환경**

* Android Studio
* ARCore (안드로이드 버전 7.0.0 (Nougat) 이상 요구)
* Galaxy S7 및 Galaxy A8

## **프로젝트 구현사항**

1. **ARCore**



구글의 ARcore 샘플 코드(<https://github.com/google-ar/arcore-android-sdk/tree/master/samples>) 중 hello\_ar를 기반으로 개발을 진행하였다. Hello\_ar 의 동작 알고리즘은Cloud Anchor를 통한 사물인식 -> Plane rendering를 통한 평면 인식 -> model rendering을 통한 평면위에 모델 랜더링 순서로 진행된다. 우리는 이 코드를 이용하여 평면인식 후 사용자의 정면에 화살표모델을 랜더링하는 방법을 사용하였다.

ARcore의 CloudAnchor와 평면인식을 이용하였기 때문에 별도의 ar마커 없이도 허공에 모델을 랜더링 할 수 있었다. 우린 이 방법을 이용해서 사용자의 정면에, 아무 표시가 없는 허공에 화살표를 생성할 수 있었다.

1. **길 찾기 알고리즘**

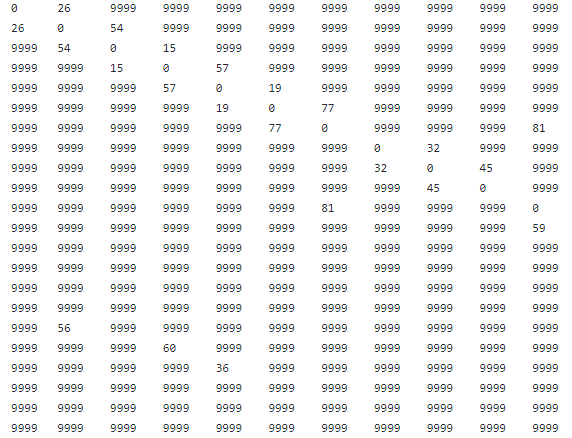
중앙대학교 캠퍼스의 GPS노드를 66개를 설정하였다. 길 찾기에 사용하기에는 노드 수가 부족하지만, GPS좌표 오차를 생각해서 갈림길, 교차로 등 최대한 효율적인 노드를 구성하고자 하였다.

이 노드들의 인접행렬, 66x66의 그래프를 만든 뒤 이 그래프를 이용해서 시작점과 도착점의 경로 노드를 계산해주는 다익스트라 알고리즘을 구현하였다. 처음에는 실시간으로 경로를 계산하는 것이 아니라, 미리 저장된 경로를 불러오는 방법으로 구현하였었지만, 개발 도중에 노드가 계속 추가되고 변경되었으며, 실시간으로 다익스트라 알고리즘을 적용해도 휴대폰에 큰 부하가 걸리지 않아서 실시간 계산 방식으로 변경하였다.

그래프의 가중치는 노드 간의 평면 상의 거리를 위도/경도를 통해 계산하였다. 그 중 경사와 계단은 평면상의 거리와 실제 이동거리, 운동량이 다르다고 생각해서 1.3배, 1.6배의 가중치를 추가하였다.



각 노드에는 위도경도와 인접한 건물ID, 노드의 이름과 인접한 노드의 번호가 저장된다.



그래프의 가중치는 실제 노드의 거리이며, 연결되지 않은 노드는 9999의 값을 가진다.

1. **GPS를 통한 현재 위치 파악**

안드로이드에서 GPS정보를 제공받는 방법은 크게2가지가 있다. 직접 GPS위성으로부터 신호를 받거나, 네트워크를 통해서 근처의 디바이스의 정보를 받아오는 것이다. 위성을 이용하는 방법은 제약사항 (실외, 날씨 등) 이 많지만 네트워크에 비해서 높은 신뢰도를 가진다. 네트워크를 통한 GPS는 네트워크에 연결만 되어 있다면 제약사항 없이 정보를 받아올 수 있지만 낮은 신뢰도를 가진다.

