

# Contents

## **Report-01**

1. Introduction.....	7
2. Problem Statement .....	7
3. Project Objectives .....	7
4. Preliminary Solution .....	8
5. Project Scopes .....	8
5.1 Cost Estimation .....	9
5.2 Estimation Cost and Time for Feasibility Study .....	9
6. Conclusion .....	10

## **Report-02**

1. Introduction.....	12
2. Background .....	12
3. Outline of the Software Project.....	13
4. Method of Analysis/Methodology .....	15
5. Overview of Alternatives.....	15
5.1 Overview of Alternative 1 .....	15
5.2 Overview of alternative 2.....	17
5.3 Overview of alternative 3 .....	18
6. Recommendation .....	20
7. Conclusion .....	20

## **Report-03**

1. Introduction.....	23
2. Project Organization .....	23
3. Risk analysis .....	25
3.1 Risk Identification .....	25
3.2 Risk reductions strategies.....	27
4. Hardware and Software Resource Requirement .....	28
5. Work Breakdown .....	28

6. Project schedule .....	30
7. Monitoring and Reporting Mechanism .....	33
8. Conclusion .....	33

## **Report-04**

1. Preface.....	36
2. Introduction.....	35
3. Glossary .....	36
4. User Requirements Definition .....	36
6. User Requirements Specification.....	37
7. System Model .....	40
7.1. Use Case.....	40
7.2. Use case diagram.....	47
8. System Evolution .....	47
9. Appendices.....	48
9.1. Appendix A: Hardware Specification .....	48
9.2. Appendix B: Database Specification.....	48
10. Index .....	50

## **Report-05**

1. Introduction.....	53
2. Architectural Pattern .....	53
3. Activity Diagram .....	54
4. Class Diagram .....	58
5. Sequence Diagram .....	59
6. State Machine Diagram.....	67
7. Conclusion .....	70

## **Report-06**

1. Introduction.....	72
2. Proposed Conceptual Class Diagram.....	72
3. Specify Interface .....	73
4. Used Design Pattern.....	75

4.1 Builder Design pattern .....	75
4.3 Facade Design pattern .....	76
4.4 Strategy Design pattern .....	77
5. Detail Class Diagram .....	79
6. Conclusion .....	83

## **Report-07**

1. Introduction.....	86
2. Architectural Design.....	88
3. Architectural Pattern .....	88
4. Detail Design .....	89
5. Screen shoot of System.....	94
6. Limitation.....	97
7.Conclusion and Future work.....	97

## **List of Tables**

## **Report-01**

Table 4.1: preliminary Solution .....	8
Table 5.1: Project Scopes.....	9
Table 5.2: Cost Estimation.....	10
Table 5.1(a): The investment of alternative 1 .....	16
Table 5.1(b): The cost/benefit analysis of alternative 1 (yearly basis) .....	16
Table 5.1(c): The cost/benefit analysis of alternative 1 (yearly basis) .....	17
Table 5.2: The investment of alternative 2 .....	18
Table 5.3(a): The investment of alternative 3 .....	19
Table 5.3(b): The cost/benefit of alternative 3 is illustrated (yearly basis) .....	19
Table 6: Economic analysis between alternative 1, alternative 2 and alternative 3 .....	20

## **Report-02**

Table i: Role of Team Members .....	<b>Error! Bookmark not defined.</b>
Table ii: Risk identification .....	26
Table iii: possibility and effects of risk.....	27
Table IV: Risk reductions strategies .....	28

Table 4.1:List of hardware and software resources .....	28
Table VI: Task management of the whole system.....	29
Table vii: Milestones and deliverables .....	30
Table viii: Schedule Representation .....	31
Table ix: Task Distribution .....	32

## **Report-03**

Table I: Glossary.....	36
Table II : Definition of User Requirements .....	37
Table III: Specification of User Requirements.....	40
Table IV : List of Use Cases .....	40
Table V: Server Specification .....	48
Table VI: Database Specification .....	49
Table VII : List of Figures .....	50

## **List of Figures**

### **Report -02**

Figure 1: illustrates the Data Flow Diagram of the system. ....	14
---	----

## **Report-03**

Figure 1: Gantt Chart .....	31
Figure 2: Staff Allocation Chart .....	32

## **Report-04**

Figure 7.1: Use Case Diagram .....	47
------------------------------------	----

## **Report-05**

Figure 2.1: Architectural Pattern.....	53
Figure 3.1: UC1 .....	54
Figure 3.2: UC2 .....	55
Figure 3.3: UC 3 .....	55
Figure 3.4: UC4 .....	56
Figure 3.5: UC5 .....	56
Figure 3.6: UC6 .....	57

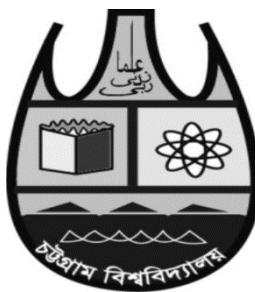
Figure 3.7: UC7 .....	57
Figure 4.1: conceptual class diagram.....	58
Figure 4.2: Relationships between conceptual classes. ....	59
Figure 5.1: Sequence diagram for UC1: Login to the system.....	60
Figure 5.2: Sequence diagram for use case 2 : Distribution Script.....	61
Figure 5.3: Sequence diagram for use case 3: Submission information .....	62
Figure 5.4: sequence diagram for use case 4: view information.....	63
Figure 5.5: Sequence diagram for UC5 : Take action against any inconsistency.....	64
Figure 5.6: sequence diagram for use case 6: Make decision.....	65
Figure 5.7: sequence diagram for use case 7: Modify information. ....	66
Figure 6.1: State machine diagram for Users object.....	67
Figure 6.2: State machine diagram for Distributor object .....	68
Figure 6.3: State machine diagram for Submitter object .....	68
Figure 6.4: State machine diagram for Viewer object .....	69
Figure 6.5: State machine diagram for TCB department object .....	69
Figure 6.6: State machine diagram for Modifier object.....	70

## **Report-06**

Figure 2.1: Conceptual Class diagram .....	72
Figure 3.1: Insertdata interface .....	73
Figure 3.2: Show data Interface .....	74
Figure 3.3: Zone Interface.....	74
Figure 4.1: Builder Design Pattern .....	75
Figure 4.2: Singleton pattern.....	76
Figure 4.3: Facade Design pattern .....	77
Figure 4.4: Strategy Design pattern .....	78
Figure 5.1: Details Class diagram.....	83

## **Report-07**

Figure 2.1: Architectural Design.....	86
Figure 2.2: Architectural Design.....	86
Figure 2.3: Architectural Design.....	87
Figure 3.1: Presentation Layer .....	88
Figure 3.2: Application Layer.....	88
Figure 3.3: Data Access Layer.....	89
Figure 5.1: System Login Demonstration .....	95
Figure 5.2: Submit and Modify information Demonstration .....	95
Figure 5.3: Submit and Modify information Demonstration .....	96
Figure 5.4: Distribute info, Statistics, and Action of Admin Demonstration .....	96
Figure 5.5: Statistics Demonstration with Pie Chart.....	97



## **Department of Computer Science and Engineering, University of Chittagong.**

<b>Course Code</b>	CSE 516
<b>Course Title</b>	Software Engineering and Design patterns Lab
<b>Assignment name</b>	Problem Definition Document
<b>Report number</b>	01
<b>Group</b>	E

### **Submitted to**

**A. H. M. Sajedul Hoque**

Assistant Professor

Department of Computer Science and Engineering,

University of Chittagong.

### **Submitted by**

ID: 18701032

ID: 18701057

ID: 18701071

ID: 17701087

ID: 17701101

## **1. Introduction**

Controlling market price fluctuations app is an android based application. By using this app, government will be much more efficient and effective for controlling market price fluctuations as it will help to bring equilibrium situation both producers and consumers. One frequently cited reason for increased prices is ‘market fundamentals’ as demand is thought to be outstripping supply and thus leading to increased prices. Although high prices can technically be good news for producers, price fluctuation is extremely dangerous for producers as well as consumers, risk losing their investments if prices fall.

Price fluctuation is not only harmful to consumers but also affects producers. Due to the observed effect of price fluctuation and its impact to small producer’s development, this brings the need to focus on how producers can improve their investment in order to sustain their development by use storage facilities and this can be done during lowest price for produce.

This report illustrates about the project. It includes a brief description about the project, importance of this project, the problem on which the project based on and what can be the primary solution of this problems.

## **2. Problem Statement**

The Ministry of Commerce of Bangladesh has been supplying products through TCB to ensure the necessary supply, including the import, supply and exploration of these products, to keep the prices of essential commodities in the public purchasing capacity. But lately, the prices of essential commodities, including onions, are unaffordable and it is imperative to take initiative to adopt a timely solution or action plan. In coordination with various government agencies by utilizing digital technology, the supply chain ensures human income by aligning with the international price of the product, allowing proper production of all essential commodities, finding suitable product sources, collecting information on tariff and transportation costs. Discrimination is minimal, standardized Nya ensure the delivery of basic supplies and ensuring quality products to create an Integrated Market Intelligence Platform necessary. The platform will include real-time information including importer's overall information, quantity and price of imported goods, sourcing, cost.

## **3. Project Objectives**

- Goal of balancing the overall price.
- Maintaining the demand and supply of daily products.
- Monitoring warehouse of products.
- Reducing the Human sufferings.
- Decreasing the dependency of TCB (trading corporation of Bangladesh).
- Giving the Budget and planning services of products.

## 4. Preliminary Solution

To overcome those difficulties which are describe before in the problem statement we have proposed three preliminary solution below:

Solution 1	Solution 2	Solution 3
Making a Software based system solution to monitor overall process of a good delivery from producer to consumer and up to date give information of government to take rapid action before a good market price hiking.	Increasing manpower in every sector of monitoring team such as mobile court.	All food warehouse governmentalizing, such that all private food warehouse controlled by government.

**Table 4.1:** preliminary Solution

## 5. Project Scopes

There are several facilities exist of those solutions. I will show you, all several functionalities of those solution. Now, describes those scope, facilities and functions. Let's see the table 5.1.

Solution No	Functions	Features	Facilities
01	Determining cost product, investigations, international prices ,countrywide prices, demand and import ,market prices etc.	Stocking measurement, shortages of product, convenient prices, distributes product place wise , information tariff and transportation etc.	All information are knowing government, taking step against dishonest people and monitoring prices etc.

02	Demand, import ,market prices, salary ,instrument etc.	Determining cost, salary, transportation, measurement of manpower etc.	Maintaining all and every things sectional offices.
03	Instrumental cost, transportation, market prices ,demand ,calculation of warehouse, salary, profit and sectional offices etc.	Demand, import, buying cost, market prices.	Government maintain all and every things.

**Table 5.1:** Project Scopes

## 5.1 Cost Estimation

- Development cost 50 lakh approximately.
- Budget of project 1 core.

## 5.2 Estimation Cost and Time for Feasibility Study

Feasibility study is a must before implementing any kind of project. It helps to pick up the best solution of the problem. It concentrates on Operational feasibility, Technical feasibility and Economical feasibility.

The main aim of the feasibility study is not thinking how to solve the problem, back to determine whether the problem is work solving.

Feasibility study leads to a decision:

- Go ahead
- Do not go ahead
- Think again

Without Feasibility study the project may lead to failure. We will need 2 week and BDT 1,80,000 to make a feasibility on our project. Let's see the table 5.2 for better knowing.

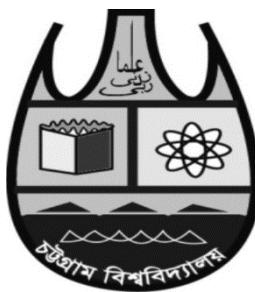
<b>Serial No</b>	<b>Tools and Requirements</b>	<b>Cost (BDT)</b>
01	Personnel Cost	1,00,000
02	Communication and Travel	30,000
03	Marketing and Administrative	20,000
04	Training	20,000
	Total	1,70,000

**Table 5.2:** Cost Estimation

Our project feasibility study will BDT 1, 80,000 taka approximately.

## **6. Conclusion**

We hope that our project will work out and it will be user friendly so that use this app perfectly.



## **Department of Computer Science and Engineering, University of Chittagong**

<b>Course Code</b>	CSE 516
<b>Course Title</b>	Software Engineering and Design Patterns Lab
<b>Assignment name</b>	Feasibility Study Document
<b>Report number</b>	02
<b>Group</b>	E

**Submitted to**

**A. H. M. Sajedul Hoque**

Assistant Professor

Department of Computer Science and Engineering,

University of Chittagong.

**Submitted by**

ID: 18701032

ID: 18701057

ID: 18701071

ID: 17701087

ID: 17701101

## 1. Introduction

Feasibility study is an assessment of the practicality of a proposed plan or method. It is the most important part in developing a software project. It leads us:

- To select the best system that meets performance requirements.
- To determine whether it would be financially and technically feasible to develop the product.
- To determine whether the problem is worth solving.
- To analyze the problem and collection of all relevant information relating to the product.

The purpose of feasibility study is not to solve the problem but to determine whether the problem is worth solving and decide whether to proceed with the project or not. Referencing to this information, the analyst does a detailed study about whether the desired system and its functionality are feasible to develop or not.

After this section, the important and necessary parts of feasibility study will be described. 1<sup>st</sup> part is the background. Background is the situation where it'll discuss the problem, need, or opportunity that has brought about this report. 2<sup>nd</sup> part is the outline of the system & it includes the description of the project. The next section or the 3<sup>rd</sup> section is methodology which involves discussion on method of analysis for feasibility study. 4<sup>th</sup> part is the overview of alternatives. Sticking with the current system, the major possible alternatives should be discussed and compared using clearly defined criteria. It will discuss the Economic, Technical and operational feasibility in details. 5<sup>th</sup> part is conclusion. It restates the individual conclusions which alternative had the best price. 6<sup>th</sup> section which is also the final part is recommendation. This part states the recommendation on one alternative emphatically.

## 2. Background

This section will describe the problem, the need and opportunity of this project.

**The problem:** The Ministry of Commerce of Bangladesh has been supplying products through TCB to ensure the necessary supply, including the import, supply and exploration of these

Products, to keep the prices of essential commodities in the public purchasing capacity. But lately, the prices of essential commodities, including onions, are unaffordable, and it is imperative to take initiative to adopt a timely solution or action plan. In coordination with the platform will include real-time information including importer's overall information, quantity and price of imported goods, sourcing, cost.

**The Need:** Controlling market fluctuations is necessary in today's corrupt market for that we need to coordinate with various government agencies by utilizing digital technology, the supply chain ensures human income by aligning with the international price of the product, allowing proper production of all essential commodities, finding suitable product sources, collecting information on tariff and transportation costs and ensuring quality products to create a necessary Integrated Market Intelligence Platform.

**Opportunity:** There are lots of reasons why a system which can control market price fluctuations is really necessary in today's market.

### **Opportunities for customers**

1. Allowing customers to have a visual information about prices and stocks of a particular product in a market.
2. Giving an opportunity to choose between alternative products if one particular product's price goes high.
3. Better understanding of insights of a particular product market.

### **Opportunities for Sellers and Governments:**

1. Sellers can refill their product stocks beforehand of any market crisis of any product.
2. Govt. Can control the market and can cut excessive price from any product.
3. Govt. Can catch the culprit businessmen who intentionally increase the product prices for illegal profits.

## **3. Outline of the Software Project**

The outline of our project is given below:

The users of the system are some government workers who are collecting information and controlling market price hiking.

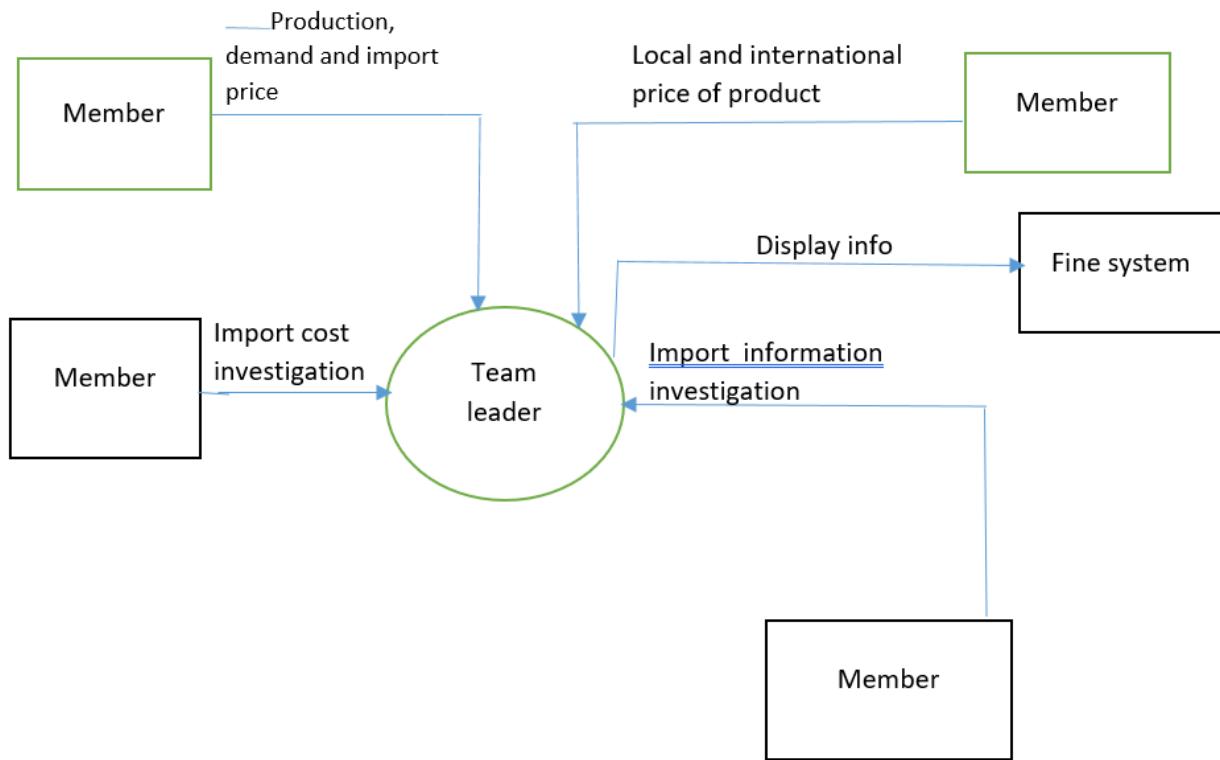
The existing system is neither cost effective nor time-efficient because people will have to meet for every reason whether they want to collect or update information. If they want to take any initiative to control the market price fluctuation time also be wasted. Thus the existing system is inconvenient. So, the following system divide workers into different part for controlling market price fluctuation easily.

The main components of the proposed system are

1. Registration
2. Login

3. Detailed info about different product (price, demand, supply ,amount of store product, source of import product, cost for import product, local and international price of product)
4. Location based search
5. Team leaders can update info
6. Members can communicate with each other by this system

#### Logical Model using DFD (Data Flow Diagram)



**Figure 1:** illustrates the Data Flow Diagram of the system.

#### **4. Method of Analysis/Methodology**

Based on the requirement of the client, we have studied our feasibility study. We have proposed two alternatives. The main objective of this feasibility study is to determine the solution based on three types of feasibilities: technical feasibility, operational feasibility, & economic feasibility.

- a) Technical Feasibility: By studying technical feasibility, we will be able to know that if the required technology to implement the alternatives is available or not.
  - b) Operational Feasibility: Operational feasibility is dependent on human resources available for the project and involves projecting whether the system will be used if it is developed or implemented. It is a measure of how well a proposed system solves the problems and takes advantage of the opportunities identified during scope definition.
  - c) Economic Feasibility: By conducting the economic feasibility, we can estimate the feedback based on our present investment. According to the cost which will be the investment on our project we can estimate the feedback of it using the formula (i),

$P = F / (1+I)^n$  ..... (i) Here P, F, n, I are present value, future benefit, year of benefit & expected rate of return, respectively.

## **5. Overview of Alternatives**

We have proposed three alternatives for the system. Alternative 1 making a Software based system solution and alternative 2 is increasing manpower in every sector of monitoring team and alternative 3 is all food warehouse governmentalizing.

## 5.1 Overview of Alternative 1

- a) **Technical Feasibility:** Our system needs a web based server which will be online 24/7. The government can monitor overall process of a daily needs goods from supplier to consumer. We can easily set up a server using MySQL and PHP which can interact with the android app, so, we can say that this system is technically feasible.
  - b) **Operational Feasibility:** As government and as well as general consumer can easily use this system, that means, the system is user-friendly, therefore the system is operationally feasible.
  - c) **Economic Feasibility:**

The investment of alternative 1 is illustrated in table I.

Serial no	Particular	Cost
01	Systems (Server + OS + workstations)	1,30,000/=
02	Personnel cost (Developer, Designer, etc)	1,20,000/=
03	DBMS	50,000/=
04	Marketing cost	50,000/=
05	Administrative cost	50,000/=
06	Initial Data Entry	1,00,000/=
	Total	5,00,000/=

**Table 5.1(a):** The investment of alternative 1

The cost/benefit of alternative 1 is illustrated in table 5.1(b)

Cost		Benefit	
Particular	Amount	Particular	Amount
Maintenance cost	70,000/=	Saving 4 persons(out of 5)	4*12*4500= 2,16,000/=
Additional cost	30,000/=	Better service time and more services	34,000/=
Total 1,00,000/=		Total 2,50,000/=	

**Table 5.1(b):** The cost/benefit analysis of alternative 1 (yearly basis)

Net return per year = (2,50,000 – 1,00,000)TK= 1,50,000 Tk.

The cost/benefit of alternative 1 is illustrated in table 5.1(c)

Year	Saving (lakh)	Present value (at 15%)	Cumulative value
1	1.5	1.30	1.30
2	1.5	1.13	2.43
3	1.5	0.98	3.41
4	1.5	0.85	4.26
5	1.5	0.74	5.00
6	1.5	0.64	5.64
7	1.5	0.56	6.20

**Table 5.1(c):** The cost/benefit analysis of alternative 1 (yearly basis)

## 5.2 Overview of alternative 2

### ➤ Technical feasibility

No other technology is used in this system. So technically feasibility issue is inconsistent.

### ➤ Operational feasibility

The solution is not operationally feasible because the problem is not worth solving completely. So the system cannot gain users satisfactions.

### ➤ Economic feasibility

After completing the economic feasibility study we know about the solution is economically feasible or not.

Table 5.2 – The cost/benefit analysis of alternative 2 (yearly basis)

Cost		Benefits	
particular	amount	particular	amount
Increasing salary	14,40,000	Consumer satisfaction	1,20,000
Bonus cost	34,000	Product Profit	10,00000
Technological cost	30,000	Machine controlling (savings)	1,00000
Travel cost	1,20,000	Additional profit	50,000
Total	16,24,000	Total	12,70,000

**Table 5.2:** The investment of alternative 2

Net return per year = $(12,70,000 - 16,24,000)TK = -3,54,000$  Tk. . Since, it has no benefit/positive return value. So, no need to calculate the present value per year

### 5.3 Overview of alternative 3

- Technical feasibility:** In this alternative solution, we can see that all warehouse need governmentalizing but technically implement this way is very tough. There are need huge people .Manually, checking all system and warehouse. Need modern technology for storing food .like as some food are possible to store timely but some food need to sell fast and take away. So, need modern instrument that is possibly tough and need more money.
- Operational feasibility:** it is useful for general people and government controlled every things. People do not hampered by the raising prices or adulteration, unhealthy condition etc. In this wa that is better for every class people. Getting benefit government and people. So, operational feasibility is useable for this alternatives.

3. **Economic feasibility:** This properties means that investment and cost effectiveness It is possible to get investment that says. Now, we get economical criterion from doing field works. See the table Figure 5.3(a)

Serial no	Particular	Cost
01	Manpower /employee	100,0000/=
02	Buying warehouse	500,000/=
03	Manufacturing	200,000/=
04	Delivery cost	100,000/=
05	Marketing prices	50,000/=
	Total	18500000/=

**Table 5.3(a):** The investment of alternative 3

Cost		Benefit	
Particular	Amount	Particular	Amount
Product transport	500,000/=	Product profit	100,000/=
Instrumental cost	800,000/=	Machine controlling(saving)	100,000/=
Additional cost	100,000/=	Additional profit	50,000/=
Total	1400,000/=	Total	250,000/=

**Table 5.3(b):** The cost/benefit of alternative 3 is illustrated (yearly basis)

Now, we calculate the net return value from the project. Total return value is given below. lets see it. Net Return value per year=  $(250,000-1400,000)$  Tk. = - 1150000TK. Since, it has no benefit/positive return value. So, no need to calculate the present value per year

## 6. Recommendation

After analyzing the alternatives in different sectors like technical, operational and economic analysis we need to recommend any one of them on the basis of different features.

The summarized comparison of alternative 1, alternative 2 and alternative 3 is shown in table 6.

<b>Serial No</b>	<b>Feature</b>	<b>Alternative 1</b>	<b>Alternative 2</b>	<b>Alternative 3</b>
01	Investment	5,00000	16,24,000	1,85,00000
02	System life cycle	7 year	continuous	continuous
03	Return value	1,20,000	Negative	Negative
04	Payback period	5 year	No	No
05	Technically feasible	Yes	Yes	Yes
06	Operationally feasible	Yes	Yes	Yes
07	Economically feasible	Yes	No	No

**Table 6:** Economic analysis between alternative 1, alternative 2 and alternative 3

So, on the basis of feasibility studies I have strongly recommendation for Alternative 1.

## 7. Conclusion

Here, we have planned three alternatives-

- Alternative 1: Making a Software based system solution to monitor overall process
- Alternative 2: Increasing manpower for monitoring overall system.
- Alternative 3: maintaining warehouse by the government.

All of these alternatives are technically and operationally feasible. But by Economical analysis, we have found that, Alternative 1 is more cost effective and efficient than Alternative 2 and Alternative 3. So finally, we have preferred Alternative 1 for our project.



## **Department of Computer Science and Engineering, University of Chittagong.**

<b>Course Code</b>	CSE 516
<b>Course Title</b>	Software Engineering and Design Patterns Lab
<b>Assignment name</b>	Project Planning Document
<b>Report number</b>	03
<b>Group</b>	E

### **Submitted to**

**A. H. M. Sajedul Hoque**

Assistant Professor

Department of Computer Science and Engineering,

University of Chittagong.

