



# **Daffodil International University**

Department of Software Engineering, FSIT

**SWE-431 Project / Thesis**

**Project Document**

**SR(Sale Representative) online delivery  
system**

Supervised by:

**Ms. Nusrat Jahan**

**Senior Lecturer**

**Department of Software  
Engineering Daffodil International University**

**Submitted by:**

**Tanvir hossain  
161-35-1519**

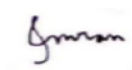
Department of Software Engineering

Daffodil International University

# Approval

This **Project** titled “SR(Sale Representative) online delivery system”, submitted by **Tanvir hossain, ID-161-35-1519** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc in Software Engineering and approved as to its style and contents.

## BOARD OF EXAMINERS



Chairman

---

**Dr. Imran Mahmud**  
Associate Professor & Head In-Charge  
Faculty of Science and Information Technology  
Department of Software Engineering  
Daffodil International University



Internal Examiner 1

---

**Dr. Md. Asraf Ali**  
Associate Professor  
Faculty of Science and Information Technology  
Department of Software Engineering  
Daffodil International University



Internal Examiner 2

---

**Mr. Md. Anwar Hossen**  
Lecturer (Senior Scale)  
Faculty of Science and Information Technology  
Department of Software Engineering  
Daffodil International University



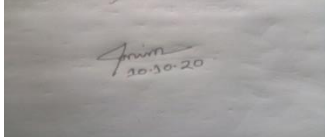
External Examiner

---

**Prof. Dr. Mohammad Abul Kashem**  
Professor  
Department of Computer Science and Engineering  
Dhaka University Of Engineering and Technology,  
Gazipur

## DECLARATION

We hereby declare that we have taken this thesis under the supervision of **Ms. Nusrat Jahan, Senior Lecturer, Department of Software Engineering, Daffodil International University**. We also declare that neither this project nor any part of this has been submitted elsewhere for award of any degree.

A photograph of a handwritten signature and the date '10-10-20' on a piece of paper.

Tanvir hossain

ID: 161-35-1519

Batch: 19<sup>th</sup>

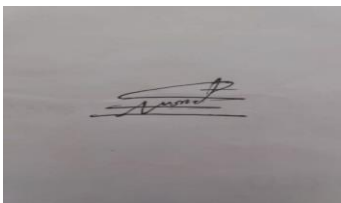
Department of Software Engineering

Faculty of Science & Information

Technology

Daffodil International University

Certified by:

A photograph of a handwritten signature on a piece of paper.

***Nusrat Jahan***

Senior Lecturer

Department of Software Engineering

Faculty of Science & Information

Technology

Daffodil International University

## Acknowledgement

First of all, thanks to ALLAH, ALHAMDULILLAH because I am very much blessed as I have successfully reached towards the final semester. From the very beginning of my university life, I have learned a lot about software engineering as well as core computer science related knowledge from my course teachers. Moreover, they teach us ethics, morality and politeness.

It is a fortunate opportunity for me as a student of the Department of Software Engineering, one of the exalted academic cancers of the Science and Information Technology Faculty of the Daffodil international University, to express my deep feelings of gratitude to the department and to my honorable teachers and also to the department staff.

Besides, I am so much thankful to my parents that, I was always supported by them. They always prioritize my opinion and inspired me as well.

I am also so many grateful to my supervisor **Ms Nusrat Jahan**, Senior Lecturer, Department of Software Engineering, FSIT. Daffodil International University, Dhaka, for his excellent guidance, inspiration, encouragement and also for allowing me to work with this project. It was not possible for me to complete my project paper successfully without his help.

Specially, I am very thankful to my friend, elder and younger brothers of our university who always appreciate me to my studies and my daily works for learning new ones.

# ABSTRACT

This project is aimed to developing an SR(Sale Representative) online delivery system. The entire project has been developed keeping in view of the distributed admin and SR, in mind. The project entitled “SR(Sale Representative) online delivery system” is a web application. It is developed using .NET and MICROSOFT SQL SERVER. This project has tried to incorporate all the advanced features of .NET such as to fulfill the aim of the project. The application has been designed such that it encompasses the managerial as well as public functionalities. Among the managerial functionalities are the task performed by the various offices. The public features are exercised by the users and outsiders according to the permissions level availed to them.

<b>Contents</b>		Page No
<b>Chapter 1: Introduction.....</b>		<b>1</b>
1.1Project Overview.....		2
1.2Project Purpose.....		2
1.2.1Background.....		2
1.2.2Benefits & Beneficiaries.....		3
1.2.3Goals.....		3
1.3Stakeholders.....		3
1.4Proposed System Model .....		4
1.5Project Schedule.....		4
1.5.1Gantt Chart.....		5
1.5.2Release Plan/Milestone.....		6
<b>Chapter 2: Software Requirement Specification .....</b>		<b>7</b>
2.1Functional Requirements.....		8
2.1.1 Crate Account For Login.....		8
2.1.2 Manage Account for admin and SR.....		8
2.1.3 Add product for SR.....		9
2.1.4 View product info for SR.....		9
2.1.5 Manage product information.....		9
2.1.6 Manage add salary information.....		10
2.1.7 View salary information.....		10
2.1.8 Manage salary information by admin.....		10
2.1.9 stock information added admin.....		11
2.1.10 adding price information for both client and manager....		11
2.1.11 Managing order information SR and admin.....		11
2.1.12 Review view b client and admin.....		12
2.1.13 Comment submitted by SR		12
2.2Data Requirements.....		13
2.3Performance Requirements.....		13
2.3.1 Speed and Latency Requirements.....		13
2.3.2 Precision or Accuracy Requirements.....		14
2.3.3 Capacity Requirements.....		14
2.4Dependability Requirements.....		15
2.4.1 Reliability & Availability Requirements .....		15
2.4.2 Robustness or Fault-Tolerance Requirements.....		16
2.4.3 Safety-Critical Requirements.....		16
2.5 Maintainability and Supportability Requirements.....		16
2.5.1 Maintenance Requirements.....		16
2.5.2 Supportability Requirements.....		16
2.6Security Requirements.....		17
2.6.1 Access Requirements.....		17

2.6.2 Integrity Requirements.....	18
2.6.3 Privacy Requirements.....	18
2.7 Usability and Human-Interaction Requirements... ..	18
2.7.1 Ease of Use Requirements.....	18
2.7.2 Personalization and Internationalization Requirements.	19
2.7.3 Understandability and Politeness Requirements.....	19
2.7.4 Accessibility Requirements.....	19
2.7.5 User Documentation Requirements.....	19
2.7.6 Training Requirements.....	19
2.8 Look and Feel Requirements.....	20
2.8.1 Appearance Requirements.....	20
2.9 Operational and Environmental Requirements.... ..	20
2.9.1 Expected Physical Environment.....	20
2.9.2 Requirements for Interfacing with Adjacent Systems...	20
2.9.3 Release Requirements.....	21
2.10. Legal Requirements... ..	21
2.10.1 Compliance Requirements.....	21
2.10.2 Standards Requirements.....	21
<b>Chapter 3: System Analysis.....</b>	<b>22</b>
3.1 Use Case Diagram..... ..	23
3.2 Use Case Description .....	24
3.1.1 Create Account.....	24
3.1.2 Manage Account.....	25
3.1.3 Add product info.....	26
3.1.4 View product info.....	27
3.1.5 Manage product info.....	28
3.1.6 Add salary info.....	29
3.1.7 View salary info.....	30
3.1.8 Manage salary info.....	31
3.1.9 Add stock info.....	32
3.1.10 add price info.....	33
3.1.11 Manage order info.....	34
3.3 Activity Diagram (for each use case) .....	37
3.2.1 Create account.....	37
3.2.2 Manage account.....	38
3.2.3 Add product info.....	39
3.2.4 View product Info.....	40
3.2.5 Manage product Info.....	41
3.2.6 Add salary Info.....	42

3.2.7View salary Info.....	43
3.2.8Manage salary Info.....	44
3.2.9Add stock Info.....	45
3.2.10add price Info.....	46
3.2.11Manage order Info.....	47
3.4System Sequence Diagram (for each use case).....	50
3.3.1Create Account.....	51
3.3.2Manage Account.....	49
3.3.3Admin info.....	52
3.3.4SR info.....	52
<b>Chapter 4: System Design Specification.....</b>	<b>58</b>
4.1Development tools and technology .....	59
4.2Database Design Diagram.....	59
4.3Development Tools & Technology.....	60
4.4.1User Interface Technology.....	61
4.4.1.1  Programming Language.....	61
4.4.1.2  JQuery UI.....	61
4.4.1.3  CSS framework or Bootstrap.....	61
4.4.2Implementation Tools & Platforms.....	
4.4.2.1Integrated Development Environment.....	62
4.4.2.2Database Server (MYSQL Server V-10.1.36- MariaDB).....	62
4.4.2.3Web Server.....	62
<b>Chapter 5: System Testing.....</b>	<b>63</b>
5.1Testing Features.....	64
5.1.1Features to be tested.....	64



5.2 Testing Strategies.....	65
...	
5.2.1 Test Approach.....	65
5.2.1.1 Black Box Testing.....	65
5.2.1.2 Equivalent Class Partitioning.....	65
5.2.1.3 Boundary Value Analysis.....	66
5.2.1.4 White Box Testing.....	66
5.2.2 Pass/Fail Criteria.....	66
5.2.3 Testing Schedule.....	67
5.2.4 Traceability Matrix.....	67
5.3 Testing Environment .....	68
5.4 Test Case... ..	68
5.4.1 Log in.....	69
<b>Chapter 6: User Manual.....</b>	<b>58</b>
6.1 User Interface.....	58
6.1.1 log In page.....	59
6.1.2 Registration page.....	59
6.1.3 Admin home page .....	60
6.1.4 Pending request .....	60
6.1.5 SR home page .....	61
6.1.6 Product adding .....	61
6.1.7 Stock adding .....	62
6.1.8 List of product .....	62
6.1.9 salary adding.....	63
6.1.10 salary list.....	64
6.1.11 Take order.....	65
<b>Chapter 7: Project Summary.....</b>	<b>66</b>
7.1 Github Link.....	67
7.2 Summary(Critical Evolution) .....	68

7.3Limitations.....	69
7.4Obstacles & Achievements.....	70
7.5Future Scope.....	70
7.6References.....	71

### **List of Figures**

	<b>Page No</b>
Figure 1.1: Proposed System Model	2
Figure 1.2: Gantt Chart	5
Figure 3.1: Use Case diagram	23
Figure 3.2: Create account	37
Figure 3.3: Manage account	38
Figure 3.4: Add product info	39
Figure 3.5: View product Info	40
Figure 3.6: Manage product Info	41
Figure 3.7: Add salary Info	42

Figure 3.8: View salary Info	43
Figure 3.9: Manage salary Info	44
Figure 3.10: Add stock Info	45
Figure 3.11: add price Info	46
Figure 3.12: Manage order Info	47
Figure 3.15: Create account	50
Figure 3.16: Manage account	51
Figure 3.17: Add product info	49
Figure 3.17: View product Info	52
Figure 3.18: Manage product Info	52
Figure 3.19: Add salary Info	53
Figure 3.20: View salary Info	53
Figure 3.21: Manage salary Info	54
Figure 3.22: Add stock Info	54
Figure 3.23: Add price Info	55
Figure 4.1: Database Diagram	59
Figure 6.1: log in	60
Figure 6.2: Registration	61
Figure 6.3: admin home page	61
Figure 6.4: pending request	62
Figure 6.5: SR home page	62
Figure 6.6: product adding	63
Figure 6.7: stock adding	63
Figure 6.8: list of product	64
Figure 6.9: salary adding	64
Figure 6.10: salary list	65



# **Chapter 1**

## **Introduction**

## **1.1 Project Overview**

Today we have computers with large computing power and almost every business is going to take the advantages of using those technologies My Project is To Take Order Via Online And Generate A Details Of Order Taken By Sr Instead Of Traditional Way.. Admin Will Upload And Edit Stock, Upload And Edit Employees Salaries, Approved User Account, Upload And Edit Outlet Info. Sale Representative Will Take Order Can Edit Profile. Sr, Will Eligible To See Their Order And Salary Details By Search. Delivery Person Can View Taken Order And See Theirs Salary History By Search Their Id.. Shopkeeper Can Drop A Comment Or Give Their Complain , Confirm Order. The Mainly User Of My Project Is Admin, Sale Representative . This Project Will Reduce Complexity In Taking Order Delivery Process.