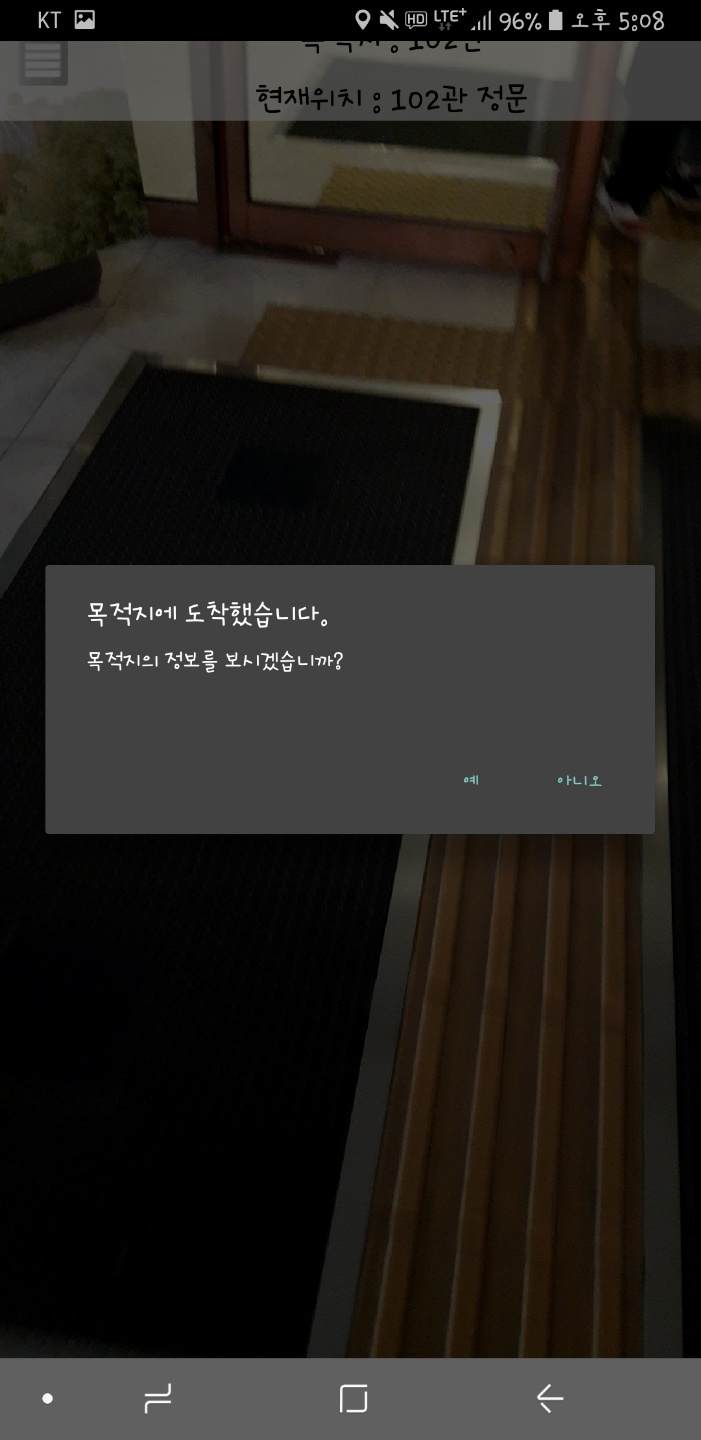
우리의 프로젝트 목표는 캠퍼스 길찾기였고, 정확한 GPS좌표가 필요했기 때문에 네트워크 정보를 받지 않고 GPS위성을 통해서만 좌표를 받아오는 방법을 선택하였다. 하지만 위성을 통한 정보라고 해도 기본 10m의 오차가 있었다. 그래서 우리는 사용자의 현재위치 파악을 GPS좌표로만 하는 것이 아니라, 66개의 노드 중 제일 가까운 노드를 계산해서 ‘그 노드 근처에 있다’ 라고 현재위치를 가정하였다.

