

# 상세 설계서

## 오픈소스 기여

-모듈 관계도

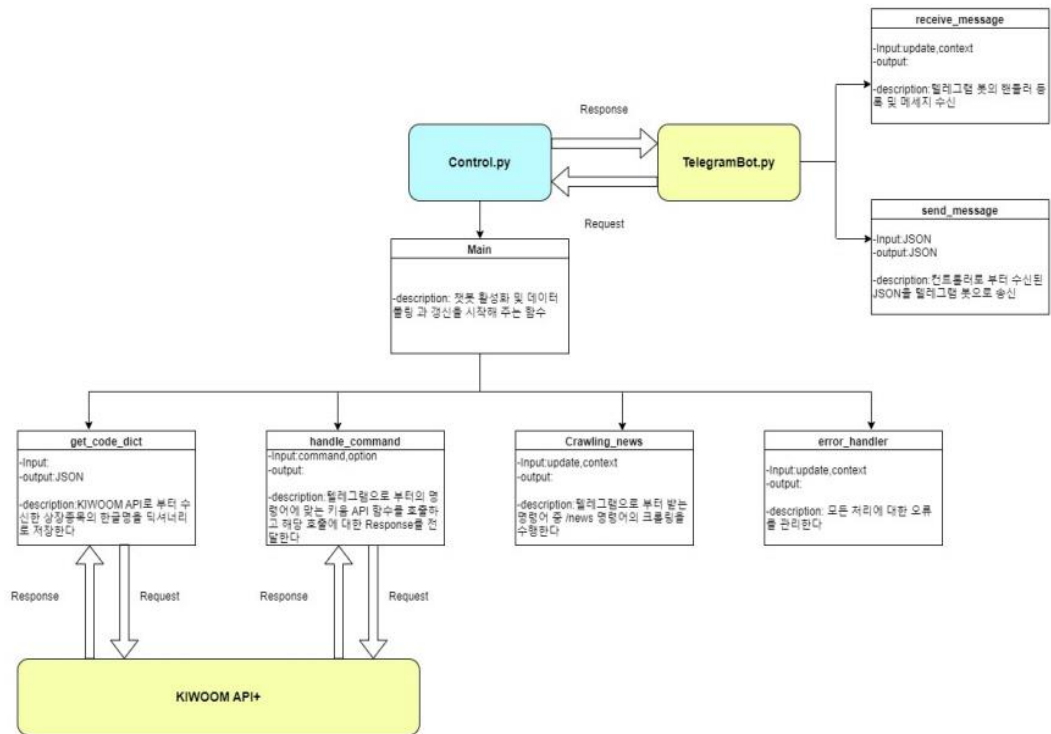
-API 상세 설계

32173154 이승현

32172086 석홍준



## 모듈 관계도



## Command) /info : 실시간 시세 데이터 반환

### GetCommRealData

함수 원형	BSTR GetCommRealData(LPCTSTR strCode, long nFid)
설명	실시간 시세 데이터 반환
입력값	strCode – 종목 코드 nFid – 실시간 아이템
반환값	수신 데이터
비고	현재가 출력 – openApi.GetCommRealData("039490", 10) strCode는 OnReceiveRealData 첫번째 매개변수를 사용

### OnReceiveRealData

함수 원형	void OnReceiveRealData(LPCTSTR sJongmokCode, LPCTSTR sRealType, LPCTSTR sRealData)
설명	실시간데이터를 받은 시점을 반환
입력값	sJongmokCode – 종목코드 sRealType – 리얼타입 sRealData – 실시간 데이터전문
반환값	X
비고	X

## Command) /search : 조건 검색의 조건 리스트 받아오기

### GetConditionNameList

함수 원형	BSTR GetConditionNameList()
설명	조건검색 조건명 리스트를 받아옴
입력값	
반환값	조건명 리스트(인덱스^조건명)
비고	조건명 리스트를 구분(',') 사용하여 받아온다. Ex) 인덱스1^조건명;인덱스2^조건명;인덱스3^조건명...

### SendCondition

함수 원형	BOOL SendCondition(LPCTSTR strScrNo, LPCTSTR strConditionName, int nIndex, int nSearch)
설명	조건검색 종목조회TR송신
입력값	LPCTSTR strScrNo : 화면번호 LPCTSTR strConditionName : 조건명 int nIndex : 조건명인덱스 int nSearch : 조회구분(0:일반조회, 1:실시간조회, 2:연속조회)
반환값	성공 1, 실패 0
비고	단순 조건식에 맞는 종목을 조회하기 위해서는 조회구분을 0으로 하고, 실시간 조건검색을 하기 위해서는 조회구분을 1로 한다. OnReceiveTrCondition으로 결과값이 온다. 연속조회가 필요한 경우에는 응답받는 곳에서 연속조회 여부에 따라 연속조회를 송신하면된다.

