감염병 전파 예측 시스템

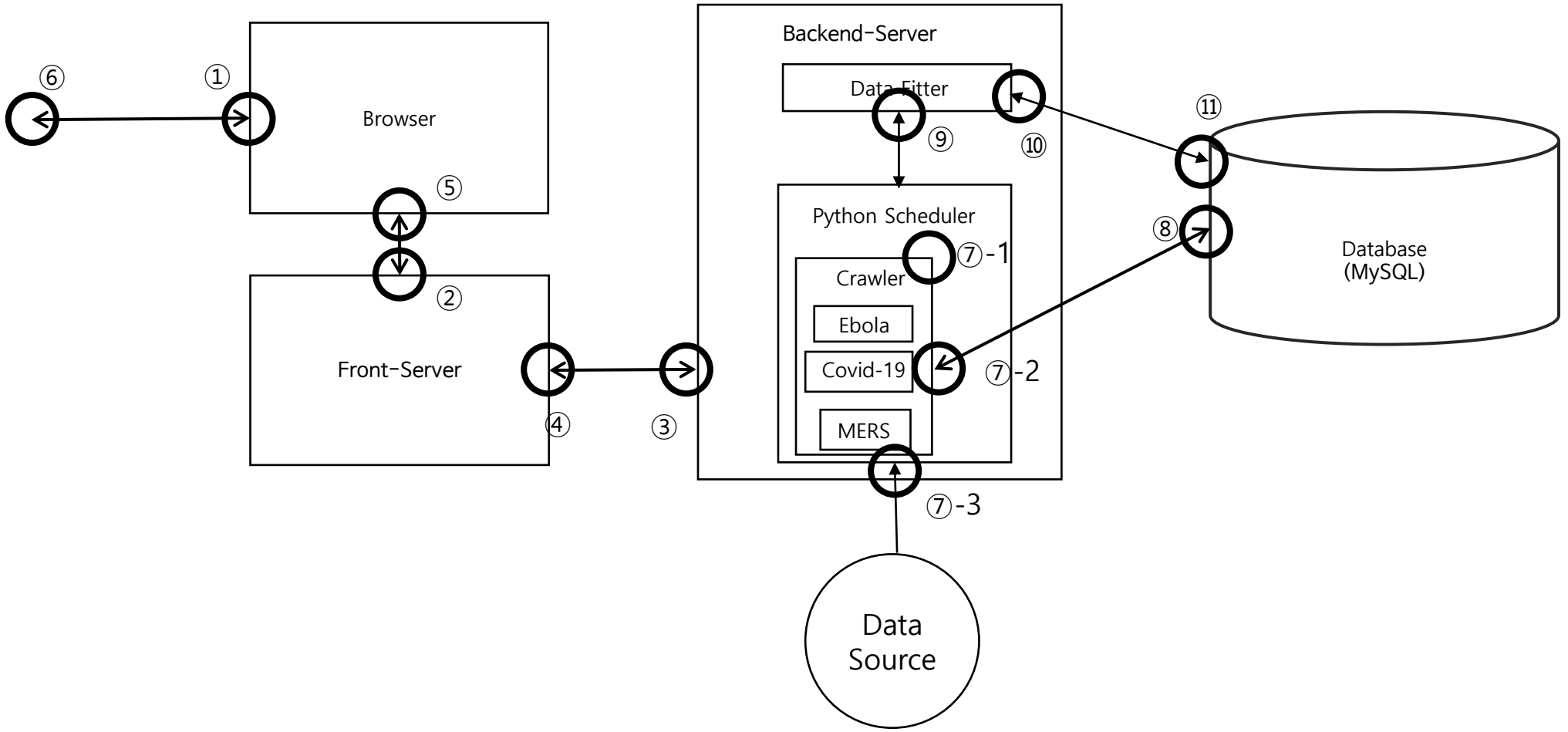
졸업프로젝트1

고길재

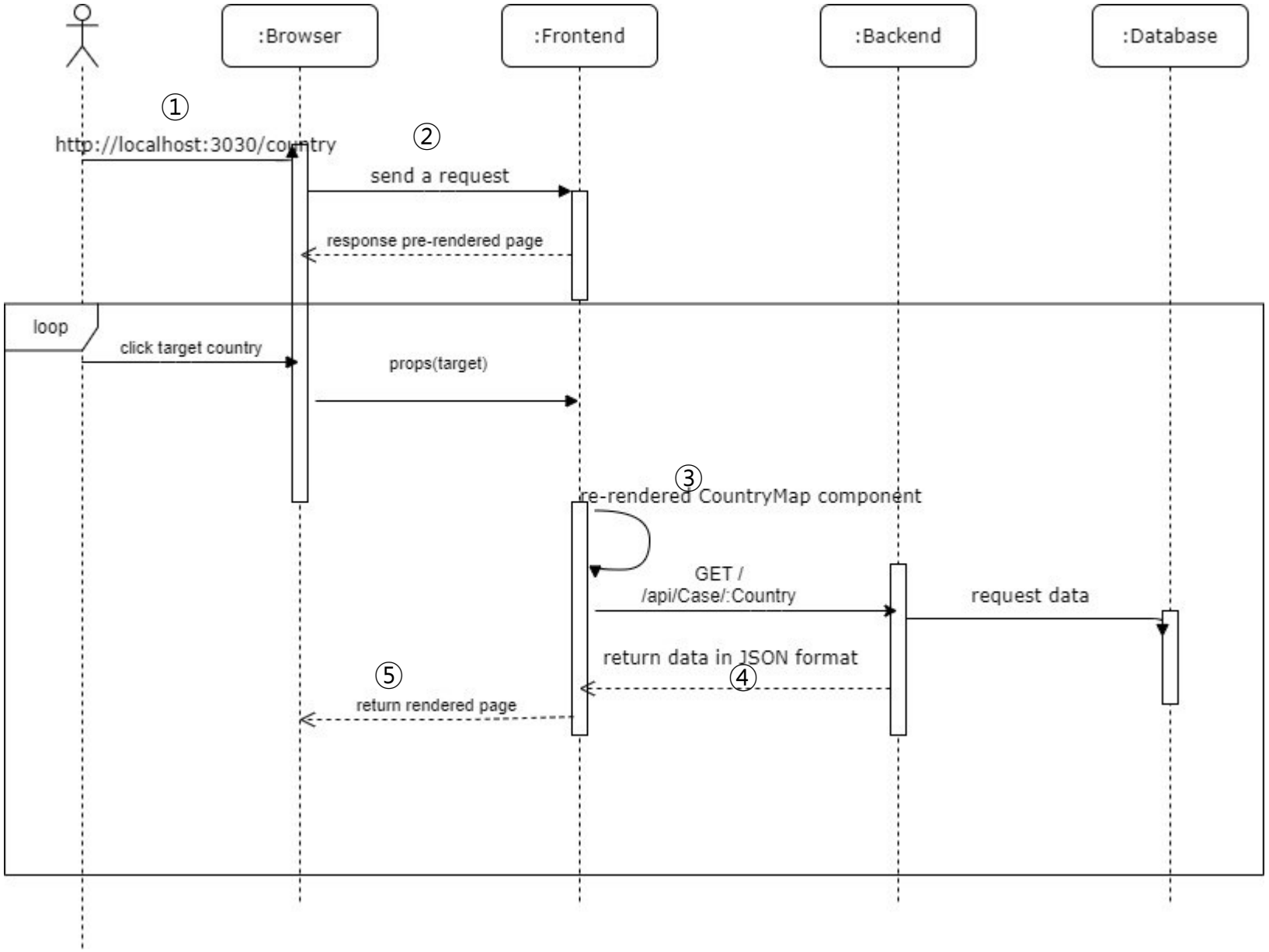
