

캡스톤 디자인 **11.09 발표**

발표자 : 김가희

팀장 : 김현성

팀원 : 김가희, 정영훈

Bicycle Application

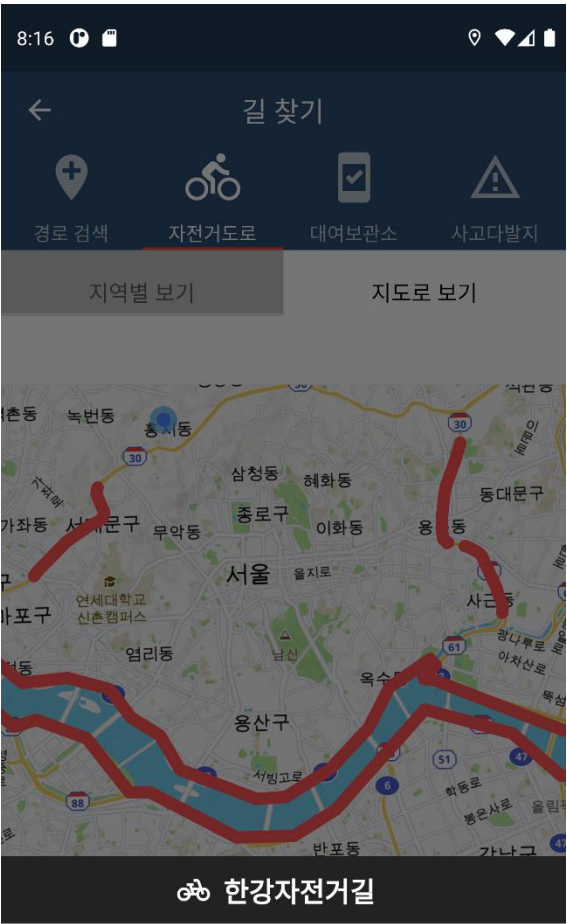
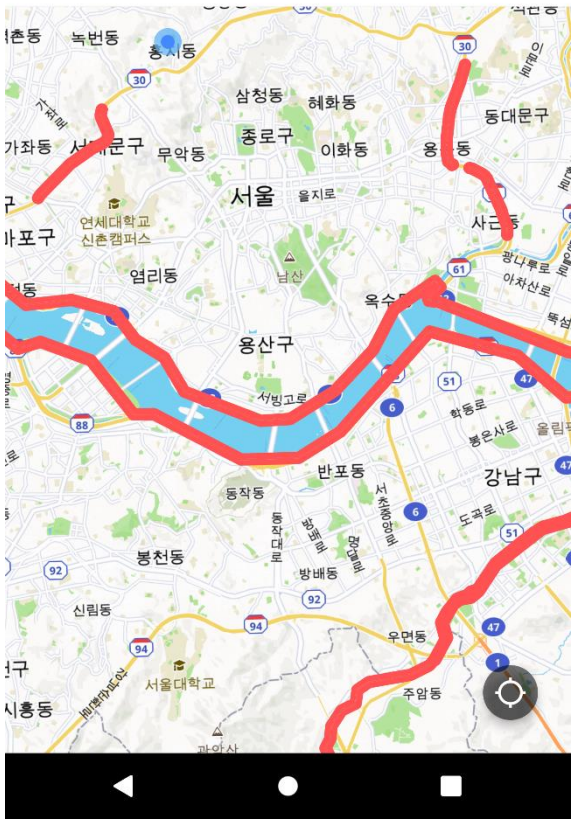
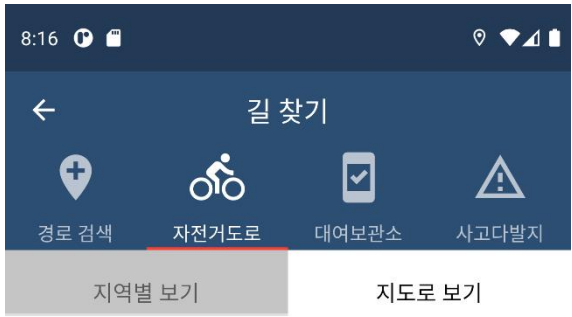
자전거 네비게이션

