# REQUIREMENTS

## Application Phase 1

* **Filters:** 
  + Stocks: F&O Stocks Only( Around 180 stocks)
  + TimeFrame : 5 min.
  + LTP above day’s open, LTP below day’s open , LTP above day’s high , LTP below day’s high, LTP above day’s low , LTP below day’s low
* **Sorting:** Only on Change(%) and Stock Name (%) using Datatables.
* **Other Functionalities:** Auto Refresh every 5minutes using AJAX and JS or Websockets, Search and Pagination, Direct Data from KITE API,

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| STOCK | LTP | CHANGE (%) | Sell / Buy | Sellers | Buyers | O | H | L | C |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |

## Application Phase 2

* **Filters:** (All visible on Front End , with Apply Filter button)
  + Stocks: F&O Stocks Only( Around 180 stocks)
  + TimeFrame : 5 min.
  + Buy / Sell Ratio: Sellers twice the Buyers , Sellers 1.5 times the buyers , Buyers twice the Sellers , Buyers 1.5 times the Sellers
  + Stochastic : Stochastic below 30, Stochastic above 70, Stochastic above 30 and below 70
  + Candle Patterns : Any Bearish Reversal Pattern , Any Bullish Reversal Pattern
* **Sorting:** Multi Sort Feature, being able to apply sorting on multiple columns at the same time.
* **Other Functionalities:** Direct Data from KITE Historical API, NO Database for now

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| STOCK | LTP | CHANGE (%) | VOLUME(%) | STOCHASTIC | Sell / Buy | Buyers | Sellers | Candle Pattern |
|  |  |  |  |  |  |  |  | Shooting Star , Hanging Man |
|  |  |  |  |  |  |  |  | Bearish Engulfing , Gravestone Doji |
|  |  |  |  |  |  |  |  | Hanging Man, Double Tops |
|  |  |  |  |  |  |  |  | Hammer |
|  |  |  |  |  |  |  |  | Bullish Engulfing |
|  |  |  |  |  |  |  |  | Double Bottom |
|  |  |  |  |  |  |  |  |  |

## Application Phase 3:

* Database with all data recorded from KITE API, no need of Historical API.
* Auto Data clearing feature after 14 days.(Storing only the data for the last 14 days).
* Hosting the application so that it keeps recording data all the time.
* Buying a domain for the application so it is accessible from anywhere at any time.

## Application Phase 4:

* Intraday Screener( All time frames: 5 min, 10 min, 15 min, etc , All exchanges and all stocks with various sectors)
* EOD Screener
* Stock Alerts Feature(SMS or Email)
* Auto Trading from Application

# Application TimeLine:

PHASE 1

17th March

PHASE 2

30th April

PHASE 3

30th June

PHASE 4

31st October

# Initial Setup:

## GIT

git init

git add README.md

git commit -m "first commit"

git remote add origin https://github.com/play-area/StockConnectStandAlone.git

git push --set-upstream origin master

username : [kulbir2408@gmail.com](mailto:kulbir2408@gmail.com)

password : ks9556462141

## ECLIPSE SETUP:

1. Add an apache server to Eclipse:
2. Import Project as Maven project into eclipse.
3. Navigate to Project’s Home Directory where pom.xml file is kept in command prompt and run the mvn clean install command.
4. Add the following settings to Eclipse for a Maven Dependencies.
5. You need to add the "Maven Dependency" in the Deployment Assembly

* right click on your project and choose properties.
* click on Deployment Assembly.
* click add
* click on "Java Build Path Entries"
* select Maven Dependencies"
* click Finish.
* Rebuild and deploy again

# KITE CONNECT API:

## Steps to connect :

1. How to get request\_token , when the redirect URL is http://127.0.0.1?
2. Login manually to https://kite.zerodha.com/connect/login?api\_key=9rrtu9ln1a00gcfp
3. This asks for client id and password, later security questions.
4. On successful login, we will be able to capture the request token from the browser address bar (Manually)
5. Later use this request token for any further interaction (programmatically) with the platform.
6. Every time the session expires, or there is a need to create a new connection then we will follow steps 1 to 4 again.

**Configuration for websocket:**

Add the following two JAR Files to : C:\jdk1.8.0\_45\jre\lib\security downloaded from <http://www.oracle.com/technetwork/java/javase/downloads/jce8-download-2133166.html>

1. local\_policy.jar
2. US\_export\_policy.jar

For websockets need to add Security Certificate from KITE from the websockets URL, follow the Link:

<https://support.cloudbees.com/hc/en-us/articles/217078498-PKIX-path-building-failed-error-message>

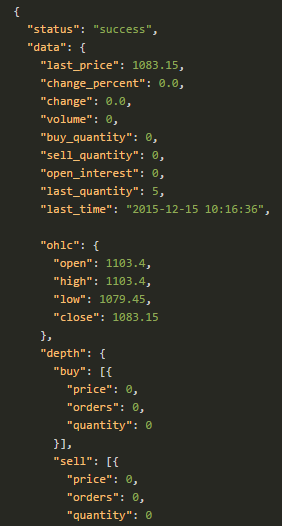
keytool -import -alias KiteConnect -keystore C:\jdk1.8.0\_45\jre\lib\security\cacerts -file C:\Users\k4304735\Desktop\websocketkitetrade.crt

## LIVE QUOTES:

### Request:

"https://api.kite.trade/instruments/NSE/INFY?api\_key=xxx&access\_token=yyy

### Response:



## HISTORICAL QUOTES :

### Request:

https://api.kite.trade/instruments/historical/5633/minute?from=2015-12-28&to=2016-01-01&api\_key=xxx&access\_token=yyy

| **parameter** |  |
| --- | --- |
| :instrument\_token | Identifier for the instrument whose historical records you want to fetch. This is obtained with the[instrument list](https://kite.trade/docs/connect/v1/#retrieving-full-instrument-list) API. |
| :interval | The candle record interval. Possible values are:  · minute · day · 3minute · 5minute · 10minute · 15minute · 30minute · 60minute |

### Response:

