Contents

[Requirements Mapping 2](#_Toc465080984)

[Design 4](#_Toc465080985)

[Implementation 5](#_Toc465080986)

[Get Company List 5](#_Toc465080987)

[Get Company History 5](#_Toc465080988)

[Create new Company 5](#_Toc465080989)

[Delete Company 5](#_Toc465080990)

[Testing 6](#_Toc465080991)

[Unit Testing: 6](#_Toc465080992)

[Functional Testing: 6](#_Toc465080993)

[Version Control 6](#_Toc465080994)

[User Guide 8](#_Toc465080995)

## Requirements Mapping

From the given documents software requirements are constructed for the required Standalone Application and Web Application with Restful API.

**Supp1:** Find an API which helps to get stock value of a company

* Supp1.1: API should return current stock value when company code is given
* Supp1.2: API should return full company name when company code is given

**Supp2:** Create a Database where companies and stock history of company can be stored

* Supp2.1: When a company is deleted all its corresponding stock values in data base should be deleted
* Supp2.2: There should be some primary key for each table created in database
* Supp2.3: Primary Key, Foreign key constrains should be properly given when creating database tables

**Supp3:** Create job which executes after every five minutes and adds current stock values to data base for all the registered companies.

**Supp4:** Create Restful API to get the stock history of a company

* Supp4.1: When company is not found in data base http response status should be 404 (Not found)
* Supp4.2: When company is found in data base http response status should be 200(OK) and should send JSON which contains name of the company and all the stock history.

**Supp5:** Create Restful API to get the list of companies in database

* Supp5.1: When data base is empty. Response status should be 204 (No Content)
* Supp5.2: When request is successful and data base has data. Then Response status should be OK (200) and should have list of companies in response body.

**Supp6:** Create a Restful API which can add company name to database

* Supp6.1: When requested company is valid and not present in data base. Add this company to database. Response Status should be 201 (Created)
* Supp6.2: When requested company is already in database. Response status should be 403 (Forbidden)
* Supp6.3: When requested company\_code is not a valid one. Then Response status should be 400 (Bad Request)

**Supp7:** Create a Restful API which deletes company in database

* Supp6.1: When requested company is not present in data base. Response Status should be 404 (Not Found). No action should be performed in database.
* Supp6.2: When requested company is in database. This company should be deleted in data base and Response status should be 200 (OK)

**Supp8:** Implement a logger which logs error messages or any required messages.

**Supp9:** Create a front end interface which shows stats of company stocks

* Supp9.1: Create a time scaled area chart. X-axis represents time and Y-axis represents stock value in dollars.
* Supp9.1: On mouse over on the graph chart. Corresponding stock value and time should be shown. On mouse out this section should be hidden.