김민혁

조수빈

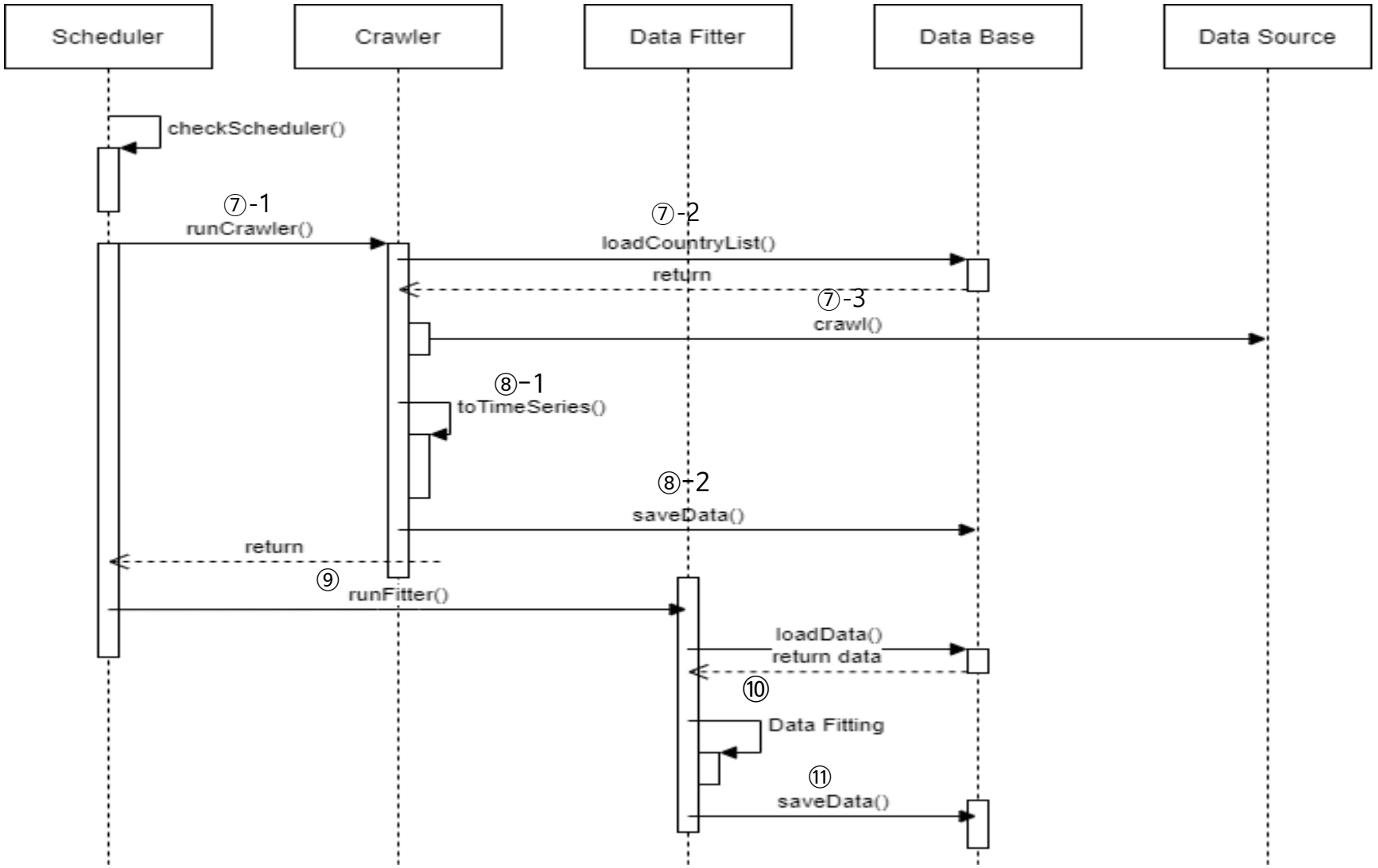
Architecture Diagram



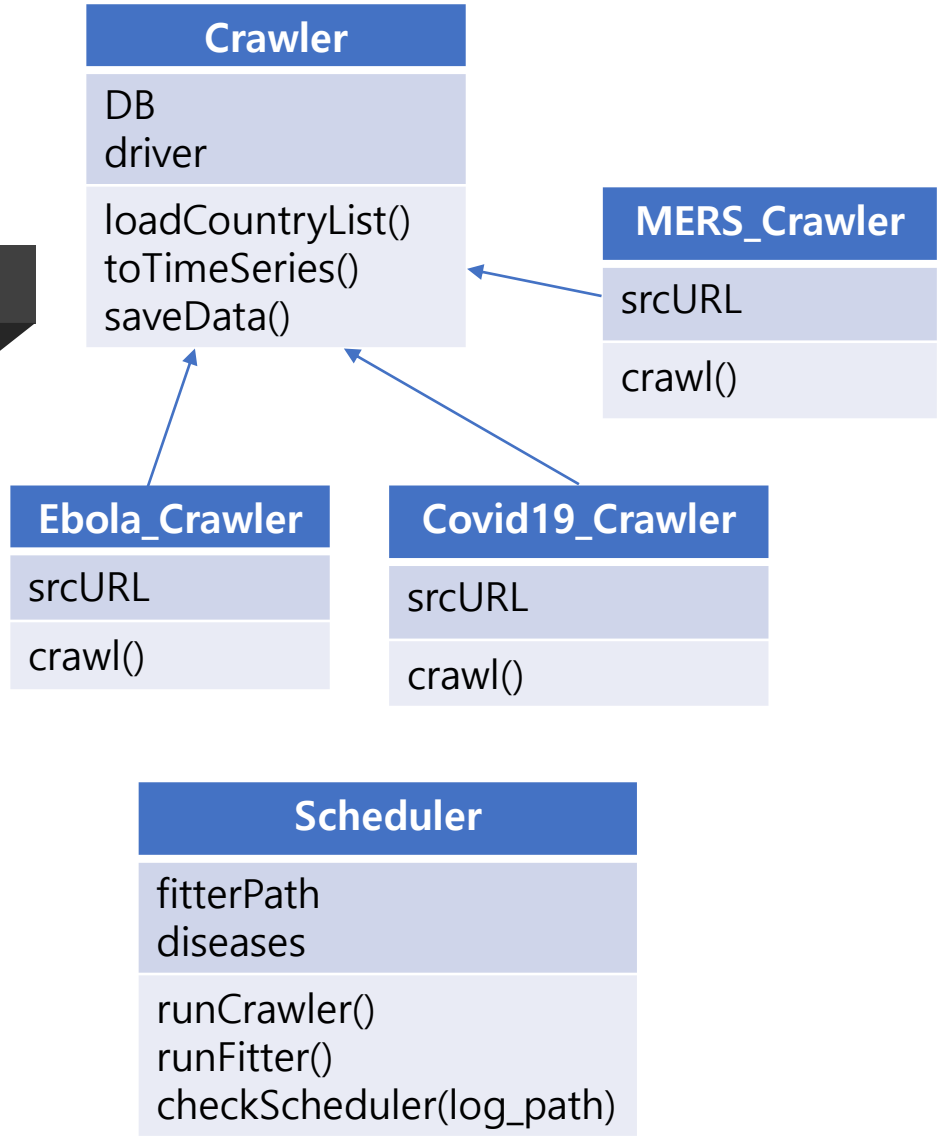
Sequence Diagram_01



Sequence Diagram_02



Class Diagram_01



Crawler

- loadCountryList : DB로부터 크롤링할 국가 목록을 가져온다.
- toTimeSeries() : 크롤링한 결과를 시계열 데이터로 변환한다.
- saveData() : 시계열로 변환된 데이터를 DB에 저장한다.

DISEASE-NAME_Crawler

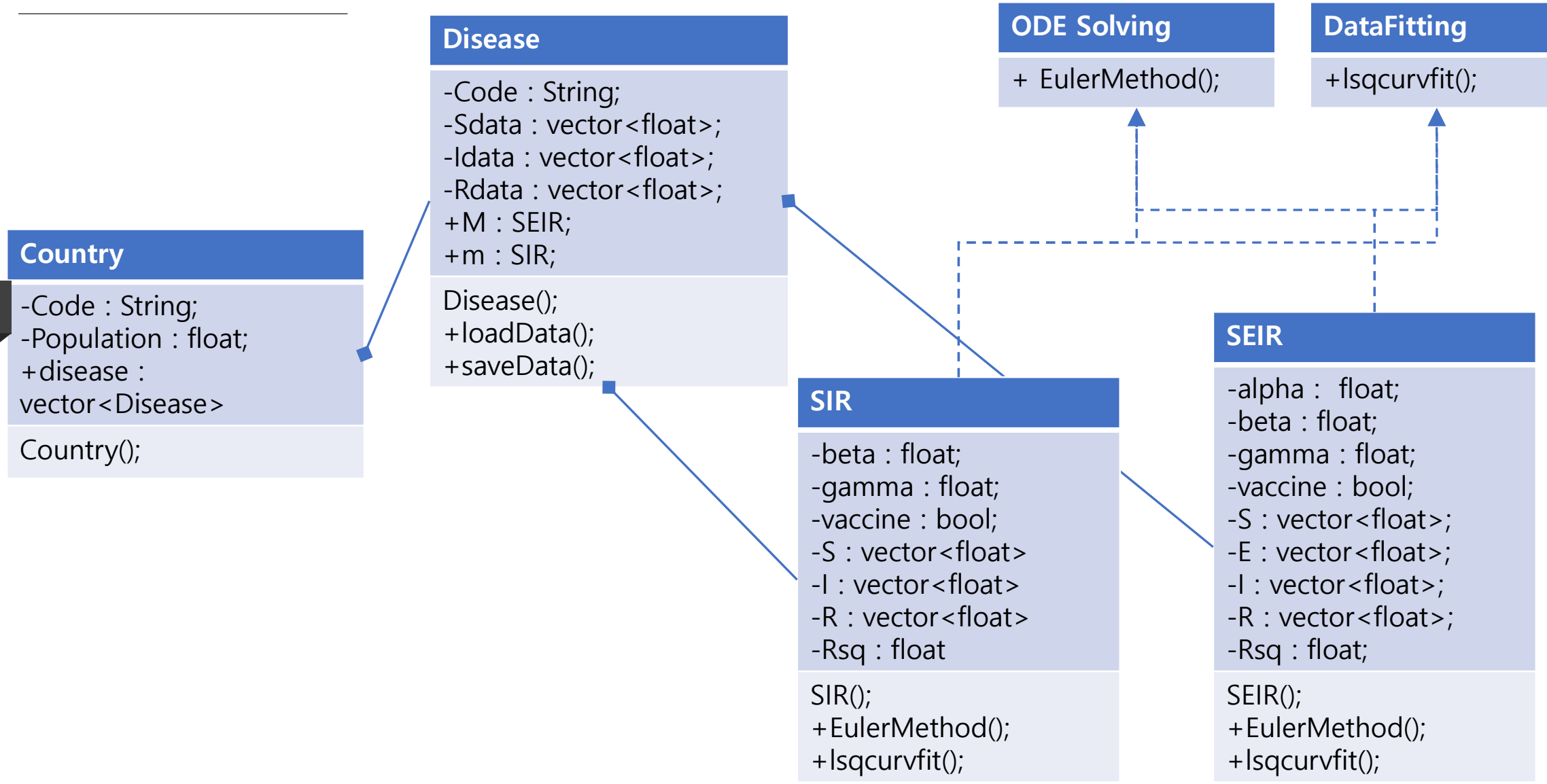
- crawl() : 해당 질병의 데이터 소스로부터 (국가명, 확진자 수, 사망자 수, 회복 인원) 데이터를 크롤링한다.

