

설계과정 1

- 안드로이드 스튜디오를 통해 어플리케이션 최소 환경 구성
- 카카오 맵 api를 어플리케이션 페이지에 탑재
- 카카오 맵 api polyline 표기 메인에 코딩 (임의의 좌표 이용)

kakao api 맵뷰 탑재

polyline 표기 매서드 실행

안드로이드 어플리케이션

설계과정 2

- 실제 등산로 좌표값 데이터를 통해 등산로 그리기
- polyline그리기 이용.
- 등산로 api데이터를 직접 옮겨 입력하여 작업 (오른쪽 그림과 같은 데이터를 직접 복사)

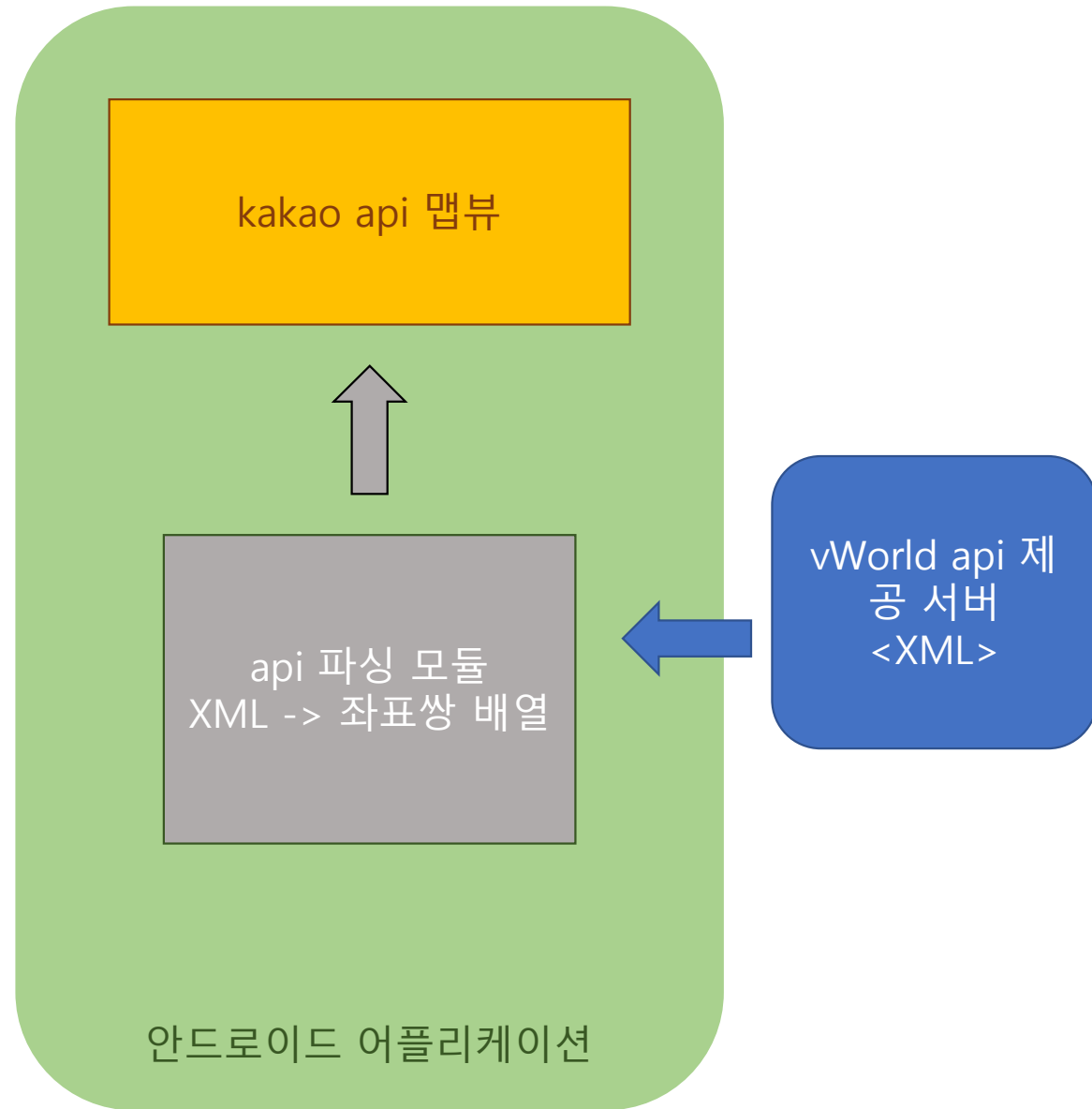
```
MapPolyline polyline = new MapPolyline();
polyline.setLineColor(Color.argb(128, i*25, 50, 0));
polyline.setTag(i);
polyline.addPoint(MapPoint.mapPointWithGeoCoord(좌표값))
mapView.addPolyline(polyline)
mapView.fitMapViewAreaToShowAllPolylines();
```

