

주식 가격 예측 사이트 (최 종 보 고 서)

2020. 12. 11

학년/반:	2학년 C반
학번:	201944072
성명 :	이성희
담당교수	조규철

인하공업전문대학
컴퓨터정보과



<목 차>

I. 프로젝트 개요	7
1. 프로젝트 목적	7
2. 프로젝트 필요성	7
3. 프로젝트 내용	7
II. 프로젝트 분석	7
1. 관련 시스템 현황 조사 및 벤치마킹	7
2. 요구사항 분석	9
3. 업무 정의	10
III. 프로젝트 설계	12
1. 개발환경 정의	12
2. 화면 설계	13
3. 시스템 아키텍처	14
4. DB 설계	15
5. 데이터 분석 설계	16
IV. 프로젝트 구현 결과	18
1. 데이터 학습 결과	18
2. 사이트 구현 결과	20
V. 개발 일정 및 주요 이슈	24
1. 개발 일정	24
2. 주요 이슈	24
VI. 개발 일정 및 주요 이슈	24
1. 프로젝트 기대 효과	24
2. 갈무리 및 향후 계획	24

<표 목차>

<표 2-1> 요구사항 분석	7
<표 2-2> 기능 분석	8
<표 3-1> 소프트웨어 개발환경	10
<표 3-2> 시스템 개발환경	10
<표 3-3> 테이블 설계	14
<표 5-2> 개발 일정	22
<표 5-3> 개발 일정	22

<그림 목차>

<그림 2-1> 벤치마킹1	6
<그림 2-2> 벤치마킹2	6
<그림 2-3> 업무절차1	8
<그림 2-2> 업무절차2	9
<그림 3-1> 화면설계1	11
<그림 3-2> 화면설계2	11
<그림 3-3> 화면설계3	11
<그림 3-4> 화면설계4	12
<그림 3-5> 화면설계5	12
<그림 3-6> 시스템 구성도	13
<그림 3-7> ERD	13
<그림 3-8> RNN구조	14
<그림 3-9> RNN구조	14
<그림 3-10> MSE 공식	15
<그림 3-11> 주식 가격 데이터	15
<그림 3-12> windows 설정	16
<그림 4-1> 손실률	16
<그림 4-2> 손실률 그래프	17
<그림 4-3> 예측 값과 실제값 그래프	17
<그림 4-4> 로그인하지 않았을 때 메인페이지	18
<그림 4-5> 로그인했을 때 메인페이지	18
<그림 4-6> 로그인 페이지	19
<그림 4-7> 로그인 실패 시 뜨는 경고 창	19
<그림 4-8> 주식 선택 전	20
<그림 4-9> 주식 선택 후	20
<그림 4-10> 주식 선택 전	21
<그림 4-11> 주식 선택 후	21

I. 프로젝트 개요

1. 프로젝트 목적

주식투자를 하는 사람들은 보통 아침에서 점심시간에 증권계좌를 개설한 증권사의 주식 어플이나 페이지 혹은 프로그램을 통해서 주식의 가격을 예측해보고 주식을 매도할지 매수할지 결정을 하여 주식을 거래하게 된다. 주식 차트를 볼 줄 알고 경제에 관심이 많은 사람이라면 주식으로 조금이나마 수익을 볼 수 있을 것이다. 하지만 주식은 하고 싶은데 주식에 대한 지식이 없는 처음 주식 거래를 시작하는 사람들에게는 언제 주식을 매도하고 매수할지 결정하는데 어려움이 있고 또 주식 투자로 수익을 내기 힘들다. 이처럼 주식투자는 진입 장벽이 높아서 초보자는 금방 포기하게 된다. 그래서 이런 사람들을 위해 단기적 수익을 낼 수 있게 도와주는 내일의 주식 가격 예측 웹 사이트를 개발하고자 한다.

2. 프로젝트 필요성

가. 단기적 수익을 낼 수 있음

주식 가격 예측 웹사이트를 통해 사용자가 주식을 매매할지 결정하는데 도움을 주어 단기적 수익을 낼 수도 있게 도와준다.

3. 프로젝트 내용

가. 내일의 주식 가격 예측 모델

내일의 주식의 가격을 예측하는 모델을 제작하여 주식 가격을 얻을 수 있게 한다.

II. 프로젝트 분석

1. 관련 시스템 현황 조사 및 벤치마킹

최근 들어 인공지능을 이용한 주식 투자 프로그램들이 많이 나오고 있다. 대부분 주식데이터를 분석하여 자신에게 맞는 최적의 종목과 매매 타이밍을 알려주는 프로그램들이 많았다. 그 중에 가장 접하기 쉬웠던 어플리케이션 두 개를 분석해 보았다.

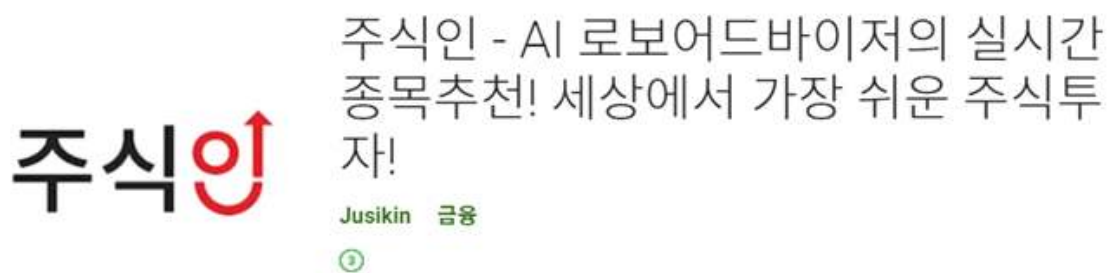
가. MK 라씨로



<그림 2-1> 벤치마킹 어플리케이션 MK라씨로

첫 번째로 인공지능 주식 투자비서 'MK라씨로'라는 어플이다. 이 어플은 인공지능을 활용하여 실시간으로 사용자들에게 적절한 매매 타이밍을 알림으로 알려준다. 또 주식 시장에서 현재 매매되고 있는 실시간 상황과 수익성이 높을 것이라고 예측되는 주식 분야들을 추천해주고 주식시장에 영향을 미칠 기사 속보를 알려준다.