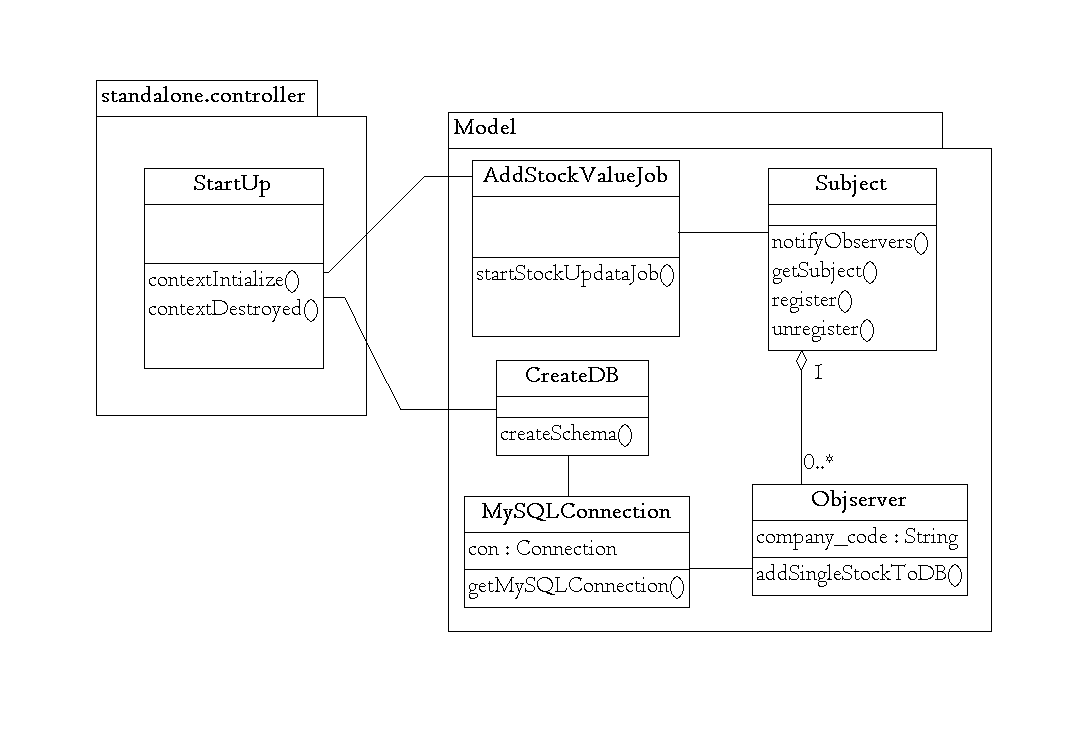
## Design

### Stand Alone Java Program

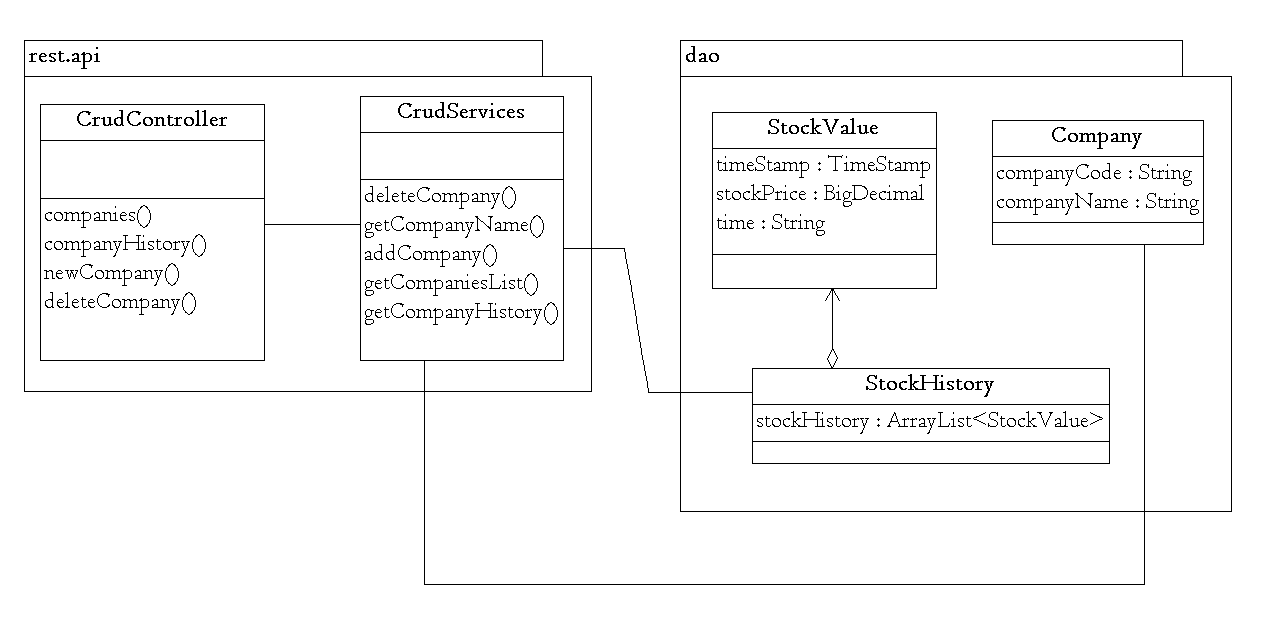
This application should get the stock values for all the companies in data base every 5 minutes and should update database.

