**[5110135 캡스톤디자인 2020S]**

**Progress Report**

팀 번호: 2팀

진행 기간: 2020.4.17. ~ 2020.4.23.

|  |
| --- |
| 주차별 진행 내용 및 결과를 자유롭게 작성. 분량 제한은 없으나 본 과제도 이번 학기 평가의 대상임에 유의. **주차별 목표**는 3주차 작성 내용과 완전히 일치할 필요는 없으므로 진행 상황에 맞게 지정. **진행 결과**는 주차별 목표 항목과 대응하도록 작성. 자료 수집을 한 경우 자료 수집 내용을 작성하고, 개발을 진행한 경우 Github에 commit한 내역을 캡쳐하여 올릴 것. |

**1. 프로젝트 주제: 비콘을 이용한 의류 매장 서비스**

**2. 주차별 목표 및 달성률**

|  |  |  |
| --- | --- | --- |
| 주차별 목표 | 담당자 | 달성률 |
| (1) 주제에 맞는 beacon 조사  (2) 관련 오픈소스 및 기술 조사 | 박병조  김소연, 이시은 | 100%  100% |

**3. 진행 결과**

**(1) 주제에 맞는 beacon 조사**

프로젝트에 사용할 비콘은 iBeacon을 사용할 예정이다. 접근하기가 쉽고 iBeacon에는 인식거리별 여러 제품이 있어 다양한 활용이 가능하기 때문이다.

**1) 실제 서비스에 이용할 Beacon**

실외 : 인식거리 약 100m 이상이 되고 장기간 이용을 해야 하기 때문에 E2 제품을 사용하면 좋을 것이라 판단

 **<iBeacon E2>**

실내 : 내부 위치파악을 위해 약 200m 이내와 행거 부착에 용이해야 하기 때문에 i3 제품이 좋을 것이라 판단

 **<iBeacon i3>**

**2) 테스트를 위한 Beacon**

실내, 실외 상관없이 연동 및 프로그램 개발용으로 가격이 싸고 적당한 인식거리가 필요하고 휴대가 좋은 i4 또는 B1제품으로 선택해 사용할 예정이다.

<iBeacon i4> <iBeacon B1>

**(2) 관련 오픈소스 및 기술 조사**

**[ iBeacon 기초 ]**

**1) Settings**

비콘을 활용하기 위해선 비콘의 영역을 감지하고 비콘과의 대략적인 거리를 판단하는 기능이 필요하다. 이를 각각 Monitoring과 Ranging이라고 한다. 먼저 위치 서비스는 앱의 사용 유무와 상관없이 항상 켜져 있어야 한다. 이는 info.plist 에 명시해주어야 하고 앱이 실행됨과 동시에 위치 서비스에 대한 권한을 사용자에게 요청해야 한다. 다음은 Background에서도 현재 위치가 계속해서 갱신되어야 한다. 또한 항상 디바이스는 블루투스 Detecting 기능이 켜져 있어야 한다.

**2) Monitoring & Ranging**

Monitoring은 비콘의 영역 내/외부를 판단하는 행위다. 이는 위에서 언급한 앱의 실행 유무와 상관없는 위치 서비스와 백그라운드 위치 갱신 서비스의 설정이 선행된 후에 가능한 기능이다. 이렇게 Monitoring을 통해 현재 디바이스가 비콘의 영역의 내부에 있다는 것이 판단되면 Ranging을 통해 해당 비콘과의 근접도를 판단한다. 근접도의 단계에는 네 가지가 존재한다.