나. MK 라씨로



<그림 2-2> 벤치마킹 어플리케이션 주식인

주식인은 인공지능을 활용하여 단기(5일), 중기(20) 주식 종목을 추천해준다. 또 실시간으로 손절가를 알려주고 수익을 낼 수 있는 주식들을 추천해준다. 그리고 경제용어와 주식관련 용어를 정리한 사전이 있다.

다. To-be 시스템

‘주식인’은 추천해줄 주식만 알려주고 ‘MK라씨로’는 자신이 관심있는 주식을 저장해 두고 매매시기를 알림 받을 수 있는 기능이 있다. 두 어플 다 자신이 매입한 주식을 매입한 시점으로부터 주식의 순수익을 알려주지 않는 점이 아쉬웠다.

아직 주식 쪽의 지식도 부족하고 기술적으로도 부족하여 실시간 매매 시기를 예측하는 알고리즘을 구현하기 힘들어서 내일의 주식 가격만 예측하는 알고리즘을 구현하고 위의 두 어플에 없었던 주식 매입 날짜로부터 순수익을 알려주는 사용자만의 매매 주식 카테고리가 포함되어 있는 주가 예측 웹사이트를 구현할 예정이다.

2. 요구사항 분석

가. 요구사항

요구사항
<ul style="list-style-type: none"> • 사용자의 매매주식 관리 카테고리 • 주식가격과 수익률 예측 • 데이터로 주식차트 표현

<표 2-1> 요구사항 분석

<표 2-1>은 프로그램에 대한 요구 사항이다. 이 사이트는 주식의 가격 데이터를 가지고 다음날의 주식의 가격을 예측하고 그 결과를 나타내야한다. 또 사용자가 현재 자신이 매입한 주식을 언제 팔아야 얼마나 수익률을 낼 수 있는지 판단을 도와주기위해 사용자 주식데이터가 있어야 한다. 따라서 사용자 주식을 볼 수 있는 사용자만의 매매 주식 관리 카테고리가 있어야한다. 그리고 매매할 주식의 가격변동이 어떻게 일어나고 있는지 확인하기 위해 주식차트가 있어야한다.

나. 기능분석

위의 요구사항을 통해 알아낸 기능은 사용자가 자신의 주식을 관리 할 수 있게 사용자만의 주식 관리 기능과, 가격을 예측하는 기능, 주식의 차트를 보여주는 기능이 필요하다.

아래의 <표 2-2>는 각 기능들에 대한 조건들이다.

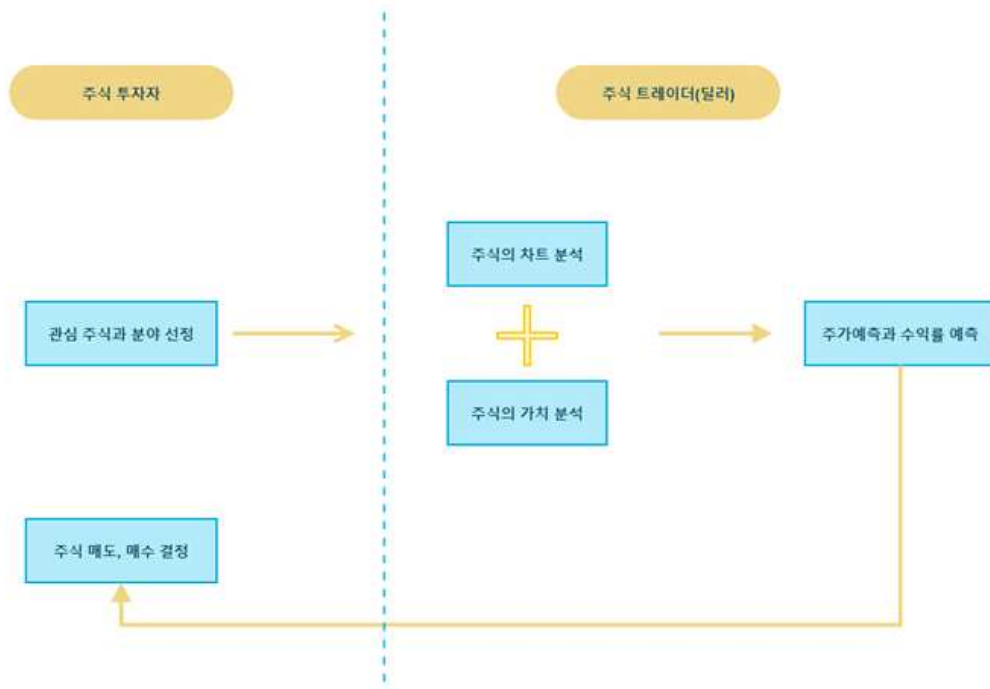
기능	조건
사용자 주식 관리 기능	매입한주식(P)와 관심있는 주식(F) 카테고리를 나누어야 한다. 매입한 주식 카테고리에 있는 주식은 매입한 날짜, 매입한 단가, 매입한 주식의 수를 저장해야 한다.
데이터 분석	주식 데이터를 분석하여 주식 가격을 예측할 수 있게 도출 한다.
예측가격 결과와 차트기능	데이터 분석 후 예측된 가격을 보여준다. 주식데이터를 이용하여 주식 차트를 보여준다.

<표 2-2> 기능 분석

3. 업무 정의

가. 주식 가격 예측 절차

주식 가격을 예측하는 절차는 아래 <그림2-3>과 같다.



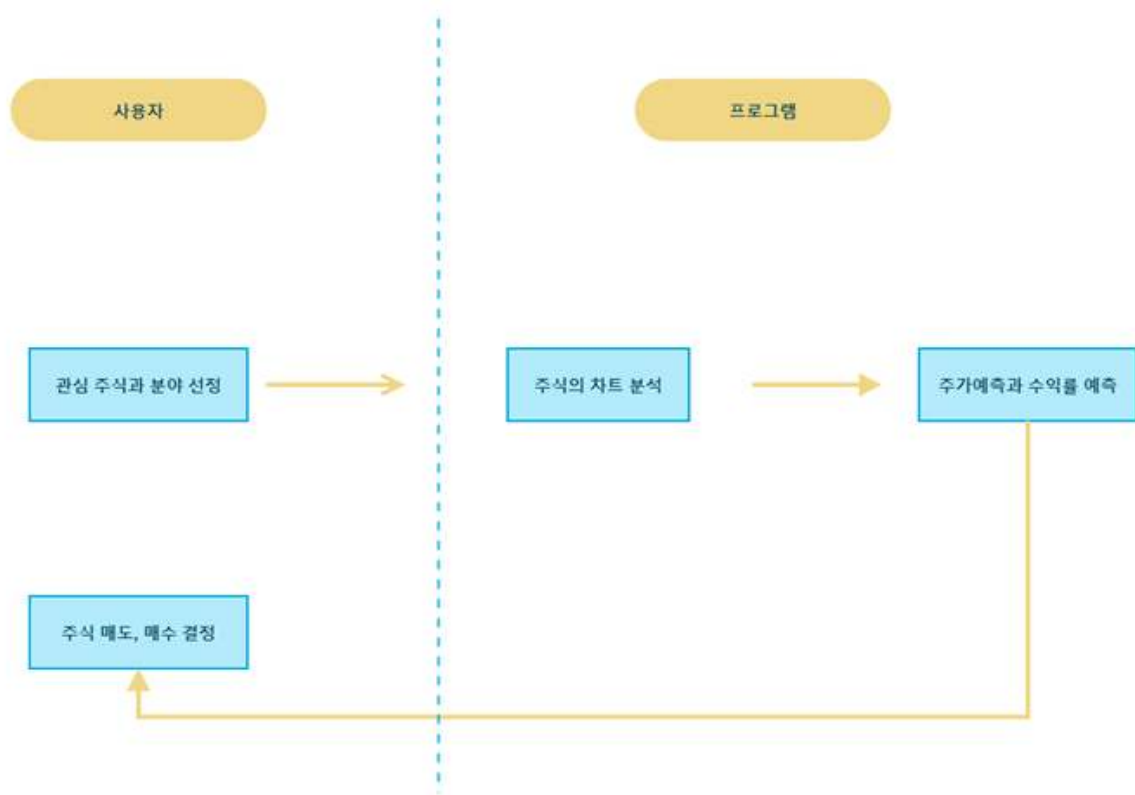
<그림 2-3> 주식 가격 예측 절차

관심있는 주식을 선정하여 주식 트레이더에게 알려주면 주식 트레이더는 주식의 차트 분석과 현재의 주식시장의 상황과 경제상황등을 분석하는 주식의 가치를 분석하여 주

가를 예측한다.

나. 프로그램에서의 주식 가격 예측 절차

계획한 프로그램 상에서 주식을 예측하는 절차는 아래 <그림2-4>과 같다.



<그림 2-4> 프로그램 상에서 주식 가격 예측 절차

가치 분석이 없으면 주식의 가격 예측률이 떨어지겠지만 주식의 가치를 분석하는 일은 나의 기술과 지식이 부족하여 구현하는 것이 힘들다. 따라서 이전 주식의 가격 데이터를 가지고 주식 가격을 예측하는 방법인 주식 차트 분석 방법만으로 주식가격을 예측하는 사이트를 구현 할 예정이다.

III. 프로젝트 설계

1. 개발환경 정의

가. 소프트웨어

구분	프로그램	비고
WEB Language	python, javascript	3.8.3
WEB Developer Tool	Atom Editor	
WEB Application Server	flask	1.1.2
DB	MySQL	8.0.22
Data Analysis	python	3.8.3
Data Analysis Tool	jupyter notebook	6.0.3
	tensorflow	cpu version 2.3.1
	keras	2.3.1