Scheduler

- runCrawler() : diseases에 있는 질병에 해당하는 크롤러를 실행한다.
- runFitter() : Data Fitting Engine을 실행한다.
- checkScheduler(log_path) : 스케쥴러가 24시간에 1번씩 실행되는지 확인한다.

01
02
03
04
05
06
07
08
09
10

Class Diagram_02



Country

-Code : String;
-Population : float;
+disease : vector<Disease>

Country();

Disease

-Code : String;
-Sdata : vector<float>;
-Idata : vector<float>;
-Rdata : vector<float>;
+M : SEIR;
+m : SIR;

Disease();
+loadData();
+saveData();

Country

Country(String counCode, float pop);

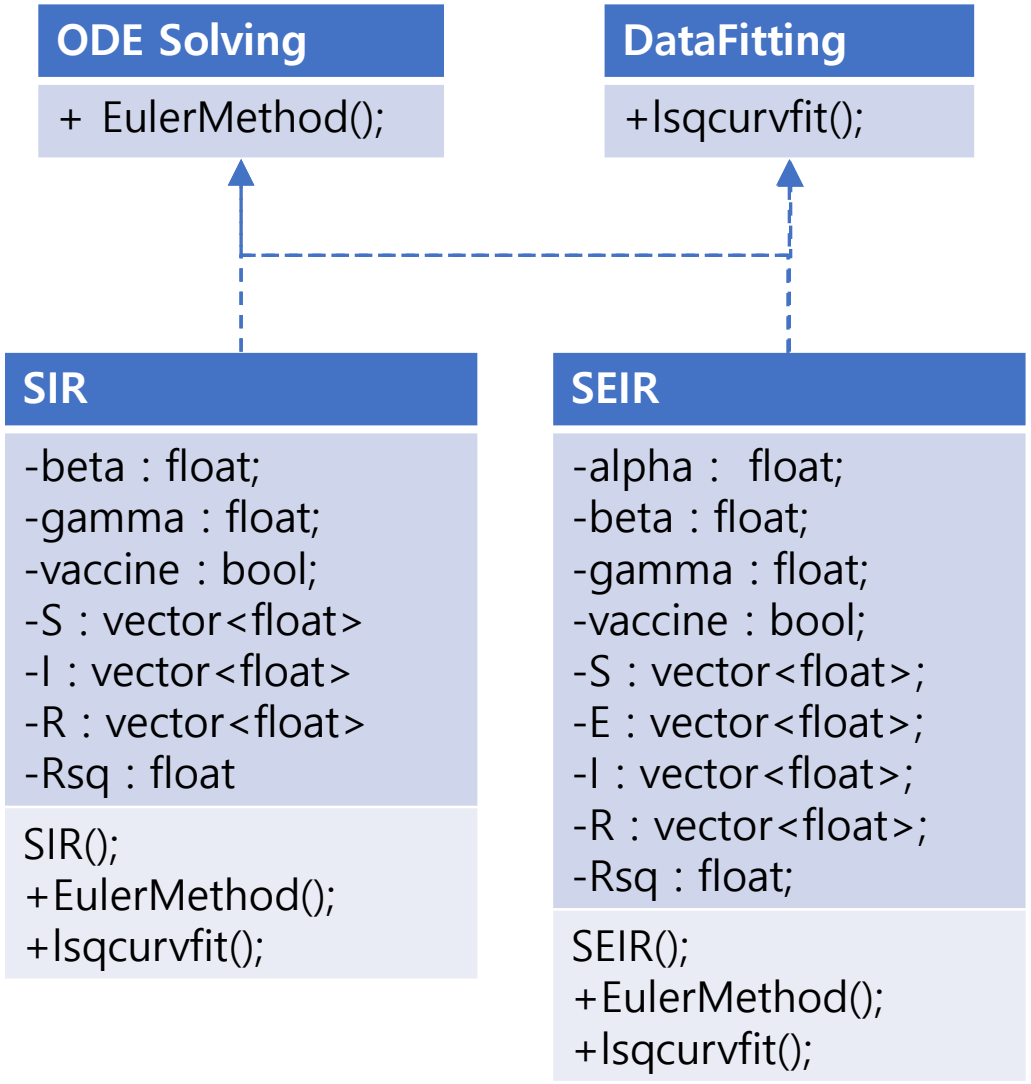
1. Country 클래스는 main함수에서 생성된다.
2. Main에서 생성 당시에 데이터 베이스에 접근하여 국가 코드를 인식하고, 코드와 인구수를 읽어 Country를 실행한다.
3. Country는 데이터 베이스의 질병 테이블에 접근하여 질병 코드를 인식하고 그에 해당하는 Disease class를 생성 갖게 된다.