## **1.2 Project Purpose**

The main purpose of this project named “**SR online delivery system**” is to make an automation system which might be helpful for millions of users from different place of country by giving them a few information. Therefore, we are going to develop such a project.

### **1.2.1 Background**

Suddenly I notice that, people from our country come shop from different places is facing a lot of problem where they live. Most of the time they select wrong order for company and there are a few shop remaining to escape those problem. Thousands of SR are being suffering. So considering that, I have brainstormed and thinking that my project is going to solve their problems.

### 1.2.2 Benefits & Beneficiaries

The module “SR online delivery system” applications would be beneficial for some point of view. Now, I am mentioning those

Below:

- My system helps SR and admin to make registration easily
- It helps authenticated access to the system.
- It would be helpful for find the product.
- It is helpful for find best value for the product.
- It is also helpful for determining required product packing.
- It helps client and SR to find product within a moment.
- This application makes authentication system for product finding system.
- Our app helps company to increase their business.
- This app helps client and SR manager to work easily.
- The system also going to reduce unwanted problems
- Only registered person can get access to the system.

I have also mentioned some benefits as well as beneficiaries. So, I think this application is very much helpful for users

### 1.2.3 Goals

The main scope of this project is to develop an application based on android device. As most of shop are giving low quality service, so I implement my system firstly for those persons who wants.

This document is also aimed at specifying requirements of software to be developed but it can also be applied to assist in the selection relation between the different stakeholders. The standard can be used to create software requirements specifications directly or can be used as a model for defining the system requirements.

## 1.3 Stakeholders

There are two types of stakeholders in my total “**SR online delivery system**” system. Such as:

<b>Admin/SR</b>	Admin can add their product , location, food salary. They can update their product information.
<b>SR/Normal User</b>	User can registered my system, they find product with specific location, give review and read review of other user. They can contact with SR and see the product information.

## 1.4 Proposed System Model

Before going to develop a system it is very important to have a system model. I have find system model. This model proposed system in brief.



## 1.5 Project Schedule

We need to prepare a scheduling plan to complete the project on time. It also refer to make communication with what task need to get done within timeframe.



## 1.5.1 Gantt Chart

Activities	Weeks->	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28
Planning	Ideas	█	█																										
	Problem definition	█	█	█	█	█																							
	Proposal planning	█	█	█	█	█	█																						
Requirements	Requirements specification		█	█	█	█	█	█																					
	Requirements Analysis			█	█	█	█	█	█																				
QA-1	Quality Assurance							█																					
System	Sketching								█	█																			
	Design Specification								█	█	█																		
	Database design									█	█	█																	
Mid defense												█	█																
Implementation - 1	Administration Part									█	█	█	█	█	█	█	█	█											
	Test case					█							█	█	█	█	█												
Implementation - 2	Client's Part & authentication part															█	█	█	█	█	█	█	█	█	█	█	█		
	Testing							█			█																		
Delivery & Final defence	Unit testing																												
	Black box Testing																												
Delivery & Final defence	Software Release																												
	Scheduled Time	█																											
	Buffered Time	█																											

## 1.5.2 Release Plan or Milestone

<b>Activities</b>	<b>Duration in week</b>	<b>Total week</b>
<b>Brainstorming</b>	Week no. (1,2)	2
<b>Problem identification</b>	Week no. (1,2,3,4)	4
<b>Requirement specification</b>	Week no. (2,3,4,5)	4
<b>Requirement analysis</b>	Week no. (3,4,5,6)	4
<b>Sketching</b>	Week no. (8,9)	2
<b>Design specification</b>	Week no. (8,9)	2
<b>Database design</b>	Week no. (9,10)	2
<b>Update product info and add product information</b>	Week no. (9,10,11,12,13,14)	6
<b>Quality assurance(1,2)</b>	Week no. (7,12,13,14,15)	5
<b>Test case</b>	Week no. (12,13,14,15)	4
<b>Impose case &amp; demerits</b>	Week no. (15,16,17,18,19,20,21,22)	8
<b>Unit testing</b>	Week no. (22,23,24)	3
<b>Black-box testing</b>	Week no. (25,26)	2
<b>Software release</b>	Week no. (27,28)	2

# **Chapter 2**

## **Software Requirement Specification**

## 2. Requirement Specification

The complete requirement specification based on the elicitation process is described in this section.

### 2.1 Functional Requirements:

Functional requirements refer to the functions which are mandatory to the system. Functional requirements must be able to perform on the software system. Every system must have some functional requirements. Now, I am going to mention functional requirements associating with my project.

#### 2.1.1 Create Account for Login

<b>Requirements 1</b>	<b>Create Account.</b>
Description	After Click on sign in button if user don't have an account then there is two option for registration, one is for admin and other one is for SR.
<b>Stakeholders</b>	Admin and SR.

#### 2.1.2 Manage Account for admin and SR

<b>Requirements 2</b>	<b>Manage Account.</b>
Description	After Complete sign in. Both user admin and SR can successfully login and log out from the system.
<b>Stakeholders</b>	Admin and SR.

### 2.1.3 Add product for SR

<b>Requirements 3</b>	<b>Add product info.</b>
Description	admin can add product info and SR can view product information.
<b>Stakeholders</b>	Admin and SR.

### 2.1.4 View product info for SR

<b>Requirements 4</b>	<b>View product info.</b>
Description	After Completing registration and login user(SR) can view the information of product.
<b>Stakeholders</b>	SR.

### 2.1.5 Manage product information

<b>Requirements 5</b>	<b>Manage product info.</b>
Description	admin can inset update and delete his product information.
<b>Stakeholders</b>	admin.

### 2.1.6 add salary information

<b>Requirements 6</b>	<b>Add salary info.</b>
Description	After Complete sign in his own account admin can add salary information..
<b>Stakeholders</b>	admin.

### 2.1.7 View salary information.

<b>Requirements 7</b>	<b>View salary info.</b>
Description	After completing all sign in issue user can view salary information..
<b>Stakeholders</b>	SR.

### 2.1.8 Manage salary information by admin.

<b>Requirements 8</b>	<b>Manage salary info.</b>
Description	After inserting salary information he can update his information .
<b>Stakeholders</b>	admin.

### 2.1.9 stock information added by admin.

<b>Requirements 9</b>	<b>Add salary info.</b>
Description	Admin can add his stock in my system. He also can update and delete his salary information.
<b>Stakeholders</b>	Admin .

### 2.1.10 add price information for admin.

<b>Requirements 10</b>	<b>View price info.</b>
Description	After adding price information by admin and admin can see the product item and product menu.
<b>Stakeholders</b>	Admin .

### 2.1.11 Managing salary information by admin.

<b>Requirements 11</b>	<b>Manage order info.</b>
Description	SR can take order and admin and SR both can view the order page .
<b>Stakeholders</b>	Admin and SR .

### 2.1.12 Review views by Admin and SR.

<b>Requirements 12</b>	<b>View review.</b>
Description	Admin and SR can view review. For this user and client have to sign in the system with their registered accounts.
<b>Stakeholders</b>	Admin and SR.

### 2.1.13 Comment submitted by SR.

<b>Requirement 13</b>	<b>Comment.</b>
Description	SR can share their experience with above system by giving their comment or we can say giving rating. Only user(SR) can comment about system.
<b>Stakeholders</b>	SR.

## 2.2 Data Requirements:

For defining data requirements, we need to build the model. For my application maximum data would be loaded from Admin and SR. And for that purpose I need to focus on some major points.

Such as:

- Types of entity of the system
- Quantity of product
- product availability
- Quantity of data
- shop service elements.



## 2.3 Performance Requirements:

It is very important to maintain performance of any software system. To ensure performance, we need to maintain some steps. Now, I will explain some perspective by which we are going to enhance the performance of my project.

### 2.3.1 Speed & Latency Requirements:

Speed and latency requirements must be ensured while retrieving data from the cloud server.

<b>SLR-1</b>	<b>Search result must be faster</b>
<b>Description</b>	When SR search for a product by giving his Quantity, then the result must show within a second.
<b>Stakeholders</b>	Admin and SR

### 2.3.2 Precision & Accuracy Requirements:

Results that is to be shown to the end user is need to be accurate. Because, wrong information might be ruined the whole process.

<b>PAR-1</b>	<b>Search result must be accurate</b>
<b>Description</b>	When Admin or SR search anything the accurate result must show , with inserted information.
<b>Stakeholders</b>	Admin and SR.

### 2.3.3 Capacity Requirements:

The developed system by us must be capable to handle user data, provide accurate information, handling database, manage http request etc.

<b>CR-1</b>	<b>The system will handle many data</b>
<b>Description</b>	The system need to handle many data every moment.
<b>Stakeholders</b>	Admin and SR

### 2.4 Dependability Requirements:

The term dependability is measured based on four dimensions. Such as:

- Availability
- Reliability
- Safety
- Security

If we want to say that my application system is dependable then it must fulfill the four dimensions. But there are other tasks. Like there is no way to make mistakes or my system should have the ability to detect and then remove errors. Besides that, it is also very important to limit the damage which might be caused by system failure.

### 2.4.1 Reliability & Availability Requirements:

Now, I will mention requirement which is related to reliability and availability.

<b>RAR-1</b>	<b>The system must be available on 24 X 7</b>
<b>Description</b>	My system must be available all day long, every day in a week <ul style="list-style-type: none"><li>• The system must be updated regularly</li></ul>
<b>Stakeholders</b>	Admin ,SR.

### 2.4.2 Robustness or Fault-Tolerance Requirements:

To ensure robustness and fault-tolerance facilities to the end users, it is urgent to ensure 0% crush. Moreover, it must show accurate results.

<b>RFT-1</b>	<b>The system handles Clients and manager access without system errors</b>
<b>Description</b>	Not more than 3 Admin and SR might hit my application system at a time. All their requests must be handled without any fault.
<b>Stakeholders</b>	Admin and SR.

### 2.4.3 Safety-Critical Requirements:

There are no safety-critical requirements in my project.

### 2.5 Maintainability & Supportability Requirements:

It is very important to provide after service or support to the end users.

### **2.5.1 Maintainability Requirements:**

Not available for this project.

### **2.5.2 Supportability Requirements:**

Supportability requirements may have related to some extends. Like:

- Testability
- Maintainability
- Configurability
- Serviceability
- Install ability

My application meets all of the above requirements related to supportability.

### **2.6 Security Requirements:**

Making system security as a requirement is very important. System security requirements should be its functional requirement. System security enforces security of an application system.