자전거 네비게이션 기능을 탑재한

자전거 전용 애플리케이션

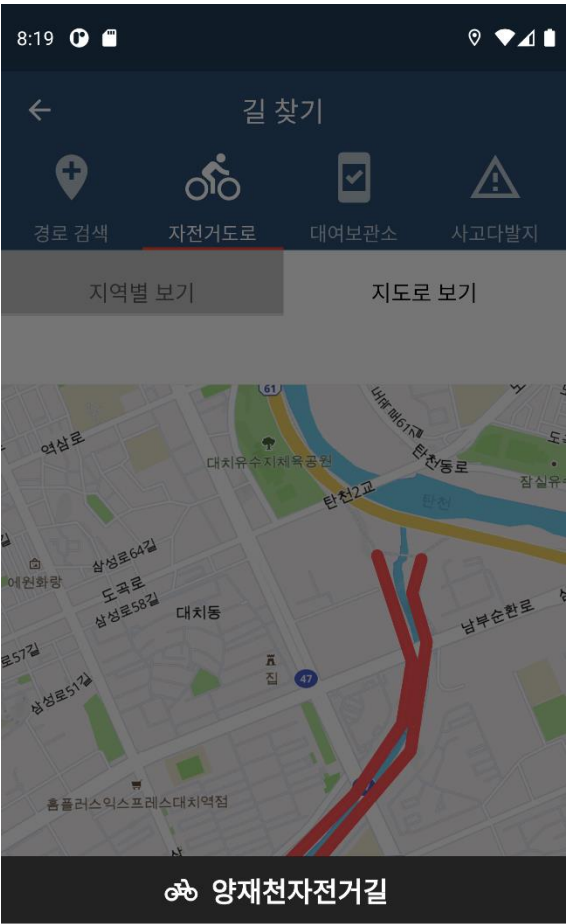


자전거 도로 표시



전호대교 ~ 팔당대교

총 거리 103km | 소요시간 약 6시간 52분

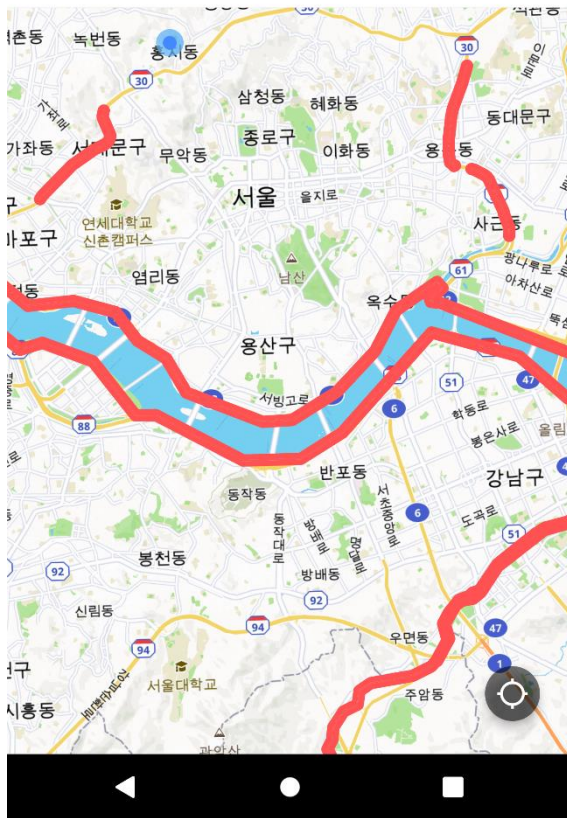
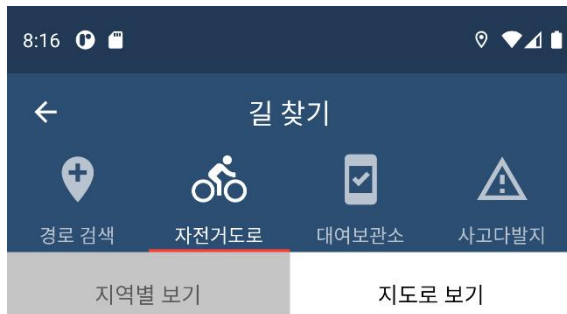