1. Immediate : 비콘과 디바이스가 상당히 가까운 상태로 거의 디바이스에 붙어 있는 수준

2. Near : 대략적으로 1 ~ 3 미터 내외의 거리로 비교적 가까운 수준이거나 신호가 Immediate에 비해 약할 때

3. Far : 물리적인 거리가 Near나 Immediate보다 떨어져 있거나 신호의 세기가 이들보다 약한 경우

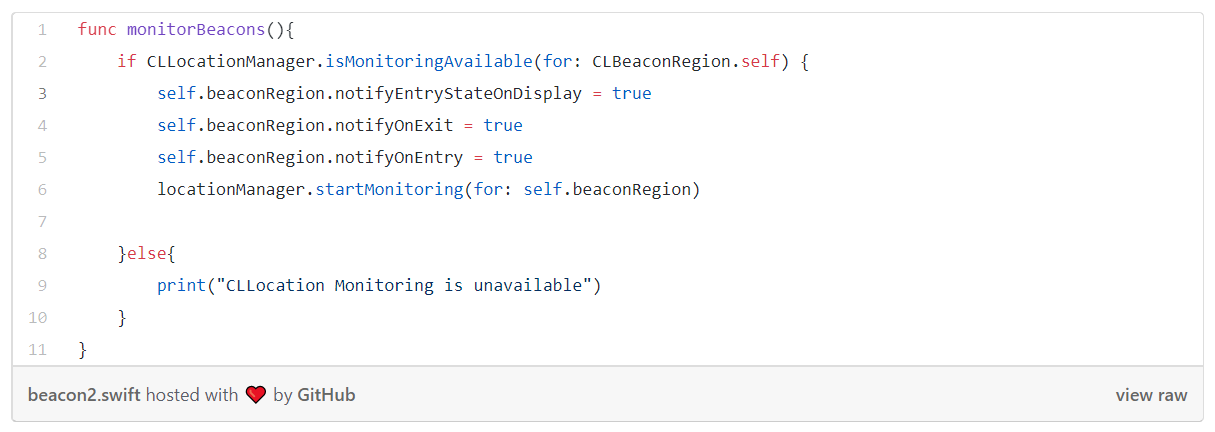
4. Unknown : 신호의 세기나 거리에 따른 근접도의 판단이 불가능할 때, 방금 막 Ranging이 시작되었을 때 등과 같이 측정을 하기 충분한 상황이 아닐 때

위 내용들을 살펴보면 근접도는 단순한 물리적 거리 뿐만 아니라 신호의 세기도 영향을 받는다는 것을 알 수 있다.

**3) Implementation**



locationManager 객체를 생성하여 프로젝트 설정과 별개로 위치 서비스 권한 요청과 백그라운드 위치 갱신 등의 코드들을 작성한다. 또한 위치 서비스에 대한 권한이 받아들여졌다면 monitorBeacons 메소드를 통해 비콘의 영역을 Monitoring 한다.



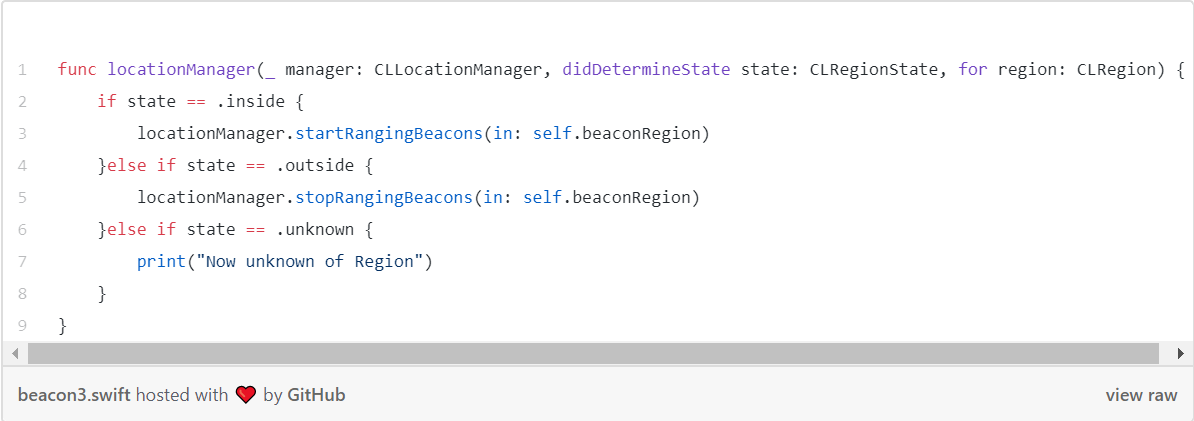
먼저 현재 Monitoring이 가능한 상태라면 Monitoring을 진행한다. 이에 앞서 notifyEntryStateOnDisplay를 true 로 설정함으로써 디바이스가 이미 영역 안에 있거나 앱이 실행되고 있지 않은 상황에서도 영역 내부안에 들어오면 백그라운드에서 앱을 실행시켜 해당 notification을 받을 수 있게 한다.