Functionality related to system security can either be directly tested or observed. Some security related requirements is given below:

- Signing in as Admin and SR.
- Signing out as Admin and SR.
- Handling encrypted passwords.

While accessing to the system, each and every user must provide a central authentication mechanism. There is also a process to prevent entering into the system by ensuring hashed password for the unauthenticated users.

### 2.6.1 Access Requirements:

For accessing to my application system, there remain some authentication and authorization techniques. And every module of my system will provide it. Now I will provide an explanation below.

<b>AR-1</b>	<b>Application provides security mechanism</b>
<b>Description</b>	Every module is designed in such a way that it only gives access to the authorized and authenticated users.
<b>Stakeholders</b>	Admin and SR.

### 2.6.2 Integrity Requirements:

Integrity requirements refer to a security system which ensures an expectation of data quality. It also ensures that all data of the system would never be exposed to the malicious modification or accidental destruction. For that reason, I will store my user passwords as encrypted format which is impossible to decrypt. It is also called hashed password.

### 2.6.3 Privacy Requirements:

It is very important to ensure privacy of the system users. Privacy requirements enhance to protect stakeholder's privacy. In this way, all data or a partial part of data are going to be disclosed according to system's privacy policy. To ensure privacy, the central database should be protected by the anonymous. Planners permitted to get access to those data which are being associated by them which can be ensured by the total log in system.

### 2.7 Usability and Human-Interaction Requirements:

The main target of developing any system is to make the system user friendly and easy to usable for the end users.

### 2.7.1 Ease of Use Requirements:

My application is easy to use and also easily understandable.

<b>EUR-1</b>	<b>Application must be usable for the end users</b>
<b>Description</b>	This app is enough usable to the Admin and SR by which he can operate this system easily.
<b>Stakeholders</b>	Admin and SR.

### 2.7.2 Personalization and Internationalization Requirements:

There are not any personalization and internationalization requirements in my system. This maiden version of my application is only be operated by a Specific company.

### 2.7.3 Understand ability and Politeness Requirements:

It is already said that the application which I am going to develop, is understandable enough. The system provides hints to users whether any error occurred or wrong. By reading those errors client and manager can be able to operate the system easily.

**2.7.4 Accessibility Requirements:** Documentations are mainly two types. One is internal documentation which is generally written by the application engineers. It is prepared to make development life cycle easier for the system engineers or system analysts.

<b>UDR-1</b>	<b>The system engineer documentation</b>
<b>Description</b>	To develop my application named Smart online order system , firstly I have make a system analysis as well as documentation.
<b>Stakeholders</b>	System analysts or software developers

### 2.7.6 Training Requirements:

Training requirements involved in after service of any application. It is very necessary to properly train up end users to the system so that they would be capable to operate easily. After launching the full package to the market, firstly I provide training to the different end users like SR, admin, Registered SR, registered admin, Unregister Admin and SR

### 2.8 Look and Feel Requirements:

Look and feel requirement mainly refers how the system will look like and how the user interface of my system will display to the users.

#### 2.8.1 Appearance Requirements:

All users must know which input fields are required and which are not. For that reason, I will use labels for all input fields. Input fields might be text type, radio, checkbox etc.

<b>AR-1</b>	<b>Labels of mandatory fields must be bold</b>
<b>Description</b>	The mandatory field's label must be bold and all input fields must have placeholder to make it easier for the users.
<b>Stakeholders</b>	All users of the system

## **2.9 Operational and Environmental Requirements:**

Operational and environmental requirement refers to the capabilities, performance measurements, process, measurements of effectiveness, measurements of performance, measures of sustainability, measurements of technical performances etc.

### **2.9.1 Expected Physical Requirements:**

There are no expected physical requirements in my system.

### **2.9.2 Requirements for Interfacing with Adjacent Systems:**

There are no requirements for interfacing with adjacent system for my project. Because it is desktop and web based system.

### **2.9.3 Release Requirements:**

There are no specific release requirements in my system. Because it is an live project.

## **2.10 Legal Requirements:**

Legal requirements normally refer to the terms and conditions or privacy policy of any organizations. The terms and condition of our application is that, no third party software or persons are allowed to engage to use my data for their business purpose.

### **2.10.1 Compliance Requirements:**

There are no specific compliance requirements for my system.

### **2.10.2 Standards Requirements:**

As I said above there are no specific standards requirements for my system.



# **Chapter 3**

## **System Analysis**

### 3.1 Use Case Diagram

I have use case diagram. And there are two actors. Each actor plays different role. And those are already indicated to this use case diagram. This diagram will clarify our system in brief.

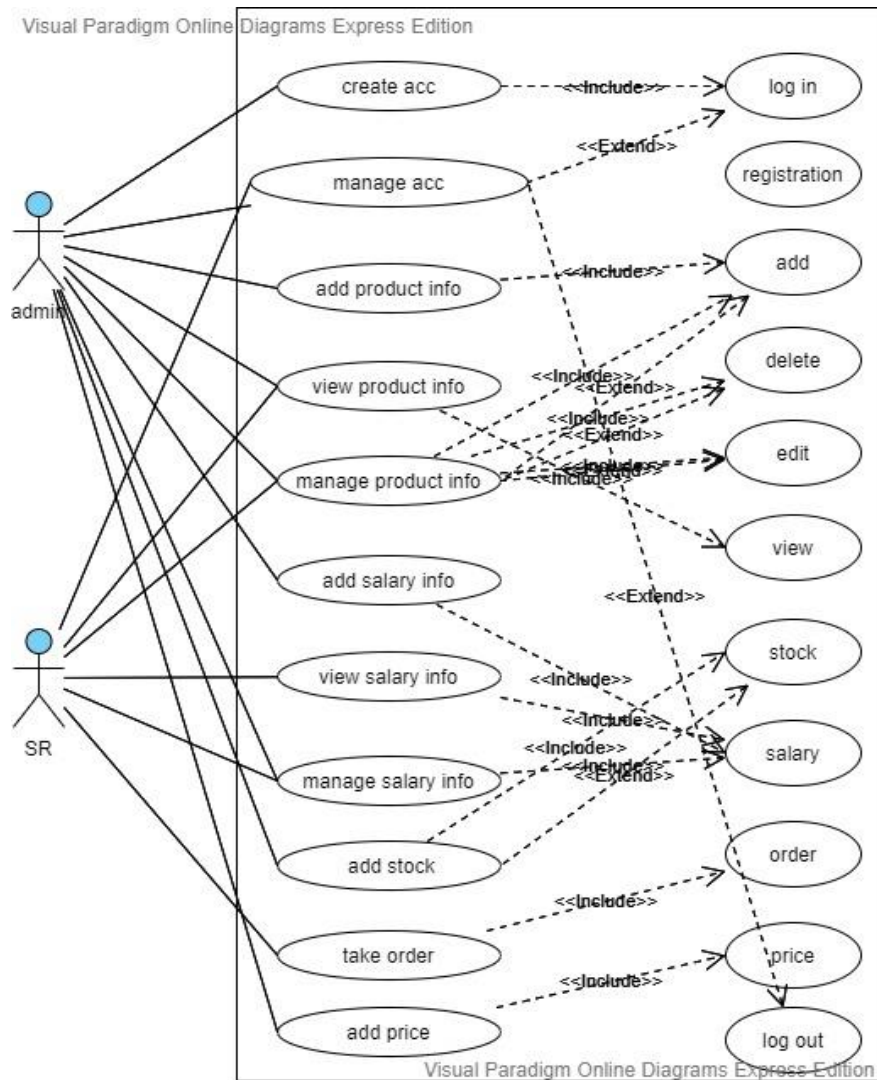


Figure 3.1: Use Case diagram for “Smart Hostel Finder”

### 3.1.1 Create Account.

<b>Use Case</b>	<b>Create Account.</b>	
<b>Goal</b>	Admin and SR can log in to the system if they don't have account, they have to choose create account option. There is two options for create account both client and manager.	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>Admin and SR must open the system.</li> <li>Admin and SR must go to the create account option.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>Admin and SR can open the system successfully.</li> <li>Admin and SR can create account successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>T&amp;A and planning admin can't login successfully.</li> <li>T&amp;A and planning admin can't logout successfully.</li> </ul>	
<b>Primary Actors:</b>	Admin and SR	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin and SR must sign in to the system for create account.	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin and SR clicks to the sign in option
	2	Admin and SR gets login form.
	3	Admin and SR enters right username password and other requirements for create account
	4	Admin and SR finally clicks create button.
	5	Admin and SR successfully create account.
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin and SR doesn't sign in system
	2	Admin and SR doesn't get create account form.
	3	Admin and SR can't give right username password

### 3.1.2 Manage Account.

<b>Use Case</b>	<b>Manage Account.</b>	
<b>Goal</b>	Admin and SR can log in to the system if they have account, they have to choose sign in option. There is one options for login account both Admin and SR	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin and SR must have user name and password.</li> <li>• Admin and SR must go to the log in account option.</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Admin and SR can open log in option in system successfully.</li> <li>• Admin and SR can login and log out successfully.</li> </ul>	
<b>Failed End Condition</b>	Admin and SR can't login successfully. Admin and SR can't logout successfully.	
<b>Primary Actors:</b>	Admin and SR	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin and SR must have a valid account for log in to the system.	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin and SR clicks to the login option
	2	Admin and SR gets login form.
	3	Admin and SR enters right username password for log in
	4	Admin and SR finally clicks sign in button.
	5	Admin and SR successfully logged in system.
	6	Admin and SR clicks on button and successfully log in and log out from the system
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin and SR doesn't logged in the system
	2	Admin and SR cant login with their account
	3	Admin and SR can't log out from the system

### 3.1.3 Product Add Info.

<b>Use Case</b>	<b>Product add info.</b>	
<b>Goal</b>	Admin can log in the system and add his product information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin and SR must have an valid account for this</li> <li>• admin must go to the add product info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• admin can add his product information in the system successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>• Admin and SR can't add product information successfully.</li> </ul>	
<b>Primary Actors:</b>	Admin	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin must have a valid account for add hostel info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin and SR clicks to the login option
	2	Admin and SR gets login form.
	3	Admin and SR enters right username password for log in
	4	Admin and SR finally clicks sign in button.
	5	Admin and SR successfully logged in system.
	6	Admin and SR clicks on button and successfully log in and log out from the system
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin doesn't add hostel info in the system
	2	Admin cant login with his account
	3	Admin can't log out from the system

### 3.1.4 View product Info.

<b>Use Case</b>	<b>View product info.</b>	
<b>Goal</b>	SR can view the hostel information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin and SR must have an valid account for this</li> <li>• SR must go to the view product info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• SR can see the all product information from the system successfully.</li> </ul>	
<b>Failed End Condition</b>	SR cant view hostel information	
<b>Primary Actors:</b>	SR	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	SR must have a valid account and sign in for view hostel info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	SR have to clicks on the view product info option
	2	SR got hostel info form.
	3	SR enters right username password for log in
	4	SR clicks on view hostel info button and successfully see the info
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	SR doesn't see product info in the system
	2	SR cant login with his account
	3	SR can't log out from the system

### 3.1.5 Manage product Info.

<b>Use Case</b>	<b>Manage product info.</b>	
<b>Goal</b>	Admin can manage his product information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin must have an valid account for this</li> <li>• Admin must go to the manage product info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Manager can add his product information successfully and updated his product info.</li> </ul>	
<b>Failed End Condition</b>	Admin can't manage his product information	
<b>Primary Actors:</b>	Admin	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin must have a valid account and sign in for manage product info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin have to clicks on the manage product info option
	2	Admin got manage product info option
	3	Admin inserts his product information
	4	Admin clicks on add product info button and successfully inserts the info and updated
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin doesn't see add product info in the system
	2	Admin cant Update his product information
	3	Admin cant delete his product information