위와 같은 내용의 코드들로 좌표 값에 해당하는 polyline 생성이 가능.

```
▼<gml:LineString srsName="epsg:4326">
  <gml:posList dimension="2">126.96469024457357 37.44317753410403 126.96491330252192 37.44294363378588 126.96493385667065
  37.44292284663924 126.9649442596329 37.44291273348717 126.96495479376809 37.44290292381797 126.96496550990969
  37.44289354539154 126.96497647545544 37.44288477011631 126.96498777873121 37.44287682657815 126.96499953402206
  37.44287001497423 126.96501188706617 37.44286472424007 126.96502358549073 37.4428614936006 126.96503560385777
  37.44285927058872 126.96505939413716 37.442855580699934 126.9650705306455 37.44285300049739 126.96508067937862
  37.442849136691336 126.96509010278409 37.442843069828804 126.96509829908773 37.44283529664343 126.96510557957207
  37.44282624120537 126.96511218724255 37.44281623473148 126.96511831101346 37.442805533470604 126.96512409832343
  37.4427943345283 126.96512966624017 37.442782789506616 126.96515155591493 37.44273519527088 126.96515737886465
  37.442723332680984 126.96516357147033 37.44271163548253 126.96517030649456 37.44270019881644 126.96517781360718
  37.44268914305173 126.96518616112967 37.442678723686676 126.96519580477081 37.442668303434964 126.96520765171226
  37.44265692607521 126.9652211001175 37.44264494053617 126.96523546217844 37.442632764749554 126.96525005555763
  37.44262081870269 126.96527724736244 37.44259931461997 126.96530497748249 37.44257785645978 126.96530749711991
  37.442579411682345 126.9653131237 37.44258427537501 126.96532091211327 37.44259204550997 126.96532985001386
  37.44260234405079 126.96533889437544 37.44261480396567 126.96534699473159 37.44262880843852 126.9653539409661
  37.442643930257816 126.96535964349624 37.44265948723941 126.96536407660346 37.44267477405269 126.96536723460599
  37.44268907817571 126.96537821626585 37.44276538929308 126.96538277453682 37.44279048243219 126.9653856104329
  37.44280286153002 126.96538900949342 37.442815069799835 126.96539316947823 37.442827046177 126.96539834452456
  37.44283871199804 126.96540461393981 37.44284981885665 126.96541183271243 37.442860632436606 126.96541972654188
  37.442871123424498 126.9654280738902 37.44288168832051 126.96546253393552 37.44292293605355 126.9654705918224
  37.44293329006845 126.9654780347701 37.44294373503859 126.96548459348432 37.442954316656326 126.9654899233698
  37.44296509052768 126.96549361462189 37.44297612555671 126.96549535755328 37.44298814055083 126.96549530436923
  37.44300044005231 126.9654939736701 37.443012941003744 126.9654917794667 37.44302557609102 126.96548320136999
  37.4430637477489 126.96548060702006 37.44307639395952 126.96547863329833 37.44308890875183 126.96547754106474
  37.44310122091318 126.9654780753897 37.44311324132401 126.96548049638615 37.44312485680334 126.96548536810246
  37.44313601582301 126.96549223434026 37.4431467698392 126.96550052584702 37.44315723581895 126.9655097893112
  37.44316750648882 126.96554984255249 37.44320803137855 126.9655591251614 37.44321833927554 126.96556745516546
  37.443228845526065 126.96557439221714 37.443239660568764 126.96557938658601 37.44325089729661 126.9655819938029
  37.4432625701525 126.96558270500475 37.44327465938165 126.96558208890293 37.44328704752372 126.96558061399317
  37.443299639154866 126.96557488994263 37.44333786553593 126.96557377569901 37.44335051593375 126.9655737528494
  37.443362982051994 126.96557535615153 37.44337515425502 126.96557889994024 37.443386588003996 126.96558416336055
  37.443397609025794 126.96559083682628 37.44340817367565 126.96559872738892 37.4434181925663044 126.96560774972161
  37.443427522318764 126.9656179213278 37.44343595460947 126.96562936210348 37.443443202177725 126.9656425830008
  37.4434491394469 126.96565667415298 37.44345386823314 126.96567165530894 37.44345772310758 126.96568710551519
  37.44346096757974 126.96574823459379 37.443471621216986 126.96576174172651 37.44347450322028 126.96577358599463
  37.44347782525935 126.96578299900413 37.443481830986094 126.96578893839511 37.44348683802568 126.96579049134255
  37.443493133160764 126.9657882571055 37.44350058026416 126.96578188521235 37.44351007689358 126.96577191289431
```