1. **지자기센서를 통한 현재 상태 파악**

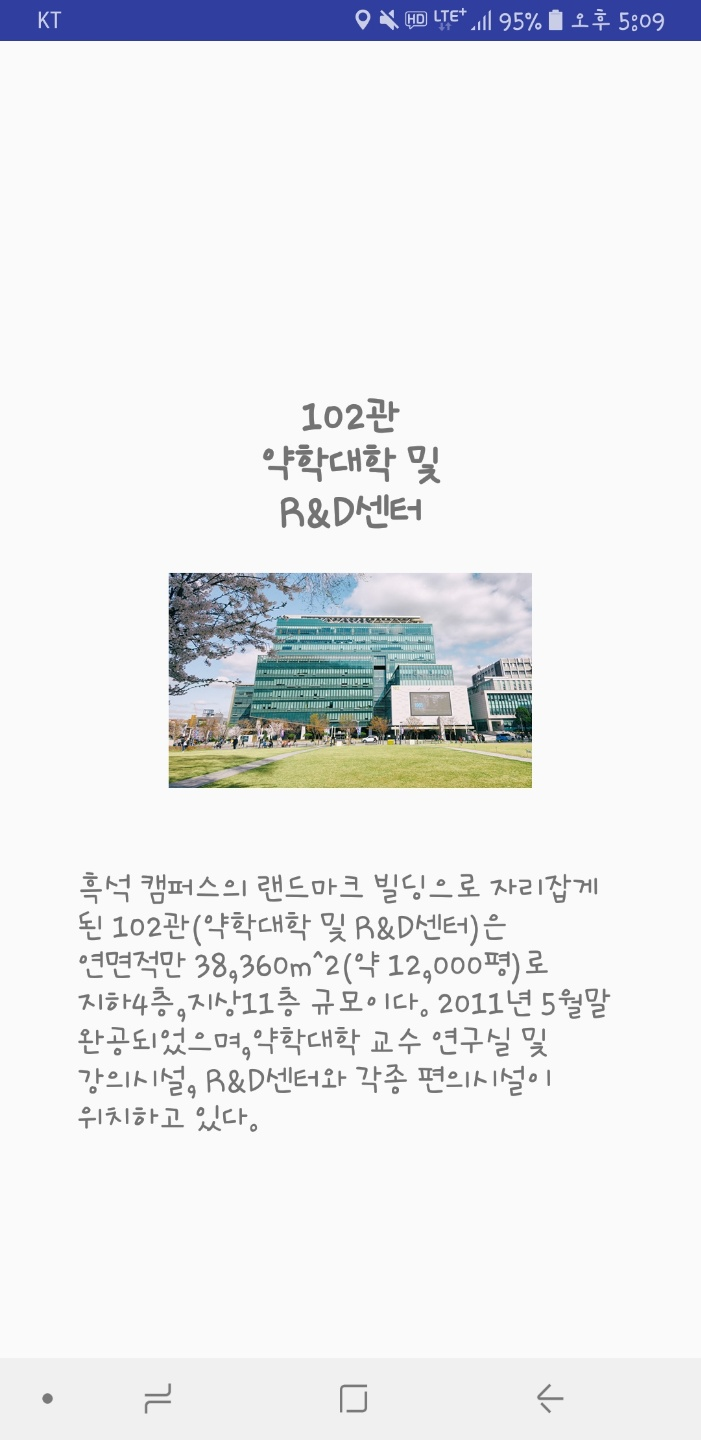
안드로이드 센서매니저를 통해서 스마트폰의 내장센서의 데이터를 받아올 수 있다. 지자기센서에는 여러가지 모드를 통해서 값을 받아올 수 있는데, 단순히 지자기센서를 이용하는 것(ORIENTATION) 과 지자기센서(MAGANETIC\_FIELD)와 가속도센서(ACCELEROMETER)를 이용해서 보정하는 2가지 방식이 있다. 정확도와 민감도는 지자기센서+가속도센서가 뛰어나지만 반환되는 각도의 값이 라디안방식이라 계산에 난항이 있었고, 너무 민감하다 보니 미세한 움직임에도 각도가 10도씩 변하는 등 우리 프로젝트와 맞지 않았다. 그래서 우리는 TYPE\_ORIENTATION 방식을 사용하여 지자기센서의 값을 받아왔다.

지자기센서는 스마트폰이 자기장을 감지해서 x축 y축 z축 3방향의 각도를 반환한다. 자기장을 이용하다 보니 주위에 전자제품이나 자동차, 자판기 등이 있으면 정확한 값을 반환하지 못한다는 단점이 있다.

1. **다양한 정보열람 접근 방법**

자유투어에서 마커를 터치하거나, 길찾기에서 목적지에 도착하거나 하면 그 화면에서 정보열람 여부를 묻도록 하였다. 단순히 메인화면에서만 접근할 수 있는 것보다, 사용자가 어플리케이션을 활용하던 도중에도 접근할 수 있어야 UX 측면에서 적합한 것이라고 생각이 되어서 이렇게 구현하였다.

그렇기에 메인화면에서 정보에 접근했을 때와 그렇지 않을 때에 가이드 시작 버튼의 활성화 / 비활성화 여부를 다르게 하였다. 사용자의 의도와 다르게 목적지가 변경되면 어플리케이션이 꼬여서 정상적으로 작동하지 않을 우려가 있었기에 이러한 조치를 하였다.

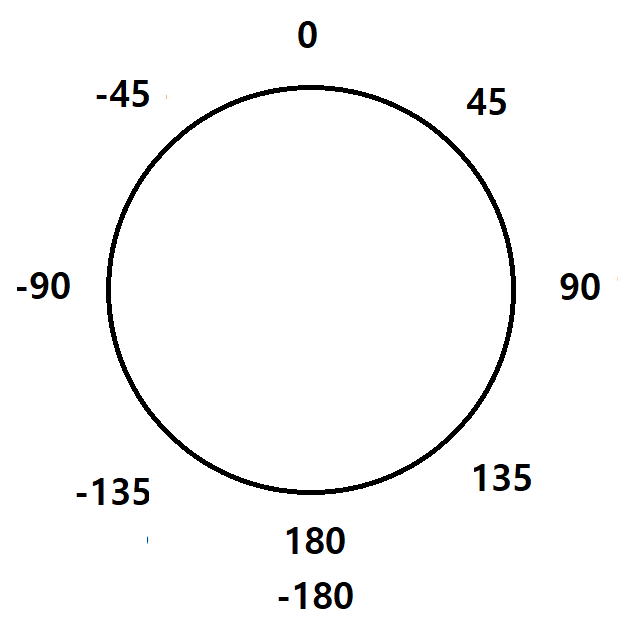
## **개발 이슈**

1. **안드로이드 쓰레드의 사용으로 인한 Toast 버그**

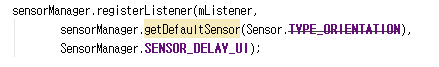
개발 도중 쓰레드 안에서 토스트를 띄우는 코드가 있었는데 액티비티가 종료돼도 계속 토스트가 발생하는 문제가 있었다. 원인은 액티비티에서 쓰레드를 사용한 뒤, 쓰레드 종료없이 액티비티를 종료하면 쓰레드가 종료되지 않고 계속 작동하는 점이였다. 액티비티에서 직접 스레드를 사용하는 것이 아니라, 다른 클래스를 이용해서 스레드를 사용하면 클래스 인스턴트가 종료되면서 스레드도 같이 종료되지만, 액티비티의 경우에는 그렇지 않았던 점이 문제였던 것 같다. 결국 액티비티 종료와 함께 스레드 종료문을 함께 넣어서 문제를 해결할 수 있었다.