### 3.1.6 Add salary Info.

<b>Use Case</b>	<b>Add salary info.</b>	
<b>Goal</b>	Admin can log in the system and add his salary information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin must have an valid account for this</li> <li>• Admin must go to the add salary info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Admin can add his salary information in the system successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>• Admin can't add salary information successfully.</li> </ul>	
<b>Primary Actors:</b>	Admin	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin must have a valid account for add salary info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks to the login option
	2	Admin manager click the add salary info option
	3	Admin clicks on button and successfully add his salary information
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin doesn't add salary info in the system
	2	Admin cant login with his account
	3	Admin can't log out from the system



### 3.1.7 View salary Info

<b>Use Case</b>	<b>View salary info.</b>	
<b>Goal</b>	SR can view the salary information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• SR must have an valid account for this</li> <li>• SR must go to the view salary info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• SR can see the all salary information from the system successfully.</li> </ul>	
<b>Failed End Condition</b>	SR cant view salary information	
<b>Primary Actors:</b>	SR	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	SR must have a valid account and sign in for view salary info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	SR have to clicks on the view salary info option
	2	SR see the salary information
	3	SR enters right username password for log in
	4	SR clicks on view salary info button and successfully see the info
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	SR doesn't see salary info in the system
	2	SR cant login with his account
	3	SR can't log out from the system

### 3.1.8 Manage salary Info.

<b>Use Case</b>	<b>Manage Room info.</b>	
<b>Goal</b>	Admin can manage his salary information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin must have an valid account for this</li> <li>• Admin must go to the manage salary info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Admin can add his salary information successfully and updated his salary info.</li> </ul>	
<b>Failed End Condition</b>	Admin cant manage his salary information	
<b>Primary Actors:</b>	Admin	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin must have a valid account and sign in for manage salary info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin have to clicks on the manage salary info option
	2	Admin got manage salary info option
	3	Admin inserts his salary information
	4	Admin clicks on add salary info button and successfully inserts the info and updated
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin doesn't see add salary info in the system
	2	Admin cant Update his salary information
	3	Admin cant delete his salary information

### 3.1.9 Add stock Info.

<b>Use Case</b>	<b>Add stock info.</b>	
<b>Goal</b>	admin can log in the system and add his product stock information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin must have an valid account for this</li> <li>• Admin must go to the add stock info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• admin can add his stock information in the system successfully.</li> </ul>	
<b>Failed End Condition</b>	<ul style="list-style-type: none"> <li>• Admin can't add stock information successfully.</li> </ul>	
<b>Primary Actors:</b>	Admin	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin must have a valid account for add stock info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin clicks to the login option
	2	Admin click the add stock info option
	3	Admin clicks on button and successfully add his stock information
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin doesn't add stock info in the system
	2	Admin cant login with his account
	3	Admin can't log out from the system

### 3.1.10 Take order Info

<b>Use Case</b>	<b>Take order info.</b>	
<b>Goal</b>	Admin and SR can take order information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin and SR must have an valid account for this</li> <li>• SR go to the order info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• SR can see the all order information from the system successfully.</li> </ul>	
<b>Failed End Condition</b>	Admin and SR cant view order information	
<b>Primary Actors:</b>	Admin and SR	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin and SR must have a valid account and sign in for view order info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	SR have to clicks on the view order info option
	2	Admin and SR see the order information
	3	Admin and SR enters right username password for log in
	4	Admin and SR clicks on order info button and successfully see the info
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin and SR doesn't see order info in the system
	2	Admin and SR cant login with his account
	3	Admin and SR can't log out from the system

### 3.1.11 add product price Info.

<b>Use Case</b>	<b>add price info.</b>	
<b>Goal</b>	admin can manage his product price information	
<b>Preconditions</b>	<ul style="list-style-type: none"> <li>• Admin must have an valid account for this</li> <li>• Admin must go to the manage price info option</li> </ul>	
<b>Success End Condition</b>	<ul style="list-style-type: none"> <li>• Admin can add his product price information successfully and updated his product price info</li> </ul>	
<b>Failed End Condition</b>	Admin cant manage his product price information	
<b>Primary Actors:</b>	Admin	
<b>Secondary Actors:</b>	System	
<b>Trigger</b>	Admin must have a valid account and sign in for add price info	
<b>Description / Main Success Scenario</b>	<b>Step</b>	<b>Action</b>
	1	Admin have to clicks on the manage product price info option
	2	Admin got manage price info option
	3	Admin inserts his price information
	4	Admin clicks on add price info button and successfully inserts the info and updated
<b>Alternative Flows</b>	<b>Step</b>	<b>Branching Action</b>
	1	Admin doesn't see add price info in the system
	2	Admin cant Update his price information
	3	Admin cant delete his price information

## 3.2 Activity Diagram

I have prepared some activity diagram according to my use case. These activity diagrams are properly referring the flow of the individual conditions of my project.

### 3.2.1 Create account

If user have account then direct going to login, otherwise click on create account option for account.

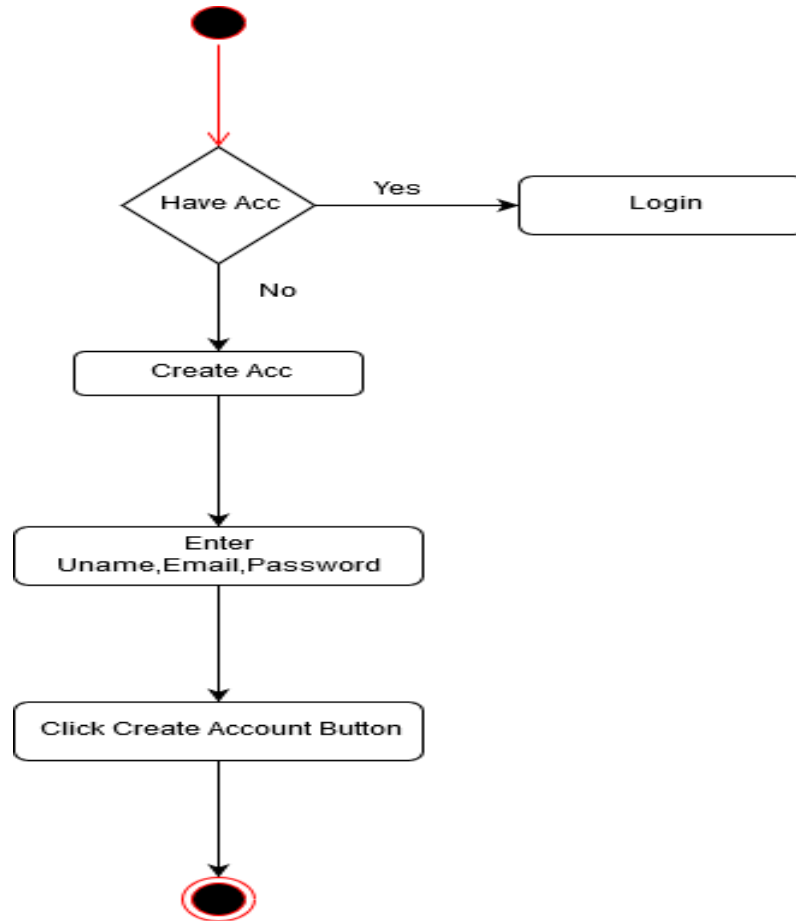


Figure 3.2.1: Activity diagram for Create Account

### 3.2.2 Manage account

Users already have an account then click on sign in option and click on enter username and password further works.

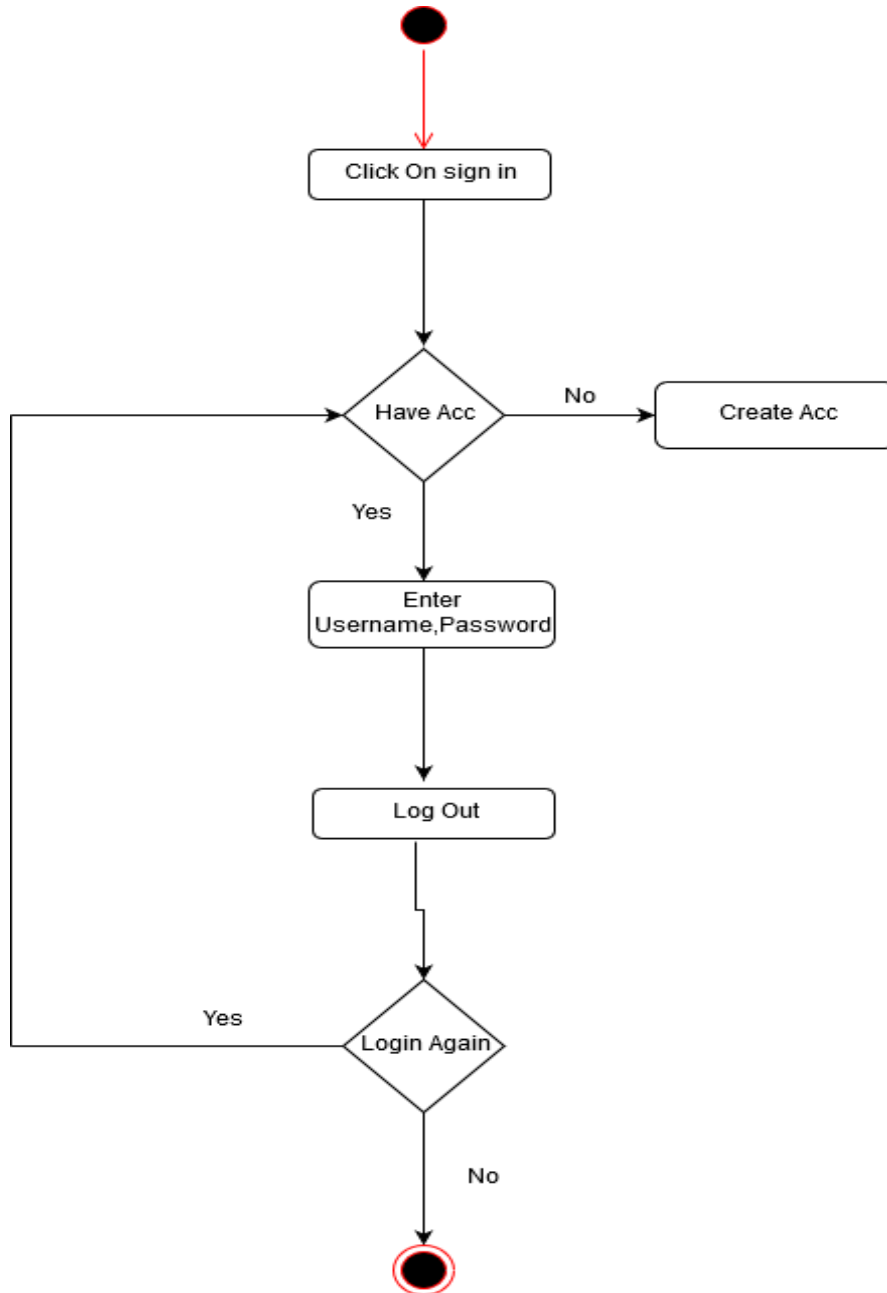


Figure 3.2.2: Activity diagram for Manage account

### 3.2.3 Add product Info

Manager can add his product information b click on add product info and others option.