### **Submitted by**

ID: 18701032

ID: 18701057

ID: 18701071

ID: 17701087

ID: 17701101

## **1. Introduction**

French writer Antoine de Saint-Exupéry quoted that, “A goal without a plan is just a wish”. The main objective of the project is to develop a system called MarketWatch which will monitor the current market situations like the price of products and what number of products are available in stock, what are the odds of increasing or decreasing of price of products. Overall the main purpose of our project is to control market price fluctuations and giving the control of markets to the govt. in efficient and completely digitalized way.

The main purpose of this report is to plan the project according to client needs, it defines the roles and responsibilities of the project management team members, it ensures that the project management team works according to the business objectives, it checks feasibility of the schedule and user requirements, it determines project constraints, estimate the whole costs, break the whole project tasks into smaller activities, schedule the project development and analyze different types of risks and prepare preliminary solutions to the problem. The budget needed for developing the system is about 3,00,000 BDT and the time needed for completing the project is approximately 3 months.

This report covers eight sections. Section 2 discusses how the development team is organized and the team member’s role in the team. Section 3 illustrates about possible project risks, the likelihood of these risks, strategies for risk reduction. Hardware and software resource requirements are discussed in section 4. In section 5, the breakdown of the project into activities, milestones, and deliverables are discussed. Section 6 is on a project schedule that shows dependencies between activities, estimated time to reach each milestone and allocation of people to activities. Section 7 discusses monitoring and reporting mechanisms. The concluding remarks are given in Section 8.

## **2. Project Organization**

Organizational Structure determines the relationship between functions and positions as well as subdivides and assigns roles, responsibilities, and authorities to carry out different tasks. A proper organizational structure is needed to balance many tasks efficiently and effectively. The project tasks are divided into some activities and the project team consists of some specific groups. The role of the team members in the development process of the system is given in Table i.

Serial No	Name	Roles	Responsibilities
01	Md. Akram Hossain	Project Manager	<ul style="list-style-type: none"> <li>1. Lead, manage and coordinate the whole project team.</li> <li>2. Recruit project staffs and consultants.</li> <li>3. Develop and maintain a detailed project plan.</li> <li>4. Manage project deliveries within constraints.</li> <li>5. Monitor project progress and performance.</li> </ul>
02	MD Rofiqul Islam	Analyst	<ul style="list-style-type: none"> <li>1. Analyze the requirement of the system and system design.</li> <li>2. Create and deliver Software Requirement Specification Document (SRS).</li> </ul>
03	Abdul Aziz	Designer	<ul style="list-style-type: none"> <li>1. Design the system as well as the database structure.</li> <li>2. Create Design Document (DD).</li> </ul>
04	Md Mohsin, Md. Akram Hossain	Coder	<ul style="list-style-type: none"> <li>1. Allocate necessary resource needed.</li> <li>2. Implement the system according to the design document</li> </ul>
05	Md Mohsin, Md. Alamin	Tester	<ul style="list-style-type: none"> <li>1. Check the system for errors.</li> <li>2. Suggests improvement for the system.</li> </ul>

**Table i:** Role of Team Members

### **3. Risk analysis**

Project risk management is the process of identifying, analyzing and then responding to any risk that arises over the life cycle of a project to help the project remain on track and meet its goal. Risk management isn't reactive only; it should be part of the planning process to figure out risk that might happen in the project and how to control that risk if it in fact occurs. A risk is anything that could potentially impact your project's timeline, performance or budget. Risks are potentialities, and in a project management context, if they become realities, they then become classified as "issues" that must be addressed. So risk management, then, is the process of identifying, categorizing, prioritizing and planning for risks before they become issues.

Risk management can mean different things on different types of projects. On large-scale projects, risk management strategies might include extensive detailed planning for each risk to ensure mitigation strategies are in place if issues arise. For smaller projects, risk management might mean a simple, prioritized list of high, medium and low priority risks.

#### **3.1 Risk Identification**

In this step, the possible risk the project may encounter are identified. This process includes analyze the possibility of the risk and the effect of the risk if occurred. Table ii lists such risk identification and their likelihood and effects are described in Table iii.

<b>Serial no</b>	<b>Risk name</b>	<b>Risk type</b>	<b>Affects</b>	<b>Description</b>
01	Requirements change	Requirement	Project and product	Changes to requirements that require major design rework is proposed.
02	Team member turn over	people	Project	Experienced member leaves before its finished.

03	Hire new staff	People	Project and product	New member acquired and also provide training
04	Necessary hardware unavailable	technology	Project	Essential hardware must be need.
05	Solve customers' complaints	People	Product	Any customer or user can complaints products
06	Compete with similar products	technical	Product	For all similar company can provide better services.
07	Database not working	technology	Project and products	Database can be failure for any operations or any complexity
08	Time estimation	estimation	Project	Time required to develop the software is underestimated
09	Money problem	organizations	Projects	Organizational money crisis can effect for this projects

**Table ii:** Risk identification

<b>Serial no</b>	<b>Risk name</b>	<b>Possibility</b>	<b>Affects</b>
01	Requirement changes	Low	Catastrophic
02	Member turn over	High	Serious
03	Hire staff	Low	Tolerable
04	Necessary hardware unavailability	Moderate	Catastrophic
05	Solve customers complaints	High	Serious
06	Compete with similar products	High	Serious
07	Database not working	High	Serious
08	Time estimation	High	Serious
09	Money problem	Moderate	Tolerable
10	Software integration problem	Moderate	Tolerable

**Table iii:** possibility and effects of risk.

### 3.2 Risk reductions strategies

The analysis of the risk is not sufficient. The strategies have to be devised for the time the risk arises to minimize its effects. The risk reductions strategies are described in Table IV.

<b>Serial no</b>	<b>Risk name</b>	<b>Reductions strategies</b>
01	Requirement changing	Discussing the impact of the requirement change with customers.
02	Member turn over	Member must be motivated, providing the high salary, security and suitable environment.
03	Hire staff	Hiring staff providing money with his/her skillfulness.
04	Necessary hardware	It must be buy for projects and apply technique to reduce the money.
05	Solve the customer complaints	At the easily feedback system are exist so that we can solve the customer complaints.
06	Compete with similar products	For better products supplying must be competitive and follow the advanced technology, hardware, facilities and services.
07	Database not working	Must checking the system before the delivery within testing data and hardware devismement.

08	Time estimation	Time estimations must be calculated before starting the working and end the task before delivery date.so, distribution must be well defined.
09	Money problem	Having powerful donor for any leakages of hardware and staff hiring, training etc.
10	Software integration	Software integrated must require for any complexity before testing functions, facilities and changing the requirements.

**Table IV:** Risk reductions strategies

#### **4. Hardware and Software Resource Requirement**

For developing an effective system some hardware and software are needed. Without appropriate hardware and software, a project continuation may be endangered. Good quality hardware ensures environment stability as well as system performance and good software helps rapid development process and reduces the possibility of system crashes. Table 4.1 lists the hardware and software resources needed for the project.

Resource Type	Serial No	Resource Name	Quantity
Hardware	01	Personal Computer (Intel Core i5 8600K 3.6 GHz, 8GB DDR4 RAM, 1000 GB HDD)	05
	02	LAN Connection (20 Mbps)	01
	03	Hard Drives (250 GB SSD)	03
	04	Keyboard and Mouse	05
Software	05	Operating System	05
	06	Database Management System	01
	07	Android Studio Bundle	01

**Table 4.1:** List of hardware and software resources

#### **5. Work Breakdown**

Our project have many collection of work. So we need to divide those work into some smaller units of task. It is important to set milestone. It also help to gain the milestone and produce the

deliverables in time. Task of the project are summarized in table vi and table vii outlines the milestones and deliverables.

<b>Task</b>	<b>Name of Tasks</b>	<b>Achievement</b>
T1	Taking opinion from Staffs	
T2	Arranging meeting with Stakeholders	
T3	Defining Function and Facilities	
T4	Preparing System Requirement Specification Document	Milestone(M1)
		Deliverable(D1)
T5	Designing System Structure	
T6	Designing Database Structure	
T7	Establish connection between System interface and the database	
T8	Preparing Design Document	Milestone(M2)
		Deliverable(D2)
T9	Allocate Resources	
T10	Database Specification	
T11	Training staffs with new technology	
T12	Implementation of the system	Milestone(M3)
T13	Testing and Debugging of the system	Milestone(M4)
T14	Preparing Installation and Implementation Manuals	Milestone(M5)
T15	Delivering of the system	Deliverable(D3)

**Table VI:** Task management of the whole system

Type	Name	Description
Milestone	M1	Analysis of the system completed
	M2	Design of the system completed
	M3	Implementation of the system completed
	M4	Complete system testing and debugging completed
	M5	Installation and Implementation Manuals completed
deliverable	D1	System Requirement Specification (SRS)
	D2	Design document (DD)
	D3	Complete system

**Table vii:** Milestones and deliverables

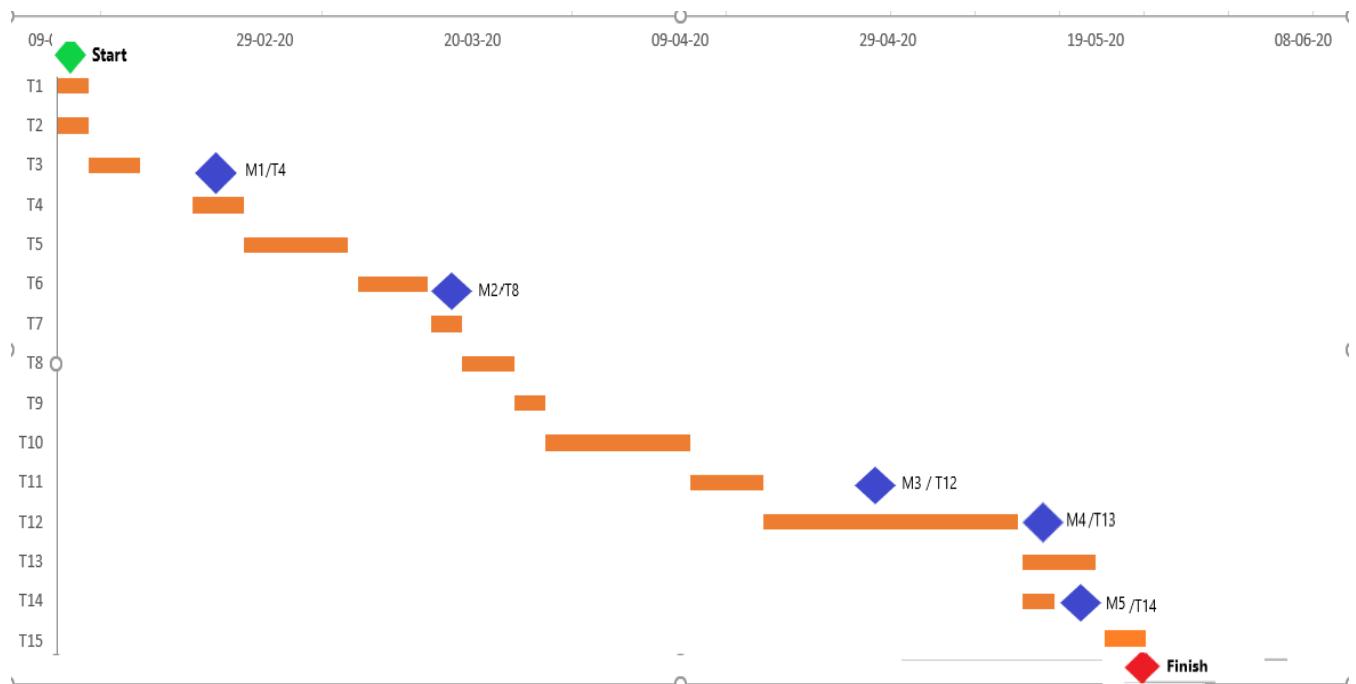
## 6. Project schedule

Project scheduling is a mechanism to communicate what tasks need to get done and which organizational resources will be allocated to complete those tasks in what timeframe. A project schedule is a document collecting all the work needed to deliver the project on time. The number of persons assigned to the task, the duration of that task and dependencies of the tasks is included in schedule representation which is shown in table viii

Tasks	Effort (Person – Days)	Start Date	Duration (Days)	Dependencies
T1	07	February 09, 2020	03	
T2	03	February 09, 2020	03	
T3	05	February 12, 2020	05	T1,T2
T4	03	February 22, 2020	05	T3(M1 &D1)
T5	05	February 27, 2020	10	T4
T6	05	March 09, 2020	07	T4 (M1)
T7	05	March 16, 2020	03	T5,T6
T8	05	March 19, 2020	05	T7(M2)
T9	03	March 24, 2020	03	T8(M2 &D2)
T10	05	March 27, 2020	14	T8(M2)
T11	02	April 10, 2020	07	
T12	05	April 17, 2020	25	T8(M3)
T13	03	May 12,2020	07	T8 (M4),T9, T10
T14	02	May12, 2020	03	T13(M5)
T15	02	May 19, 2020	07	T13(D3)

**Table viii:** Schedule Representation

The task dependency and milestone are best represented through a graphical chart called ‘Gantt chart’ which is a bar diagram. Gantt chart of the project is demonstrated in Figure 1

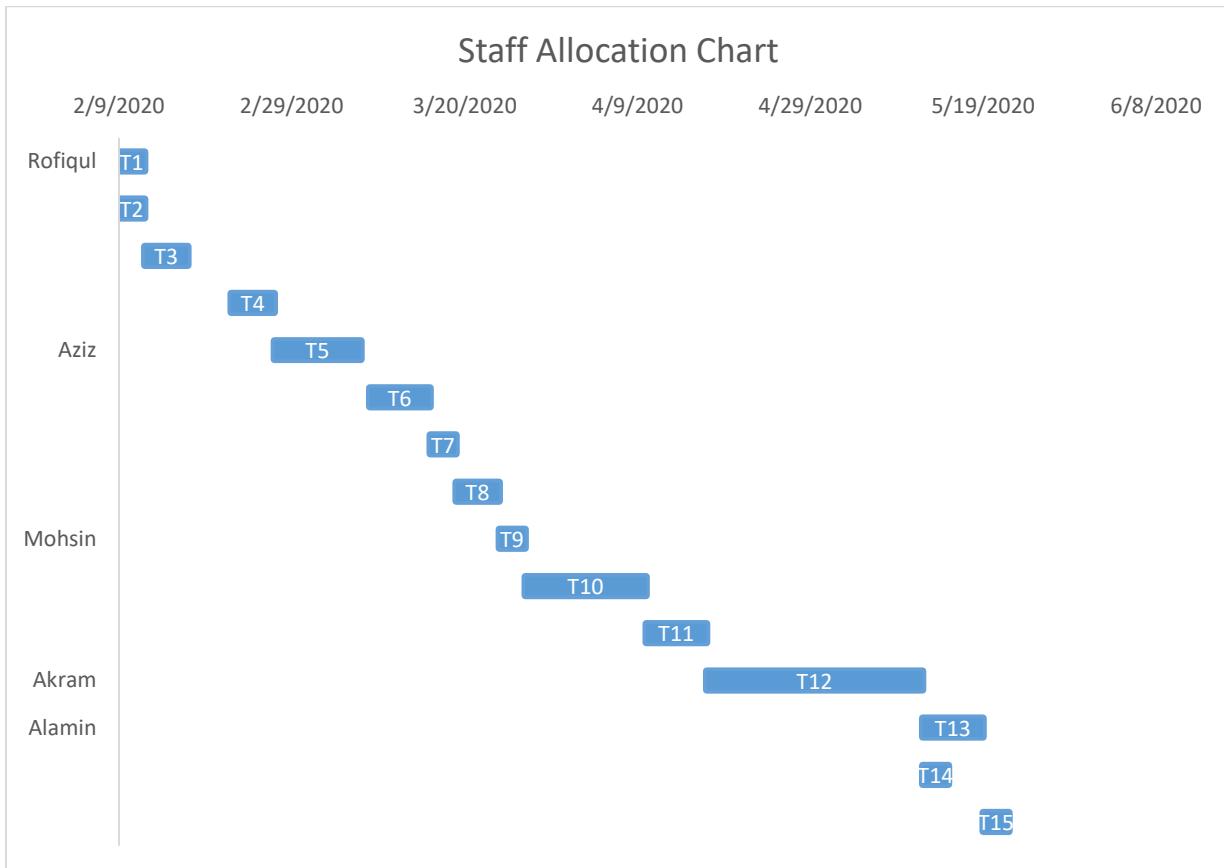


**Figure 1:** Gantt Chart

The tasks are distributed among project members to make the work done efficiently and effectively. The Task distribution is shown in Table ix and staff allocation is shown graphically in Figure 2

Serial No	Task	Assigned Member
1	T1, T2, T3, T4,	Md Rofiqul Islam
2	T5, T6, T7, T8	Abdul Aziz
3	T9,T10,T11	Mohsin Khan
4	T12	Md. Akram Hossain
5	T13,T14,T15	Md. Alamin

**Table ix:** Task Distribution



**Figure 2:** Staff Allocation Chart

## **7. Monitoring and Reporting Mechanism**

Project Monitoring refers to the process of keeping track of all project-related metrics including team performance and task duration, identifying potential problems and taking corrective actions necessary to ensure that the project is within scope, on budget and meets the specified deadlines. To simply put, project monitoring is overseeing all tasks and keeping an eye on project activities to make sure we are implementing the project as planned.

It is systematic and purposeful observation. Monitoring also involves giving feedback about the progress of the project to the donors, implementers and beneficiaries of the project. Monitoring is very important in project planning and implementation. Reporting enables the gathered information to be used in making decisions for improving project performance.

A project manager can monitor the project in a various way. The project can be monitored in following way:

- Monitoring the progress of the project daily by collecting information about progress, problems, and difficulties.
- Holding meetings weekly and monthly with the project staffs where everyone will show their progress.
- Comparing the project environment with potential risk indicator situations.
- Comparing the performance of the staff with a standard and take necessary action to correct it.

The project staff can report to the project manager as mentioned below:

- The project staff will generate and deliver important documents to the manager.
- The project team will notify the manager of the completion of the task.
- The staffs will inform and generate a report about the milestones completed.

## **8. Conclusion**

A plan the first step in completing a project successfully. This report has provided all the information about project organization, risk analysis, hardware and software requirements, work breakdown, scheduling, monitoring and report mechanism of the project. All those tasks are so important for the completion of the project. This will help the project staffs to complete the project in time. Finally, we expect that this project to be completed successfully.



## **Department of Computer Science and Engineering, University of Chittagong.**

<b>Course Code</b>	CSE 516
<b>Course Title</b>	Software Engineering and Design patterns Lab
<b>Assignment name</b>	System Requirement Specification Document
<b>Report number</b>	04
<b>Group</b>	E

### **Submitted to**

**A. H. M. Sajedul Hoque**

Assistant Professor

Department of Computer Science and Engineering,

University of Chittagong.

### **Submitted by**

ID: 18701032

ID: 18701057

ID: 18701071

ID: 17701087

ID: 17701101

## **1. Preface**

This is 'System Requirements Specification' document (generally known as 'SRS' document) of a software system named 'MarketWatch'. It is primarily intended to be proposed to a customer for its approval and a reference for developing the first release of the system for the development team.

With version 1.0, the users will experience a completely stable release that includes high authentication of the users like Admin, Staff of Agriculture department, staff of TCB, staff of NBR, staff of CCI&E. So, there is no worry about unauthenticated users. The version 1.0 will also provide a very secure and encrypted data submission and preview. Users will experience a very fast, smooth, and accurate and reliable system using this release.

## **2. Introduction**

A software requirements specification (SRS) is a description of a software system to be developed. It lays out functional and non-functional requirements, and includes a set of use cases that describe user interactions that the software must provide. Software requirements specification establishes the basis for an agreement between customers and contractors or suppliers on what the software product is to do as well as what it is not expected to do.

The purpose of this document is to give a detailed description of the requirements for the “Controlling market price fluctuation system”. This document will completely illustrate the purpose and the features of the system. It will provide a complete declaration for the development of the system in precise and explicit manner. It will also explain system constraints, what system will do and how the system will react to external interactions.

The “Controlling market price fluctuation system” is an online based system which will monitor the current market situations like the price of products and what number of products are available in stock, what are the odds of increasing or decreasing of price of products. Overall the main purpose of our project is to control market price fluctuations and giving the control of markets to the govt. in efficient and completely digitalized way.

The document contains ten sections. Section 1 gives preface to this document while section 2 introduces the readers with the system as well as with this document. Section 3 familiarize the readers with the technical terms used in this document. Section 4 narrates the user requirements briefly while section 5 gives a brief description to the system architecture. The specific description of user requirement is described in section 6. Use case scenarios of the system are given in section 7 and section 8 bears the anticipated change or evolution of the system due to hardware changes. Section 9 and section 10 contains the appendices and index for this document consecutively.

### 3. Glossary

Glossary is an alphabetical list of terms in a particular domain of knowledge with the definitions for those terms. It lists the technical terms used in the document. The glossary for this document is given in table I.

Technical Term	Description
Authentication	The process or action of verifying the identity of a user or process.
Backup	A copy of a file or other item of data made in case the original is lost or damaged.
Constraints	The limiting barrier of an action or a system.
Credentials	A group of information proving a user's identity or qualifications.
Database	A collection of information organized into rows, columns and tables, such a way that a computer program can quickly access, manage or update desired pieces of data.
Encryption	The process of converting information or data into a code, especially to prevent unauthorized access.
Login	The process by which an individual gains access to a computer system by identifying themselves.
On-line	Operating being connected to a computer or telecommunication system such as internet.
Response Time	The length of time taken for a system to react to a given event.
Server	A computer or computer program which manages access to a centralized resource or service in a network.

**Table I:** Glossary

### 4. User Requirements Definition

Requirements are physical or functional need that a particular design, product or process aims to satisfy. After meeting with the client and properly discussing with them, some requirements are discovered. The requirements are divided into two categories such as, *functional requirements*, which defines the functions of the system required by the client, and, *nonfunctional requirements*, which defines the characteristics as well as constraints of the system.

The user requirements are defined in table II.

<b>Requirement Type</b>	<b>Definitions of Requirement</b>
Functional Requirement	1. Admin will distribute information-scripts against other actors.
	2. Actors will be able to submit information his/her information-scripts.
	3. Without Admin other Actors will be able to view information against his/her information-scripts and Admin will be able to see whole system information.
	4. Admin will be able to take action against any inconsistency of all actors given information.
	5. Admin will be able to make a decision to use the system.
	6. Admin will be able to modify information of whole system.
Non Functional Requirement	7. System must be highly secure and reliable.
	8. Every operation must be protected by authentication.
	9. System must response quickly.
	10. Software must require low hardware resources to operate.
	11. Every user must be able to performed required operations online.
	12. Software must be operable on Android Mobile Phone.

**Table II:** Definition of User Requirements

## 6. User Requirements Specification

The software requirements specification document enlists necessary requirements that are required for the project development. To derive the requirements, the developer needs to have clear and thorough understanding of the products to be developed. Also, the requirements must be described at length for making the client clear and concise about what is going to be developed. The user requirements are described in great detail in table III.

<b>Resource Type</b>	<b>Specification of Requirements</b>
Functional Requirement	<p>1.1. The software will check the overall information and show the button of red, green and black product.</p> <p>1.2. The software will generate red and green button product information script.</p> <p>1.3. Admin will give the generated information scripts against each department.</p>
	<p>2.1 Staff will input marks against each answer-script he is given to.</p> <p>2.2 staff will submit the marks to the central server</p>
	<p>3.1 Actors can only see the information available for public viewing.</p> <p>3.2 Actors can see the information they've submitted.</p> <p>3.3 Admin can view all the information of the whole system.</p> <p>3.4 Admin can also view the user submitted information.</p>
	<p>4.1 Checking the information validation.</p> <p>4.2 Having any wrong or mismatch then command to change it.</p> <p>4.3 Checking the product demand and supply information.</p> <p>4.4 Views the measurement information of product.</p> <p>4.5 Seeing the country price and information price.</p> <p>4.6 Viewing the actor or user details information.</p> <p>4.7 Views the transportation of cost.</p>
	<p>5.1 admin will login to his own account using his own ID.</p> <p>5.2 admin will check is their have any inconsistency in the system.</p> <p>5.3 Admin only can use make decision when system detect any inconsistency.</p>
	.

Functional-Requirement	<p>6.1 Actors view their information      6.2 Checking the information      6.3 Modify the data within Authentication      6.4 Refreshing the page      6.5 Admin gets the modification messages</p>
	<p>7.1 Highly secured RSA and PGP encryption will be used to send and receive data to and from the central server.      7.2 High performance backup server must be configured.      7.3 Failure rate must be as low as possible.      7.4 Startup must be rapid.      7.5 Quick re-spawning capability must be implemented.      7.6 Without Admin other Actors cannot view information unless it is not information of his/her department.</p>
Non-Functional Requirements	<p>8.1 Every actor must have a unique user-account.      8.2 Each user will have to login his own account submitting his credentials.      8.3 No operation will be allowed to unauthorized actors.      8.4 Only main controller can generate, assign and distribute works.      8.5 An staff can submit information's of his corresponding working site only.      8.6 Submission of information's must be approved by main controller.      8.7 Only main controller can publish information's.      Transcript generation option will be shown to exam controller only.</p>
	<p>9.1 Overall response time will be less than 1 second.      9.2 System must be able to serve at least 300 users simultaneously.      9.3 System must be able to generate 15 transcripts per second.</p>
	<p>10.1 System must be low power consuming.      10.2 The software must be operable with a minimum of 900MHz dual core CPU.      10.3 Software must be restricted within 512MB of RAM.</p>
	<p>11.1 Users will be able to connect their device to the server from anywhere.      11.2 Examiners will be able to submit marks using Internet connection.</p>

	11.3 Ability to submit marks through a LAN will also be provided
	<p>12.1 The software must be able to run on android API 19 or higher.</p> <p>12.2 User-interface must be nice-looking and user-friendly.</p> <p>12.3 The software must be able to run in a device with 480dpi screen.</p>

**Table III:** Specification of User Requirements

## 7. System Model

System model describes the system through some scenarios. A scenario is a narration of measurable interactions of user and the technical system, which is usually, includes computer hardware and software. It is useful for adding details to the requirement description. The system is described narratively by use cases and is presented graphically by use case diagram.

### 7.1. Use Case

Use case is a list of actions or event steps typically defining the interactions between a role and a system to achieve a goal. Use cases are important to understand how the system interacts with the user or other systems. The use cases of the system is listed in table IV.