또한 notifyOnExit과 notifyOnEntry를 통해 영역 안에 들어온 순간, 나간 순간에 해당 notification을 받을 수 있게 한다. 그리고 하나의 비콘 영역의 객체는 다음과 같이 생성한다.



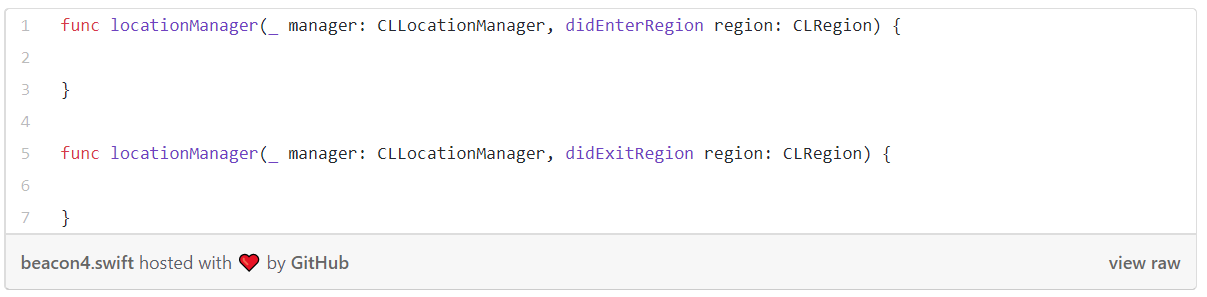
UUID는 비콘 디바이스에 설정된 값으로 작성해야 하고 identifier는 임의의 식별할 수 있는 값으로 설정하면 된다.

이렇게 Monitoring이 실행된 후 영역의 판단이 이루어지는 순간에 다음 메소드가 실행된다.