<표 3-1>

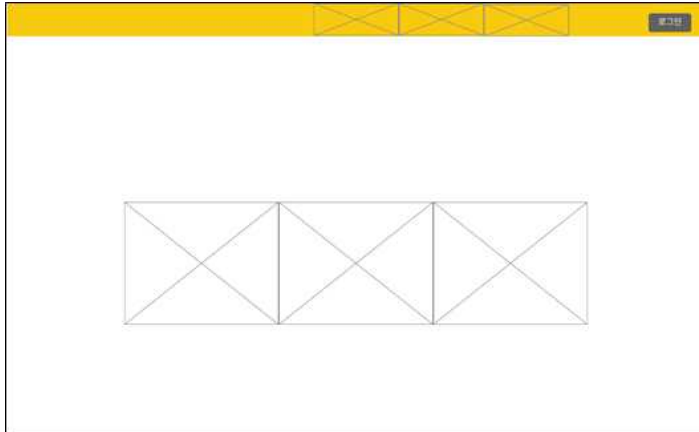
나. 시스템

구분	이름
사용모델	LG gram
CPU	1.6 GHz 4 core Intel Core i5-8265U
RAM	16GB
OS	Window 10

<표 3-2>

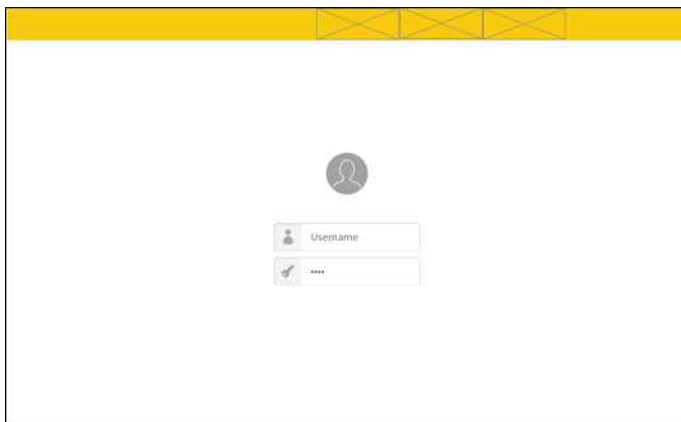
2. 화면 설계

가. 메인 화면



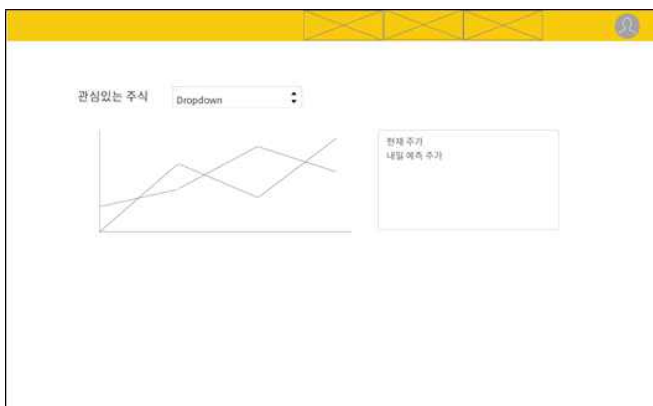
<그림 3-1>

나. 로그인 화면



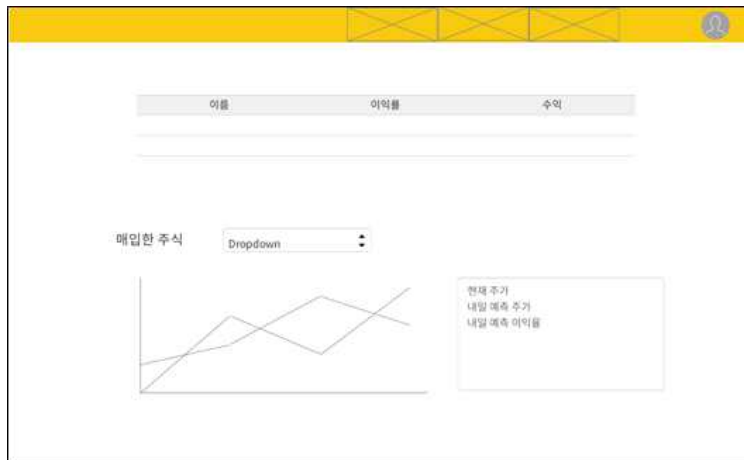
<그림 3-2>

다. 매입한 주식 화면



<그림 3-3>

라. 관심있는 주식 화면



<그림 3-4>

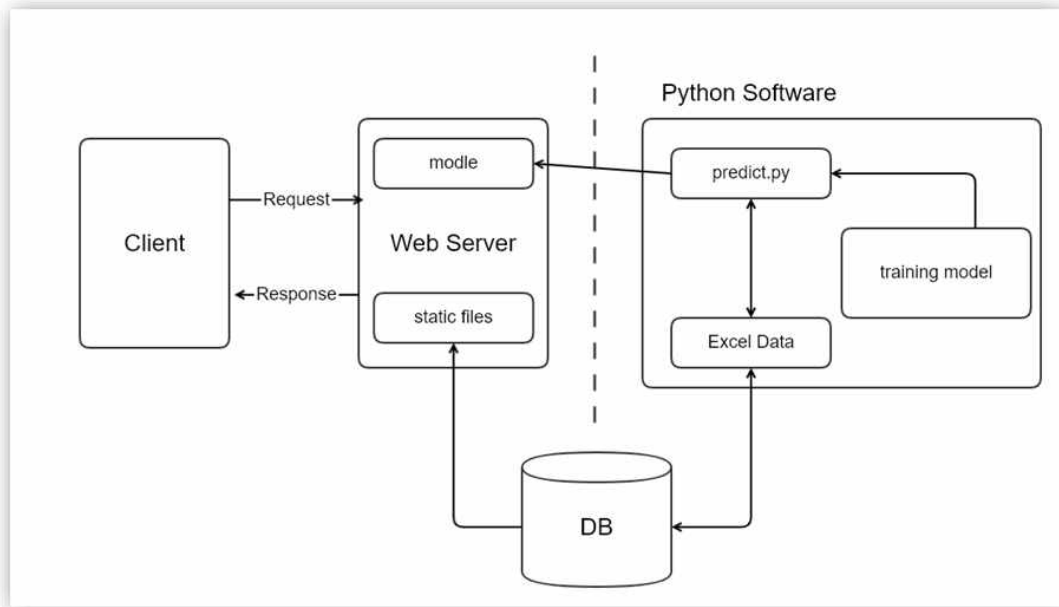
마. 주식 저장 화면

<그림 3-5>

3. 시스템 아키텍처

가. 시스템 구성

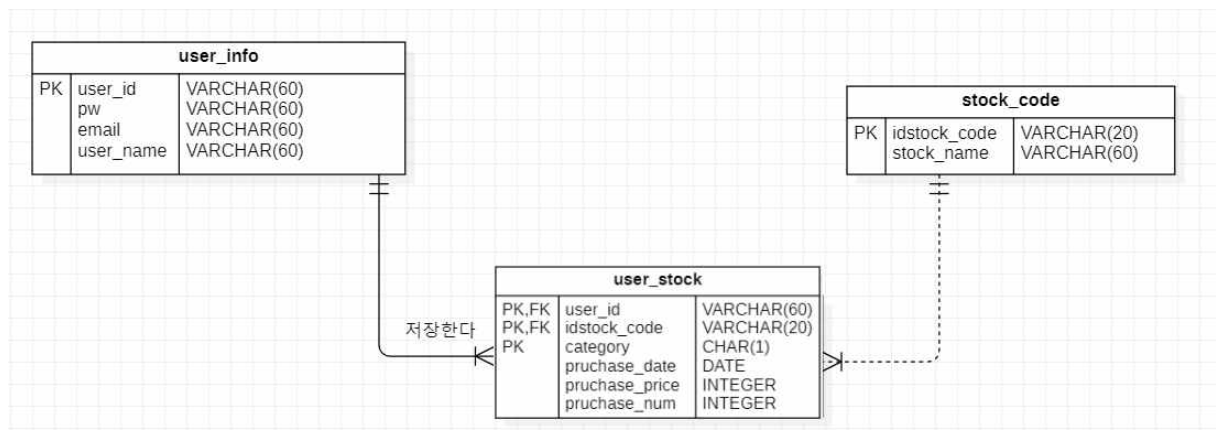
<그림 3-6>은 시스템 구성도이다. 사용자가 주가를 예측하라고 서버에 요청하게 되면 파이썬 소프트웨어에서 학습된 모델과 주식 가격 데이터를 이용해 사용자에게 응답해주도록 시스템을 구성해 주었고 사용될 고정적인 데이터는 데이터베이스와 연동하게 구성하였다.