Use Case	Title
UC1	Login to the system
UC2	Information scripts distribution
UC3	Submit information
UC4	view information
UC5	Take action against any inconsistency
UC6	Make a decision
UC7	Modify Information

**Table IV:** List of Use Cases

**UCI:** Login to the System.

Actors:

- 1 . Admin
2. Staff of Agriculture department
3. Staff of TCB department
4. Staff of NBR department
5. Staff of CCI&E department

Preconditions:

1. Device must be powered on.
2. Software must be installed in the device.
3. Device is unlocked and in standby mode.
4. Device is connected to the central server via LAN or Internet.
5. Staff of Agriculture department, staff of TCB, staff of NBR, staff of CCI&E  
Must be already stored in the database.

Main Success Scenario:

1. Click *main menu* icon of the device.
2. Phone shows all installed applications.
3. Click *MarketWatch System* application icon.
4. The application will be launched and shows *login* options.
5. Click the proper option from these five options:
  - *Login as Admin.*
  - *Login as staff of Agriculture department.*
  - *Login as staff of TCB department.*
  - *Login as staff of NBR department.*
  - *Login as staff of CCI&E department.*
6. Make sure you permit for login if you are not login as a Admin
7. Phone shows *login* window.
8. Enter *username* and *password*.
6. Tap the *Login* button and wait for a while.

Post Condition: Phone shows logged-in user dashboard.

Alternative Course:

- 6.a. Phone does not show user dashboard.

6. a.1. Make sure you selected correct login option.
6. a.2. Follow step 6
6. a.3. Press the *back* button and select correct login option.
- 6.a.4. Enter correct *password* and *username*.
- 6.a.5. Tap *Login* button.

## **UC2:** Information-Script Distribution

Actor:

- 1 . Admin

Preconditions:

- 1 . Admin is logged in.
2. The phone is unlocked.
3. Phone is showing Admin dashboard.

Main success scenario:

1. Click on Distribute Information-Scripts Option.
2. Phone shows Agriculture department, TCB department, NBR department, CCI&E department selection window.
3. Select one of the option (Agriculture department, TCB department, NBR department, CCI&E department) from drop-down menu.
4. Tap *Choose File* button.
5. Choose proper information-script file for each department.
6. Click *submit* button.

Post Condition:

Phone shows "script were successfully sent to server" message.

Alternative Course:

- 2.a. Phone shows no option (Agriculture department, TCB department, NBR department, CCI&E department).
  - 2.a. 1 . Make sure your connection to server is OK.
  - 2.a.2. Tap *Refresh* button.
- 3.a. Don't select any option.
  - 3.a.1. make sure you select the proper option of them.
  - 3.a.2. click the option which option you want to select.
- 5.a. Accidentally I don't choose any file.
  - 5.a. 1 . Click on *choose File* button.
  - 5.a.2. Follow step 5-6.

### **UC3: information's Submission**

Actor:

1. Staff of agriculture department
2. Staff of TCB
3. Staff of NBR
4. Staff of CCI&E

Preconditions:

1. User is logged in.
2. The phone is unlocked.
3. Phone is showing user dashboard.
4. Connection to the server is OK.

Main Success Scenario:

1. Click on Submit button.
2. Phone shows a form to enter information-script code and information.
3. Enter information-script code and corresponding information.
4. Click *Add* button.
5. Go to step 2 to enter more information.
6. Click *done* button.
7. Phone asks for confirmation to send information to the server.
8. Click *Yes*.

Post-Condition: Phone shows “Information Submission successful”

### **UC4: View information**

Actors:

1. Staff of Agricultural Dept.
2. Staff of TCB
3. Staff of NBR
4. Staff of CCI&E
5. Admin

Preconditions:

1. Staff/Admin must be logged in.
2. The phone must be unlocked.
3. The phone is showing the front page of the system.
4. The device must be connected to the Internet.

Main success scenario:

1. Go to system website or open the system app.
2. Navigate to homepage of the system.
3. The device fetches the information from the server.
4. Click on specific product
  - View information as Admin
  - View information as Staff of Agricultural Dept.
  - View information as Staff of TCB
  - View information as Staff of NBR
  - View information as Staff of CCI&E
5. Make sure you permit for viewing information if you are not a member as a admin.
6. The device shows the details of that product.

**Post Condition:**

Phone successfully views the information of all the products available in the database.

**Alternative Course:**

- 3.a: phone doesn't show the products information
  - 3.a.1: make sure the actor/admin is logged in.
  - 3.a.2: users/admin has the access to view the specific products.
- 4.a: Clicking on specific product shows nothing
  - 4.a.1: make sure the item is clicked precisely.
  - 4.a.2: make sure the product is available on the database.
  - 4.a.3: make sure you choose the proper option.
  - 4.a.4: follow steps 5 to 6

#### **UC5:** Taking the action any inconsistency

**Actors:**

- (1) Admin

**Precondition:**

1. Admin login the mobile.
2. Phone is unlocked.
3. Phone is showing data.
4. Connection is ok.

**Main scenario:**

1. Click action inconsistent button.
2. Showing the all sectional report.
3. Showing the previous report sheet.
4. Matching between them.
5. Checking the differentiation and mismatching.
6. Identifying the inconsistent departments, wrong, error, mismatching etc.
7. Go to the views information
8. Checking the information.
9. Checking the previous history sheet of Specific views information of error sections.
10. Recommending information for mismatching.
11. Go to the decision button.

Post condition:

1. Phone shows the inconsistency sheet so that Government see the inconsistency.

Alternative course:

- 5.a phone does not show the inconsistency.
- 5.a.1 Wait for a while.
- 5.a.2 Click Refresh button.
- 5.a.3 Restart the application and retry use case 5

## **UC6: Make a decision**

Actor:

- 1 . Admin

Preconditions:

- 1 . User is logged in.
- 2.The phone is unlocked.
- 3.Phone is showing user dashboard.
- 4.Connection to the server is OK.
5. Other actors have done their work.

Main Success Scenario:

- 1 . Click on *Make Decision* icon.
2. Phone fetches and shows list of tasks what we need to do control market products price.
3. Select a task.
4. It will show details about the task and also show future statics of market after the task done.
5. Click Save button to save this decision report on local device .

Post Condition: Phone sends command to the server and shows “success” popup message.

Alternative Course:

2.a. Phone does not show any task list.

    2.a.1 . Wait for a while.

    2.a.2. Click Refresh button.

### **UC7: Modify the information**

Actor:

1. Admin
2. Staff of Agriculture department.
3. Staff TCB.
4. Staff of CCI and E
5. Staff of NBR.

Precondition :

1. Admin login the mobile.
2. Phone is unlocked.
3. Phone is showing data.
4. Connection is ok.

Main scenario:

1. Actor Click the Modify Button.
2. Shows their specific information within their department.
3. Checking the actual value and error defined.
4. Update their specific chart.
5. AS an admin enter all pages without the login or register.
6. Then show the all report of department.
7. Goto the view option.
8. Sheet shows the all information of department.

Post condition:

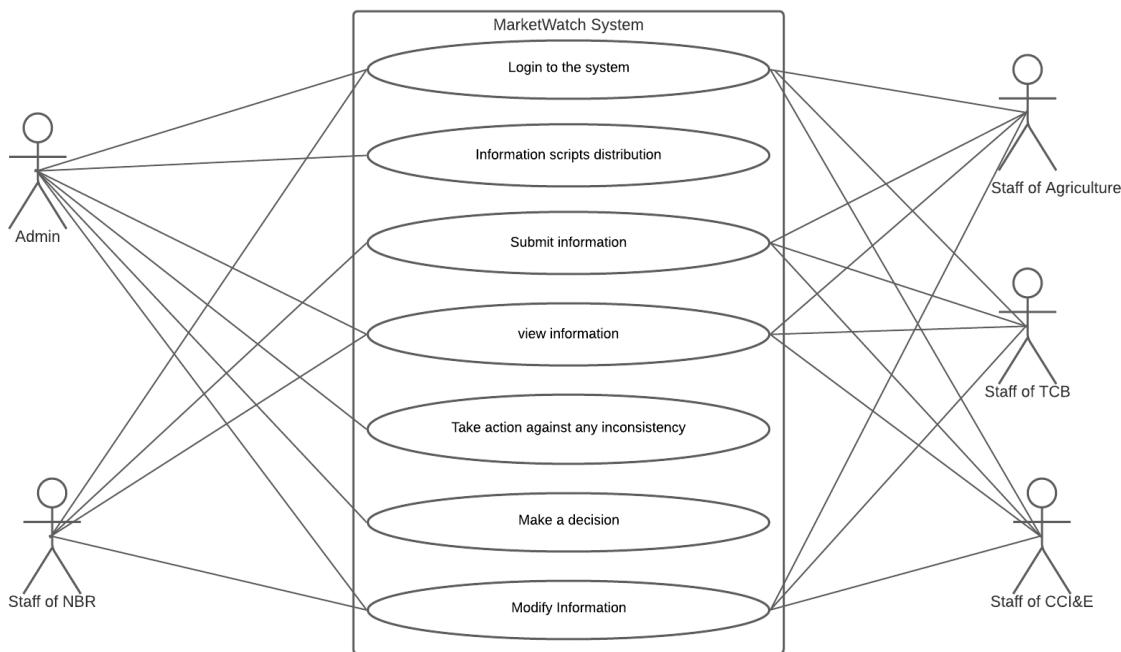
1. Server automatically updating the information of all section.
2. Show their updating information so that modification show of all department.

Alternative course:

- 7.a. phone does not show the inconsistency.
- 7.a.1 Wait for a while.
- 7.a.2 Click Refresh button.
- 7.a.3 Restart the application and retry use case 7

## 7.2. Use case diagram

The use case diagram shows the interaction between user and the system graphically. The use case diagram is shown in figure 7.1.



**Figure 7.1:** Use Case Diagram

## 8. System Evolution

Over time, software systems, programs as well as applications, continue to develop. These changes will require new laws and theories to be created and justified. Software evolution is the term used in software engineering to refer to the process of developing software initially, then repeatedly updating it for various reasons.

*Controlling market price fluctuation* is developed to be an adaptive system. It is implemented in such way that it adjusts its performance with respect to the specification of the hardware such as servers. With version 1.0, the system can process as much as 500 requests per second for a server with processor speed of 5 GHz, consisted of 4 logical cores and maximum memory of 4GB. Any

change to the hardware would change the performance of the system in proportion to the change of processor speed, number of cores and maximum memory.

## **9. Appendices**

Appendices contains the texts that is explanatory, statistical, or bibliographic in nature. The appendix for this document contains the hardware specification, database specification for the system.

### **9.1. Appendix A: Hardware Specification**

The system is developed using the server “HPE ProLiant ML10 Gen9 Tower Server”. The specification of the server is given in Table V:

Processor	Intel® Xeon® E3-1225 v5
Number of Processors	1
Processor Core Available	4
Processor Cache	8MB (1 x 8MB) Level 3 cache
Processor Speed	3.3GHz
Chipset	Intel® C236 Chipset
Power Supply Type	300W Multi-Output Power Supply
Memory	4GB DDR4
Memory Slots	4 DIMM slots
Memory Type	1R x8 PC4-2133P-E-15
Memory Protection Features	Un-buffered ECC
Included Hard Drives	LFF SATA; 1TB
Maximum Internal Storage	24TB
Optical Drive Type	SATA 9.5mm DVD RW
System Fan Features	Non-Pluggable Fan
Network Controller	Intel® Ethernet Connection I219-LM
Storage Controller	Integrated SATA RAID
Infrastructure Management	Intel® Active Management Technology (Intel® AMT 11.0)

**Table V:** Server Specification

### **9.2. Appendix B: Database Specification**

The system uses the “MySQL Enterprise Edition” as database management system. The technical specification of the database management system is given in Table VI.

Version	5.7
Data Type	Static
Architecture	Relational Model
Operating System	Linux, Solaris, FreeBSD, Mac OS, Windows
Software License	GNU General Public License
Security	<ul style="list-style-type: none"> <li>• SSL Support</li> <li>• Built-in Data Encryption/Decryption</li> <li>• View Support</li> <li>• Triggers for auditing</li> <li>• Query Logs for auditing</li> </ul>
Access Control	<ul style="list-style-type: none"> <li>❖ Enterprise Directory Compatibility</li> <li>❖ Native Network Encryption</li> <li>❖ Run Privilege</li> <li>❖ Security Certification</li> </ul>
Indexes	<ul style="list-style-type: none"> <li>• R-/R+ Tree</li> <li>• Hash</li> <li>• Full-text</li> <li>• Spatial</li> </ul>
Partitioning	<ul style="list-style-type: none"> <li>▪ Range, Hash, List, Key</li> <li>▪ Composite</li> <li>▪ 8k partitions per table</li> <li>▪ Portable partitions between tables</li> <li>▪ Explicit querying by partition</li> <li>▪ Transparent Pruning</li> </ul>
Max Database Size	Unlimited
Max Table Size	256 TB (MyISAM) 64 TB (Innodb)
Max Row Size	64 KB
Max Column Per Row	4096
Max Blob/Clob Size	4 GB
Max CHAR Size	64 KB
Max NUMBER Size	64 Bit

**Table VI:** Database Specification

## **10. Index**

### A. List of Figures

Figures are graphical representation of information. The figures used in this document is listed in table VII.

Figure Name	Name of Figure	Page No
Figure 5.1	System Architecture	23
Figure 7.1	Use Case Diagram	31

**Table VII:** List of Figures

### B. Alphabetical Index

#### A

Authentication, 21-23

#### F

Failure Rate, 25

#### C

Constraint, 21-22

#### L

Credentials, 22,25

Login, 22, 24-27

#### D

Database, 22, 26, 32, 33

Online, 22-23

Operation, 23,25

#### O

#### E

Encryption, 22, 25,33

,

#### R

Requirement,

    System Requirement, 21

    Functional Requirement, 21-24

    Non-functional Requirement, 21-22

    User Requirement, 21-24

Response Time, 22, 25

S

Scenario, 22, 26-31

Server, 22, 26-30, 32

Software, 21, 23-26, 32-33

System Architecture, 22-23

System Model

Conceptual Model, 23

U

Use Case, 21, 22, 26, 31



## **Department of Computer Science and Engineering, University of Chittagong.**

<b>Course Code</b>	CSE 516
<b>Course Title</b>	Software Engineering and Design patterns Lab
<b>Assignment name</b>	Conceptual Design Document
<b>Report number</b>	05
<b>Group</b>	E

**Submitted to:**

**A. H. M. Sajedul Hoque**

Assistant Professor

Department of Computer Science and Engineering,  
University of Chittagong.

**Submitted by**

ID: 18701032

ID: 18701057

ID: 18701071

ID: 17701087

ID: 17701101

## 1. Introduction

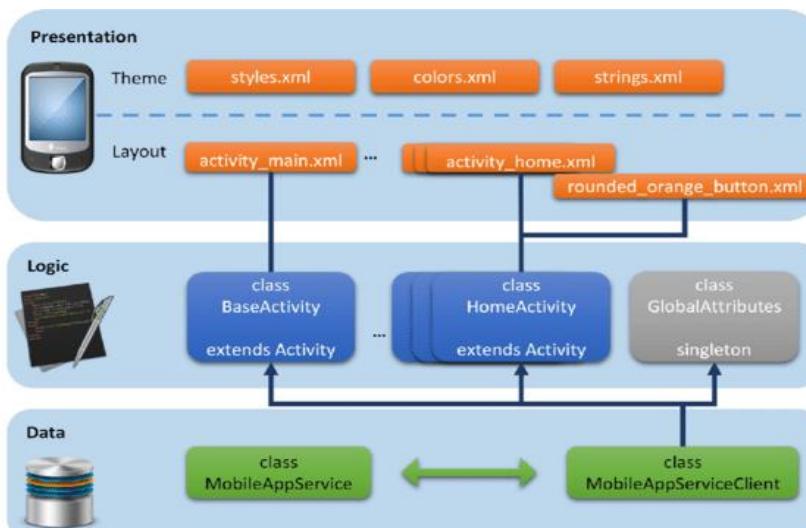
Market Watch System (MWS) is a computerized system developed for automating government market monitoring quickly and effectively to reduce the lengthy-time processing for finding why the price of a daily good are hiking and also enable the administrators to monitor the progress of the market condition and the related department work also which give real-time information to take effective steps to improve their performance. The purpose of this document is to provide design details of the Market Watch System (MWS). This document includes the architectural design of the system and also the system model from external, structural, interactional and behavioral perspectives.

This document is composed of seven sections. Section 2 describes the architectural pattern of the system, activity diagram of the system is shown in section 3, Section 4 shows the class diagram of the system, Sequence diagram for the system is illustrated in section 5, Section 6 represents the state machine diagram of the system, the conclusion for this document is given in section 7.

## 2. Architectural Pattern

An architectural pattern is a general, reusable solution to a commonly occurring problem in software architecture within a given context. An architectural pattern defines the systems in terms of structural organization, components, connectors and constraints on how they can be combined. It also addresses various issues in software engineering, such as performance, availability, reliability, maintainability and security of a system.

Market Watch System (MWS) has a 3-tier architecture pattern. The functions of the MWS are separated into three layers. The presentation layer handles the user interaction whereas Application-layer controls the system functionality by performing detailed processing and data access layer includes data persistence mechanisms. The architectural pattern of the MWS system is graphically shown in figure 2.1.



**Figure 2.1: Architectural Pattern**

### 3. Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. It describes the system from an external perspective showing the process view of the system architectural pattern.

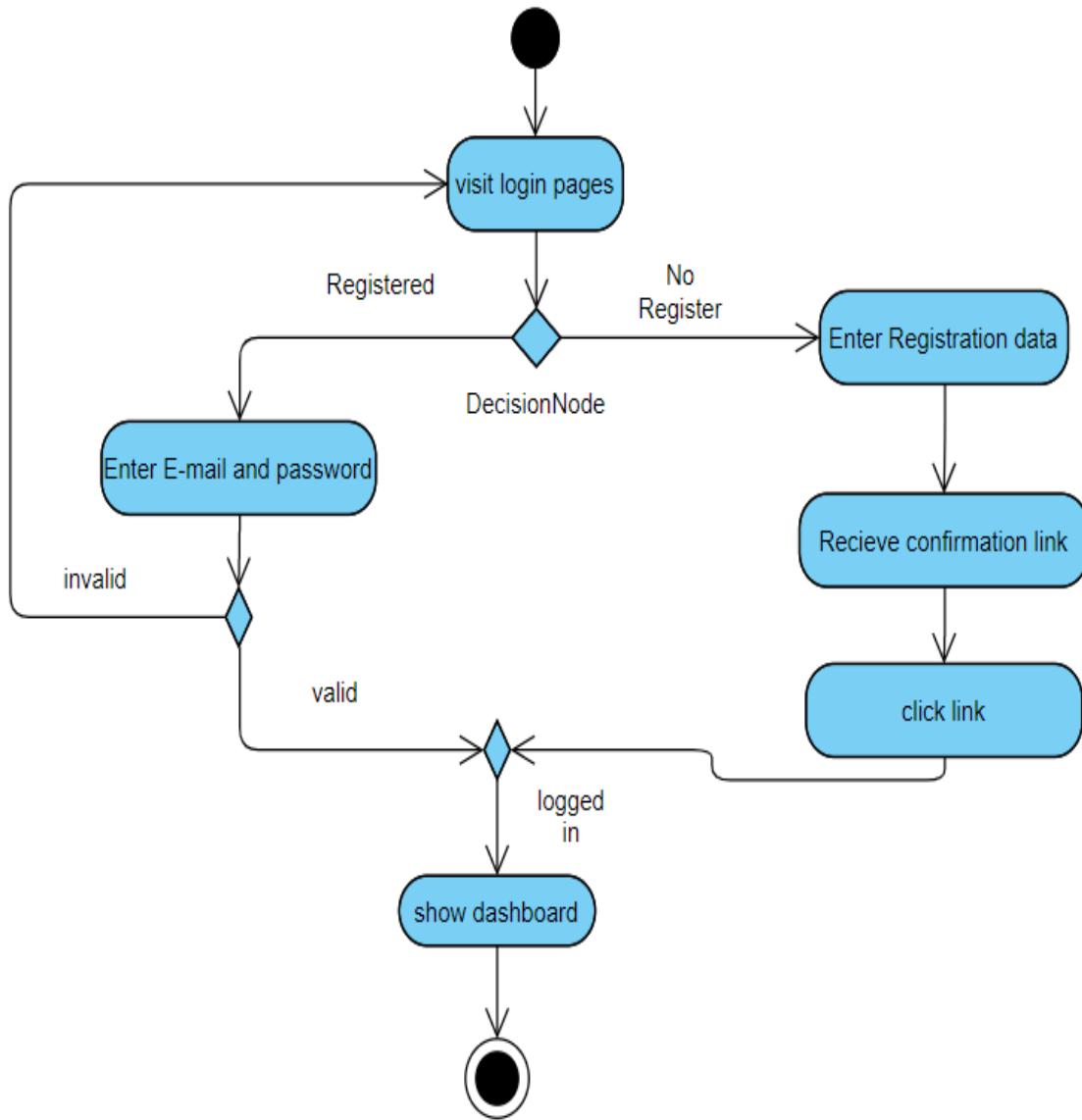
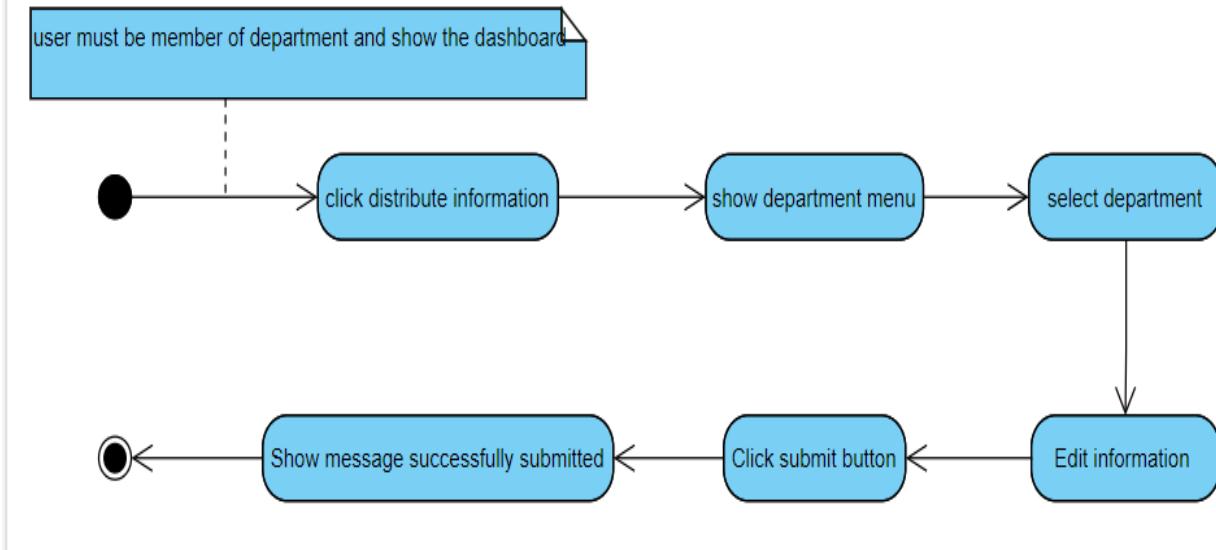
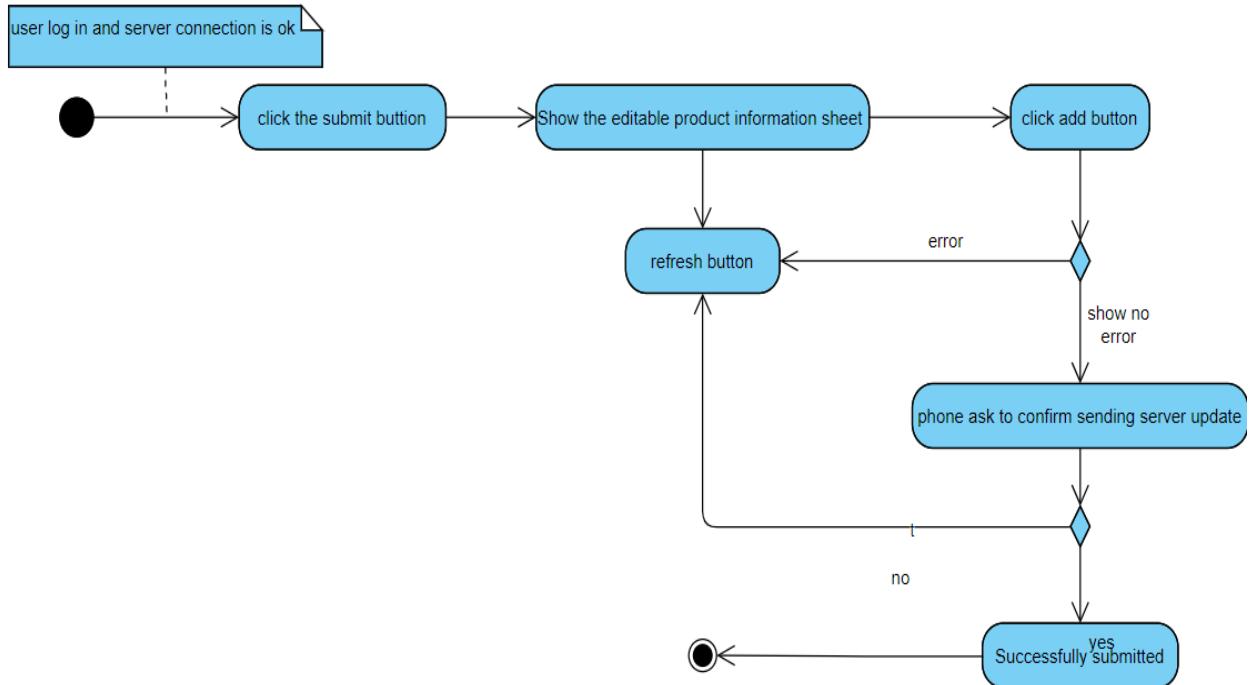