Disease

Disease(String counCode, String disCode, float pop);

1. Disease 클래스는 Country클래스 생성 당시에 생성되며 Country 클래스가 갖는다.
2. 데이터 베이스에서 얻은 국가 코드와 질병 코드를 Primary Key로 데이터베이스의 Case테이블에 접근하여 데이터를 load한다.
3. 로딩된 데이터를 가공한다. 데이터 베이스의 데이터와 Disease에 저장된 데이터의 관계는 다음과 같다.

$$Sdata = pop - Infectious - Death - Recovery$$
$$\mathrm{I}data = Infectious - Death - Recovery$$
$$\mathrm{R}data = Death + Recovery$$



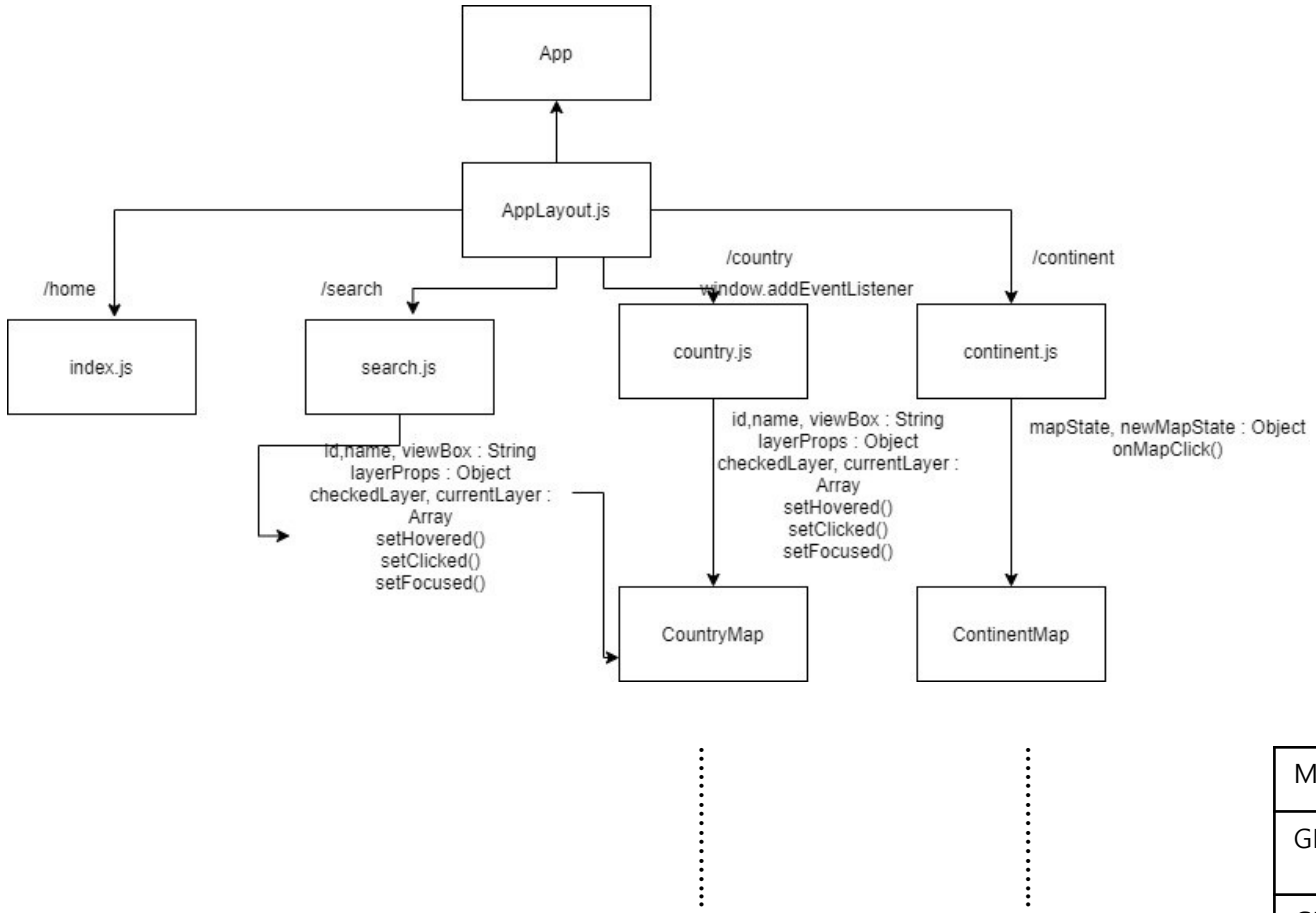
SIR & SEIR

SIR(String disCode, vector⟨float⟩ SD, vector⟨float⟩ ID, vector⟨float⟩ RD);

SEIR(String disCode, vector⟨float⟩ SD, vector⟨float⟩ ID, vector⟨float⟩ RD);

1. SIR과 SEIR 클래스는 Disease 클래스 생성 중에 생성된다.
2. 주어진 disCode를 키로 데이터베이스의 질병테이블에 접근하여 질병의 백신 개발 여부와 잠복기간을 불러온다.
3. 주어진 SD, ID, RD(Data Base에서 불러온 데이터)를 기반으로 lsqcurvfit()을 실행한다.