<그림 3-6>

4. DB 설계

가. ERD



<그림 3-7> ERD

나. 테이블 설계

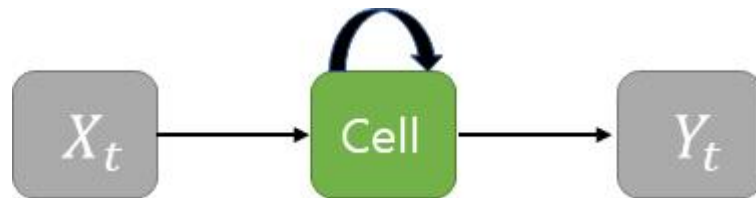
Table	Primary Key	Foreign Key
user_info	user_id	
stock_code	idstock_code	
user_stock	user_id, idstock_code, category	user_id, idstock_code

<표 3-3>

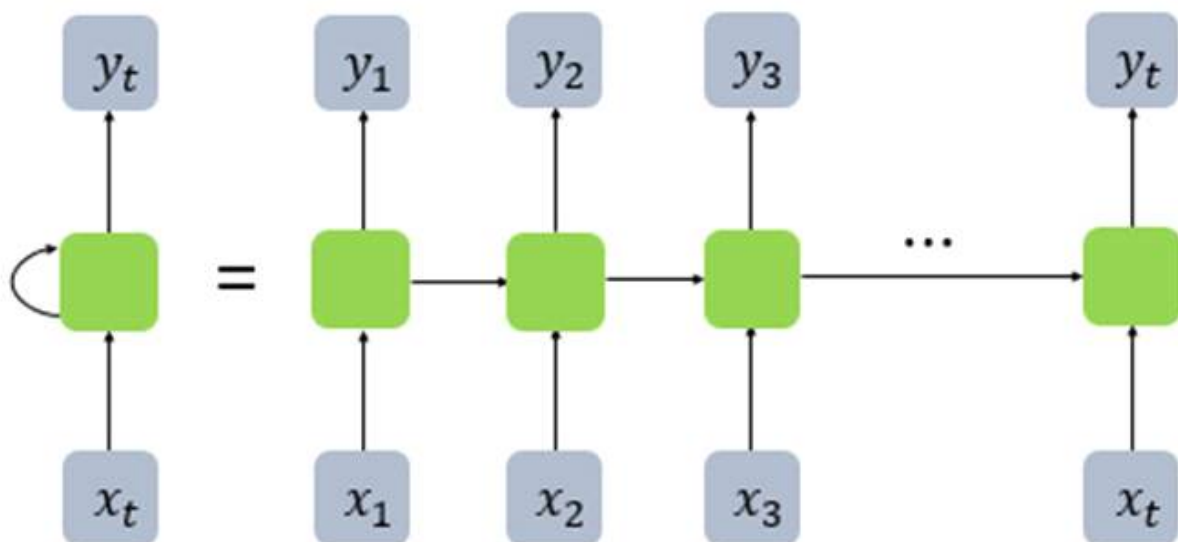
5. 데이터 분석 설계

가. LSTM

데이터 분석으로는 RNN(Recurrent Neural Networks)의 한 종류인 LSTM(Long Short-Term Memory models) 모델을 사용해 주었다. RNN은 순차 데이터를 이용해서 학습시킬 때 많이 사용하는 모델로 앞에서 입력 받는 데이터를 기억해 두어 현재 데이터에 적용 시키는 방법이다. <그림 3-8>와 <그림 3-9>는 RNN의 구조를 나타내준다.



<그림 3-8>



<그림 3-9>

RNN의 길이가 길어지게 되면 앞에 정보가 뒤에까지 충분히 전되지 않아서 정보량이 소실되는 문제가 발생하는데 이를 해결한 것이 바로 LSTM이다. LSTM은 앞에서 기억된 정보를 넘기기 전에 기억된 값을 다음 층으로 넘길지 결정을 하여 넘긴다.

나. 손실 함수

손실함수로는 MSE(Mean Squared Error, 평균제곱오차)를 사용하였다. 일반적으로 선형회기에서 많이 사용하는 손실함수이다. 손실 함수는 실제 데이터 값과 예측한 결과 값의 오차를 줄이기 위해 사용한다. <그림 3-10>은 MSE 공식이고 MSE의 값이 작을 수록 정답에 가까이 나오게 된다.

$$MSE = \frac{1}{n} \sum \left(y - \hat{y} \right)^2$$

The square of the difference
between actual and
predicted

<그림 3-10 > MSE 공식

<https://bitcodic.tistory.com/108> (최종 검색일 : 2020-12-09)

다. 학습 데이터

주식가격을 학습시키기 위해 yahoofinance가 제공하는 데이터를 사용하였다.