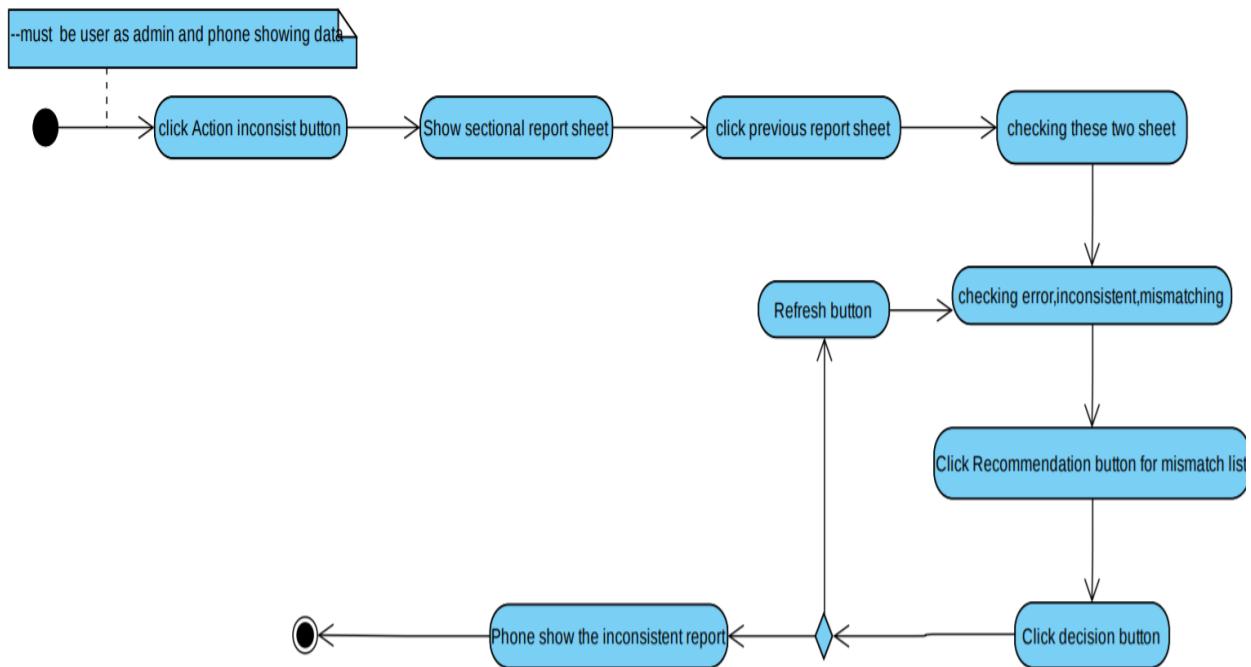
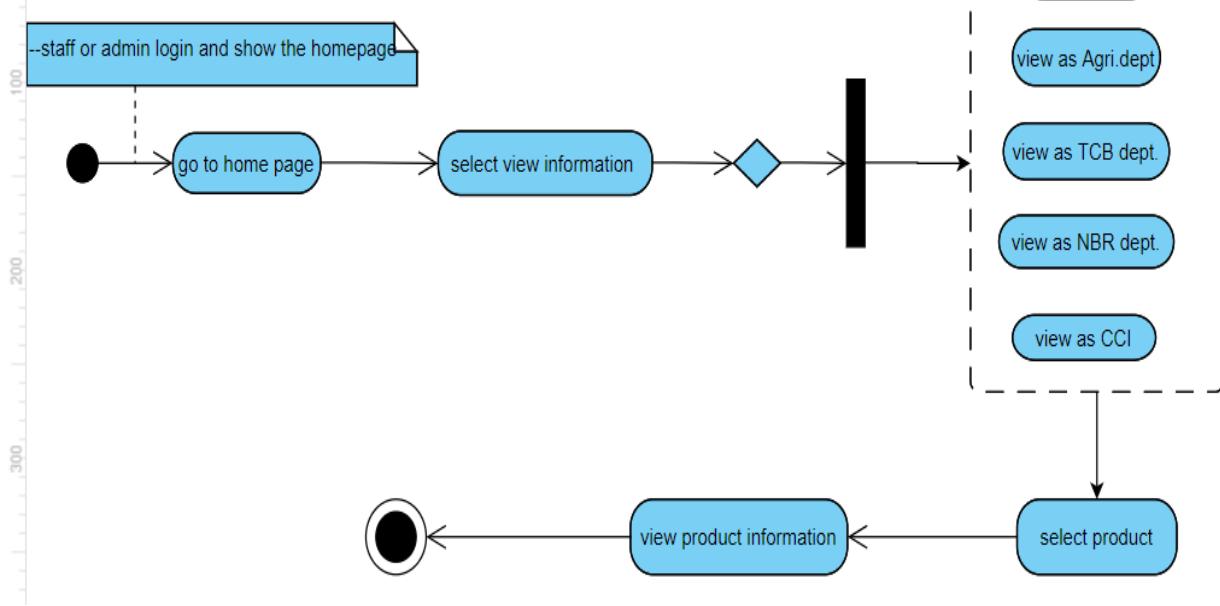
Figure 3.1: UC1

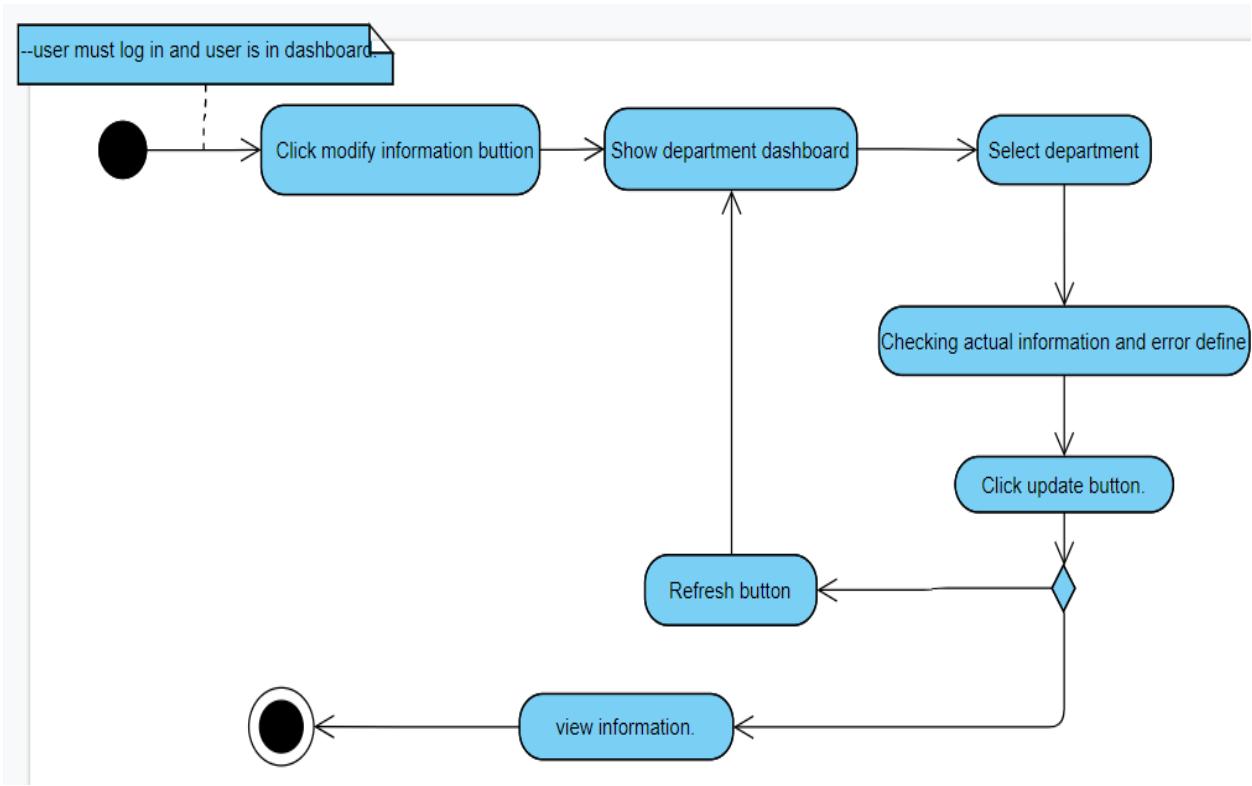


**Figure 3.2:** UC2

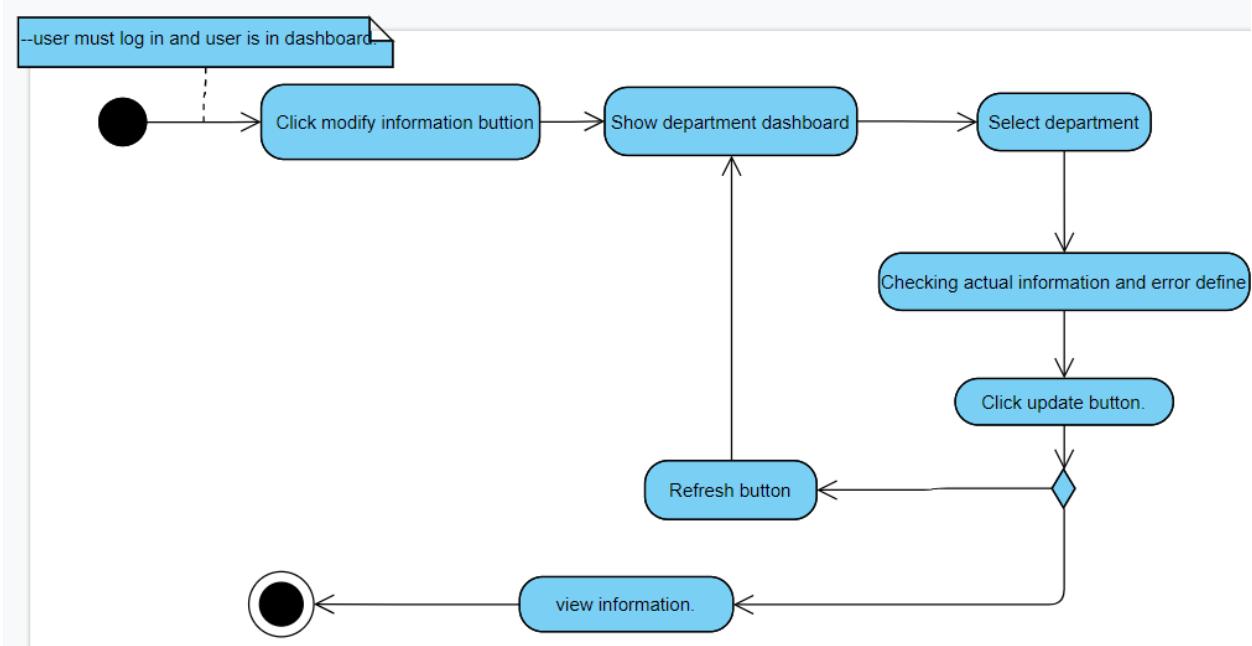


**Figure 3.3:** UC 3





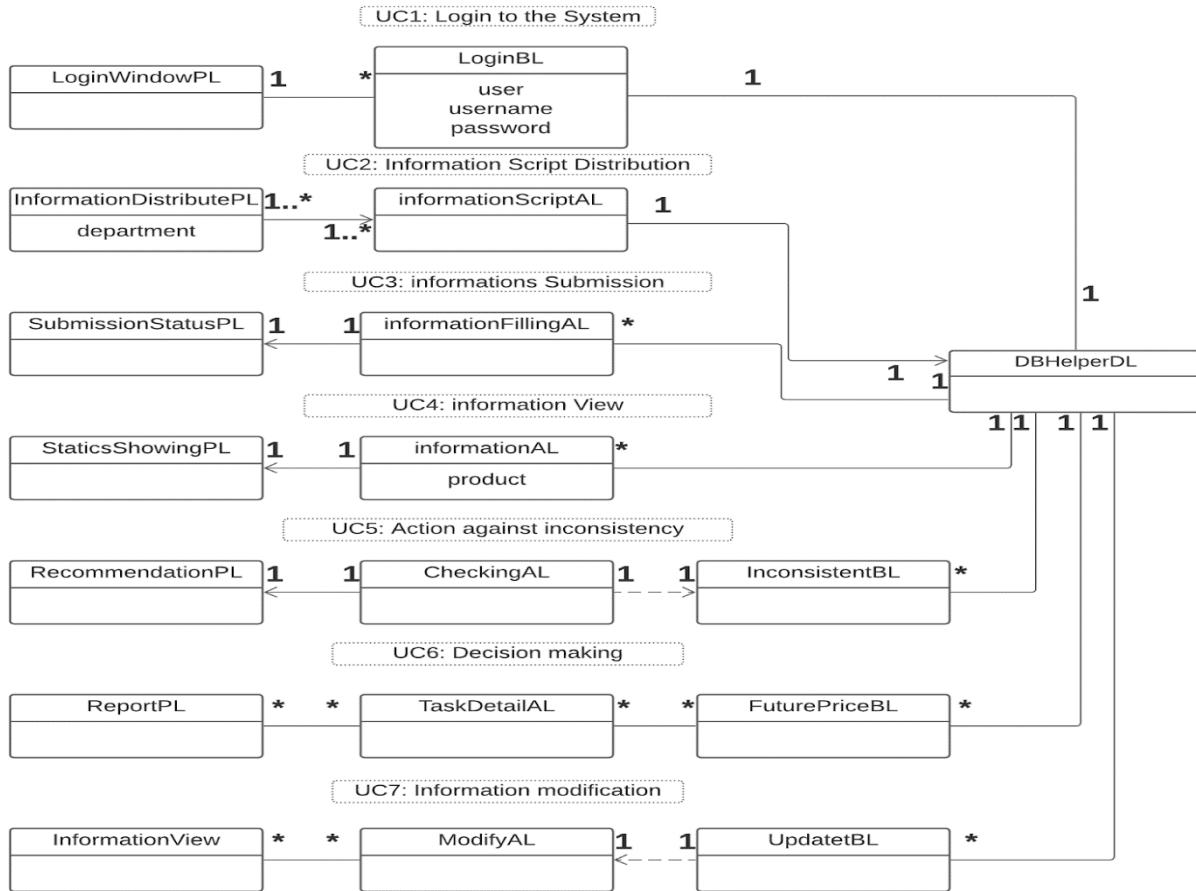
**Figure 3.6:** UC6



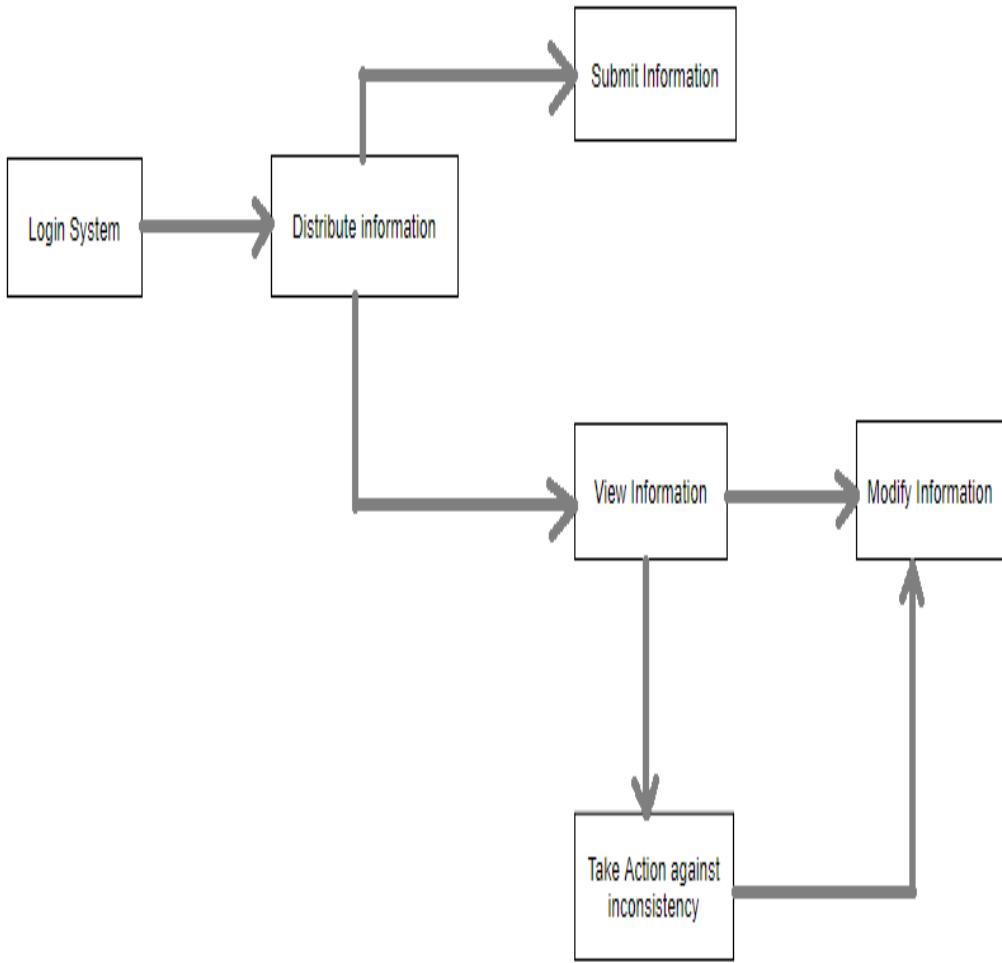
**Figure 3.7:** UC7

## 4. Class Diagram

Class diagrams describe the structure of a system by showing classes, attributes, operations of the system and also the relationships among objects. It is the main building block of object oriented modelling as it not only gives a conceptual model of the organization of the system but also is needed for translating the models into programming code. Class diagrams for MWS system is illustrated in figure 4.1 through 4.



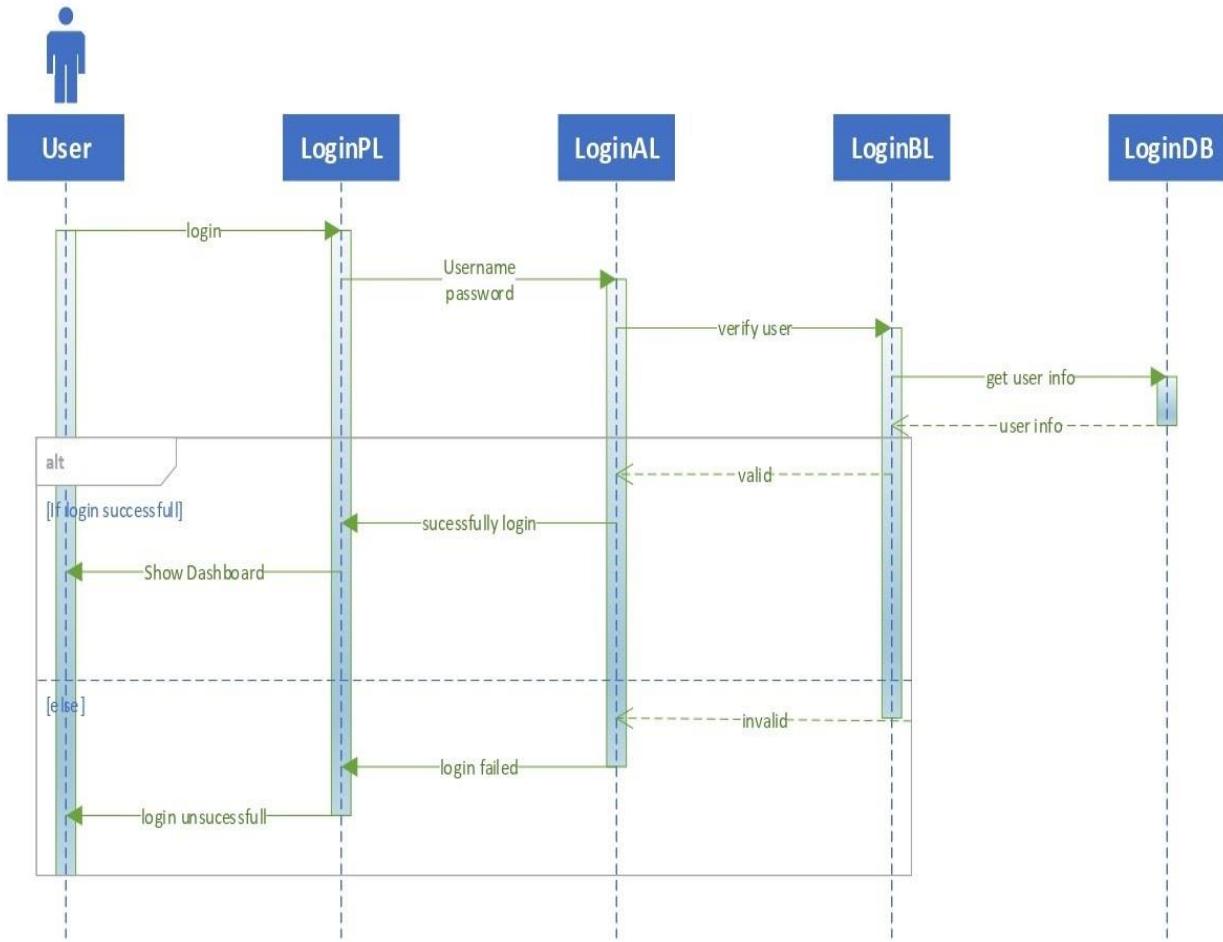
**Figure 4.1:** conceptual class diagram



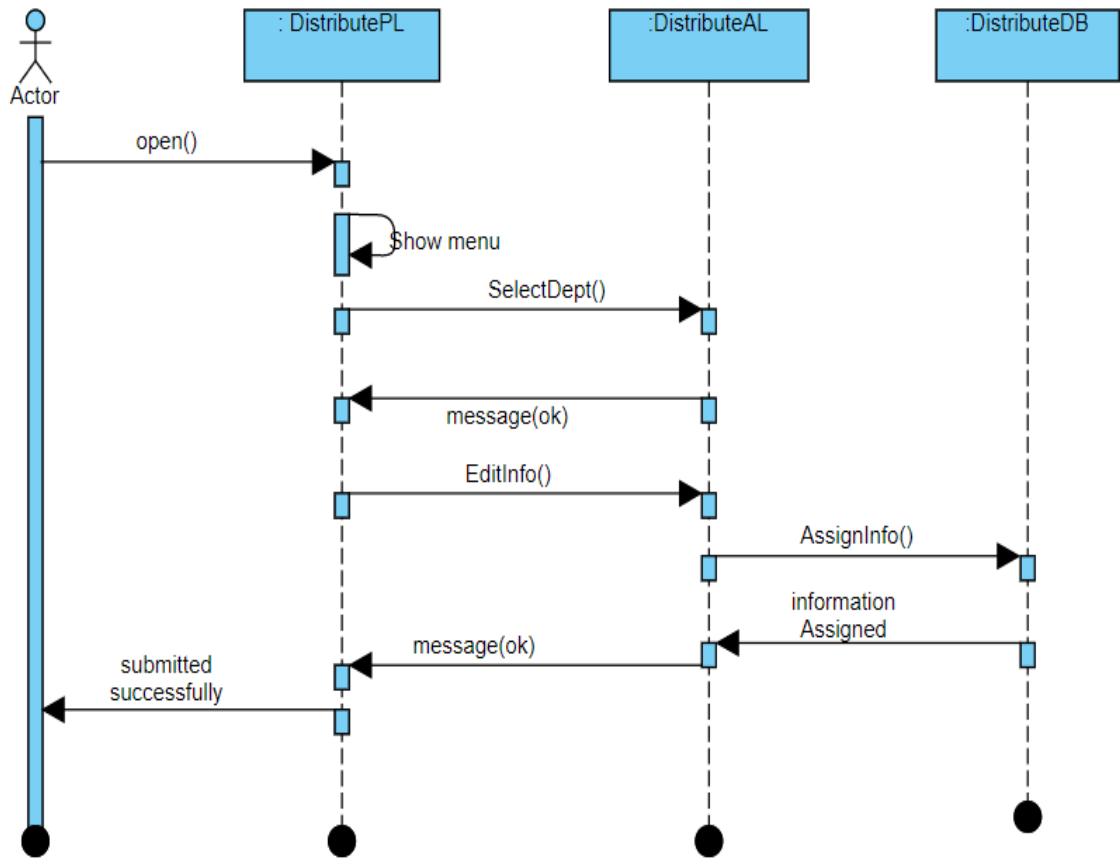
**Figure 4.2:** Relationships between conceptual classes.