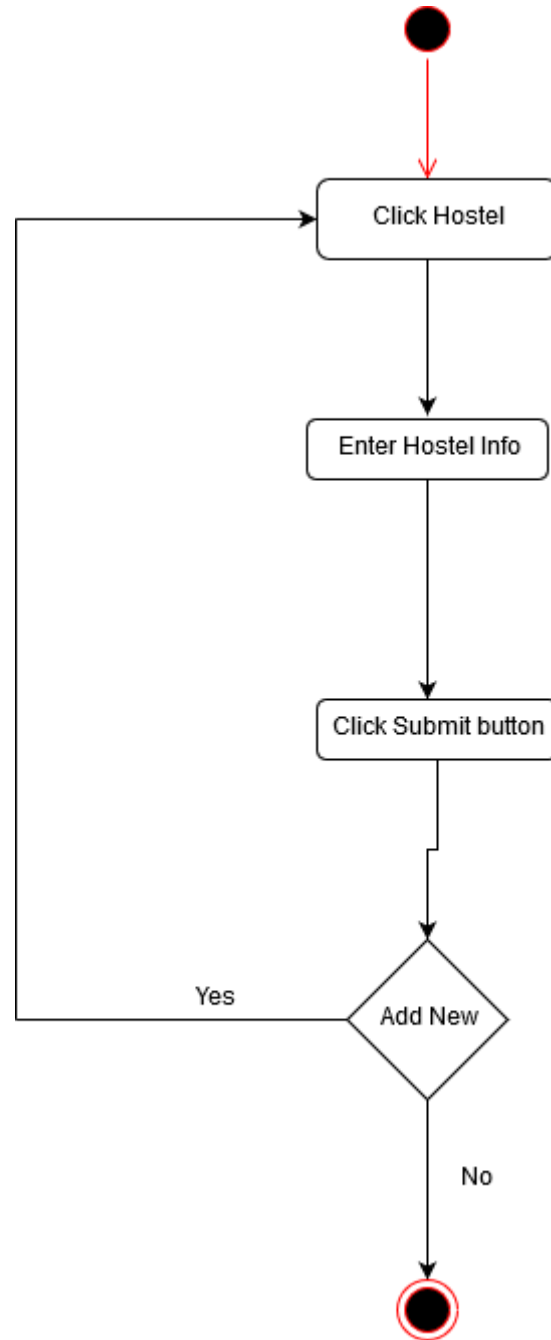


Figure 3.2.3: Activity diagram for Add product Info



### 3.2.4 View product Info

Users can view product information , if they have already registered accounts.

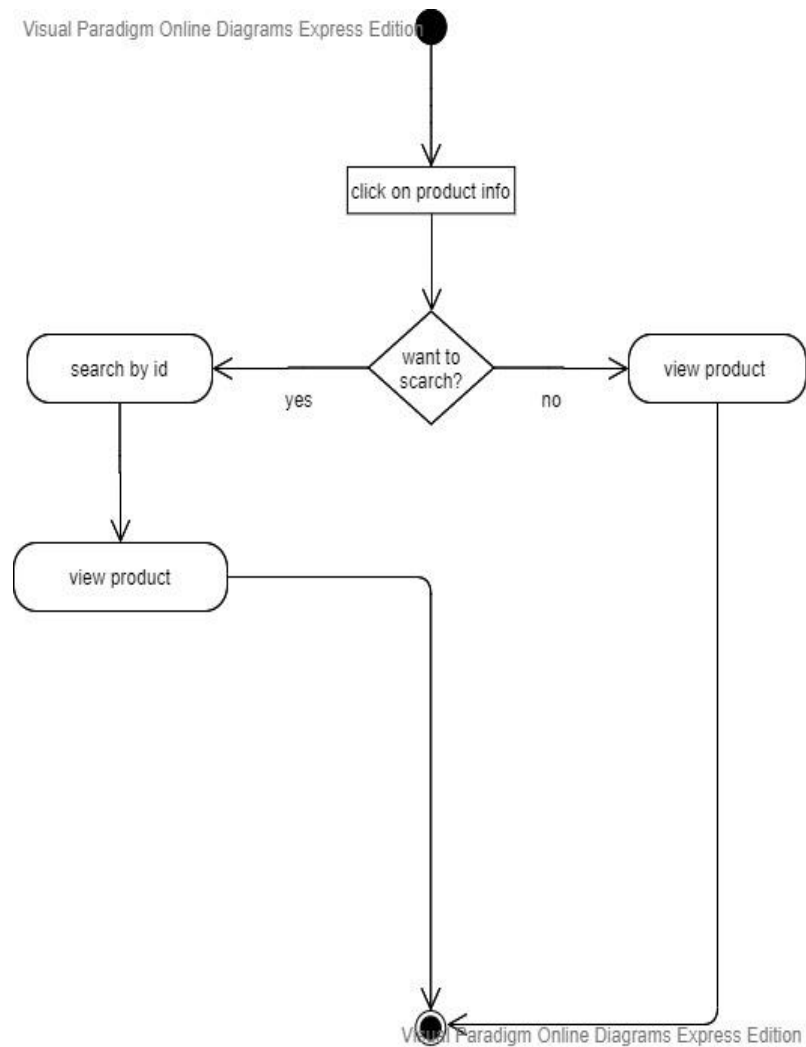


Figure 3.2.4: Activity diagram for view product

### 3.2.5 Manage Hostel Info

Hostel manager can update his uploaded hostel information to click on hostel.

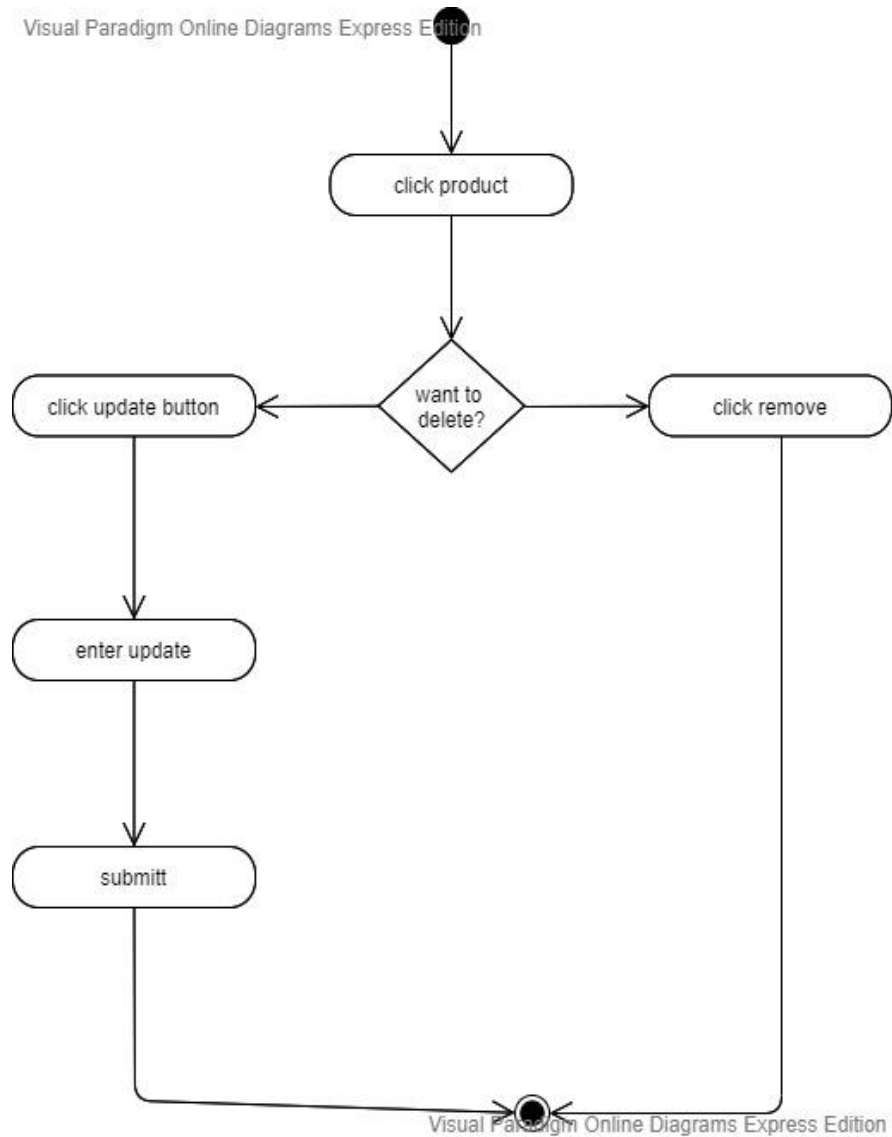


Figure 3.2.5: Activity diagram for Manage Hostel Info

### 3.2.6 Add salary Info

admin can add his salary information.

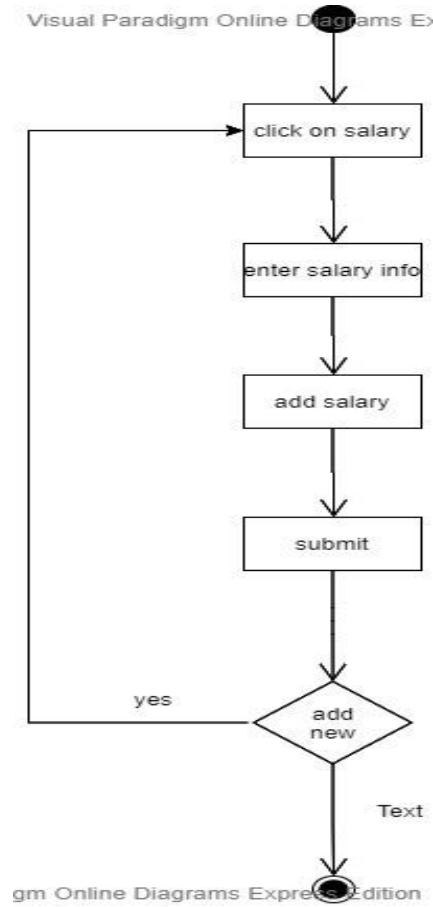


Figure 3.2.6: Activity diagram for Add salary

### 3.2.7 View salary Info

Users can views salary information.

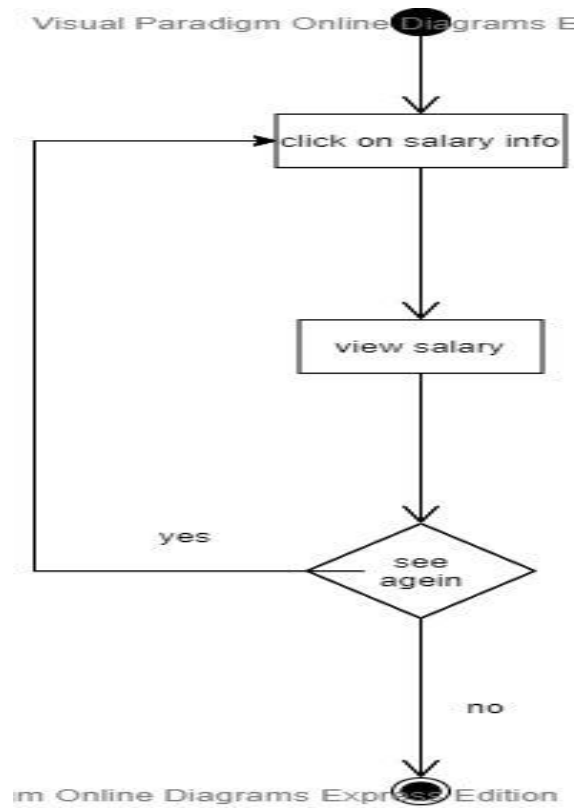


Figure 3.2.7: Activity diagram for view salary

### 3.2.8 Manage salary Info

admin can add and update his salary information.