대치교 ~ 과천중앙공원

총 거리 19km | 소요시간 약 1시간 16분



자전거 도로 표시

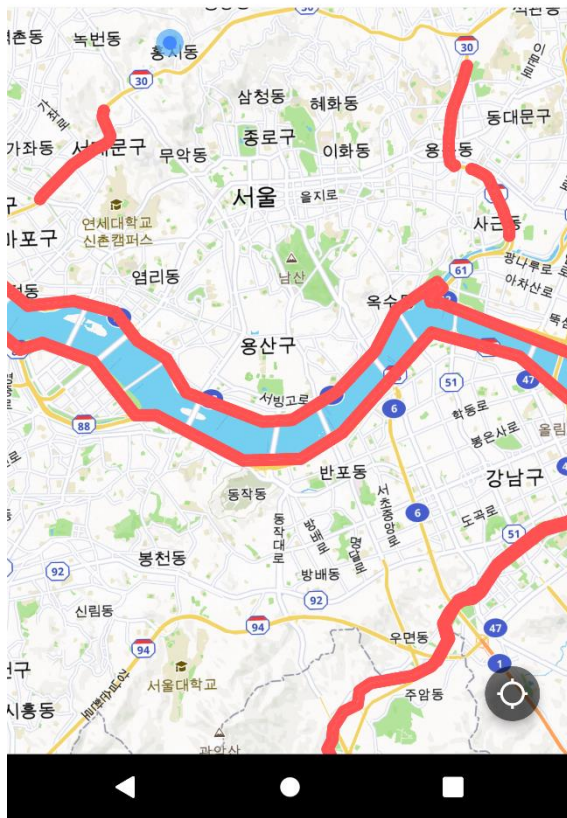
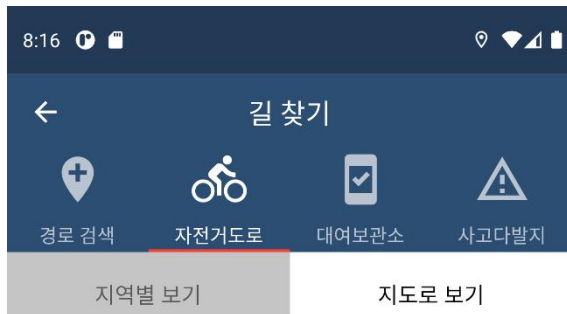


```
List<LatLng> path = [];  
List<List<LatLng>> _path = [];  
late TaggedPolyline tagpolyline;  
List<TaggedPolyline> tagpolylines = [];  
List<List<List<double>>> points = [  
  [  
    [37.57194333901629, 127.04112402679019],  
    [37.5561451814551, 127.05126534255436]  
  ],  
  [  
    [37.58763104838888, 126.93563995786636],  
    [37.56781543114469, 126.91648342338715]  
  ],  
  [  
    [37.59881040240862, 126.79958536259103],  
    [37.52483027168649, 127.10243133586648],  
    [37.56213369187951, 127.12819667163534],  
    [37.56800785620553, 127.1526071686583],  
    [37.5829667245429, 127.19582768274562],  
    [37.553787420381944, 127.22148904660556],  
    [37.542377513054426, 127.23831767806406]  
  ],  
  [  
    [37.428785293781544, 126.99356409669875],  
    [37.45917094095935, 127.02632487461509],  
    [37.45926168828694, 127.02673282355559],
```

flutter_map
tappable_polyline
패키지 이용

=> Polyline에
Tag 기능과 onTap 기능이
추가된 플러그인

자전거 도로 표시

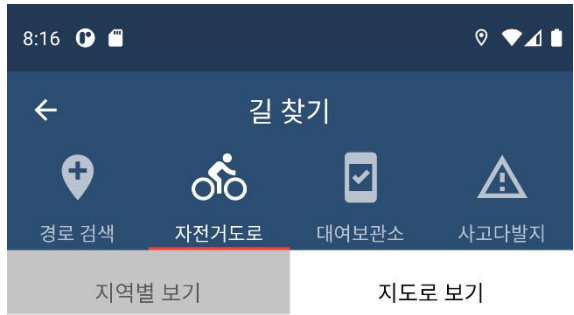


```
loadPolylines() async {  
  for (int pos=0; pos<=6; pos++){  
    DirectionsApiResponse response = await mapbox.directions.request(  
      profile: NavigationProfile.CYCLING,  
      steps: true,  
      language: 'ko',  
      coordinates: points[pos],  
    );  
  }  
}
```

```
//if (response.routes!.isNotEmpty) {  
  final route = response.routes![0];  
  final polyline = directions.Polyline.Decode(  
    encodedString: route.geometry as String,  
    precision: 5,  
  );  
  final coordinates = polyline.decodedCoords;  
  for (var i = 0; i < coordinates.length; i++) {  
    path.add(  
      LatLng(  
        coordinates[i][0],  
        coordinates[i][1],  
      ),  
    );  
  }  
  _path.add(path.toList());  
  print(_path[pos]);  
  path.clear();  
}
```