## 5. Sequence Diagram

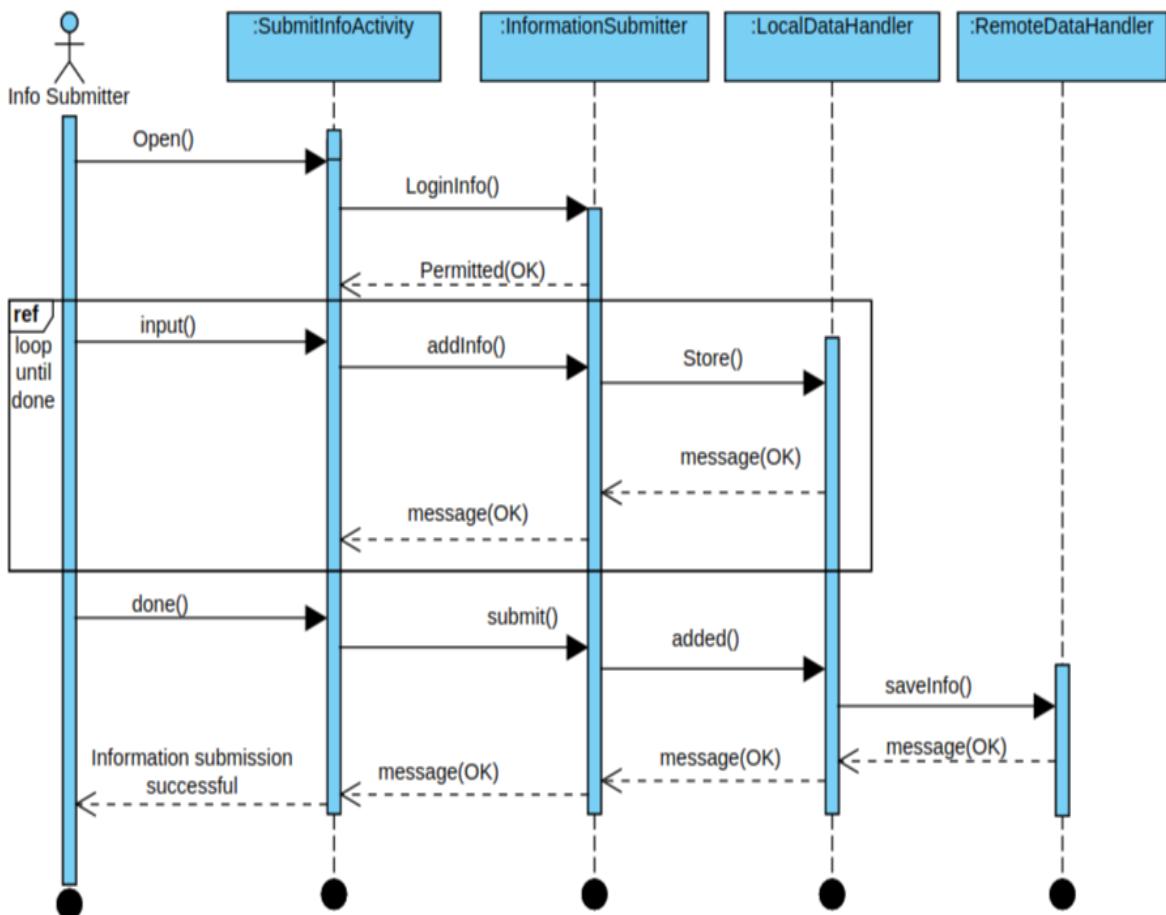
A sequence diagram shows object interactions arranged in time sequence. It portrays the classes and objects involved and the sequence of messages exchanged between the objects for carrying out the functionality of the system. It is used for showing the logical view of the system. Sequence Diagrams for MWS System is demonstrated in Figure 5.1 through 5.7



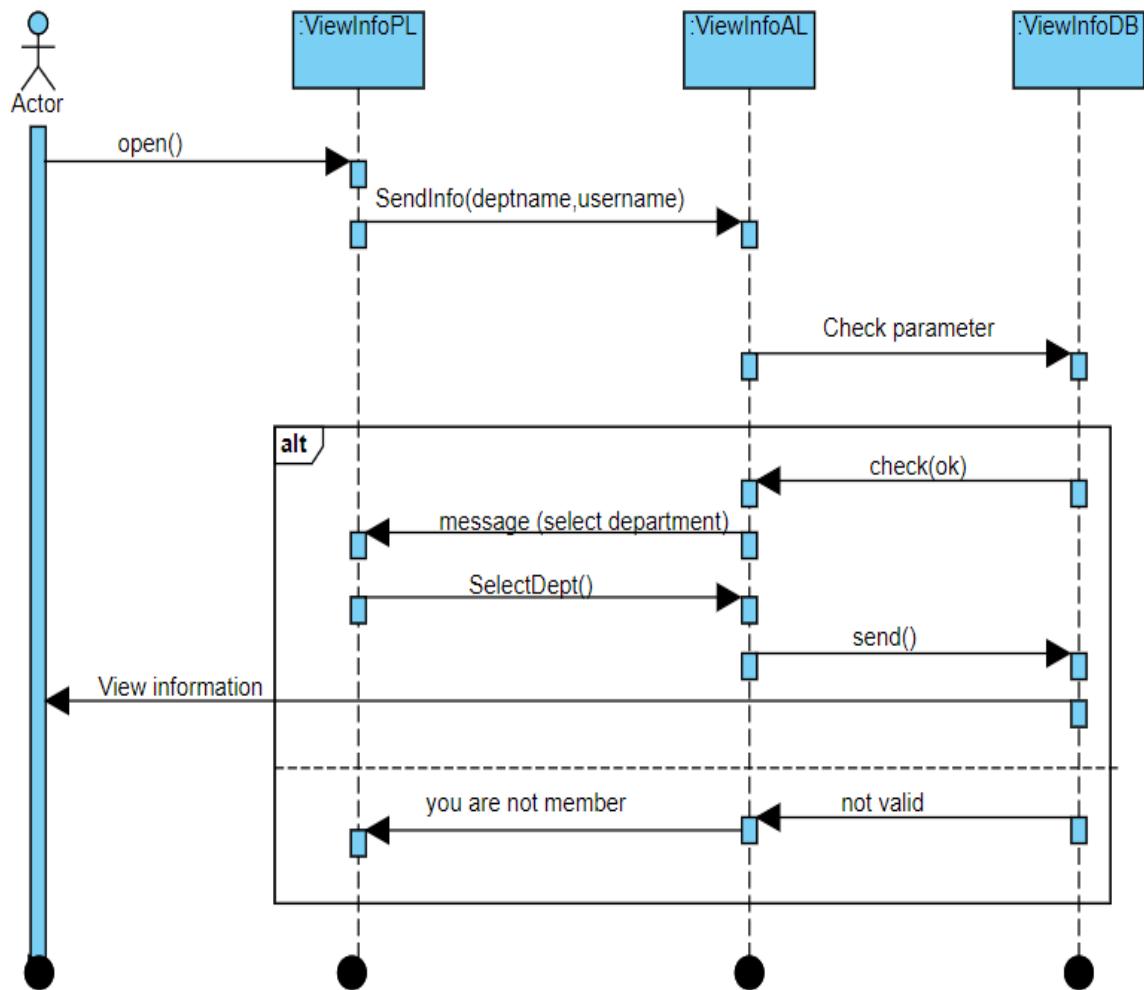
**Figure 5.1:** Sequence diagram for UC1: Login to the system



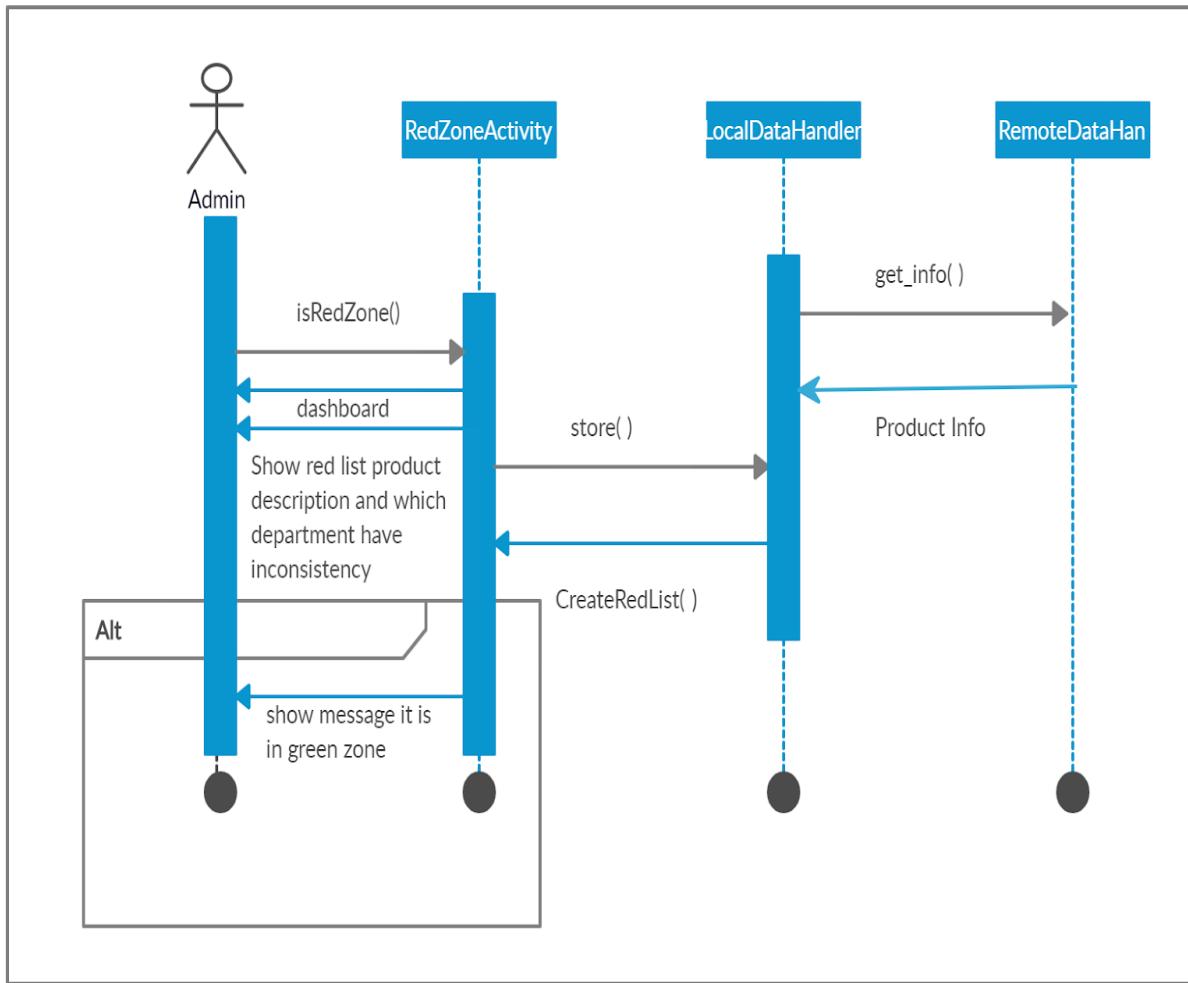
**Figure 5.2:** Sequence diagram for use case 2 : Distribution Script



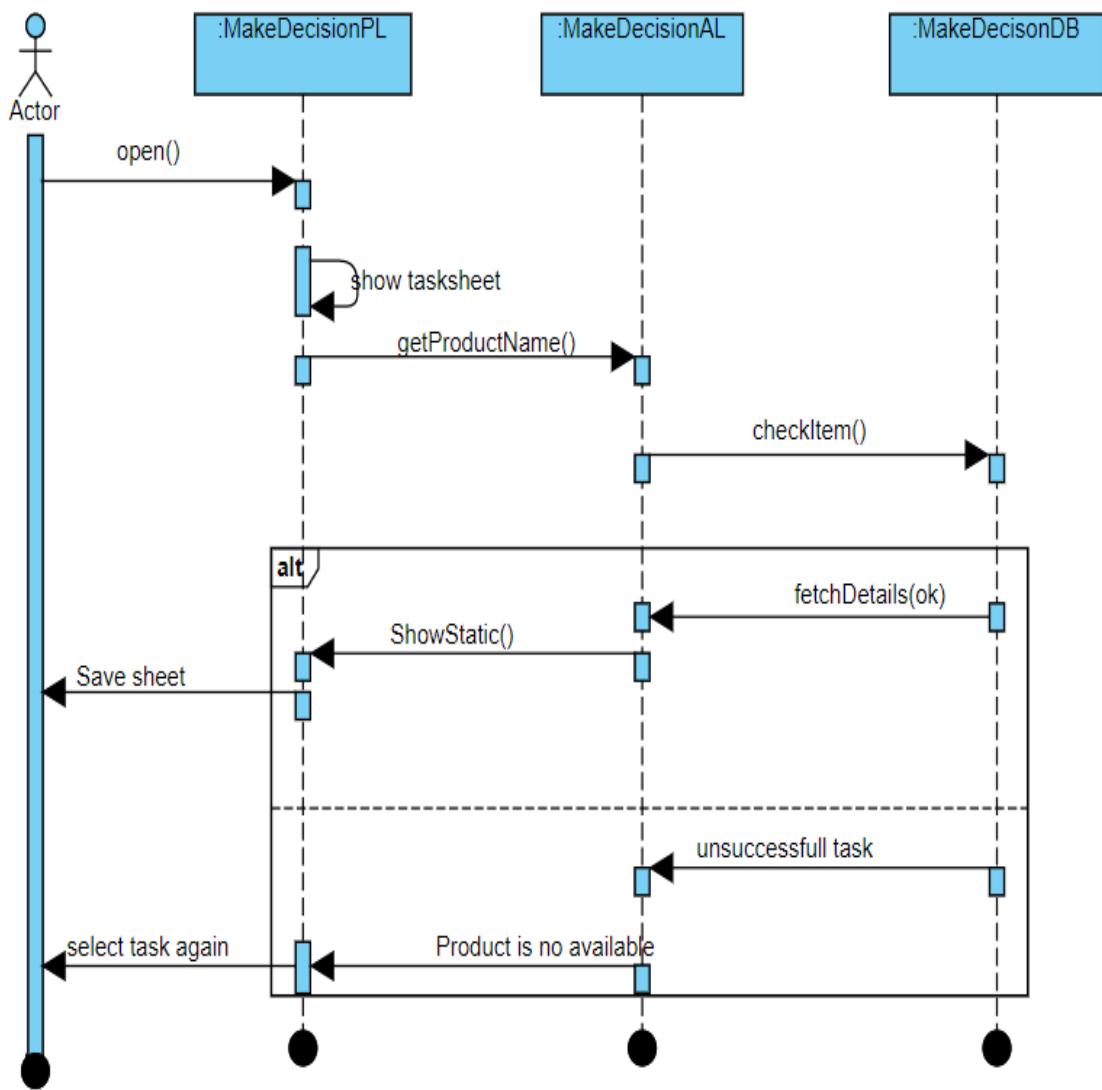
**Figure 5.3:** Sequence diagram for use case 3: Submission information



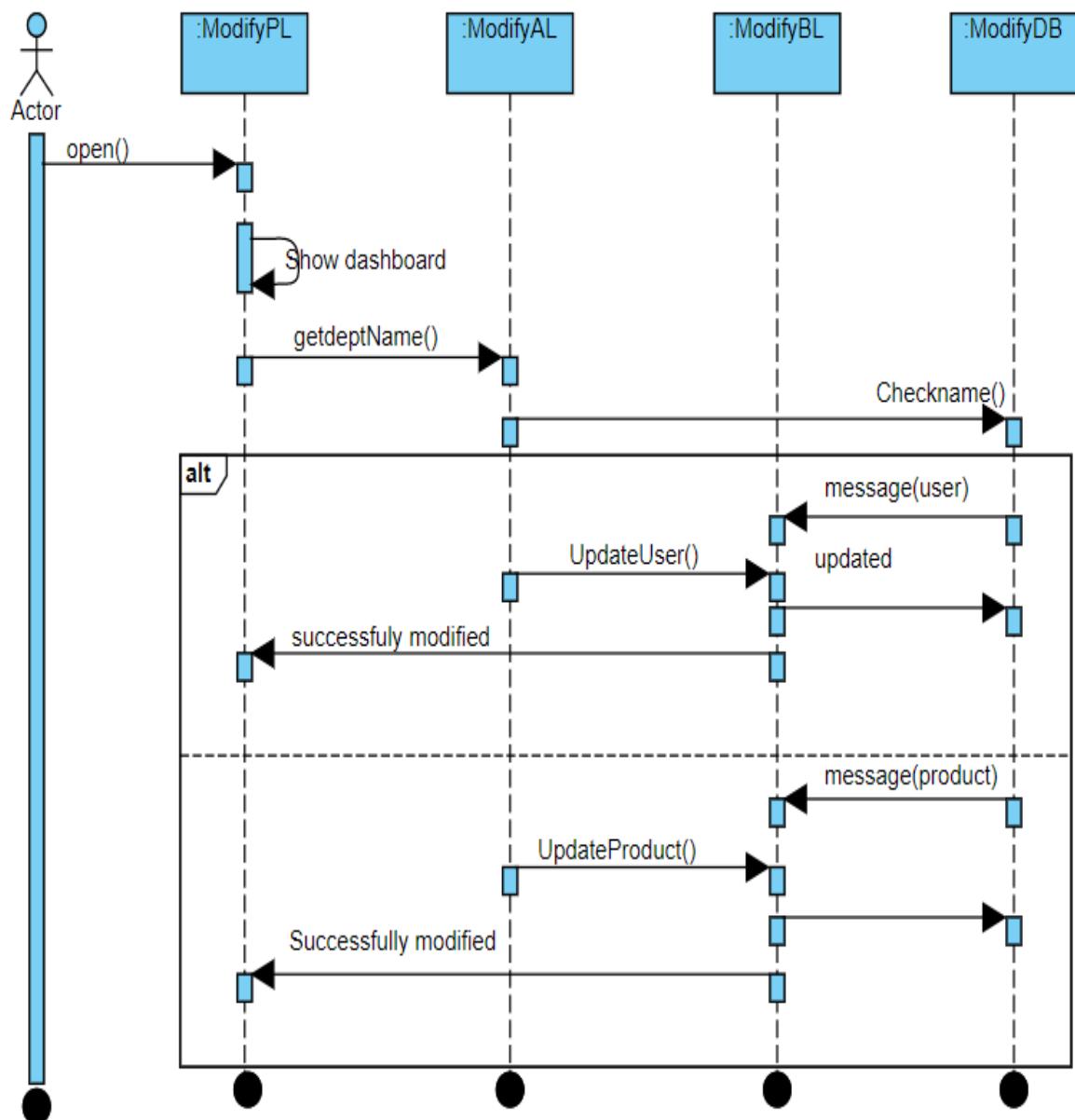
**Figure 5.4:** sequence diagram for use case 4: view information



**Figure 5.5:** Sequence diagram for UC5 : Take action against any inconsistency



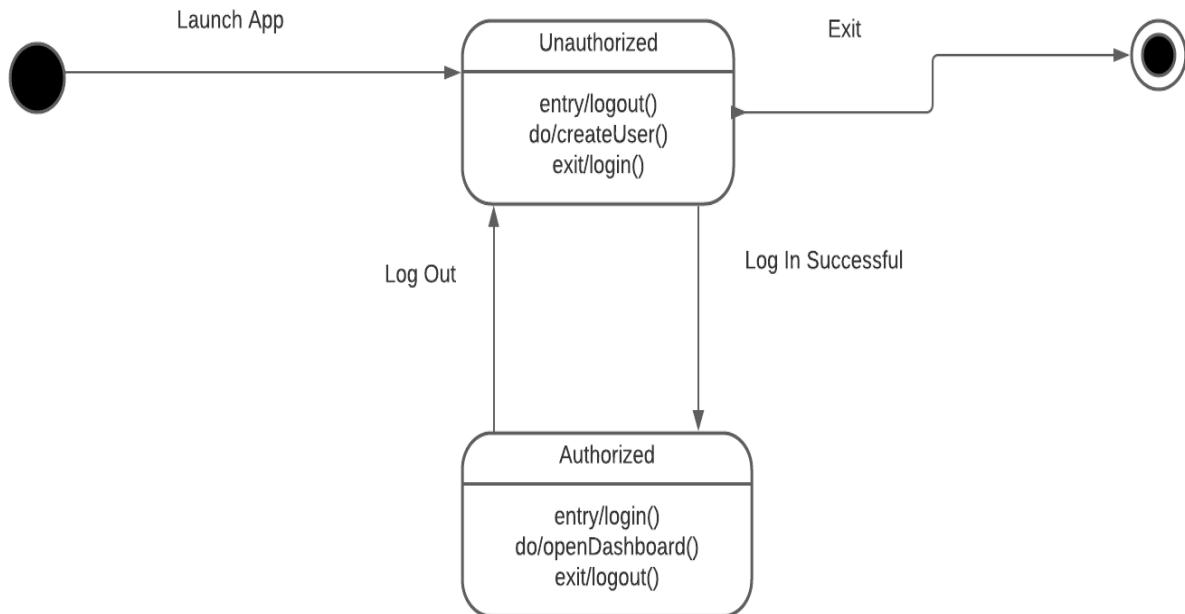
**Figure 5.6:** sequence diagram for use case 6: Make decision



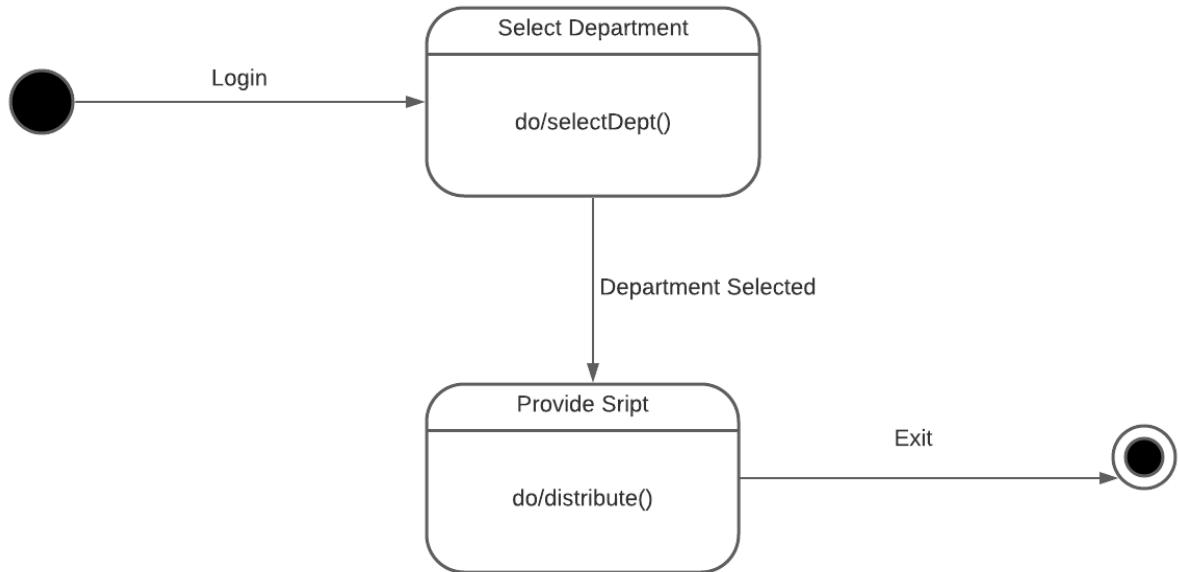
**Figure 5.7:** sequence diagram for use case 7: Modify information.

## 6. State Machine Diagram

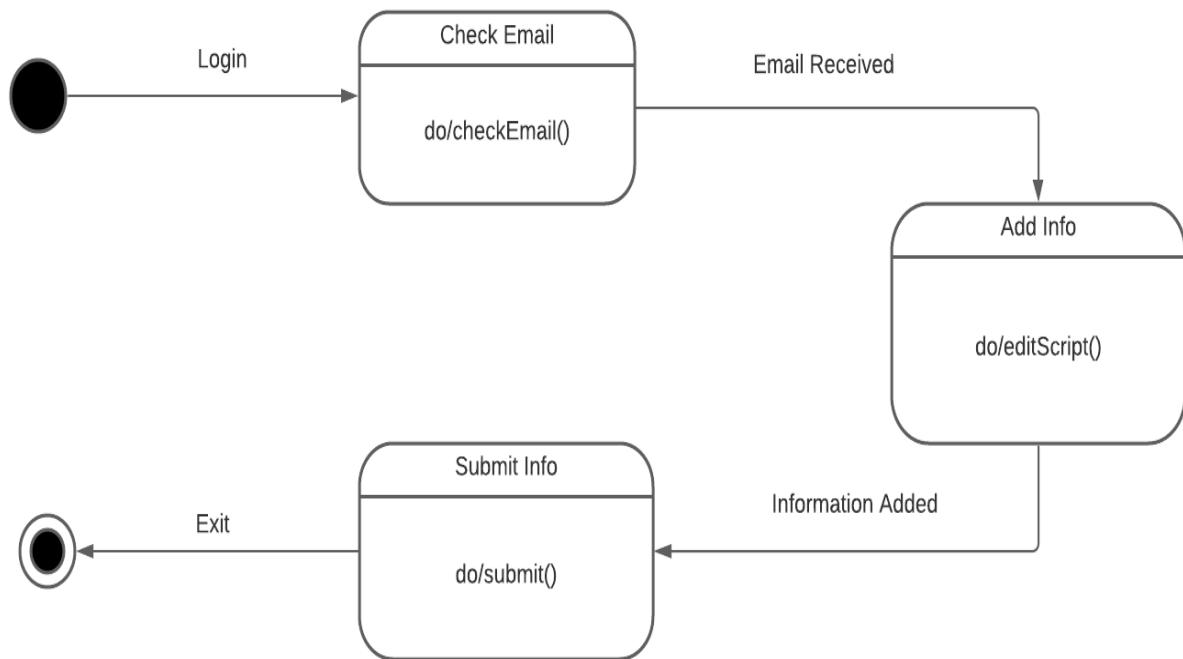
A sequence diagram shows object interactions arranged in time sequence. It portrays the classes and objects involved and the sequence of messages exchanged between the objects for carrying out the functionality of the system. It is used for showing the logical view of the system.