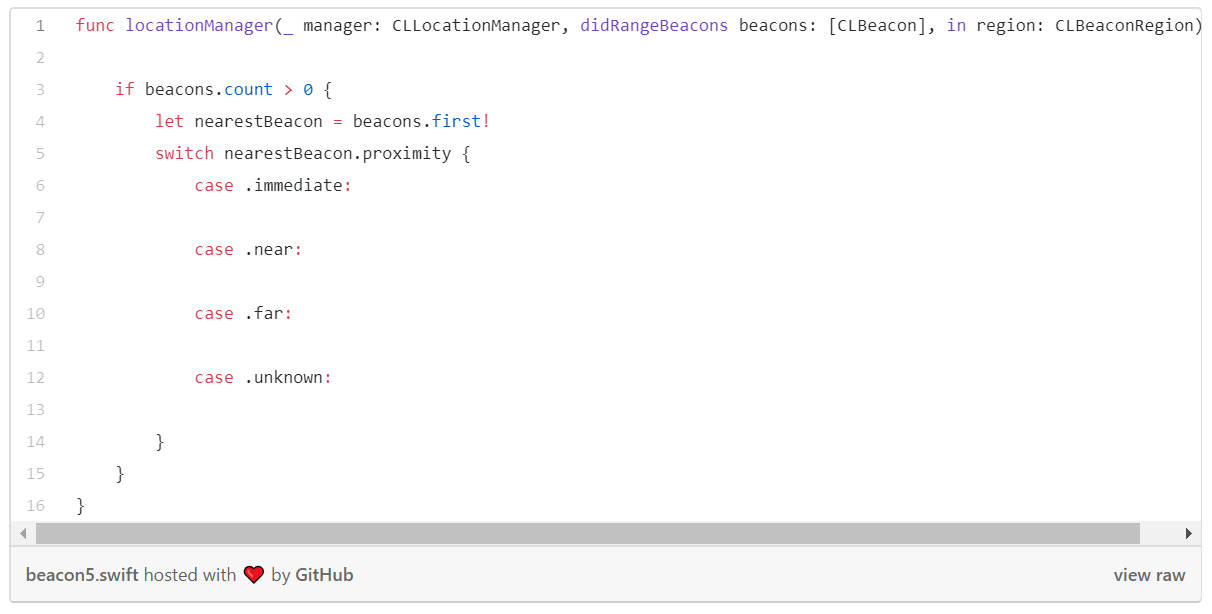


.inside와 .outside 는 각각 영역 안에 들어온 순간과 나간 순간을 의미하며 그 순간에 동작해야할 코드들을 이에 맞게 작성한다. 위의 코드는 영역안에 들어왔을 때 해당 비콘과의 근접도를 측정하는 Ranging을 시작하고 영역을 나가는 순간 Ranging을 멈춘다.

그리고 위의 메소드는 다음의 두 메소드를 추가로 호출한다.



그리고 다음 메소드를 통해 Ranging의 결과에 따른 코드를 작성한다.



현재 Ranging 이 가능한 비콘들 중 하나를 뽑아 근접도에 따른 코드를 작성할 수 있다.

참고자료 – <https://baked-corn.tistory.com/m/121>

**[ 안드로이드 Beacon 연동 ]**

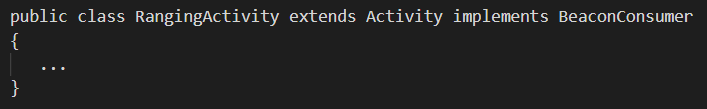
안드로이드와 Beacon을 연동하기 위한 비콘 전용 오픈소스는 크게 두 가지가 있다. 첫 번째는 AltBeacon의 android-beacon-library이고, 두 번째는 estimate이다.

1. **android-beacon-library**

android-beacon-library는 다음과 같은 특징을 갖는다.

* manifest에서 permission, service 설정할 필요가 없다 (aar에 포함되어있다)
* 스캔시 1초마다 didDetermineStateForRegion()에 비콘신호가 올라온다.
* BeaconManager 싱글턴으로 객체 생성후 bind 하면 callback으로 onBeaconServiceConnect()가 호출된다.

**< BeaconConsumer 구현 >**



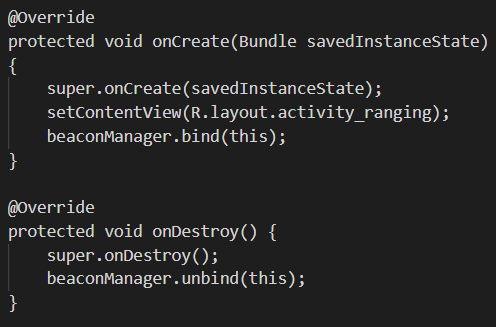
Beacon과 상호작용을 할 Activity는 BeaconConsumer 인터페이스를 구현해야 한다. BeaconManager와 연계하여 사용하면 BeaconService 사용이 준비되었을 때 알려준다. BeaconManager의 bind 메소드를 호출한 후에 서비스가 사용 가능할 때, onBeaconServiceConnect 메소드가 호출되고 이 메소드에서 Monitoring이나 Ranging을 시작한다.

**< BeaconManager 인스턴스 생성 >**



위와 같이 BeaconManager 인스턴스를 생성한다.