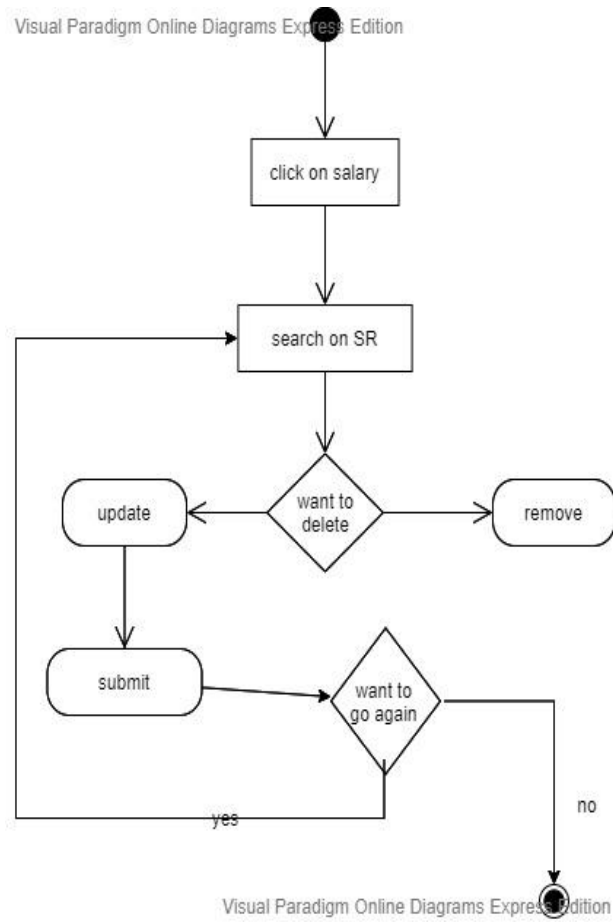


Figure 3.2.8: Activity diagram for Manage salary Info

### 3.2.9 Add stock Info

admin can upload stock info for available system.

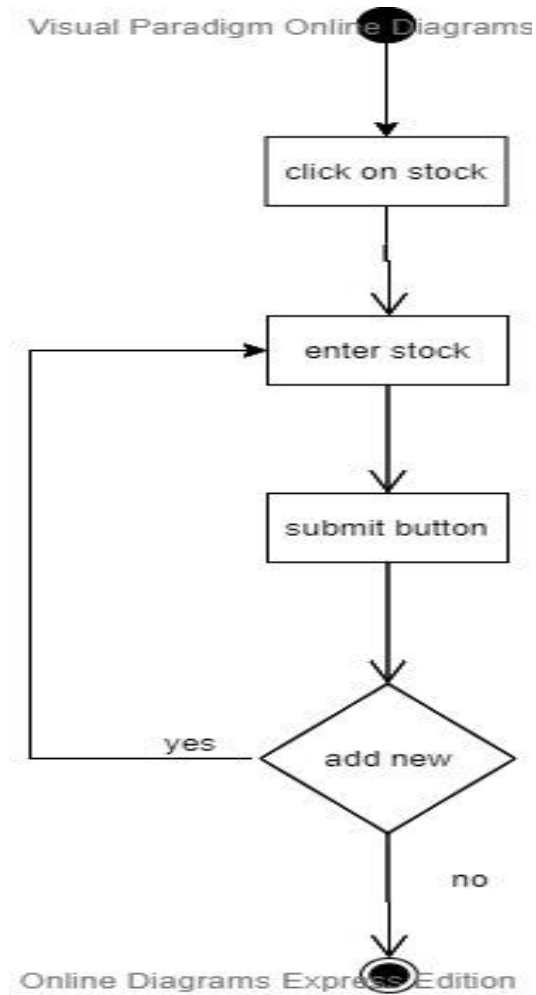


Figure 3.2.9: Activity diagram for Add stock Info

### 3.2.10 Add price Info

Admin can add all type of price information.

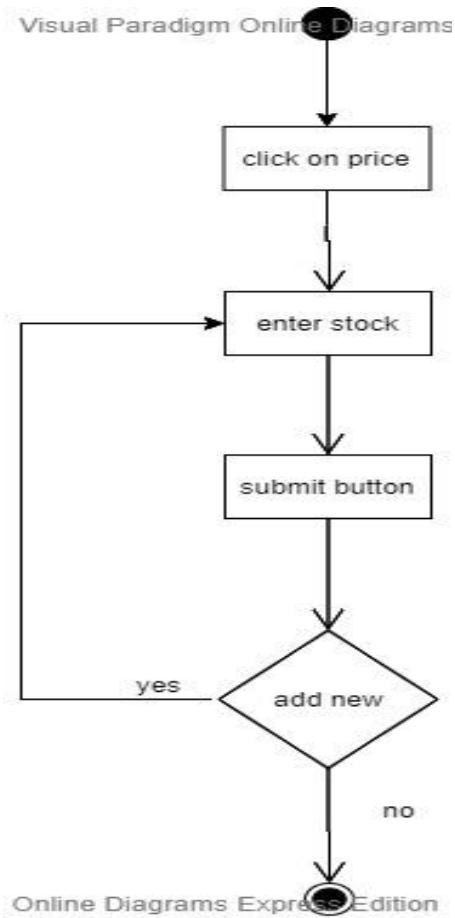


Figure 3.2.10: Activity diagram for add price

### 3.2.11 Manage order Info

Admin and SR can manage his order info.

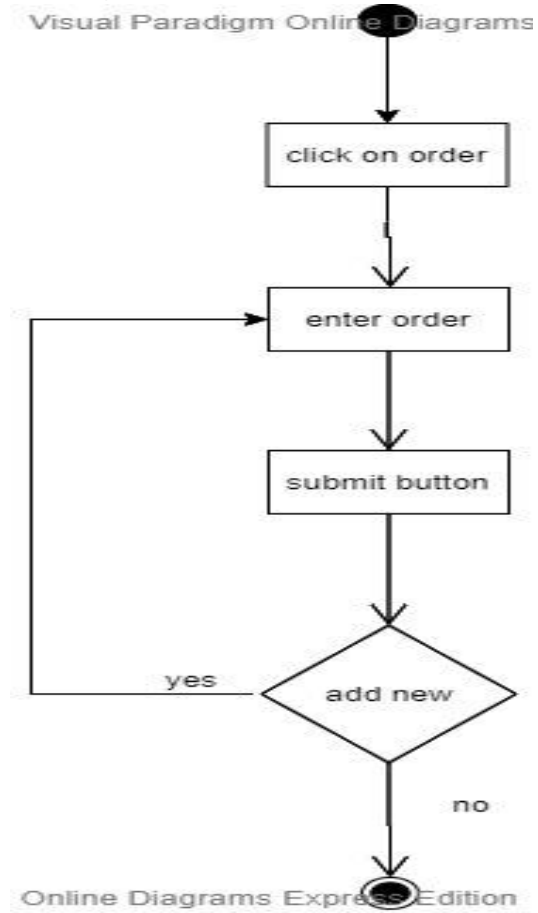


Figure 3.2.11: Activity diagram of order



### **3.3 System Sequence Diagram**

Mainly sequence diagrams understand us how the data will be followed in any application. Now I am going to show some sequence diagrams.