	Date	Open	High	Low	Close	Adj Close	Volume
0	2015-10-26	25960.0	25960.0	25440.0	25840.0	260.791168	7608250.0
1	2015-10-27	25640.0	25980.0	25620.0	25960.0	262.002289	6612050.0
2	2015-10-28	25880.0	26160.0	25820.0	26160.0	264.020813	13565150.0
3	2015-10-29	26600.0	27840.0	26480.0	26500.0	267.452271	31523550.0
4	2015-10-30	26900.0	27800.0	26820.0	27440.0	276.939209	25469250.0

<그림 3-11 > 삼성전자 주식 가격 데이터

학습 시킬 데이터는 <그림 3-11>처럼 삼성전자 주식 가격 데이터를 사용했고 독립변수와 종속변수 둘 다 Close(종가)를 사용해 주었다. 그리고 학습을 해주는 학습 셋과 테스트를 해주는 테스트 셋으로 나누어서 학습과 테스트를 같이 진행하도록 설계하였다.

라. Windows

최근 10일간의 주식 데이터를 기반으로 다음날의 주식 가격을 예측 할 수 있게 <그림 9-11>처럼 window의 크기를 11개로 설정하고 한 개의 window에서 앞에 10개의 데이터는 독립변수로 마지막 데이터는 예측하는 값인 종속변수로 설정해 주었다.

	Open	High	Low	Close
2018-10-01				
2018-10-02				
...				
2018-10-20				
2018-10-21				
...				
...				

<그림 3-12 > windows 설정

IV. 프로젝트 구현 결과

1. 데이터 학습 결과

가. 손실률

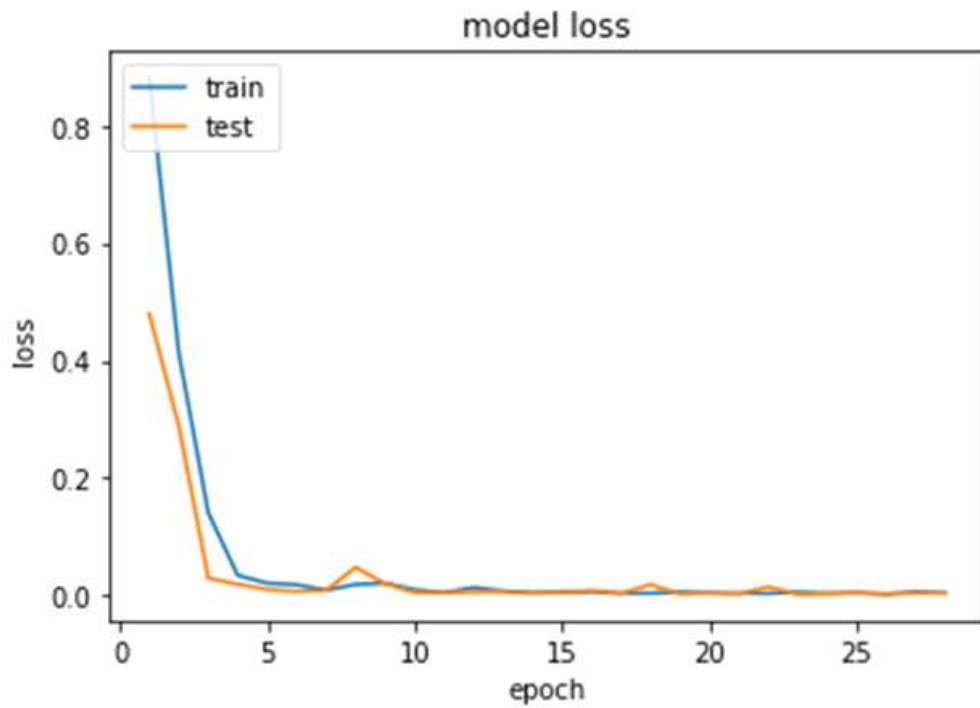
학습 후 손실률은 <그림 4-1>처럼 0.00143이 나왔다.

```
95/97 [=====>.] - ETA: 0s - loss: 0.0037
Epoch 00028: val_loss did not improve from 0.00143
97/97 [=====] - 2s 16ms/step - loss: 0.0038 - val_loss: 0.0039
```

<그림 4-1 > 손실률

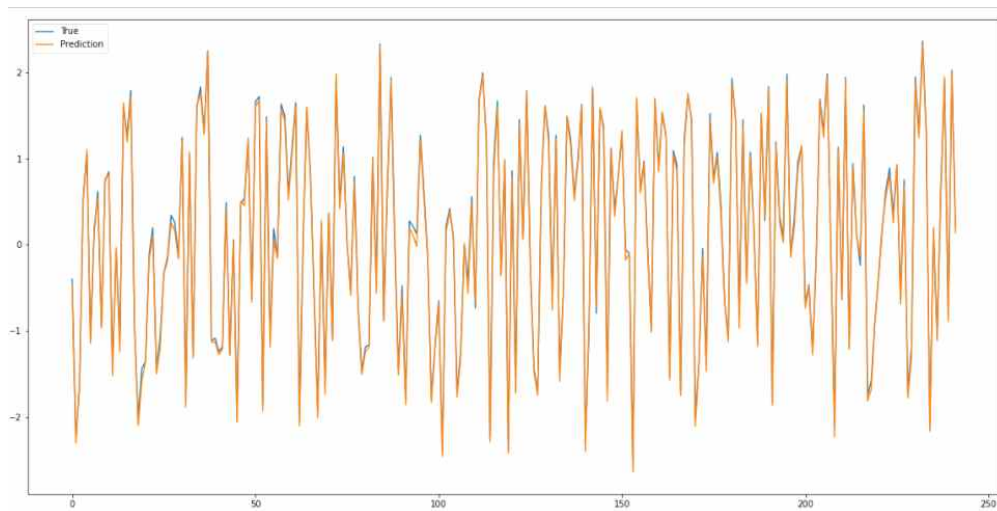
나. 학습 셋과 테스트 셋의 손실률

학습 셋과 테스트 셋의 손실률을 손실율과 학습 횟수에 따른 그래프를 나타내면 <그림 4-2>와 같다.



<그림 4-2 > 학습 셋과 테스트 셋의 손실률 그래프

다. 예측 값과 실제 값 그래프

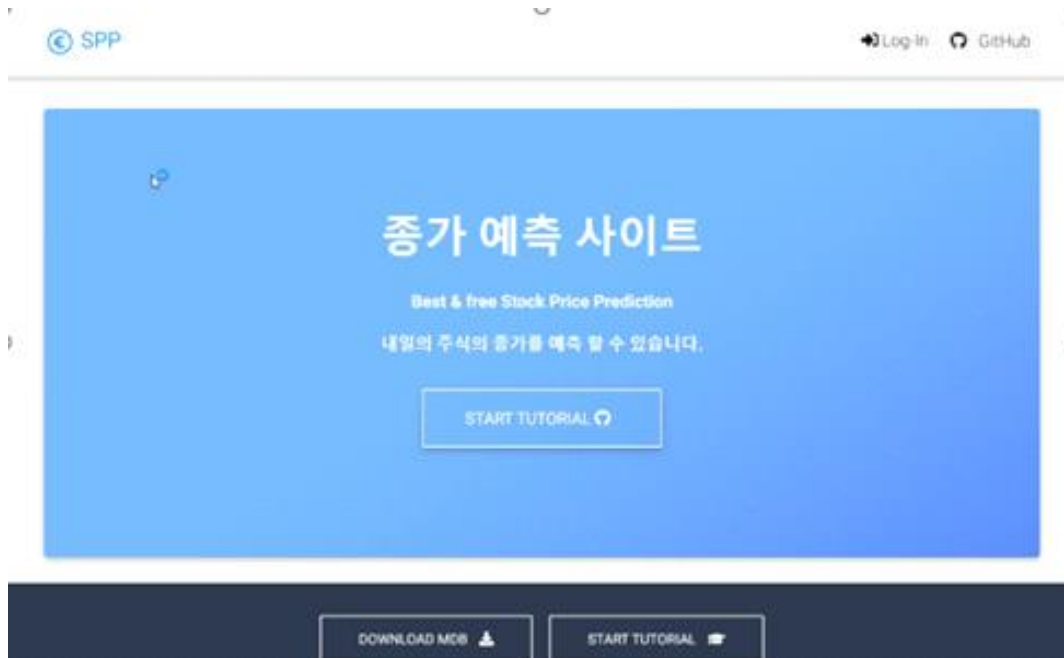


<그림 4-3 > 예측 값과 실제 값 비교 그래프

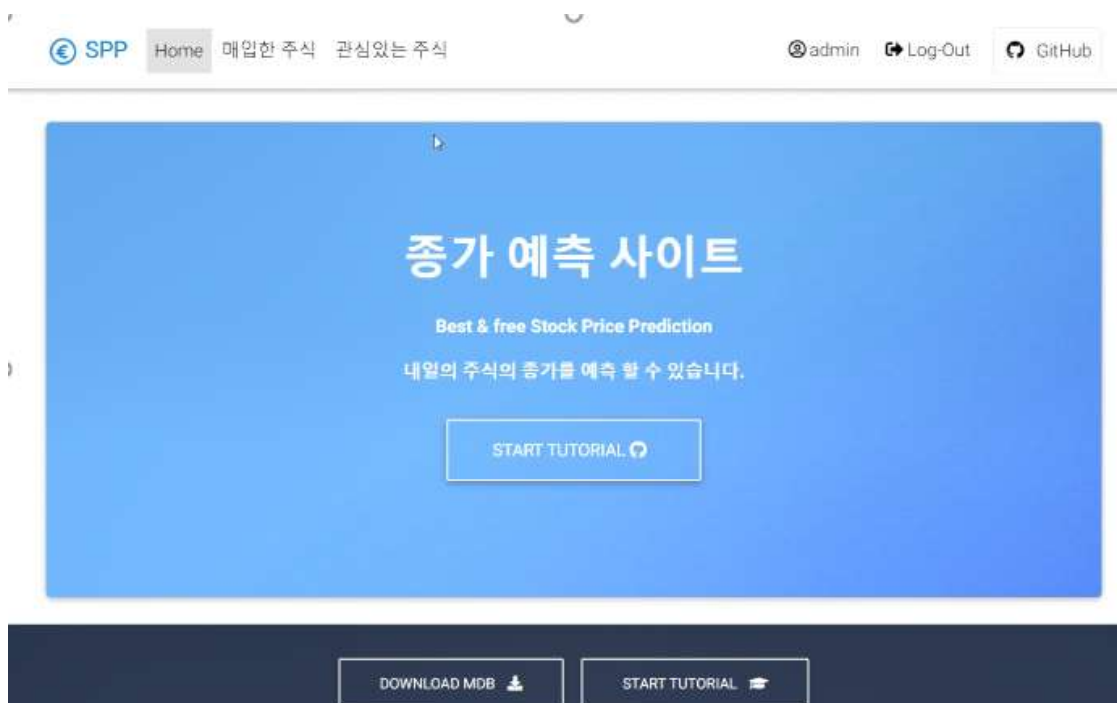
예측한 값과 실제 값을 비교한 그래프는 <그림 4-3 >과 같다.