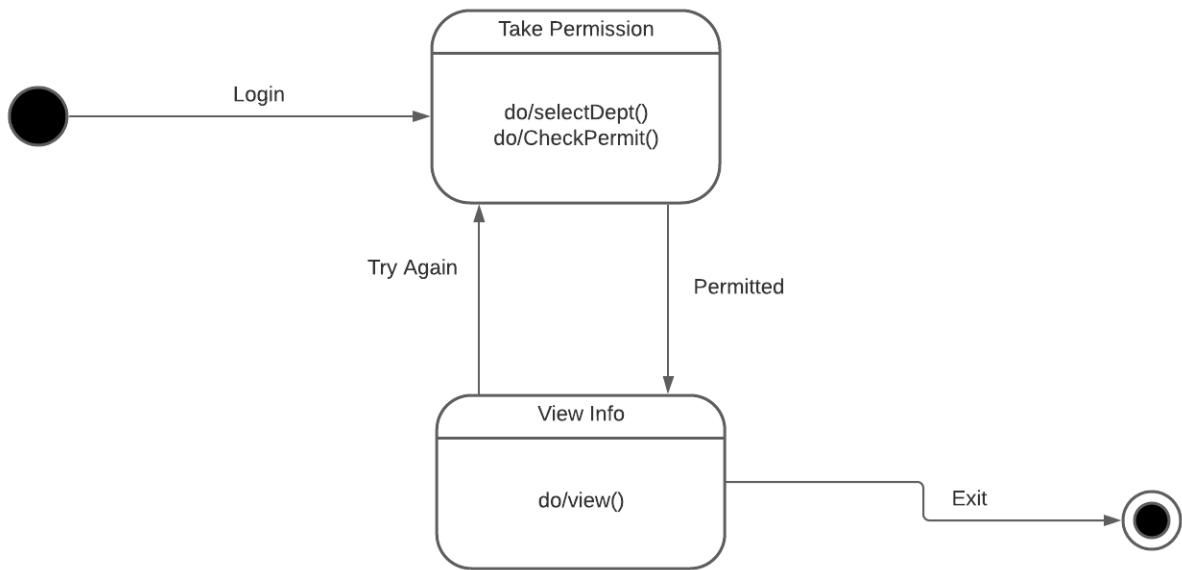
**Figure 6.1:** State machine diagram for Users object



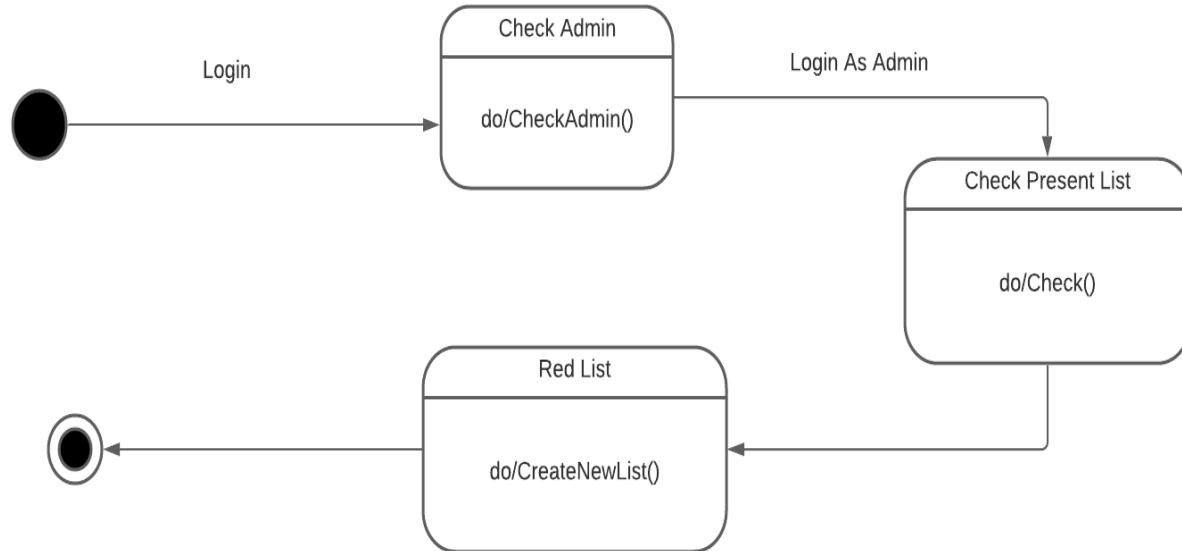
**Figure 6.2:** State machine diagram for Distributor object



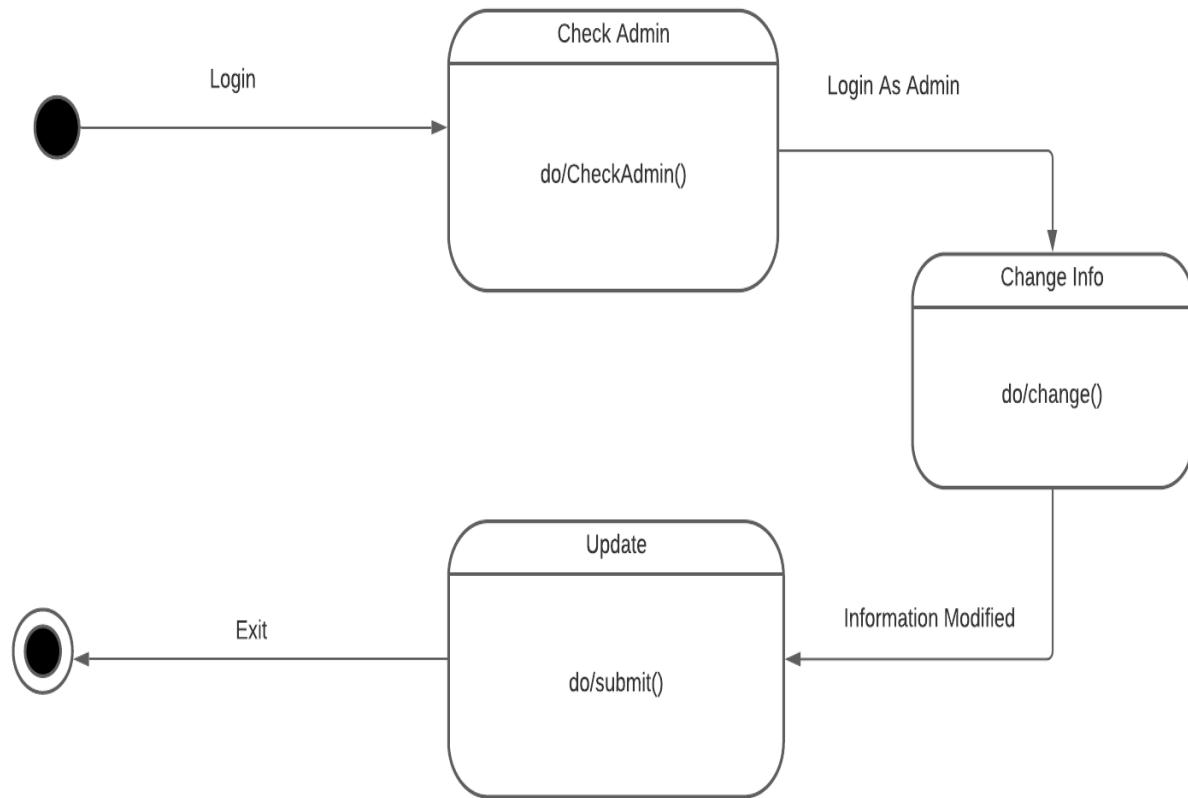
**Figure 6.3:** State machine diagram for Submitter object



**Figure 6.4:** State machine diagram for Viewer object



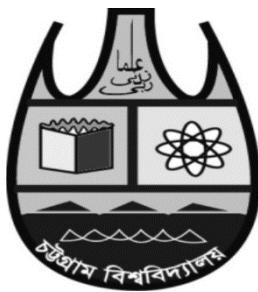
**Figure 6.5:** State machine diagram for TCB department object



**Figure 6.6:** State machine diagram for Modifier object

## 7. Conclusion

A detailed design document gives the overall structure of the system. Implementation is the process of converting the design into an executable program. The design document for the Market Watch System will help the developers to develop the system efficiently and also enable the testers to verify and validate the system precisely, therefore creating a system that fulfills the user requirements and meeting the stakeholder's expectation entirely.



## **Department of Computer Science and Engineering, University of Chittagong**

<b>Course Code</b>	CSE 516
<b>Course Title</b>	Software Engineering and Design patterns Lab
<b>Assignment name</b>	Details Design Document
<b>Report number</b>	06
<b>Group</b>	E

**Submitted to:**

**A. H. M. Sajedul Hoque**

Assistant Professor

Department of Computer Science and Engineering,

University of Chittagong.

**Submitted by**

ID: 18701032

ID: 18701057

ID: 18701071

ID: 17701101

## 1. Introduction

In software engineering, a detailed class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. In details class diagrams specify the last stage of implementation of the model. In computing, an interface is a shared boundary across which two or more separate components of a computer system exchange information. The exchange can be between software, computer hardware, peripheral devices, humans, and combinations of these. A design pattern is a general repeatable solution to a commonly occurring problem in software design. A design pattern isn't a finished design that can be transformed directly into code. It is a description or template for how to solve a problem that can be used in many different situations.

## 2. Proposed Conceptual Class Diagram

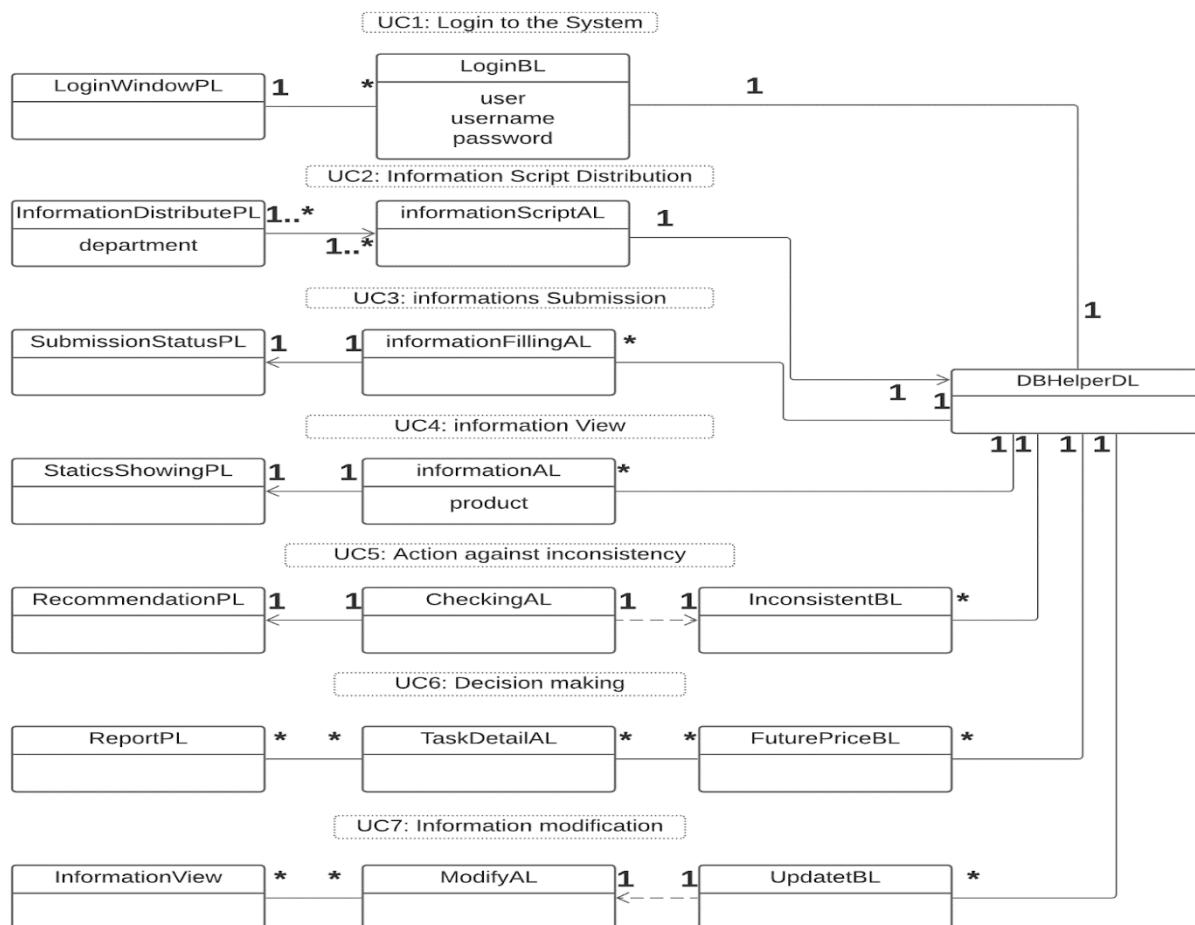
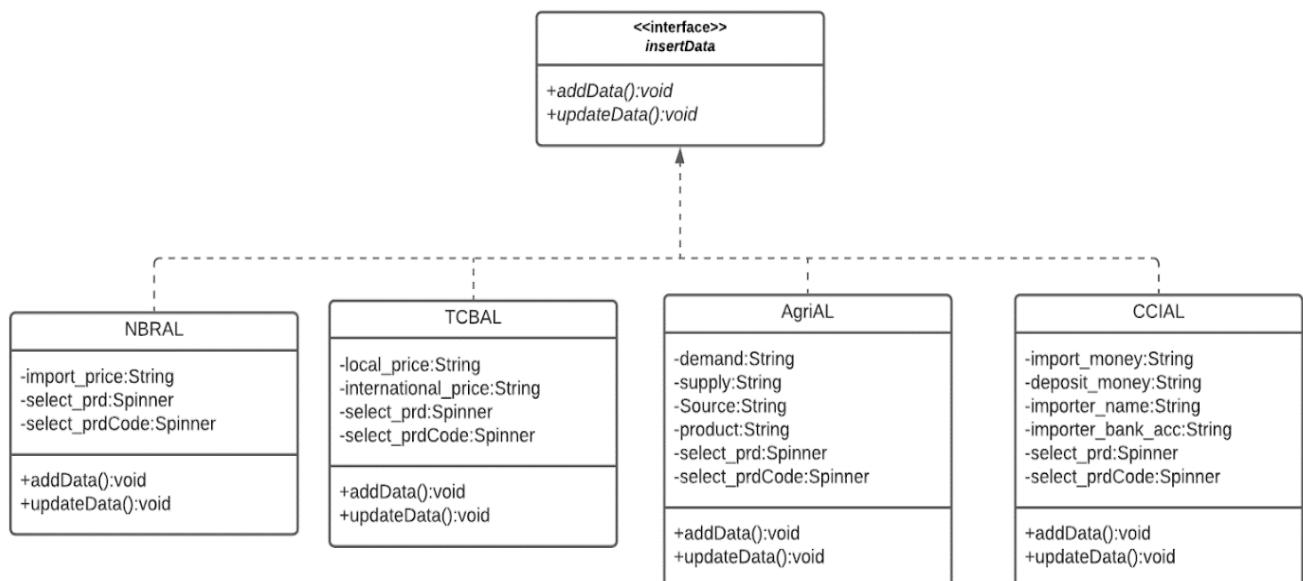


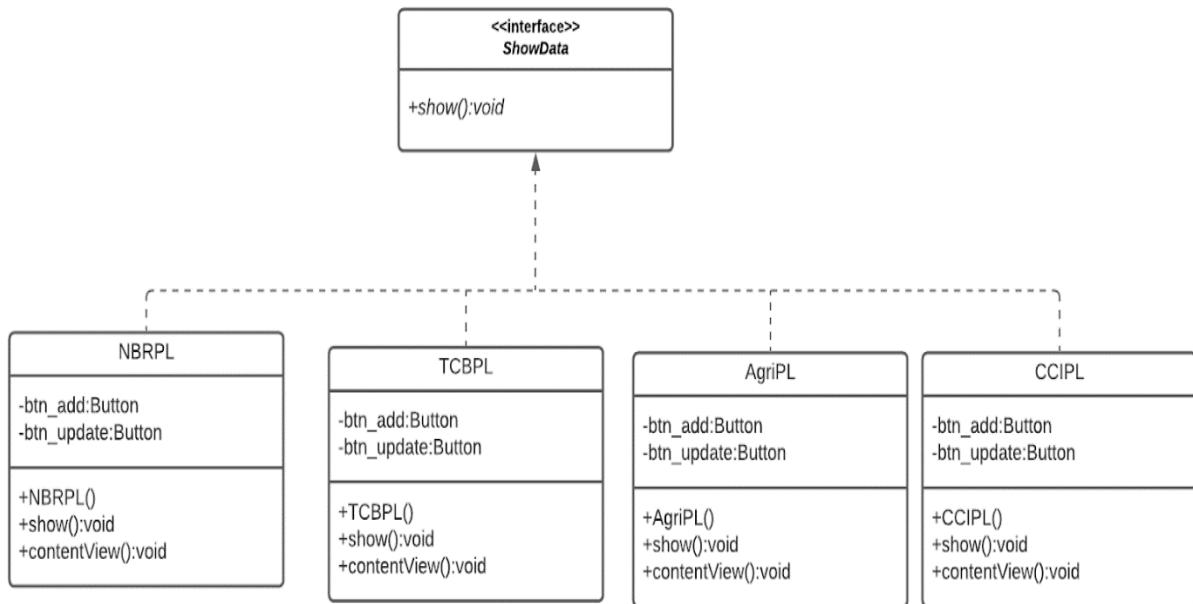
Figure 2.1: Conceptual Class diagram

### 3. Specify Interface

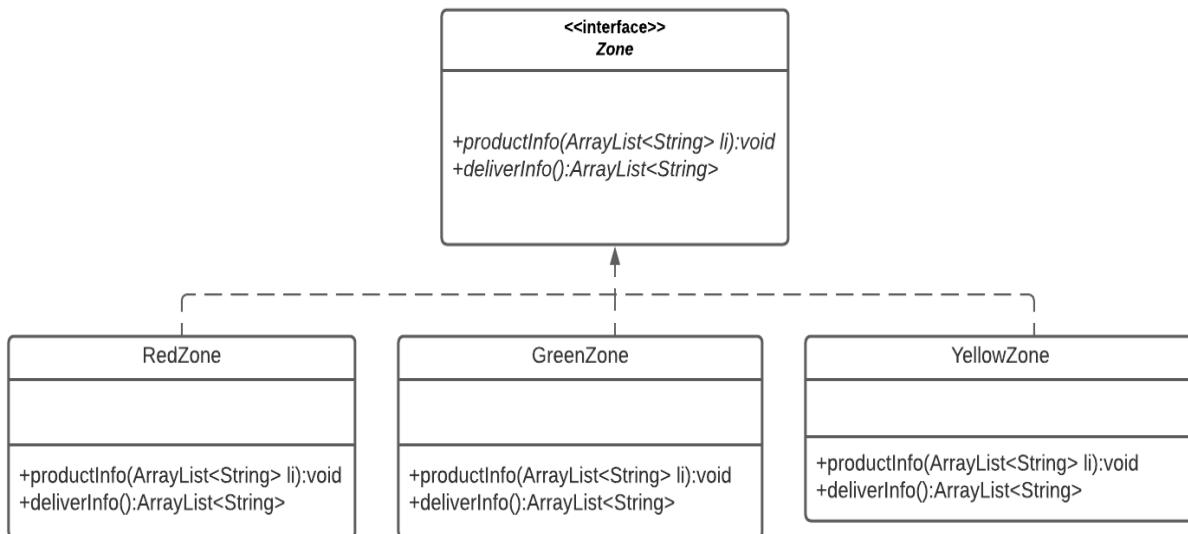
Interface class methods do not have a body; the body is provided by the "implement" class. Interface methods are by default abstract and public and its attributes are by defaults public, static and final. An interface cannot contain any constructor, as it cannot be used to create objects. For our Market Watch system, we used three interfaces to construct our system. One is InsertData, which has four "implement" classes: AgricultureDeptActivity, NBRDeptActivity, TCBDeptActivity and CCIEDeptActivity. The ShowData interface class also has four 'implement' classes: Agri, Tcb, Nbr and Ccie and Zone interface has three "implement" classes: GreenZone, RedZone and Yellowzone. These interfaces class provide our model loose coupling and up casting facilities which provide more independence to a programmer to change the source code in the future.



**Figure 3.1:** Insertdata interface



**Figure 3.2:** Show data Interface



**Figure 3.3:** Zone Interface

## 4. Used Design Pattern

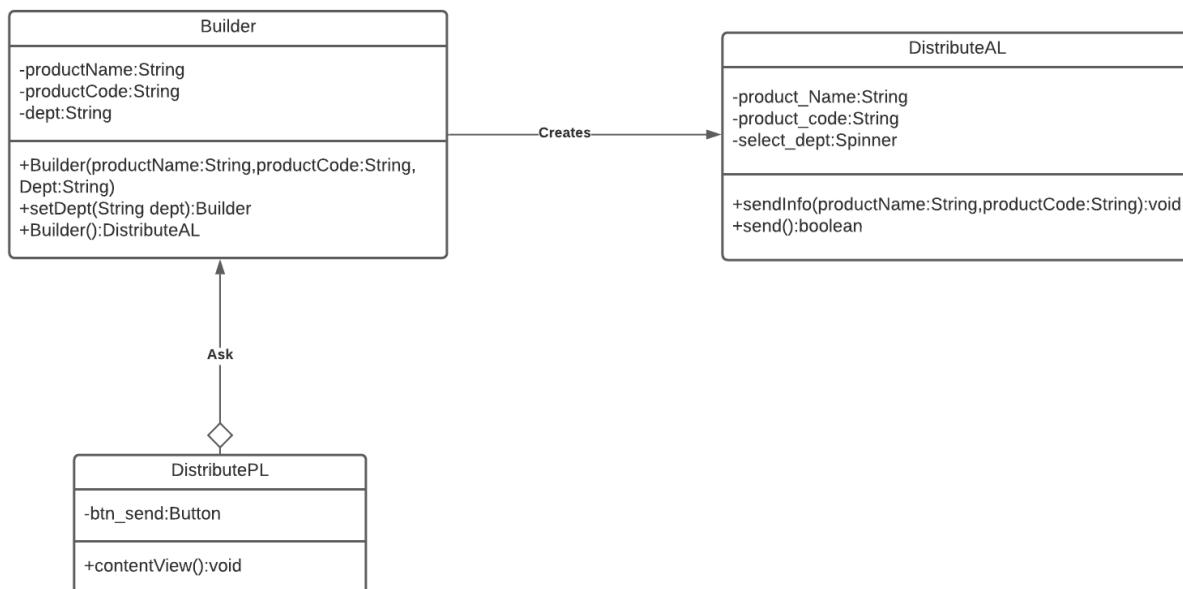
Design pattern is an architecture and formal way to solve particular design problems. Normally, Design pattern relies on object characteristics such as inheritance and polymorphism to provide the generality. Patterns help to facilitate good design by defining good object granularities, interfaces, it already defines design patterns, it makes our code easy to understand and debug. It leads to faster development and new members of the team understand it easily. I would like to propose three design patterns in my android application. There are three groups of design patterns such as creational, structural and behavior. Here is the using design pattern of my android application.

- Builder design pattern (Creational)
- Singleton design pattern (Creational)
- Facade design pattern (Structural)
- Strategy design pattern (Behavioral)
- Adapter Design pattern (Structural )

Firstly, I will explain about the Builder design pattern and sequentially others.

### 4.1 Builder Design pattern

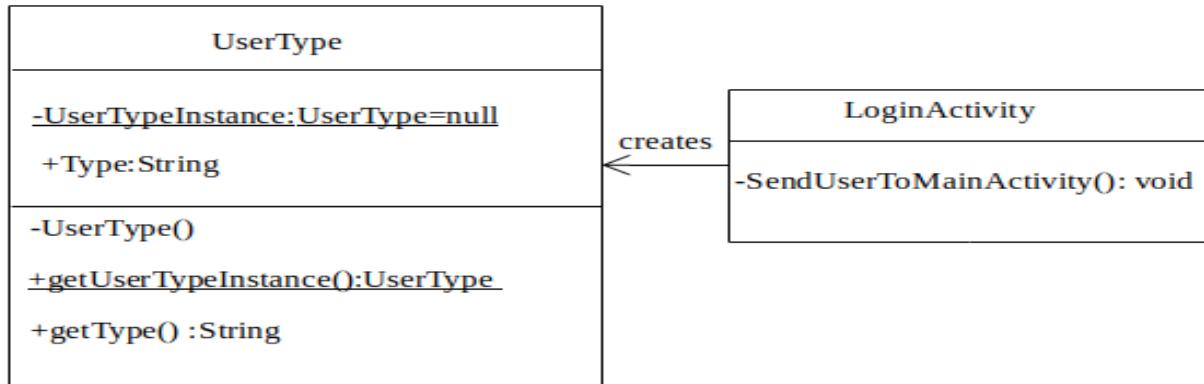
Builder design pattern is the most common design technique to create an object in class by building an object. This pattern is used to solve some problems in factory and abstract factory pattern when the object contains a lot of attributes. Builder pattern solves the issue with a large number of optional parameters and inconsistent state by providing a way to build the object step-by-step and provide a method that will actually return the final Object. I will explain the class DistributeAL, Builder and DistributePL where exist the relation with builder design



**Figure 4.1:** Builder Design Pattern

## 4.2 Singleton Design pattern

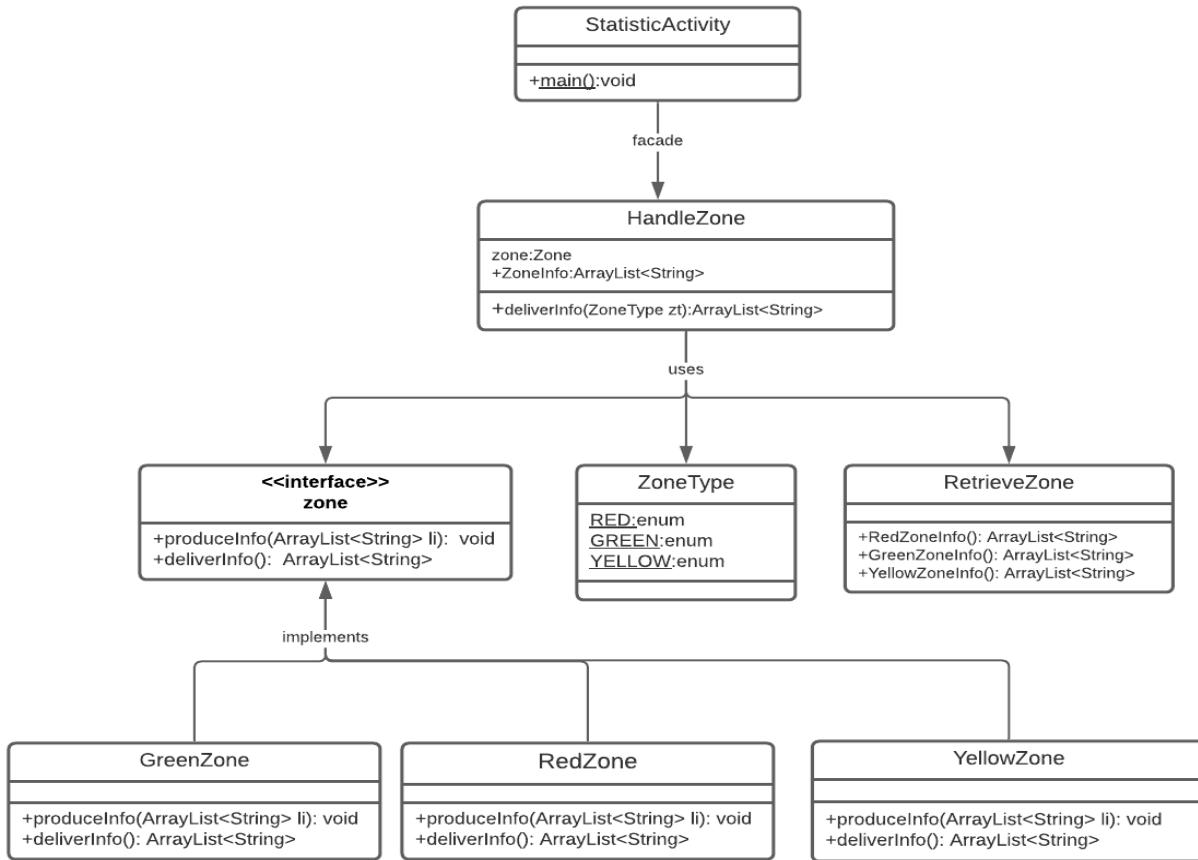
The singleton design pattern is used to restrict the instantiation of a class. It is a creational design pattern. It defines a class that has only one instance and provides a global point of access to it. We use singleton design pattern to restrict the logging type (Admin or non Admin) of our system.