Implemented this with **Observer Pattern**. Used observer pattern because in this application addition and deletion of companies should be performed. If observer pattern is used there is no need to change or changes in code will be very minimum when implementing restful API in web application. Components are loosely coupled so modification of code will be easy and will be specific to only one or two modules.

**Object Oriented Design**

****

### Web Application with Rest API



**Travis CI**

## Implementation

Get Company List – OK (200), No Content (204)

Get Company History – OK (200), Not Found (404)

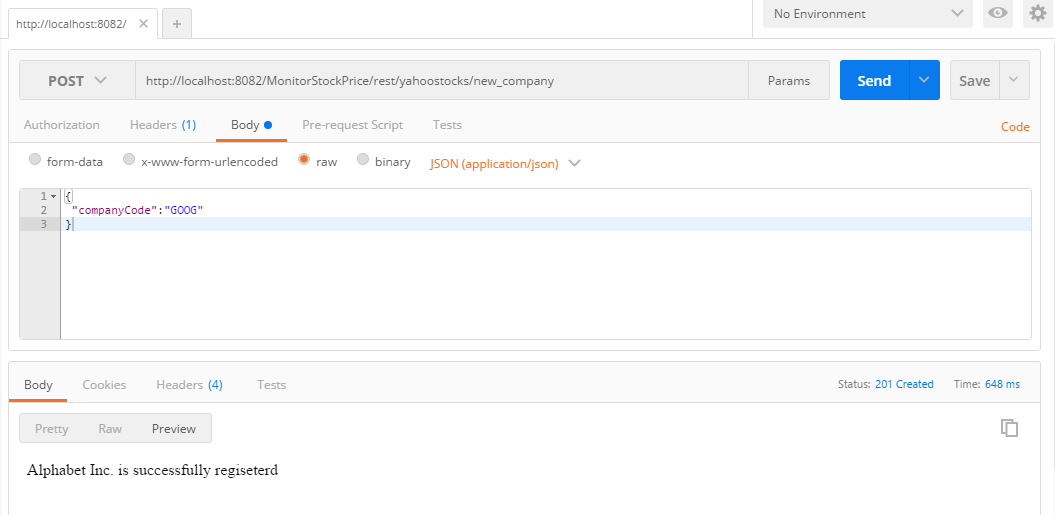
Create new Company – Created (201), Forbidden (403) – company already exist, Bad Request (400) Company Not exist

{

"companyCode":"GOOG"

}

http://localhost:8080/MonitorStockPrice/rest/yahoostocks/new\_company



Delete Company – Not Found (404), OK (200)

Maven build -- clean test

http://localhost:8082/MonitorStockPrice/rest/yahoostocks/getCompanies

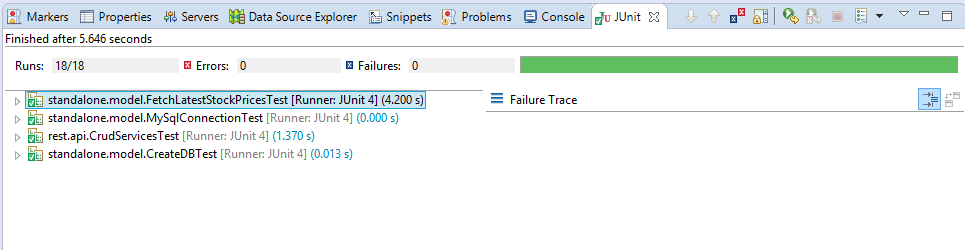
System.*setProperty*("rootPath", context.getRealPath("/"));

**C:\Users\eswar\Desktop\LM\_WS\.metadata\.plugins\org.eclipse.wst.server.core\tmp0\wtpwebapps\MonitorStockPrice\WEB-INF\logs**

## Testing

### Unit Testing:

Added Junit test cases for all service layer modules. For each module possible success scenarios and failure scenarios are tested



Restassured

### Functional Testing:

After implementing the application. Developed Restful apis are tested manually from Postman.

Tested apis with possible inputs and validated them by comparing expected output and actual output

## Version Control

Used Git for version control. Created branches for tasks and after competing each task merged with master branch.

## User Guide