1. **현재 상태를 알기 위해 사용한 센서의 변화**

AR길찾기는 화살표를 올바른 방향으로 만들기 위해서는 휴대폰이 현재 어느 방위를 향하고 있는지, 사용자가 휴대폰을 어떻게 들고 있는지를 알아야한다. 안드로이드는 이러한 정보를 담고 있는 센서들을 관리하는 매니저를 갖는데, 그것이 SensorManager 클래스이다. SensorManager는 지자기 센서를 비롯한 가속도 센서, 자이로 센서 등의 데이터를 관리한다. 맨 처음 조사를 하였을 때, 대부분의 문서에서는 방위를 알기 위해서는 가속도 센서와 지자기 센서를 조합해서 쓸 것을 추천하였다. 그렇기에 프로젝트 초반에는 그렇게 진행하였고, 거기에서 알 수 있었던 것은, 방위각을 우리가 익숙한 모양과는 다르게 표현한다는 것이었다.



가속도 센서와 지자기 센서를 같이 사용하게 될 경우 위와 같이 동서남북을 나타내는 값이 우리에게 익숙한 0 ~ 360이 아닌 0 ~ 180, -180 ~ 0의 값을 가지게 된다. 휴대폰의 상태에 따라 값이 너무 민감하게 변하여서 프로젝트를 진행하면서 계속 이부분에 대한 보정이 필요했기에, 많은 시간이 소요되었다.



그러다가 바꾼 현재의 방식은 deprecated 된 그냥 Orientation을 쓰는 방식이다. 가속도 센서와 지자기 센서를 조합하여서 상태를 알아내는 것보다 값의 변화가 덜 민감하였고, 정확도 또한 어플리케이션이 요구하는 수준을 만족하였기에 이 값을 사용하기로 하였다. Orientation은 방위를 0 ~ 360으로 나타내주기 때문에 값을 이용하기도 더욱 직관적이었다.

1. **ARActivity 진입 시 카메라가 기동하지 않는 문제**

ARCore의 깃허브 내에서도 이슈(<https://github.com/google-ar/arcore-unity-sdk/issues/332>)가 있었으며, ARCore가 기본 스마트폰에 내장된 기본 카메라 어플을 사용하는데 삼성의 카메라 어플과 ARcore의 충돌이라고 생각된다.

1. **ARCore의 광원 인식으로 인한 어두운 곳에서의 사용 제한.**

ARCore의 광원 인식 기능이 기본적으로 활성화 되어있어서, 렌더링 되는 물체에 자동으로 적용되었다. 그로 인해 어두운 곳에서 어플리케이션을 동작할 경우 화살표가 외부 광원에 영향을 받는 현상이 생겼다. 실제로 밤에 캠퍼스로 나가서 실행을 해보았을 때, 가로등 같은 광원이 있는 곳에서는 화살표가 정상적으로 볼 수 있었지만 주변이(카메라 시야가) 어두운 곳에서는 화살표가 어둡게 렌더링 되어서 생성되어도 보기 힘들다는 이슈가 있었다. 광원 인식 기능에 관한 부분을 찾지 못해서 해결을 하지 못한 부분이 아쉽다.

## **느낀 점**

**김기환**

요즘 협업능력이 강조된다는데 한학기동안 프로젝트를 진행하면서 일정조율, 업무분담 등 서로 맞춰가는 것을 배울 수 있었고 정해진 기간에 맞춰서 진행하는 게 얼마나 어려운 일인지 깨달을 수 있었던 것 같다.

**김현빈**

한학기동안 팀끼리 선택한주제로 개발을 진행해보니 재미도 있고 많은 것을 배울 수 있었다. 특히 개발일정을 정하고 그 일정에 맞춰 개발을 진행하는 것이 쉽지 않다는 것을 느꼈다. 다음 캡스톤 때는 더 노력해야겠다.

**양종만**

준비 단계에서 사전 조사가 충분하지 못했던 것 같아서 아쉬운 점이 있었다. 그렇지만 어떻게든 완성을 시켰기에 만족스러운 프로젝트였다.