**< BeaconService bind와 unbind >**

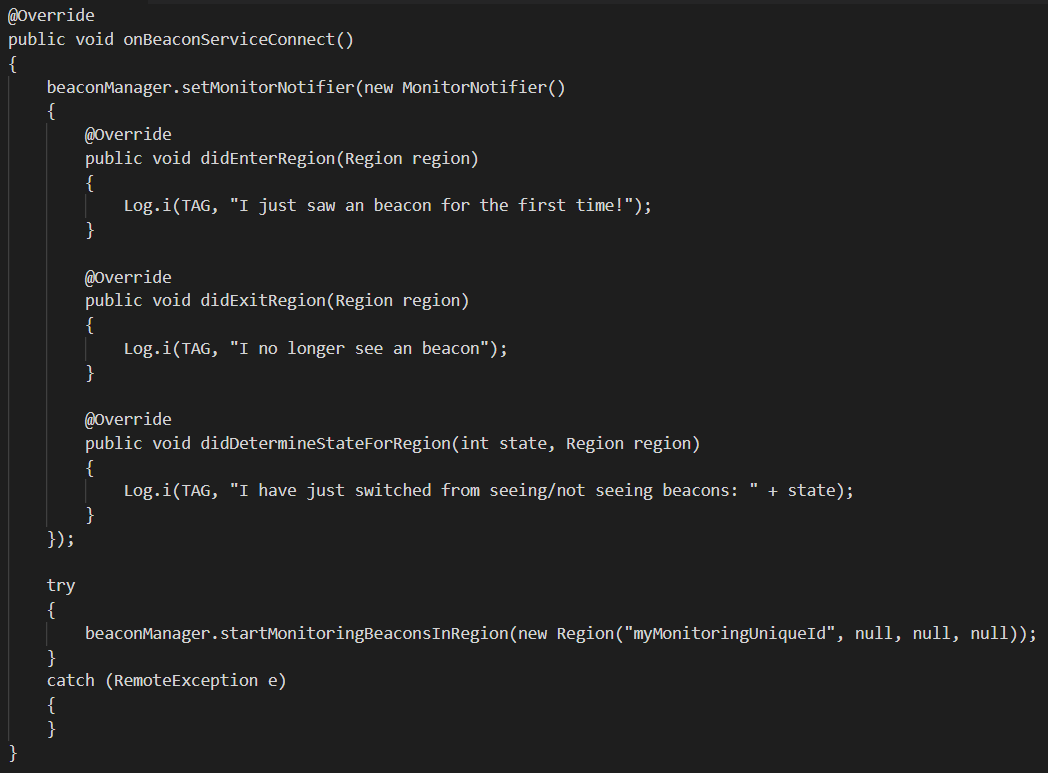


위와 같이 onCreate와 onDestroy 메소드에서 BeaconManager의 bind와 unbind 메소드를 각각 호출한다.

**< Notifier 설정 >**

Beacon 서비스에 연결되면 onBeaconServiceConnect 메소드가 호출되는데, 이 메소드는 MonitorNotifier 또는 RangeNotifier를 설정하고, 이는 각각 Monitoring 또는 Ranging을 시작하는 역할을 한다.

**< MonitorNotifier 설정 >**



MoniterNotifier는 위와 같이 설정할 수 있다.

Region은 Beacon의 UUID를 통해서 특정 지역을 나타내는데 사용된다.

MoniterNotifier는 Region에 들어가거나 나왔을 때 Notify를 받기 위한 인터페이스이다.

didEnterRegion 메소드는 Region을 구성하는 Beacon 중에 하나라도 발견되었을 때 호출된다.

didExitRegion 메소드는 Region을 구성하는 Beacon들이 하나도 발견되지 않았을 때 호출된다.

didDetermineStateForRegion 메소드는 Region을 구성하는 Beacon 중에 하나도 발견되지 않았을 때는 state가 MonitorNotifier.OUTSIDE 값으로 호출되고 Region을 구성하는 Beacon 중에 하나라도 발견되었을 때는 state가 MonitorNotifier.INSIDE 값으로 호출된다.

**< RangeNotifier 설정 >**



RangeNotifier는 위와 같이 설정할 수 있다.

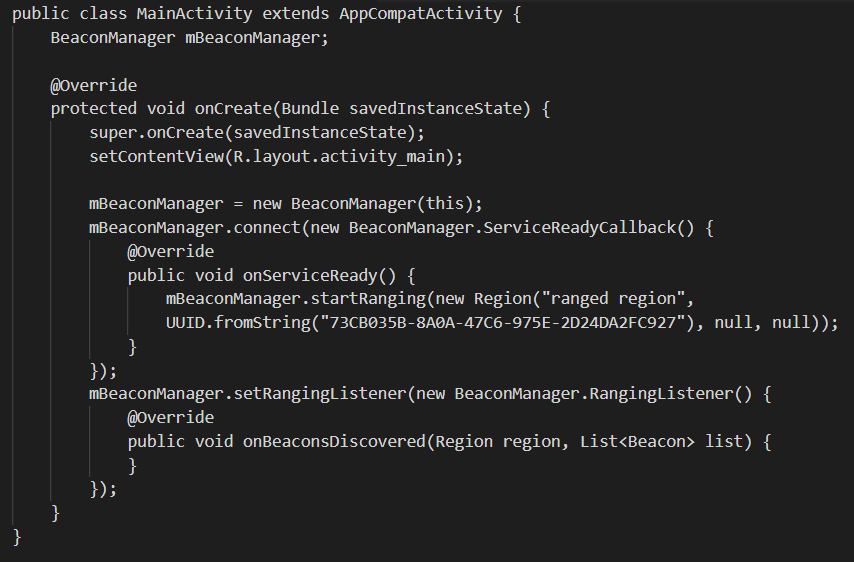
didRangeBeaconsInRegion 메소드는 초당 한번 Beacon에 대한 정보 및 거리를 산출하여 제공한다.