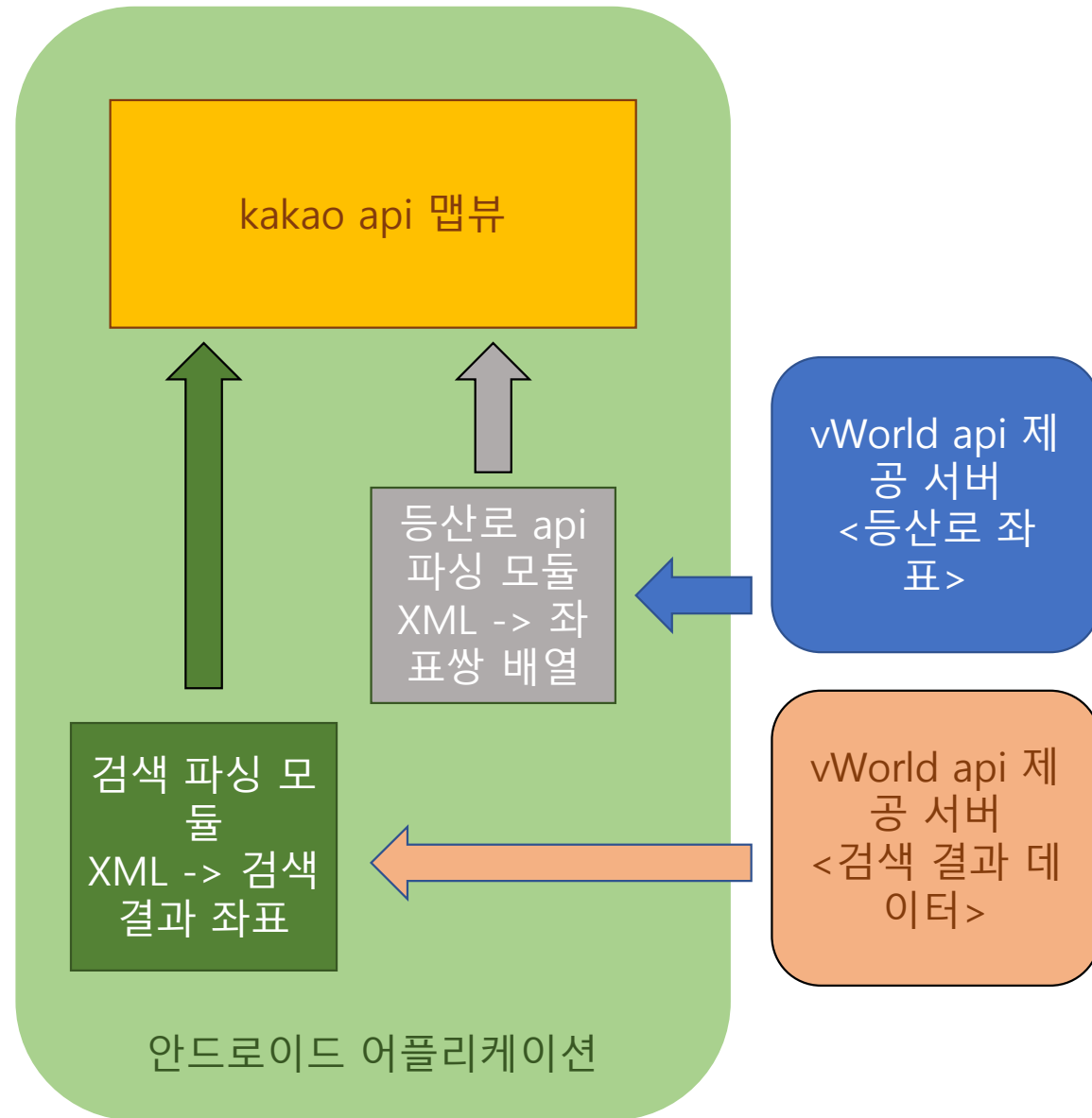
설계과정 3

- vWorld 국토교통부 등산로 2D 좌표 데이터 파싱 모듈 구현



설계과정 4

- 검색을 통해 등산로 표기 구현
- 검색 스트림 구현
- 검색 결과 xml 파싱 구현
- 한 클래스로 묶어서 모듈화



설계과정 5

- 등산로 좌표 데이터 api 직접 사용 대신 데이터 가공 저장후 사용으로 전환
- 검색 기능 자체는 유지하고 콘텐츠에 필요한 데이터를 가공 저장 후 사용으로 전환
- 가공 저장 후 사용 하기 위해 DB를 참조하여 맵뷰 상에 등산로를 표기하도록 전환