### OnReceiveTrCondition

함수 원형	void OnReceiveTrCondition(LPCTSTR sScrNo, LPCTSTR strCodeList, LPCTSTR strConditionName, int nIndex, int nNext)
설명	조건검색 조회응답으로 종목리스트를 구분자(",")로 붙어서 받는 시점
입력값	LPCTSTR sScrNo : 종목코드 LPCTSTR strCodeList : 종목리스트(",")로 구분 LPCTSTR strConditionName : 조건명 int nIndex : 조건명 인덱스 int nNext : 연속조회(2:연속조회, 0:연속조회없음)
반환값	X
비고	X

**Command) /alarm : 사용자가 설정한 조건이 만족되었을 경우 알림을 전송**

#### SendCondition

함수 원형	BOOL SendCondition(LPCTSTR strScrNo, LPCTSTR strConditionName, int nIndex, int nSearch)
설명	조건검색 종목조회TR송신
입력값	LPCTSTR strScrNo : 화면번호 LPCTSTR strConditionName : 조건명 int nIndex : 조건명인덱스 int nSearch : 조회구분(0:일반조회, 1:실시간조회, 2:연속조회)
반환값	성공 1, 실패 0
비고	단순 조건식에 맞는 종목을 조회하기 위해서는 조회구분을 0으로 하고, 실시간 조건검색을 하기 위해서는 조회구분을 1로 한다. OnReceiveTrCondition으로 결과값이 온다. 연속조회가 필요한 경우에는 응답받는 곳에서 연속조회 여부에 따라 연속조회를 송신하면 된다.

#### SendConditionStop

함수 원형	Void SendConditionStop(LPCTSTR strScrNo, LPCTSTR strConditionName, int nIndex)
설명	조건검색 실시간 중지TR을 송신
입력값	LPCTSTR strScrNo : 화면번호 LPCTSTR strConditionName : 조건명 int nIndex : 조건명인덱스
반환값	X
비고	해당 조건명의 실시간 조건검색을 중지하거나, 다른 조건명으로 바꿀 때 이전 조건명으로 실시간 조건검색을 반드시 중지해야한다. 화면 종료시에도 실시간 조건검색을 한 조건명으로 전부 중지해줘야 한다.

- /search에도 동일하게 적용

**Command) /news : 사용자가 입력한 종목에 관련된 뉴스 정보를 출력함**

## **사용 모듈**

```
from selenium import webdriver
```

```
from bs4 import BeautifulSoup
```

## **chrome 브라우저를 사용한 데이터 크롤링**

```
# 크롬 드라이버 경로 설정
```

```
driver_path = "/path/to/chromedriver"
```

```
# 크롬 드라이버 실행
```

```
driver = webdriver.Chrome(driver_path)
```

```
# 네이버 주식 뉴스 URL
```

```
url = "https://finance.naver.com/news/news_list.nhn?mode=RANK&date=20220418"
```

```
# 페이지 로드
```

```
driver.get(url)
```

```
# 페이지 소스코드 가져오기
```

```
html = driver.page_source
```

```
# BeautifulSoup 객체 생성
```

```
soup = BeautifulSoup(html, "html.parser")
```

```
# 뉴스 제목 추출
```

```
titles = soup.select("ul.realtimeNewsList dl dt a")
```

## 에러처리

리턴 코드		설명
0	OP_ERR_NONE	정상처리
-10	OP_ERR_FAIL	실패
-11	OP_ERR_COND_NOTFOUND	조건번호 없음
-12	OP_ERR_COND_MISMATCH	조건번호와 조건식 틀림
-101	OP_ERR_CONNECT	서버접속 실패
-105	OP_ERR_INPUT	함수입력값 오류
-203	OP_ERR_NO_DATA	데이터 없음
-205	OP_ERR_DATA_RCV_FAIL	데이터 수신 실패
-300	OP_ERR_ORD_WRONG_INPUT	입력값 오류
-500	OP_ERR_ORD_SYMCODE_EMPTY	종목코드 없음

## 종목 코드와 종목 명 dict처리

### GetCodeListByMarket

함수 원형	BSTR GetCodeListByMarket(LPCTSTR sMarket)
설명	시장 구분에 따른 종목 코드 리스트 반환
입력값	sMarket – 시장구분
반환값	종목코드 리스트, 종목간 구분은 ';'이다.
비고	sMarket – 0:장내, 3:ELW, 4:뮤추얼펀드, 5:신주인수권, 6:리츠, 8:ETF, 9:하이일드펀드, 10:코스닥, 30:K-OTC, 50:코넥스(KONEX)

### GetMasterCodeName

함수 원형	BSTR GetMarsterCodeName(LPCTSTR strCode)
설명	종목 코드 한글명 반환
입력값	strCode – 종목코드
반환값	종목한글명
비고	장내외, 지수선물옵션, 주식선물옵션 검색 가능

# Telegram API

**Telegram – Updater, MessageHandler, Filters 모듈 사용**

**Telegram.Bot(token)** : 챗봇 토큰으로 봇 생성

Token: string

**Updater(token, use\_context)** : 업데이터 생성

Token: string, use\_context: boolean

**Updater.dispatcher** : 디스패처 객체 생성

**updater.start\_polling()** : 지속적으로 채팅방 채팅 받아옴

**Update.message** : 챗봇에서 넘어온 메시지의 값을 받아옴

Return: list

**Bot.sendMessage(chat\_id, text)** : 채팅을 보낸 사람의 id에 text 채팅 값을 전송

Chat\_id: string, text: string

**MessageHandler(Filters.text, handler)**

**Dispatcher.add\_handler(echo\_handler)** : 디스패처에 핸들러 등록