관련 링크 : <https://altbeacon.github.io/android-beacon-library/index.html>

1. **Estimote**

Estimate 라이브러리는 다음과 같은 특징을 갖는다.

* manifest에서 permission, service 설정할 필요가 없다. (aar에 포함되어있다)
* BeaconManager 객체 생성 후(싱글턴 아님) connect() 호출 시 callback으로 onServiceReady()가 호출된다.
* setRanging() 설정 후 startRanging() 호출 시 1초마다 onBeaconsDiscovered()가 콜백 호출된다.
* background 스캔 가능하다.



위와 같이 샘플 코드를 작성할 수 있다.

관련 링크 : <https://developer.estimote.com/>

**[ 비콘 관련 안드로이드 소스 ]**

1. 비콘 플랫폼

<https://github.com/google/beacon-platform>

2. 안드로이드 비콘 라이브러리

<https://github.com/AltBeacon/android-beacon-library>

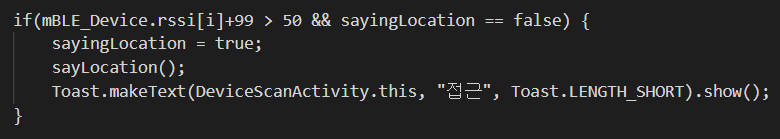
3. 안드로이드와 비콘을 연동하는 방법

<http://blog.naver.com/PostView.nhn?blogId=alice_k106&logNo=220533957852&parentCategoryNo=&categoryNo=&viewDate=&isShowPopularPosts=false&from=postView>

**[ 비콘 기능을 활용한 음성 메세지 안내 및 진동 ]**

링크 : <https://github.com/dimplejuno/BotCle110>

DeviceScanActivity에 비콘을 검색하고 세기를 검사하는 부분이 있다. 블루투스 서비스는 발견된 비콘의 내용을 담아서 콜백에 넘겨주면 콜백 루틴은 LeDeviceListAdapter라는 어댑터를 통해서 리스트를 만들어 각 비콘의 세기를 표시해주는 부분이다. 이 부분에서 RSSI의 세기를 측정한다.



진동을 위해서는 Manifest에 아래와 같이 권한을 등록해야 한다. Morse code 형태로 진동하기 위해서 MorseCodeConverter 소스도 추가한다.



테스트의 음성 안내를 위해서 TextToSpeech 객체를 사용할 수 있다.



참고자료 :

<https://hojunpark.wordpress.com/2015/06/26/%EB%B9%84%EC%BD%98-%EA%B8%B0%EB%8A%A5%EC%9D%84-%ED%99%9C%EC%9A%A9%ED%95%9C-%EC%9D%8C%EC%84%B1-%EB%A9%94%EC%84%B8%EC%A7%80-%EC%95%88%EB%82%B4-%EB%B0%8F-%EC%A7%84%EB%8F%99/>

**[ 참고해볼 사이트 ]**

안드로이드 음성인식 사용법 : <https://dsnight.tistory.com/15>

안드로이드 TTS(텍스트 읽어주는) 서비스 사용법 : <https://norwegianboy.tistory.com/4>

java를 이용한 안드로이드 비콘 (BLE) 스캐너 Android Beacon Library 사용 방법 :

<https://altongmon.tistory.com/449>

안드로이드 비콘 연동 : <https://loveiskey.tistory.com/m/207>