### 3.3.1 .Create Account

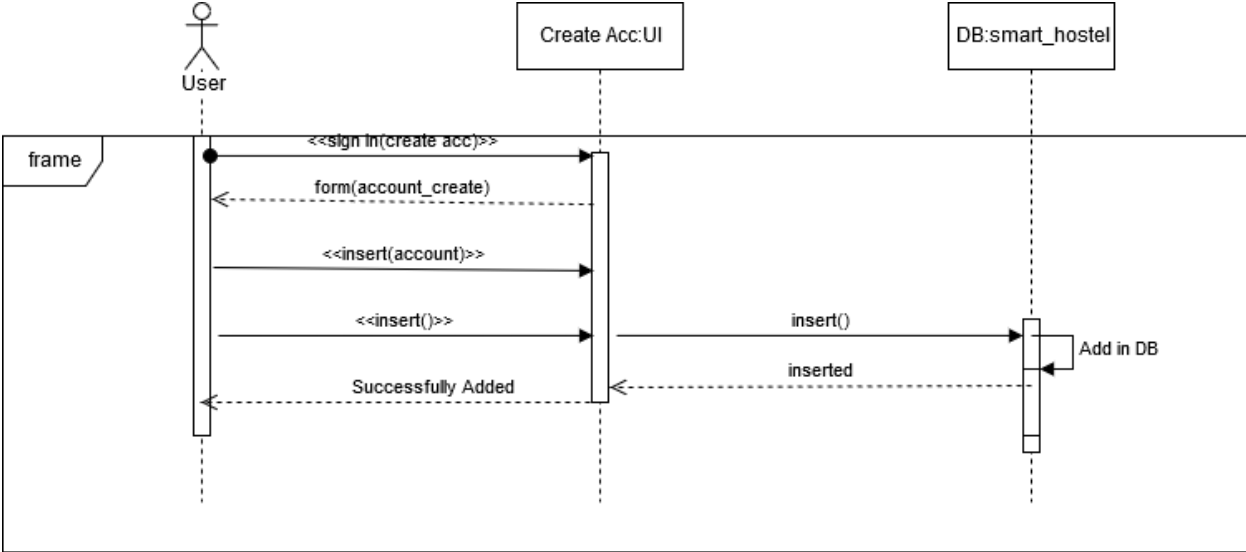


Figure 3.14: Sequence Diagram for Create account

### 3.1.1 .Manage Account

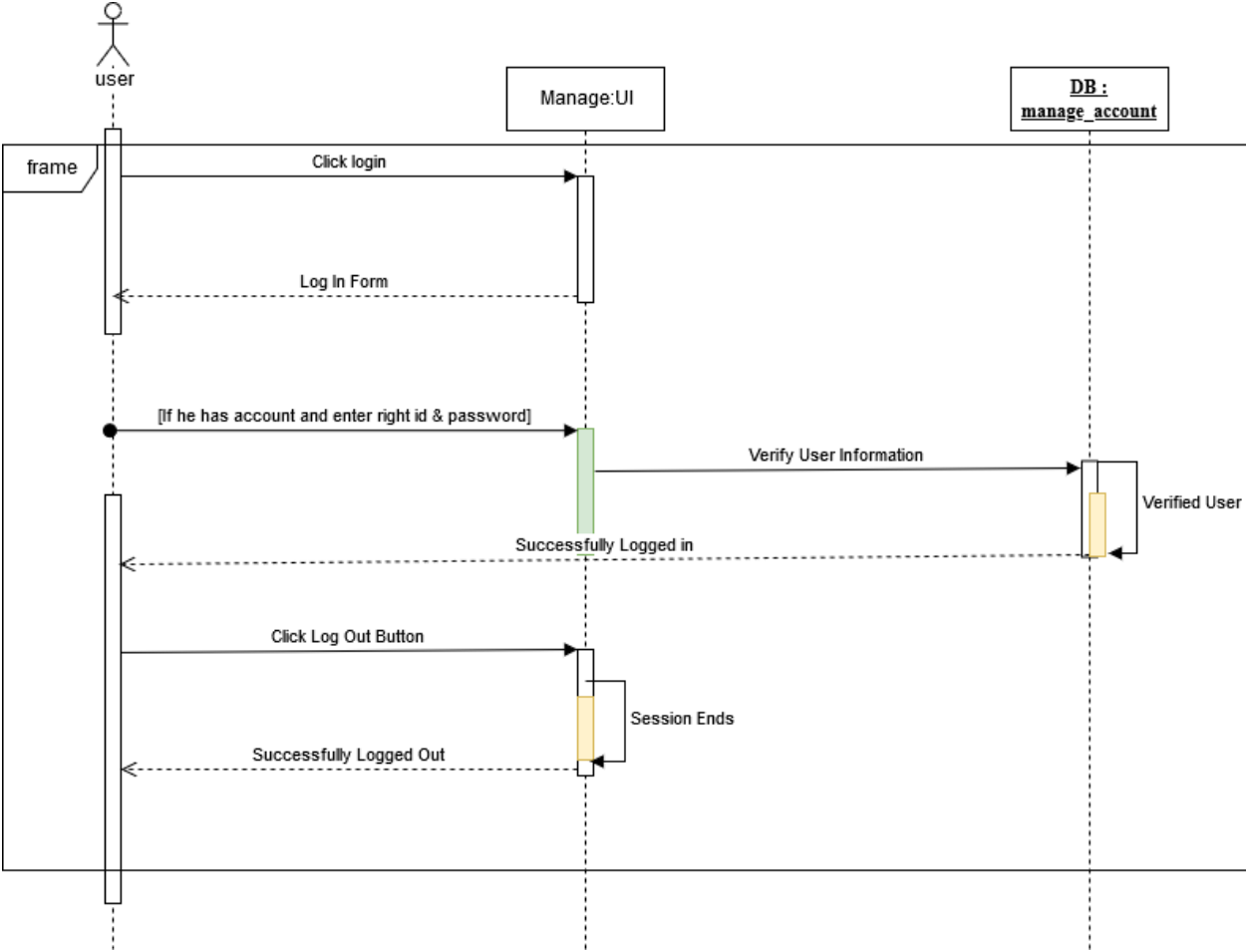


Figure 3.15: Sequence Diagram for Manage Account

### 3.1.2 Sequence Diagram for admin

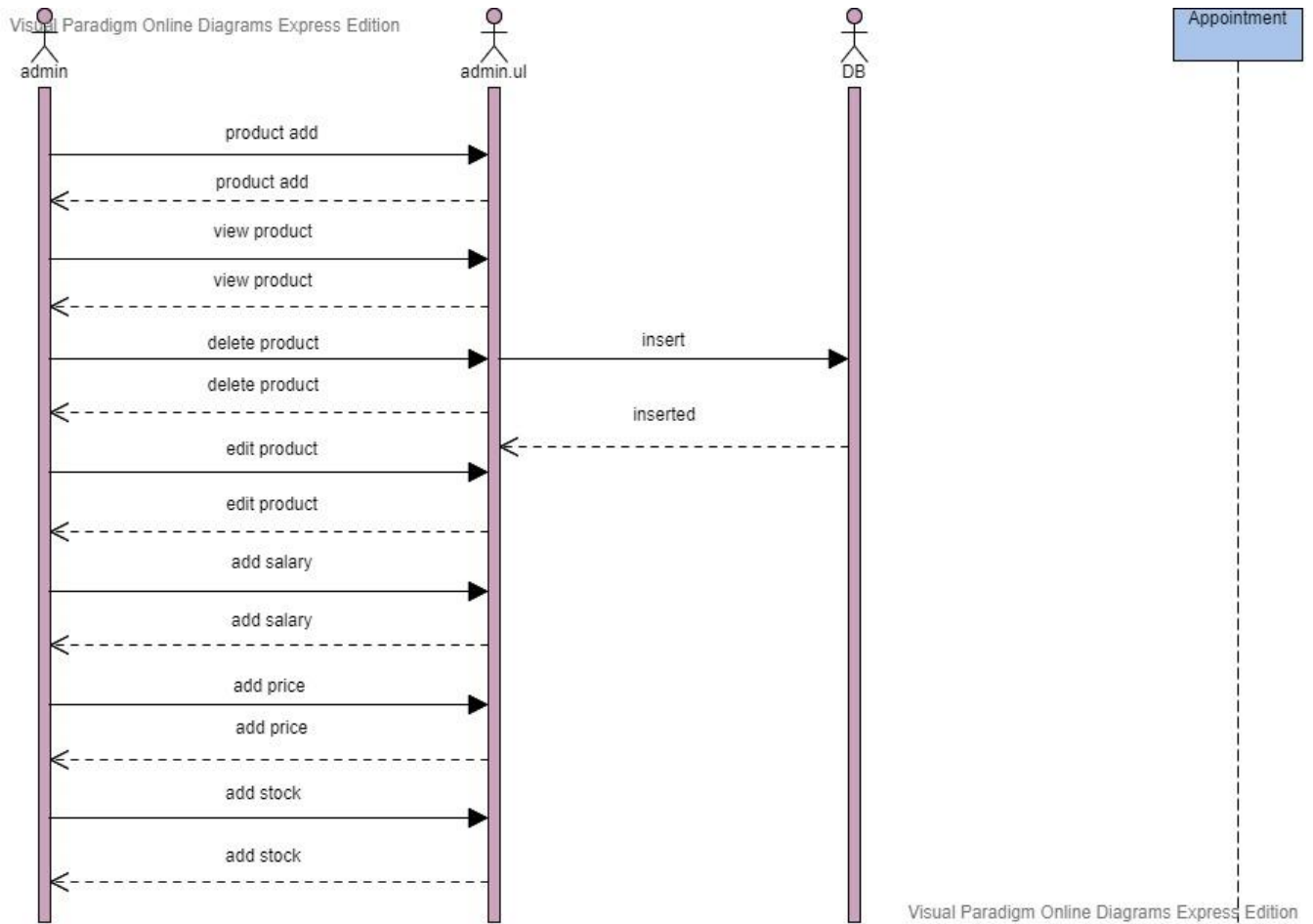


Figure 3.15: Sequence Diagram for admin

### 3.3.3 Sequence Diagram for SR

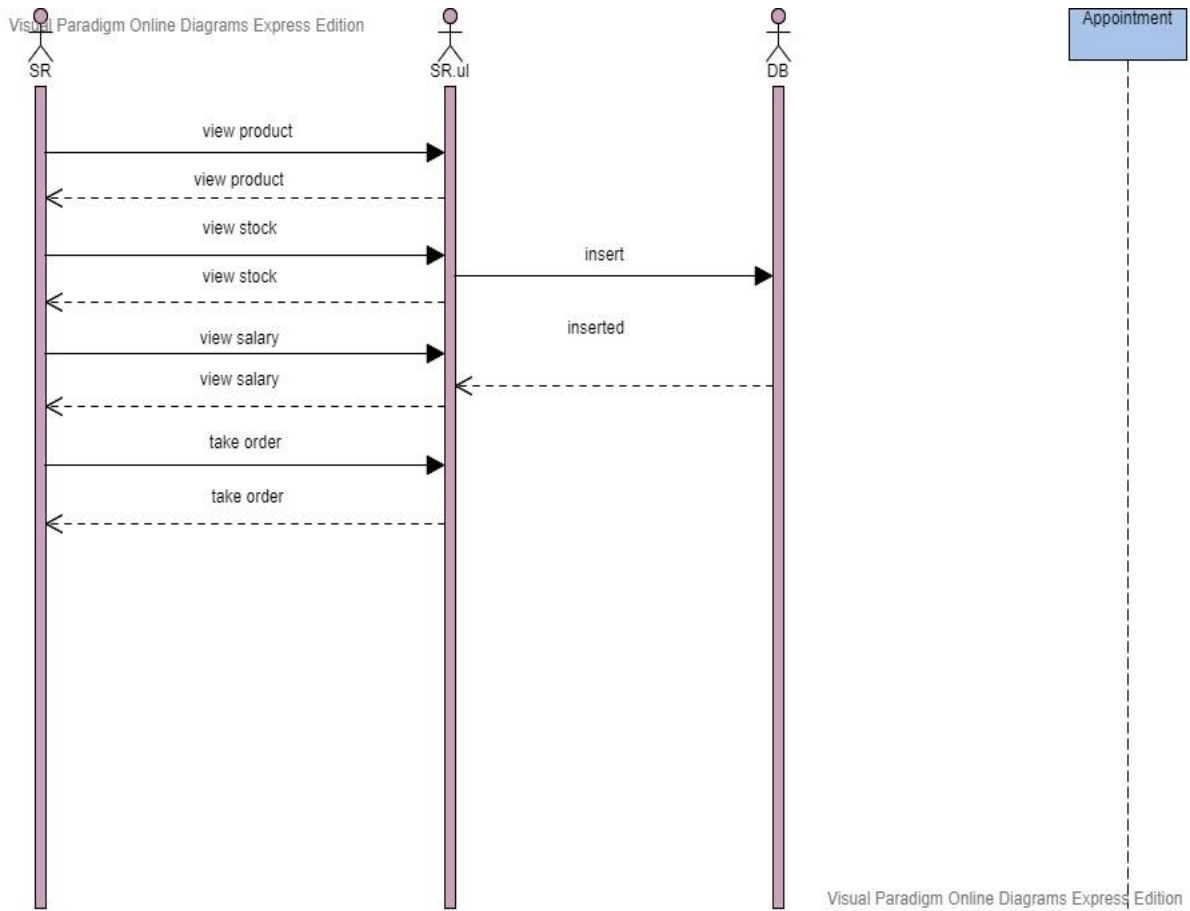


Figure 3.15: Sequence Diagram for sr

# **Chapter 4**

## **System Design Specification**

## 4.1 Development tools and technology

Software development tools are used to develop software. There are different kinds of tools for software developers to make the process of software development very smoothly. But now I will mention some of tools that I am using to develop my project