4. lsqcurvfit()에서는 Gauss-Newton Algoritm이 실행된다. 이는 데이터와 예측 자료의 차이를 이용하여 Residual을 구하고 Parameter의 변화량에 따른 Residual의 변화량을 구하기 위해서 ODE Solver가 지속적으로 실행된다.
5. ODE Solver는 Forward Euler Method를 실행한다.
6. lsqcurvfit()이 완료되면 다시 ODE Solver를 이용하여 예측 데이터를 작성 vector⟨float⟩에 저장하며, 이를 데이터 베이스에 저장한다.

Class Diagram_03



GET | http://localhost:8080/api/country....

-AppLayout은 레이아웃 컴포넌트로 모든 페이지에 공통적인 부분이 미리 정의되어있다.(Menu ...)
-index.js, search.js, continent.js, country.js는 웹의 pages
-ContinentMap : svg로 되어있는 world map을 클릭 시, 클릭된 ,continent의 id를 통해 클릭된 대륙의 state가 변하며 해당하는 대륙을 다른 색으로 보여준다.
(아프리카 대륙 클릭 시 -> this.mapState : {af : "map-selected"})
-CountryMap : svg로 되어있는 worldmap을 클릭 시 클릭된 country의 id를 통해 클릭된 나라의 이름을 띄워준다.
이 컴포넌트 안에서 backend 서버와 통신하여 해당 국가의 데이터들을 불러와 웹에 전시하게 된다.

백엔드로부터 router 통해 데이터 전달

Method	Routes	Description
GET	/api/country/	DB에 저장된 모든 국가의 데이터
GET	/api/case/:country_name	해당 국가의 전염병 시작-종식날짜까지의 데이터

Traceability Matrix