**Figure 4.2:** Singleton pattern

## 4.3 Facade Design pattern

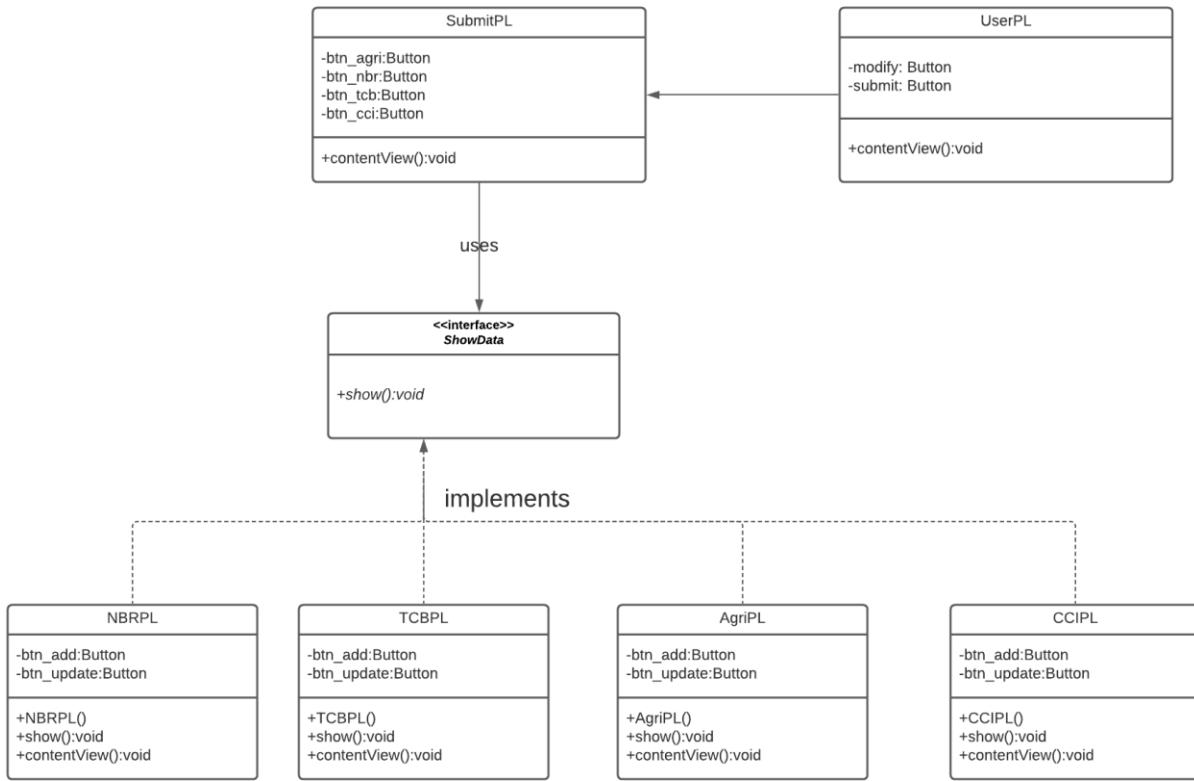
Facade is a structural design pattern that provides a simplified (but limited) interface to a complex system of classes, library or framework. While Facade decreases the overall complexity of the application, it also helps to move unwanted dependencies to one place. We can use the façade pattern to collate all the complex method calls and related code blocks and channelizes it through one single Façade class. It makes easier to use and maintain creating a more structured environment and reduces dependencies between libraries or other packages. I will explain façade design pattern in these class like as StatisticActivity, HandleZone, Zone (interface), Zonetype, RetreiveZone, GreenZone, YellowZone and RedZone where exist the relationships and uses of those class.



**Figure 4.3:** Facade Design pattern

#### 4.4 Strategy Design pattern

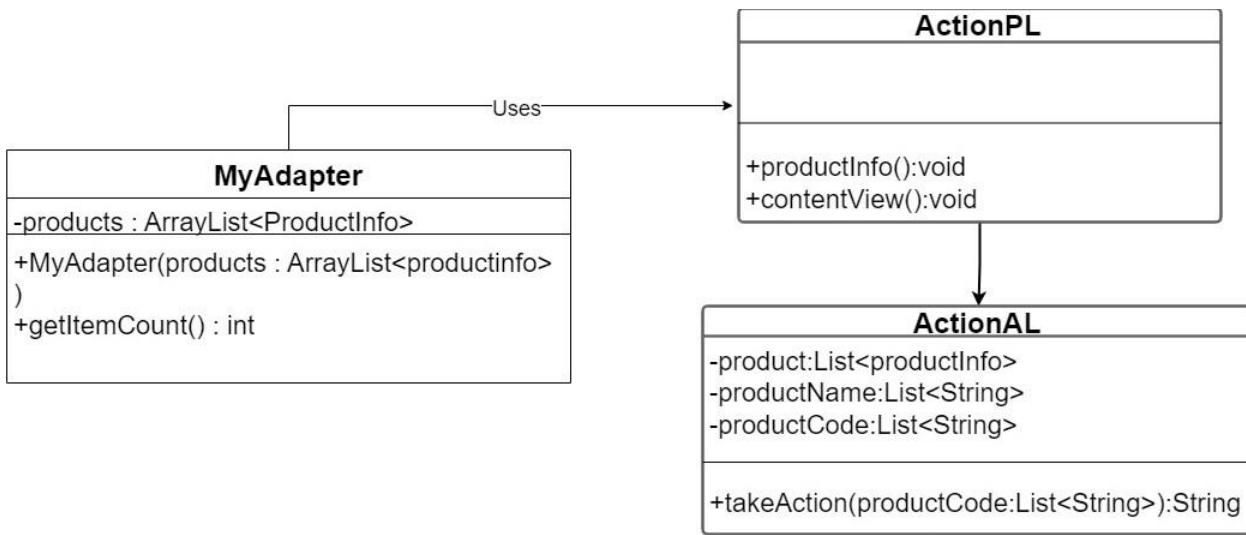
Strategy is a behavioral design pattern that turns a set of behaviors into objects and makes them interchangeable inside the original context object. The original object called context, holds a reference to a strategy object and delegates it executing the behavior. The strategy pattern is used to solve problems that might (or is foreseen they might) be implemented or solved by different strategies and that possess a clearly defined interface for such cases. I will explain the Strategy design pattern in these class like as SelectDepartmentActivity, InsertData (interface), MainActivity, Agri,Nbr,Tcb and Ccie where exist the relationship and uses those classes. Figure 4.3 attached the below.



**Figure 4.4: Strategy Design pattern**

## 4.5 Adapter design pattern

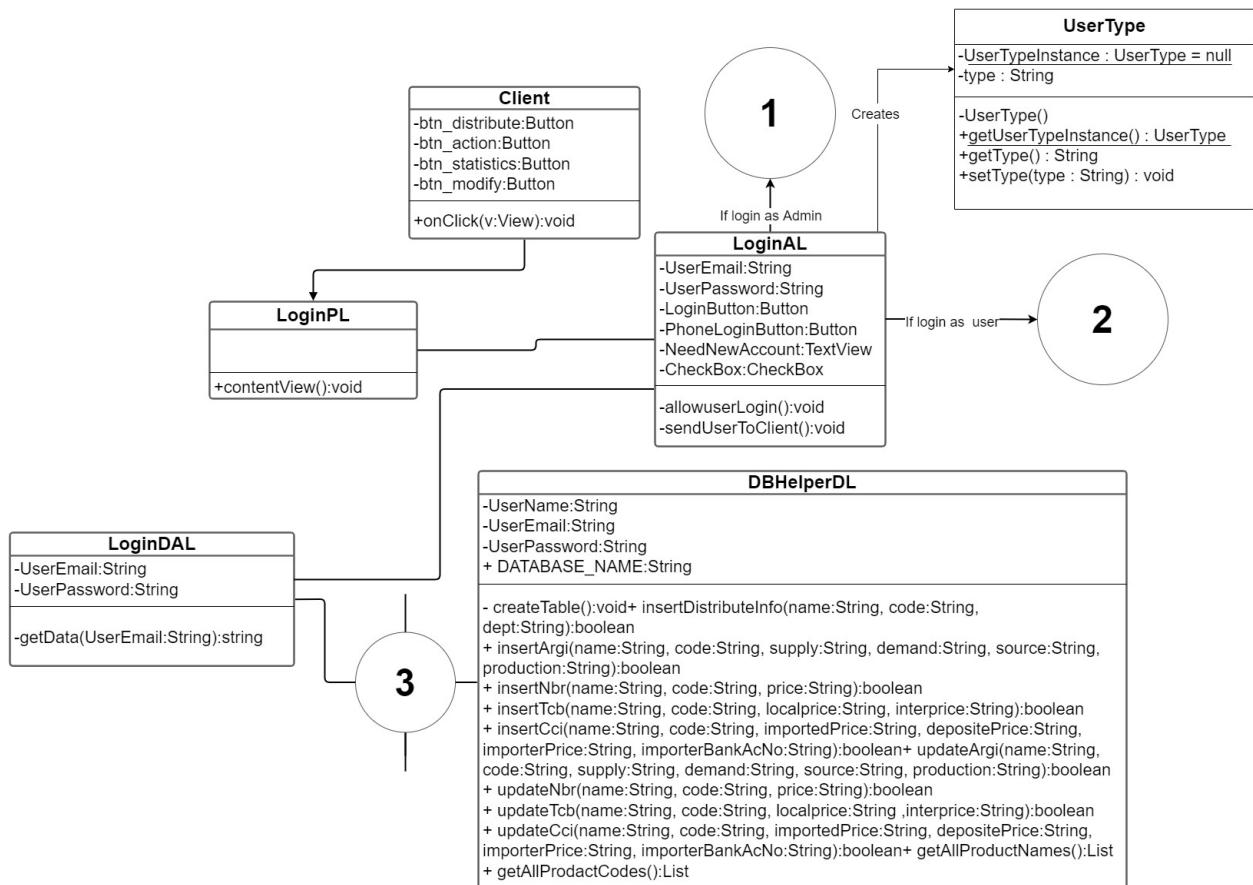
It is the structural design pattern where MyAdapter, ActionPL, ActionAL class exists

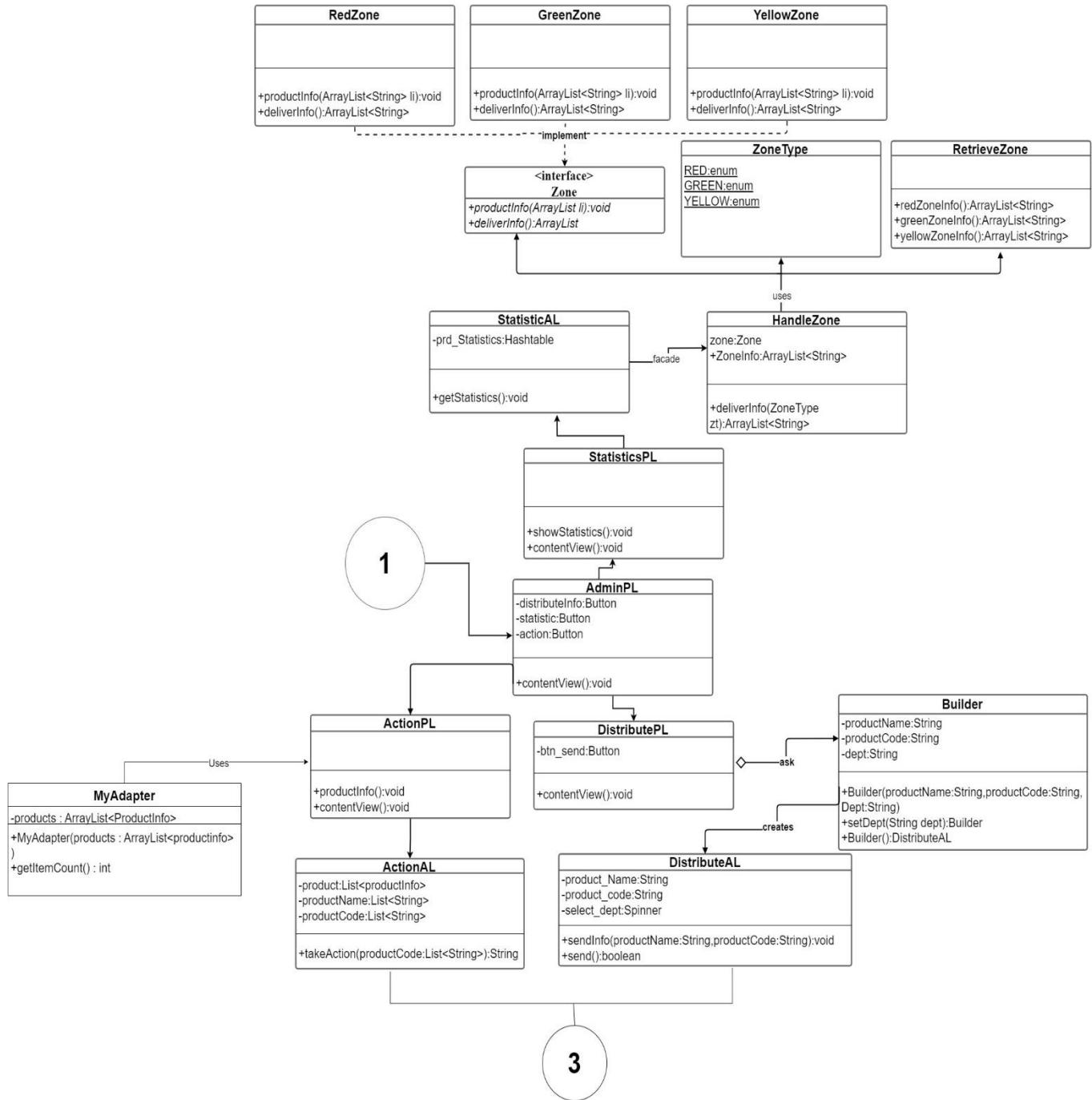


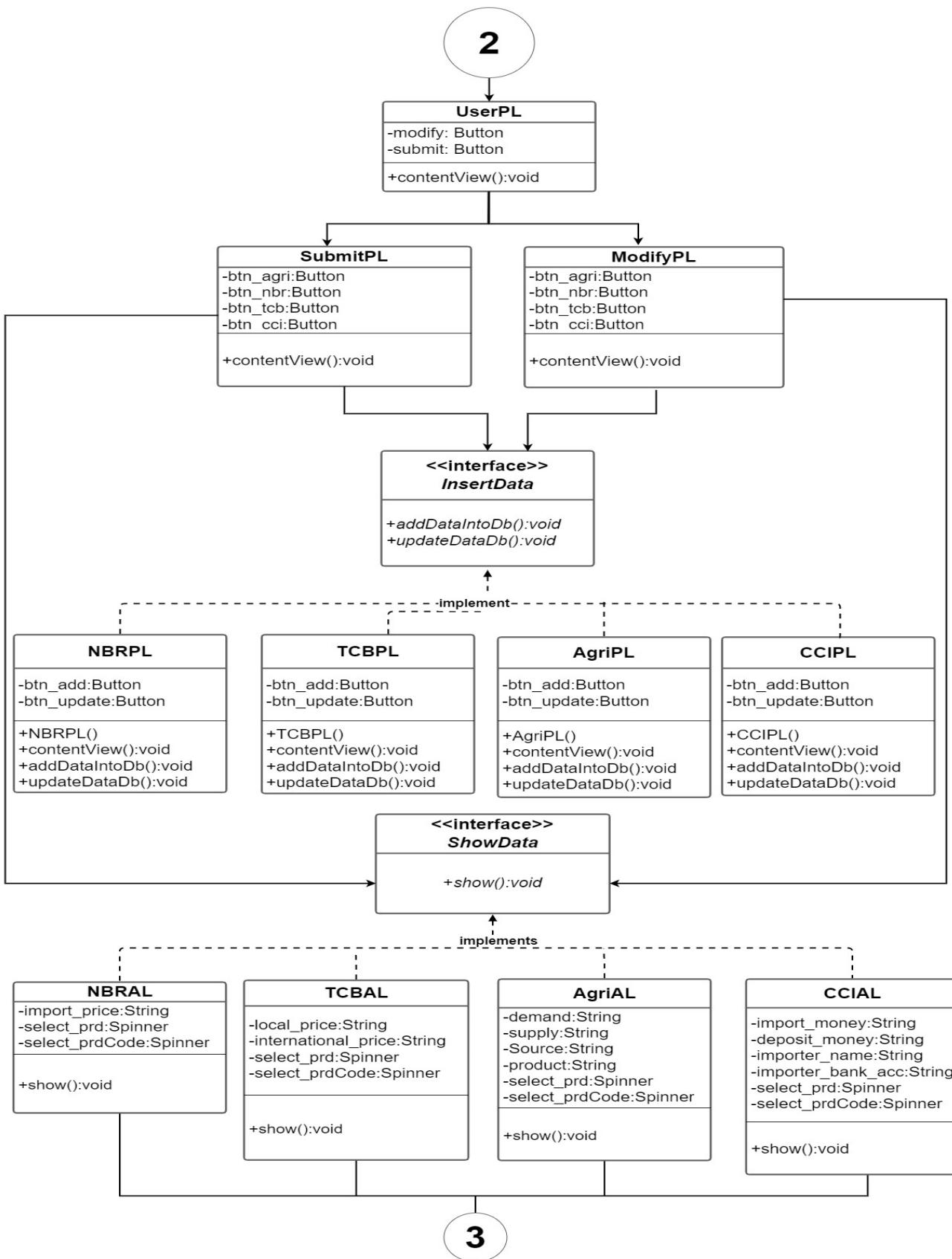
**Figure 4.4:** Adapter Design Pattern

## 5. Detail Class Diagram

Class diagram is the fundamental building block in object-oriented modeling. It is used to show the different objects in a system, their attributes, their operations and the relationships among them. It shows detail class diagram for MarketWatch system in figure 5.1







**Figure 5.1:** Details Class diagram

## 6. Conclusion

A detailed design document gives the overall structure of the system. It's a guideline for the coders. Coders use this document to implement the system so that it perfectly fulfils the user requirements and meets the stakeholder's expectations.



## **Department of Computer Science and Engineering, University of Chittagong.**

<b>Course Code</b>	CSE 516
<b>Course Title</b>	Software Engineering and Design pattern Lab
<b>Assignment name</b>	Implementation Document
<b>Report number</b>	07
<b>Group</b>	E

**Submitted to:**

**A. H. M. Sajedul Hoque**

Assistant Professor

Department of Computer Science and Engineering,

University of Chittagong.

**Submitted by**

Name: Abdul Aziz ID: 18701032

Name: MD. Mohsin Hossain ID: 18701057

Name: MD. Akram Hossain ID: 18701071

Name: MD. Rofiqul Islam ID: 17701101

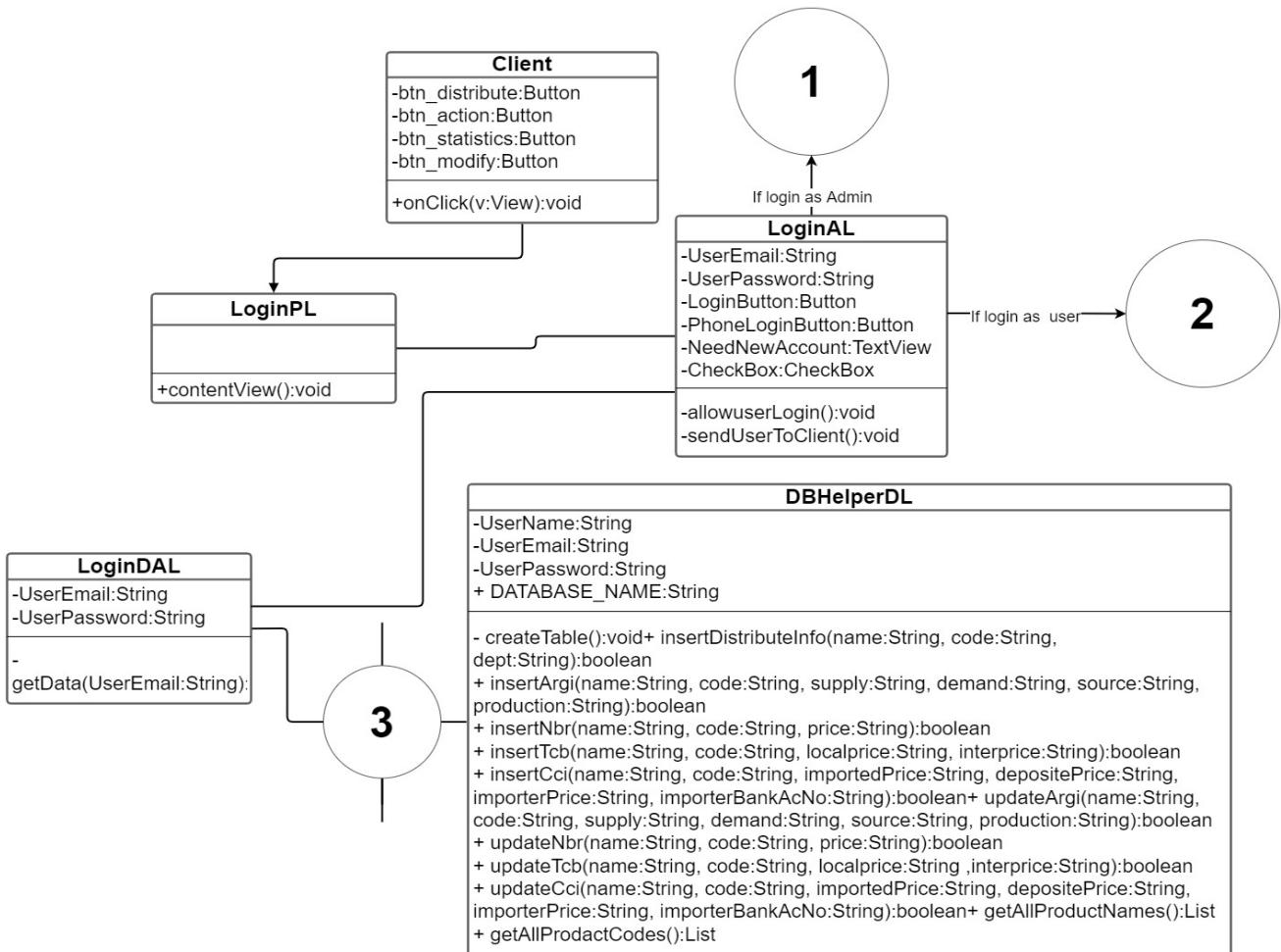
## 1. Introduction

MarketWatch System is a computerized system which aims to show market statistics, control price hiking, making the product inconsistency list easily, fast, secure and reliable. It gives the administration to distribute product information scripts, modify the product information and users can submit product information. Rather, users can show the market statistics, view product information and action activity. Using this system, manage the market strategy will be more time efficient and error free.

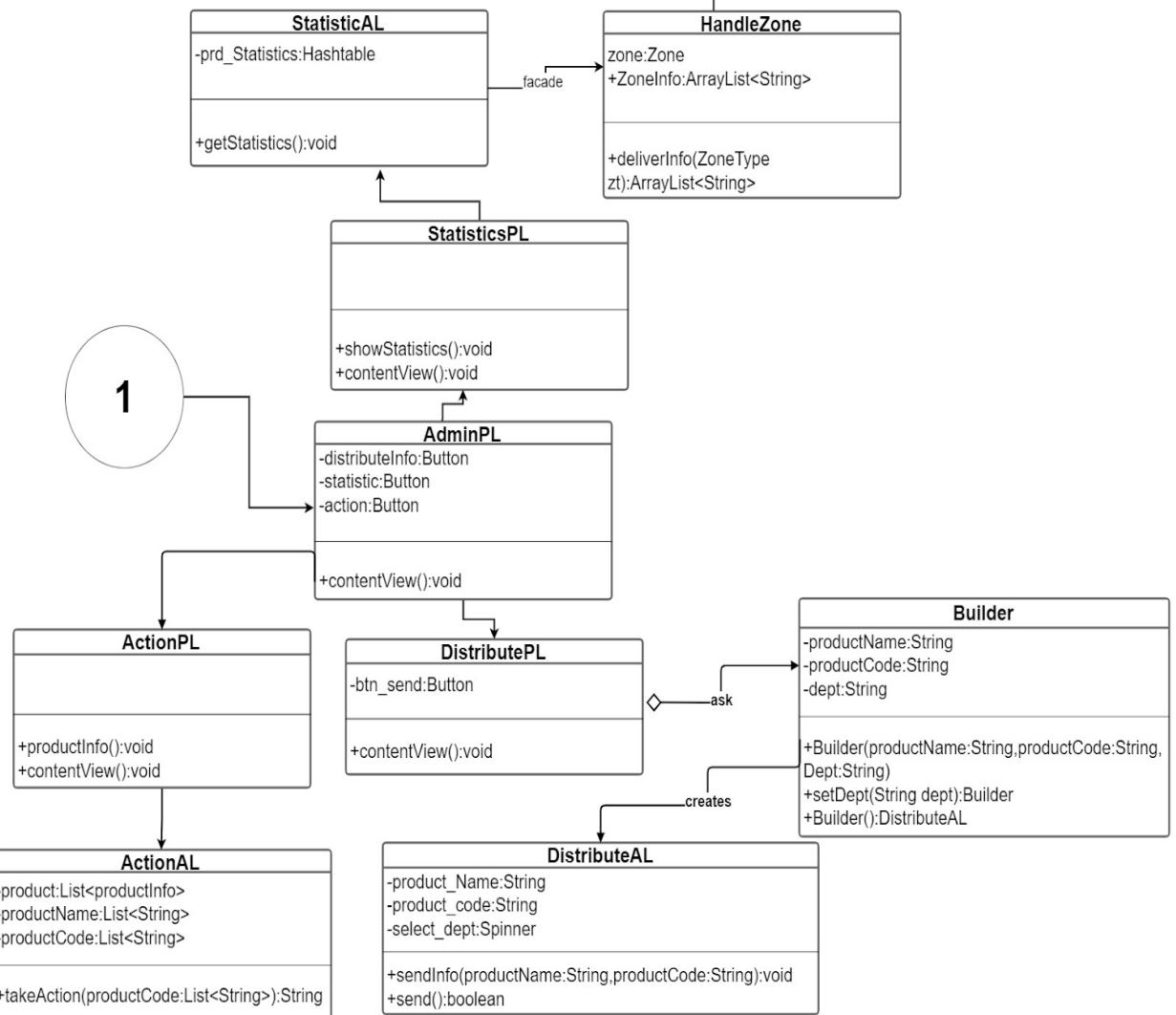
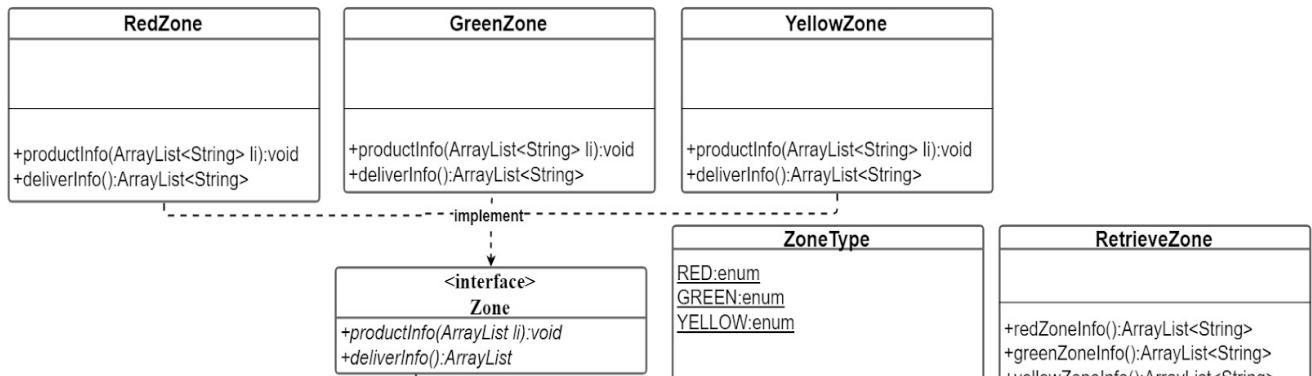
This report consists of seven sections. Section 2 shows the architectural design which specify the connection of class diagram and layout of class. in section 3, includes the architectural pattern used in 3 tier architecture. Rather, section 4 show the details and implantation class diagram. The brief graphical demonstration of the system is shown in section 5.in Section 6, includes the limitations of the system and the future works for the next versions are described at Section 7. The concluding remarks for the report is given at section 7.