2. 사이트 구현 결과

가. 메인 페이지

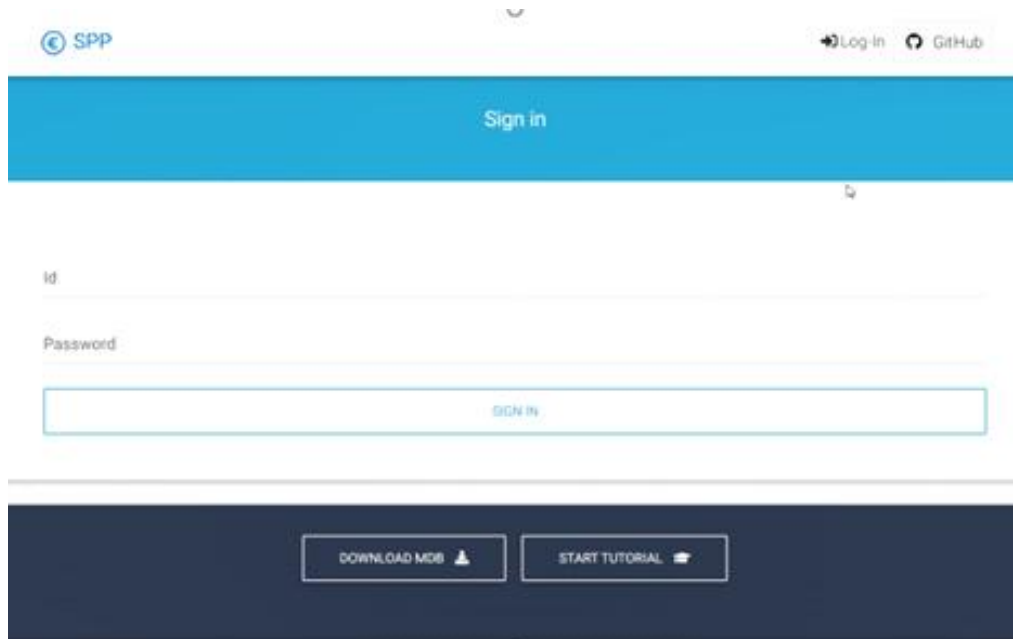


<그림 4-4 > 로그인하지 않았을 때 메인페이지



<그림 4-5 > 로그인했을 때 메인페이지

나. 로그인 페이지



Sign in

Id

Password

SIGN IN

DOWNLOAD MDS

START TUTORIAL

<그림 4-6 > 로그인 페이지



Sign in

Id

Password

SIGN IN

DOWNLOAD MDS

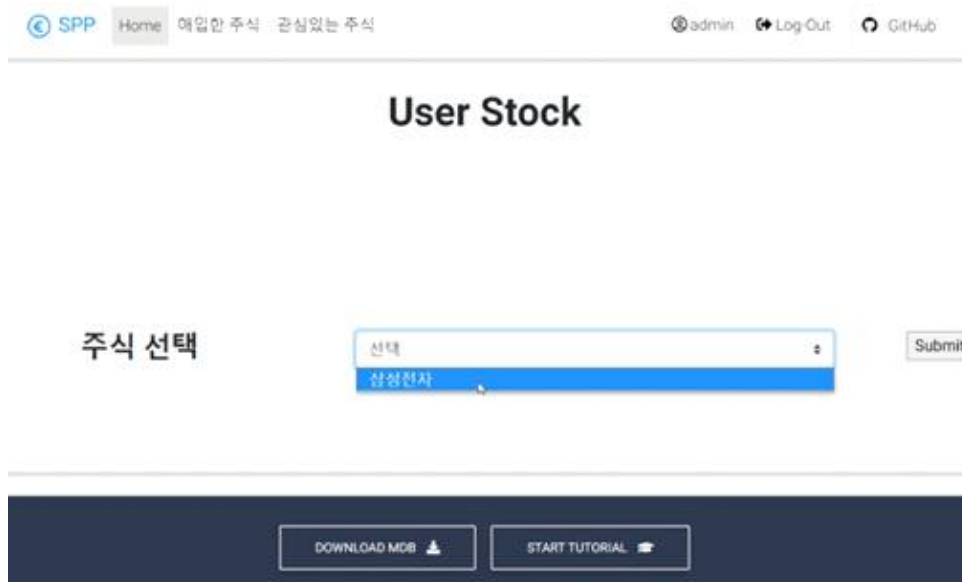
START TUTORIAL

127.0.0.1:5000 내용:
비밀번호와 아이디가 틀립니다.

확인

<그림 4-7 > 로그인 실패 시 뜨는 경고 창

다. 매입한 주식 페이지







<그림 4-8 > 주식 선택 전



<그림 4-9 > 주식 선택 후

라. 관심있는 주식 페이지


[Home](#)
[매입한 주식](#)
[관심있는 주식](#)

 admin
  Log-Out
  GitHub

주식 선택

선택

sk하이닉스

Submit

DOWNLOAD MDB

START TUTORIAL

<그림 4-10 > 주식 선택 전



<그림 4-11 > 주식 선택 후

V. 개발 일정 및 주요 이슈

1. 개발 일정

	1주차	2주차	3주차	4주차	5주차	6주차	7주차	8주차
공부								
기획								
분석								
설계								
구현								

<표 5-1>

	9주차	10주차	11주차	12주차	13주차	14주차
구현						
테스트						

<표 5-2>

2. 주요 이슈

처음에 웹언어로는 jsp와 java로 사용하도록 계획 하였으나 tesnsorflow.js가 개발 시스템 환경에 설치가 안돼서 웹을 jsp가 아닌 python과 flask를 이용하여 개발하게 되었다. flask에 대한 이해 부족과 전체적인 웹에 관한 이해 부족으로 구현이 지연 되었다.

차트를 반응형으로 구현하도록 기획하였으나 js에 대한 지식이 부족하여 반응형 차트로 완성하지 못하였다.

VI. 개발 일정 및 주요 이슈

1. 프로젝트 기대 효과

본 프로젝트를 통해 예측한 주식가격으로 사용자는 주식을 언제 매입하고 매수할지 결정을 도와주어 단기적인 수익을 낼 수도 있다. 또 주식투자를 처음하는 사용자에게는 주식 시장흐름에 대해 조금이나마 공부할 수 있는 기회가 될 수 있다고 기대 된다.

2. 갈무리 및 향후 계획

프로젝트 분석과 설계, 계획 등을 자세히 하지 않아서 구현 시 시간이 많이 걸렸다. 그래서 프로젝트 기간이 모자라 사용자 주식을 저장하는 기능을 구현하는 것과 차트를 반응형으로 구현하는 것을 못하여서 아쉬웠다. 향후에 이 기능을 다 구현하여 새 버전의 시

시스템을 구현할 계획이다.