Mapbox 길 찾기 api 이용하여
Cycling 모드로
여러 개의 polyline 생성

자전거 도로 표시

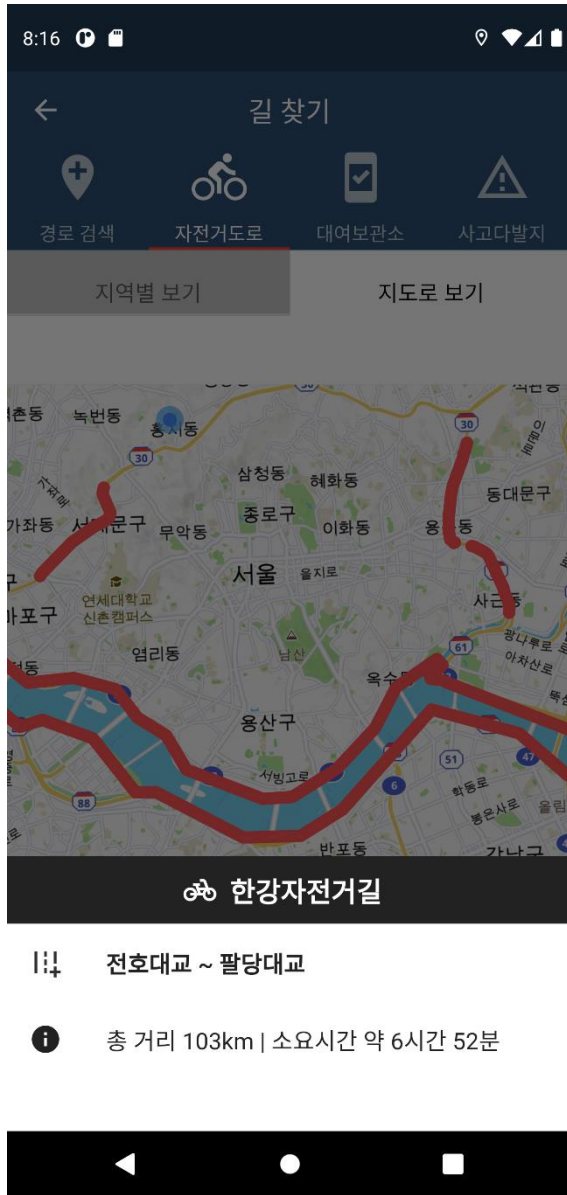


```
List<String> Addr = [
    "청계천자전거길", "홍제천서자전거길", "한강자전거길", "양재천자전거길",
    "정릉천자전거길", "한강자전거길"
];
```

```
for(int k=1; k<=6; k++){
    tagpolylines.add(TaggedPolyline(
        points: _path[k],
        tag: Addr[k],
        borderStrokeWidth: 8.0,
        color: Colors.redAccent,
        borderColor: Colors.redAccent
    ));
}
```

자전거길 위치명을 List로 생성 후
해당 polyline의 tag에 차례대로 추가

자전거 도로 표시



```
new TappablePolylineLayerOptions(  
    polylineCulling: false,  
    polylines: tagpolylines,  
    onTap: (polylines, tapPosition) => {  
        _showModalBottomSheet(polylines.map((e) => e.tag).toString())  
    }  
) // TappablePolylineLayerOptions
```

Polyline을 tap했을 때 polyline의 tag 정보를 showModalBottomSheet 함수로 넘겨주면 함수에서 문자열을 다듬어서 출력

```
void _showModalBottomSheet(String tag){  
    String _tag = tag.substring(1, tag.lastIndexOf(''));  
    for (int j=0; j<=6; j++){  
        if (_tag == Addr[j]){  
            AddrN = ListAddrN[j];  
            distance = Ldistance[j].toString();  
            if (Ltime[j] >= 60){  
                int temp=0;  
                temp = (Ltime[j] / 60).toInt();  
                time = temp.toString() + '시간 ' + (Ltime[j] - temp*60).toString();  
            }  
            else {  
                time = Ltime[j].toString();  
            }  
        }  
    }  
}
```

THANK YOU