## 2. Architectural design

In the architectural design, we provide the detail class diagram where we include variable and methods with data type. We have also included interface and applied design patterns.



**Figure 2.1: Architectural Design**



**Figure 2.2: Architectural Design**

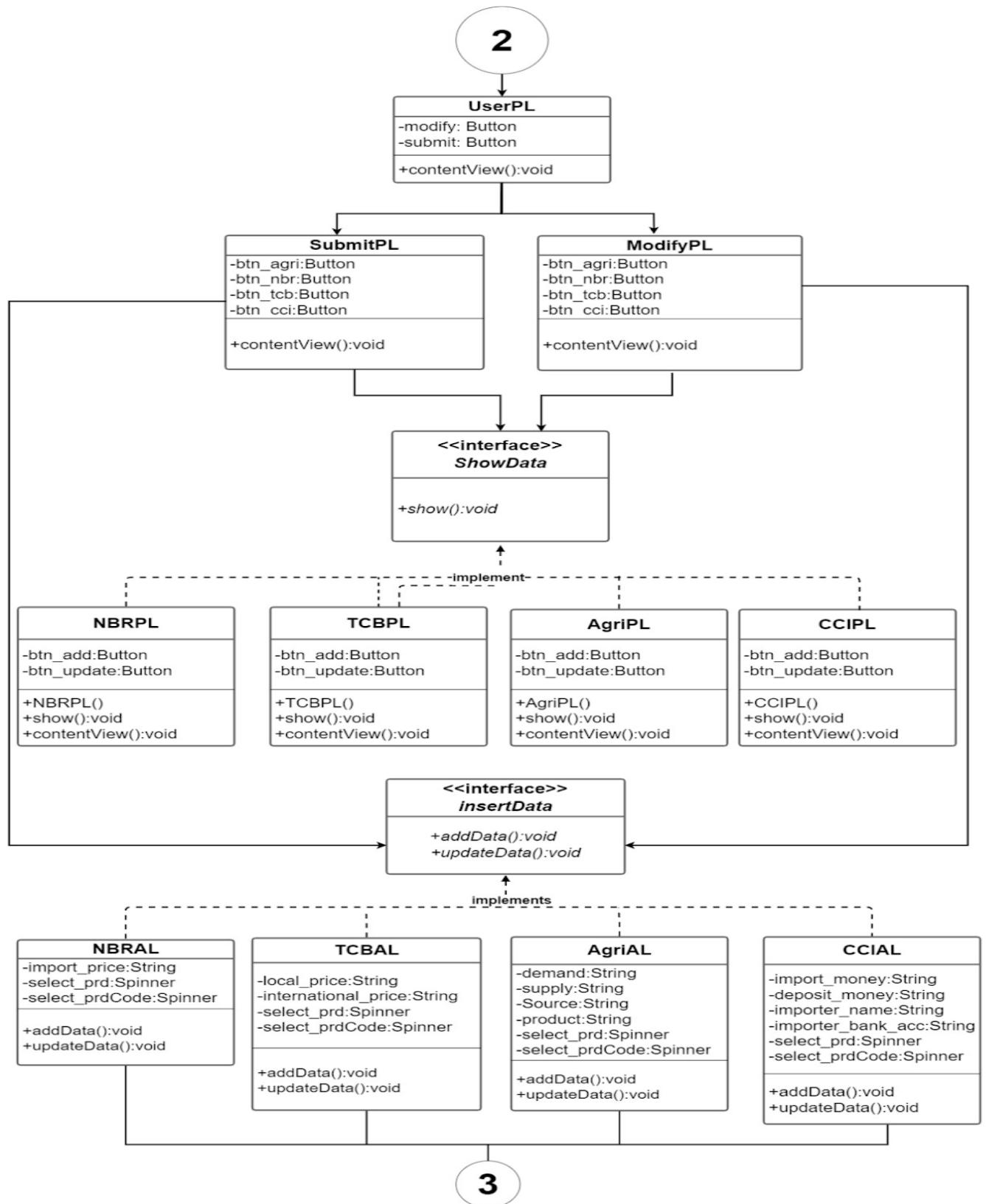
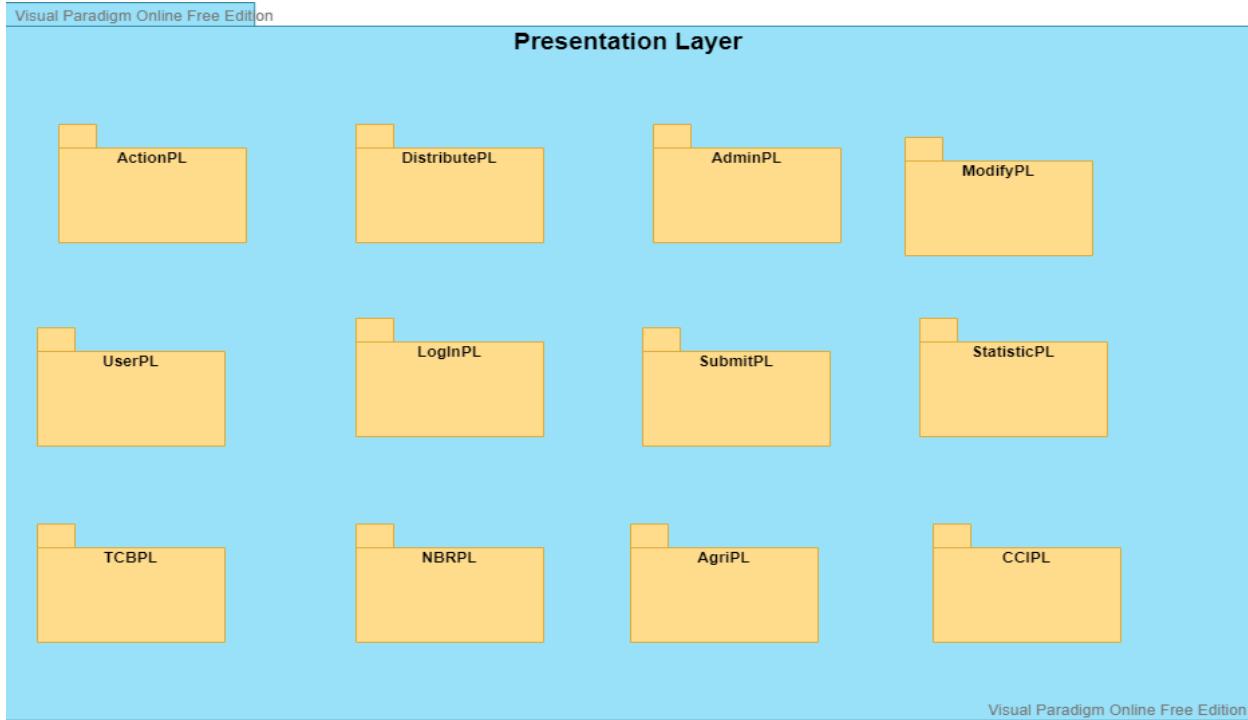


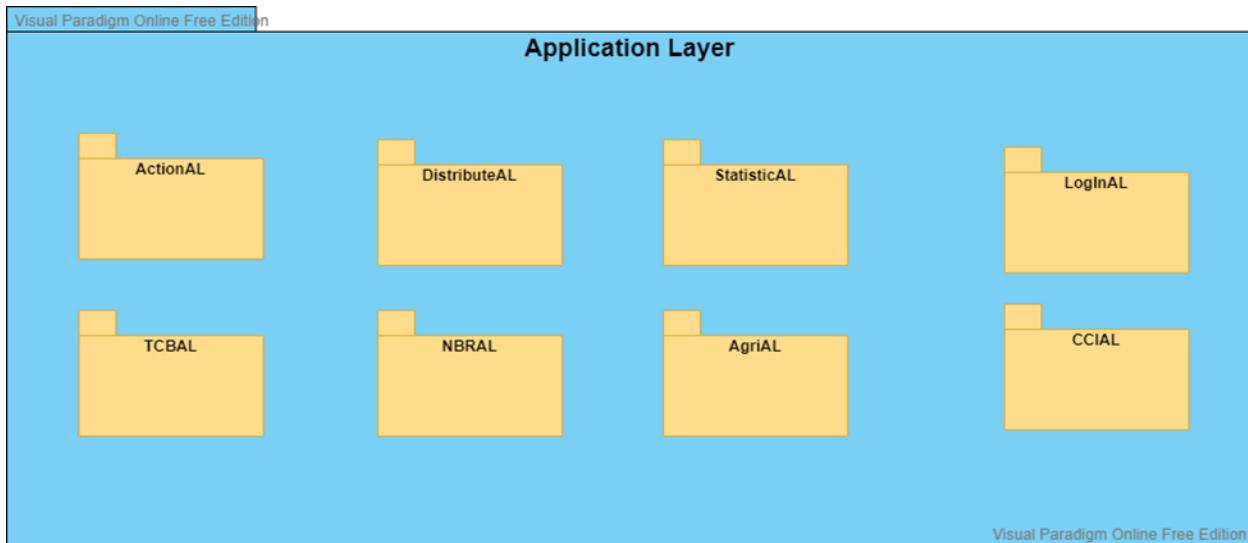
Figure 2.3: Architectural Design

### 3. Architectural Pattern

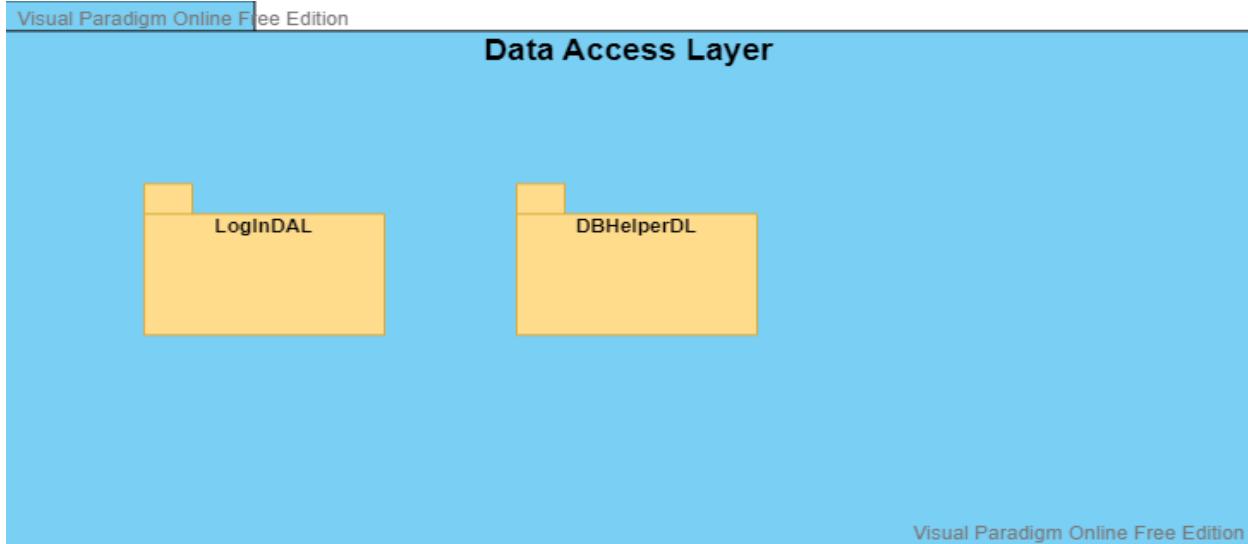
MarketWatch is a computer based system which is controlling market price fluctuations. To ensure the security we choose the 3-tire architectural pattern. This architecture pattern have 3-layer including presentation layer, Application layer and data access layer. Our system's architectural pattern package diagram is given below.



**Figure 3.1:** Presentation Layer



**Figure 3.2:** Application Layer



**Figure 3.3: Data Access Layer**

#### 4. Detail Design

Since, we develop our system using object oriented paradigm. In detail design, we use program description language (PDL) representation to determine each method what it does. Those method are including addDataIntoDb(), updateDataIntoDb(), AllowUserLogin(), getData(), productInfo(), getProductName(), getProductCode(), getMessage(), setProductName(), setProductCode(), setMessage(), sendInfo(), send(), and takeAction().

##### **addDataIntoDb() Method**

```
public void addDataIntoDb()
```

the method insert data into the database

```
{
```

create an instance of DBHelperDL;

```
DBHelperDL mydb = new DBHelperDL();
```

```
if(mydb.insert(respective department parameter) is true)
```

```
{
```

invoke insert method for inserting department-wise product information show appropriate message for successfully inserting data

```
}
```

else

show appropriate message for unsuccessful attempt

}

### **updateDataIntoDb() Method**

public void updateDataIntoDb()

the method update data into the database

{

    create a mydb instance of DBHelperDL

    if(mydb.update(respective department parameter) is true)

    {

        invoke update method for updating deparatmen-twise product information show appropriate message for successfully inserting data

    }

else

    show appropriate message for unsuccessful attempt

}

### **AllowUserLogin() Method**

public void AllowUserLogin(String email, String password)

the method takes email and password

{

    String email, password;

    check whether email and password are empty or not

    if(email is empty)

        show appropriate message for empty email

    else if(password is empty)

        show appropriate message for empty password

    else

        check email and password with database email and password

        if(email and password are correct)

            show appropriate message for successful login

        else

```
        show appropriate message for unsuccessful attempt  
    }
```

### **getData() Method**

```
public String getData( String email )
```

the method takes email

```
{  
    String password;  
    create an instance of DBHelperDL  
    DBHelperDL db = new DBHelperDL();  
    invoke method to fetch password  
    password = db.getPassword(email);  
    return password;  
}
```

### **productInfo() Method**

```
public void ProductInfo(String productName, String productCode, String message)
```

The method creates the product info model

```
{  
    assign the data in parameter list to instance data member  
    set productName equal to productName;  
    set productCode equal to productCode;  
    set message equal to message;  
}
```

### **getProductName() Method**

```
public String getProductName()
```

The method returns productName

```
{  
    get instance's data productName  
    return productName;  
}
```

**getProductCode() Method**

```
public String getProductCode()
```

The method returns productCode

```
{
```

```
    get instance's data productCode
```

```
    return productCode;
```

```
}
```

**getMessage() Method**

```
public String getMessage()
```

The method returns message

```
{
```

```
    get instance's data message
```

```
    return message;
```

```
}
```

**setProductName() Method**

```
public void setProductName(String message)
```

The method assign productName to instance variable

```
{
```

```
    set productName to instance variable productName
```

```
}
```

**setProductCode() Method**

```
public void setProductCode(String productCode)
```

The method assign productCode to instance variable

```
{
```

```
    set productCode to instance variable productCode
```

```
}
```

### **setMessage() Method**

```
public void setMessage(String message)
```

The method assign message to instance variable

```
{
```

```
    set message to instance variable message
```

```
}
```

### **sendInfo() Method**

```
public void sendInfo( String ProductName ,String ProductCode, String Dept)
```

the method set the Products information and send the product information into the database

```
{
```

set the parameter of method

create an instance of Database helper

```
DBHelperDL db =new DBHelperDL();
```

```
}
```

### **send() Method**

```
public Boolean send()
```

this method check the product information insertion

```
{
```

check whether product information either insert or not insert.

```
if(db.insertDistributeInfo(productName,productCode,dept) is true)
```

```
{
```

```
    return true;
```

```
}
```

```
else
```

```

        return false;
    }

takeAction() Method

public ArrayList<ProductInfo> takeAction()
{
    ArrayList<ProductInfo> products;
    ArrayList<String> productName;
    ArrayList<String> productCode;
    ArrayList<String> productMessage;

    create an instance of Database
    Database db = new Database();

    invoke getAllProductNames method to fetch productName list from Database
    productName = db.getAllProductNames();

    invoke getAllProductCodes method to fetch getAllProductCodes list from Database
    productCode = db.getAllProductCodes();

    invoke getAllProductMessage method to fetch getAllProductMessage list from Database
    productMessage = db.getAllProductMessage();

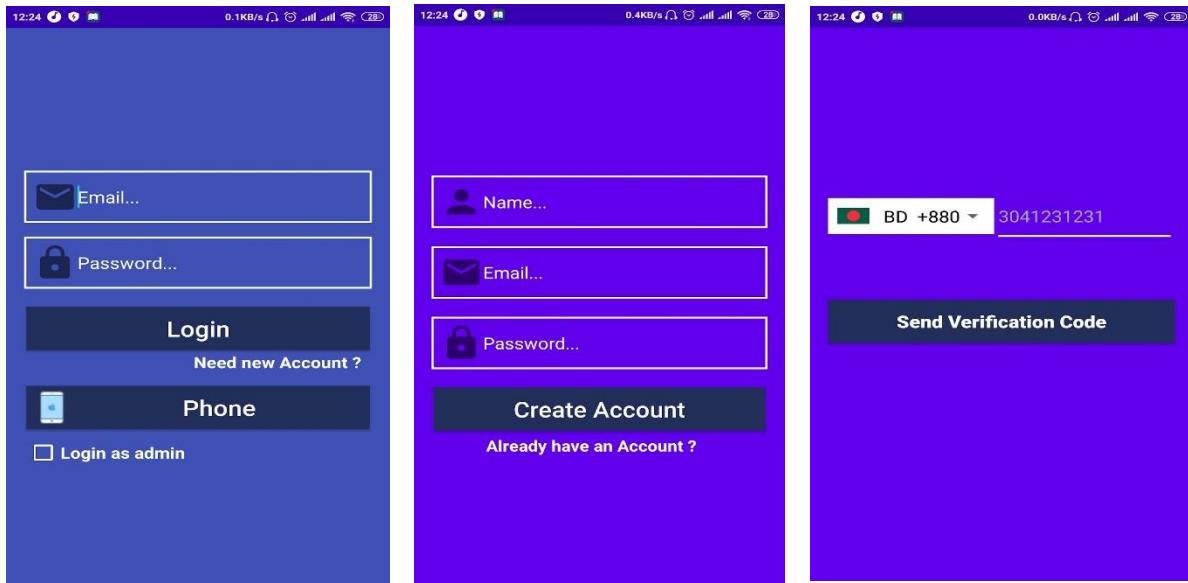
    check number of products in database
    if(number of products greater than 0){

        add productName,productCode and productMessage into products list
    }
}

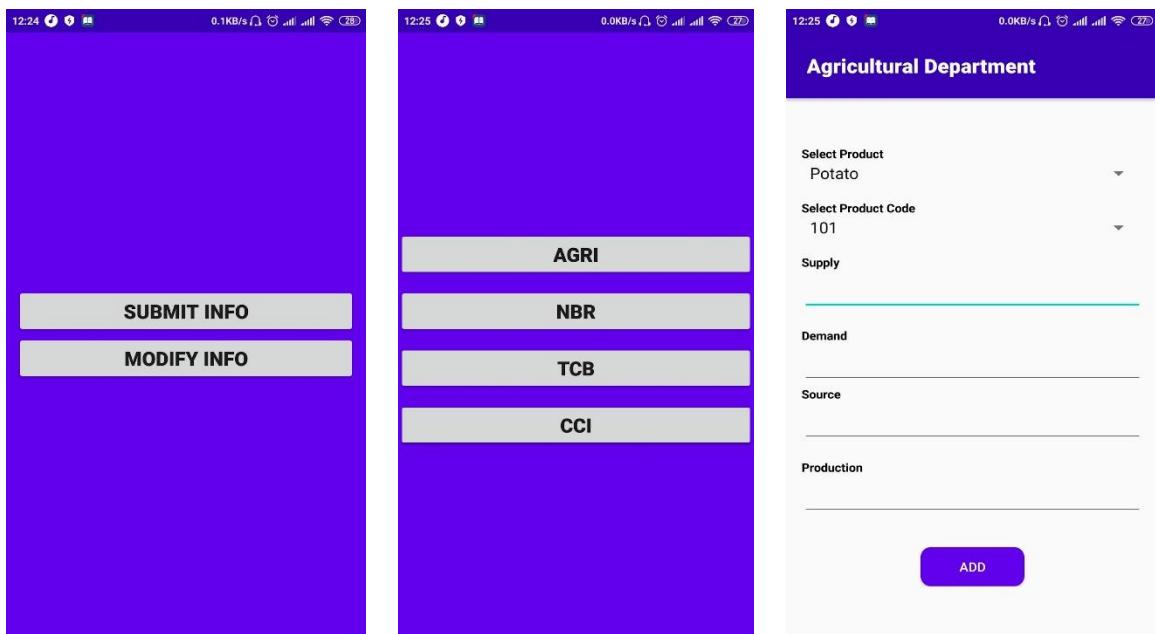
```

## 5. Screen shoot of System

MarketWatch is an android based solution which have functionalities including login, Information scripts distribution, submit and view statistics, and take action against inconsistency. These functionalities of our system are shown below via screen shots in figure 5.1 through 5.5.



**Figure 5.1:** System Login Demonstration



**Figure 5.2:** Submit and Modify information Demonstration

**NBR Department**

- Select Product: Potato
- Select Product Code: 101
- Imported Price:
- ADD button

**TCB Department**

- Select Product: Potato
- Select Product Code: 101
- Local Price:
- International Price:
- ADD button

**CCI and E Department**

- Select Product: Potato
- Select Product Code: 101
- Imported money:
- Deposit money:
- Importer name:
- Importer Bank Account no:
- ADD button

**Figure 5.3:** Submit and Modify information Demonstration

**DISTRIBUTE INFO**

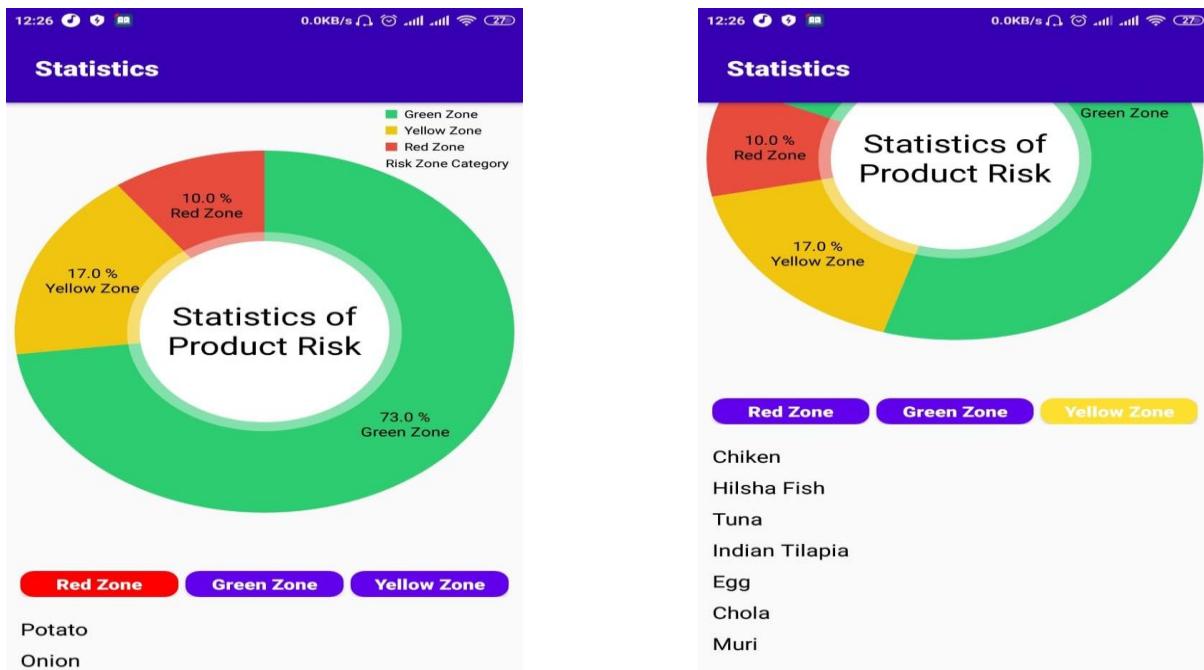
- Product Name:
- Product Code:
- Select Department: TCB
- SEND button

**STATISTIC**

**ACTION**

Prod. Name	Prod. Message
Potato	Everything is ok
Onion	Everything is ok

**Figure 5.4:** Distribute info, Statistics, and Action of Admin Demonstration



**Figure 5.5:** Statistics Demonstration with Pie Chart

## 6. Limitation

Since MarketWatch is a real-time system, its performance depends on the user's data integrity. The major limitations of our system are pointed below:

- Data integrity may be hampered the system performance
- Mass amount of data need to be processed so system performance may be slow down
- Various types of actors may not do their job in time will hamper the overall performance.
- Action may be affected due to a non-standardized tolerance value used in the system.
- Since it contains some sensitive information of the government then data security is a vulnerability of our system.

## 7. Conclusion and Future work

MarketWatch System is an automated computer-based solution to controlling market price fluctuations. We try a Software-based system solution to monitor the overall process of a good delivery from producer to consumer and up to date give information to the government to take rapid action before a good market price hiking. Using the MarketWatch system admin can able to send distributed scripts for the needed product. Also, departments including Agriculture, TCB, NBR, and CCI&E can submit their information regarding the distribution script. Our system is able to show the statistics of products which helps the government to take action if needed.

In future, we have a plan to develop a machine learning model which will predict the earlier action that has to be taken to control the upcoming price hiking problem for a particular good. It will also alert the admin

about the probable unrestrained horse and jockey of the market. Also, We have a plan to connect all zones via IoT to collect real-time information.