## 4.2 Class Diagram

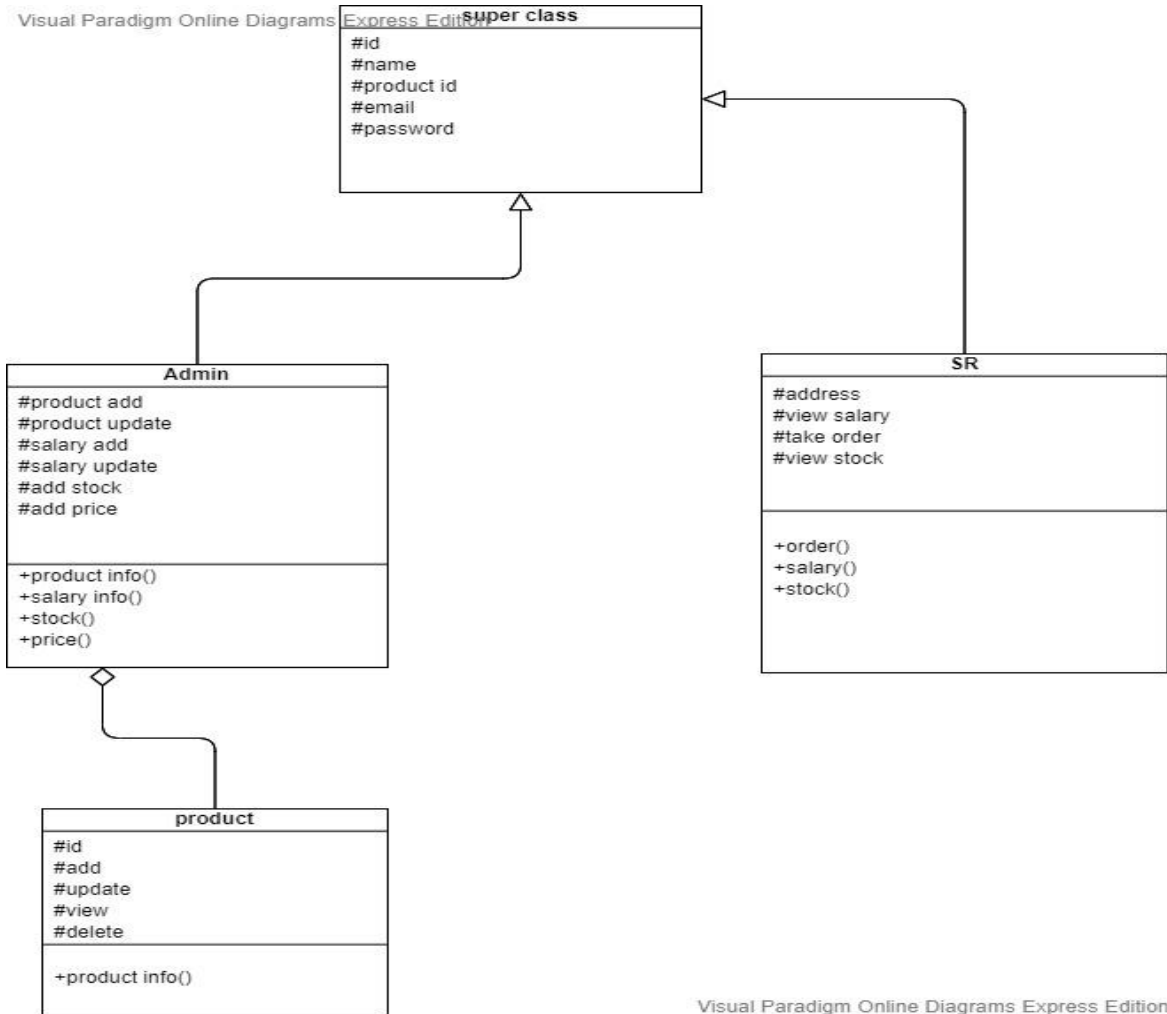


Figure 4.1: Class diagram(SR online delivery system )

### 4.4.1 User Interface Technology

First of all, after landing any application, interface of that software is appeared to the user. So, the importance of user interface is very high. For success of any software application, a good looking user interface plays a vital role. User interface includes using good image, graphics, typography, style sheet scripting etc.

#### 4.4.1.1 Programming Language

For developing any application system minimum one programming language is essential. In my application, there are two different programming language is used. One is for front-end side. And another is for server side. The front-end language that I have used to my application is HTML. And the server side programming language name is PHP which stands for Hypertext Preprocessor. Both of them are open source general purpose scripting language.

#### **4.4.1.2 JQuery UI**

JQuery UI is a name of library which uses JavaScript as core programming language. It simplifies codes of a programming language named JavaScript. Generally, it interacts with Graphical User Interface (GUI). It also provides visual effects with animation. It can also override cascading style sheets. It is also a subset of JavaScript. By using AJAX no page reloading is required. It is also compatible with any browser like Google Chrome, Mozilla Firefox, Opera, Safari, Internet Explorer etc.

#### **4.4.1.3 CSS framework or Bootstrap**

After completing markup, cascading style sheets are designed. It generally explains how HTML elements will display. There are three ways to write style sheet. One internal CSS, another is external CSS. And last one is inline CSS. But most of time external CSS is used. Because, by using external CSS, all CSS data can be kept in different files. But nowadays CSS framework seems to be very popular. In our application, I have used Bootstrap4. This framework is developed mainly for CSS and JavaScript. It is a free and open source front-end framework. This framework is highly used for designing website and web application also. This framework supports almost every browser. Like Google Chrome, Mozilla Firefox, Opera, Safari, Internet Explorer etc. Bootstrap also provides media query which features responsive layout for different devices with different screen size. Bootstrap provides a set of some files which contains style sheets which gives basic definition. Bootstrap also provides some JavaScript components also. There are some built in components like JQuery UI. By using Bootstrap framework, we get both CSS and JavaScript facilities with a single platform. But before start designing an application interface with Bootstrap, one may have some basic knowledge about this framework. It will increase the efficiency.



## **4.4.2 Implemented tools and platform**

As I have said before, there are some tools and technologies that need to be used for developing software. It is very important to determine which tools and platforms are the best match of my requirements. After making a proper decision, one need to start using them.

### **4.4.2.1 Integrated Development Environment**

IDE stands for Integrated Development Environment. Programmers write code on IDE. After that IDE provide the feature to execute the source code. For developing my web application, I have used an IDE. To develop my web version, I have used SublimeText which is powered by Sublime HQ Company. It is a commercial IDE for cross platform environment. It is able to suggest code to the programmers also.

### **4.4.2.2 Database Server (MYSQL Server V-10.1.36-MariaDB)**

For developing our whole project, I have followed Relational Database Management System or RDBMS. And I find that MySql provides the feature of RDBMS. So I should not have any issue to use MySql database. It is also very easy to use. It can also ensure the security, scalability, high performance and many things.

# **Chapter 5**

## **System Test**

## **5.1 Testing Features**

Feature testing can be considered as making change to add or modify the new functionalities to the existing project. To test the features and functionality, a new test set is to be written for testing purpose. Almost every feature and functionality has different characteristics. Those are designed to make the application more useful, intuitive, reliable, secured, scalable, effective and efficient.

## 5.2 Testing Strategy

Testing strategy is to be considered as a general purpose of testing process. Testing object, testing function methods, total available resources are included to the testing strategy. It is also the indicator of test levels that are to be performed on the whole software development life cycle. Those strategies are prepared by the quality assurance team that should be reviewed by the developers of the application. After that it should be reviewed by the test team leads. Different kinds of testing strategies can be performed according to the type of application system need to be tested.

### 5.2.1 Test approach

To complete the test process, testers must take some approach. There are mainly two type of test approach.

- **Automation testing:** Automation testing is a name of testing technique by which test engineers prepare some scripts according to test plan and after that they use suitable tools to perform testing of the software. Nowadays, almost every software company follow the approach of automation testing.
- **Manual testing:** Manual testing is also a name of technique of testing by searching out the bugs or vulnerability in an application. In this process, test engineers manually test and execute the test cases without having any automation tools.

#### 5.2.1.1 Black Box Testing

Black box testing is a software testing method by which test engineers test the software without having knowledge of the internal architecture of the application that need to be tested. It is also known as behavioral testing. Black box testing can be either functional or non-functional. It ignores the internal mechanism of a system. We have decided to perform the equivalent class partitioning and Boundary value analysis techniques to implement.

#### 5.2.1.2 Equivalent Class Partitioning

Equivalent class partitioning is also known as equivalent partitioning. In this technique input data is to be divided into groups. Those groups of data are expected to perform as similar behavior. Each group works as same as other groups also.

The main advantage of following the approach of equivalent class partitioning is to reduce the total number of test cases from infinite to finite. Another advantage is that it can be applied to all testing levels also.

For legal input value output will produce a meaning data. But for illegal input value, output will not produce meaningful data.

### **5.2.1.3 Boundary Value Analysis**

Boundary value analysis is a name of software testing by which input data can be included within a boundary range. Following this strategy, test engineers firstly define the boundary value. After that they take a specific boundary and then write the test cases that will be go through the application for testing purpose. After completing those tasks, testers use test cases to test applications software.

### **5.2.1.4 White Box Testing**

White box testing is also a name of testing approach which is also known as clear box testing, glass box testing, open box testing, transparent box testing, code based testing or structural testing. It is opposite to the black box testing. In black box testing, the internal architecture or algorithm is not known to the testers whether the whole software architecture is known to the testers while white box testing.

Testers can also predict the output of every test cases for white box testing. White box testing can be classified into some levels. Such as:

- Unit Testing
- Integration Testing
- System Testing

The main advantage of white box testing is that testing is more throughout and the testing can be started from the very beginning stage.

### **5.2.2 Pass / Fail Criteria**

Pass or fail criteria will be set by the test engineers. They will prepare the pass / fail criteria on the basis of which input data are worked and which are not works well. Those data that are worked well will considered as pass criteria. And rest of input data will be considered as fail criteria.

Now I will give the pass / fail criteria below.

- System crash will not be considered as pass case.
- If any criteria pass 100% times, then it will be considered as pass criteria only.
- If data can't be displayed to the application properly, then it is also to be considered as fail criteria.

### 5.2.3 Testing Schedule

<b>Test Phase</b>	<b>Time</b>
Testing plan create	1 week
Test specification	2 week
Unit testing	During development time
Component test	1 week
<b>Test Phase</b>	<b>Time</b>
Integration testing	1 week
Validating use cases	1 week
Testing user interfaces	1 week
Load testing	1 week
Performance testing	1 week
Release to production	1 week

### 5.2.4 Trace Ability Matrix

<b>Project admin</b>			<b>Business Analyst Lead</b>		
<b>QA Lead</b>			<b>Target Implementation Date</b>		
<b>BR#</b>	<b>Category / Functionality / Activity</b>	<b>Requirement Description</b>	<b>Use Case Reference</b>	<b>Test Case Reference</b>	<b>Comments</b>
Br-1	Functional	Login	Use case 3.1.2	Test case 5.6.1	
BR-2	Functional	Log out	Use case 3.1.2	Test case 5.6.3	
BR-3	Functional	Search product	Use case 3.1.4	Test case 5.6.4	
BR-4	Functional	Insert product info	Use case 3.1.3	Test case 5.6.5	

### **5.3 Testing Environment**

Testing environment means to prepare the environment with hardware and software so that test engineers can be able to execute test cases as required. Besides hardware and software usage, network configuration might be needed to execute test plans.

For making the environment for testing, some key area need to setup. Those are:

- Test data
- Database server
- Client's operating system
- Front end running environment
- Browser
- System and application
- Network
- Hardware with server operating system
- Documentation is also required. Like: user manuals, installation guides, configuration guides, documents etc.

### **5.4 Test Cases**

A test case refers to some rules and regulations or conditions by which it can be determined whether a system can be able to meet the works or requirements under test cases properly. We know, there is a chance to have some fault or break holes in any application. This is very common scenario. And those issues are solved usually by software testing approaches. But if we don't care of those issues, then the full application development might be ruined. So proper testing must be needed. For testing our application, I have prepared some test cases. Now, I am going to provide them.

## 5.4.1 Log In

<b>Test case #1</b>		<b>Test case name:</b> Log in			
<b>System:</b> SR online delivery system		<b>Subsystem:</b> N/A			
<b>Designed by:</b> tanvir		<b>Designed date:</b> 2/1/20			
<b>Executed by:</b>		<b>Executed date:</b>			
<b>Short description:</b> The registered users need to login to the application. And before that our application will check the authentication and authorization.					
<b>Pre-conditions:</b>					
<ul style="list-style-type: none"> <li>• Users are always redirected to the login page whether they have not authenticated by our application while entering into the dashboard.</li> <li>• Assume that, the Username is 'sa' and password is 'sa@12'</li> </ul>					
<b>Step</b>	<b>Username</b>	<b>Password</b>	<b>Expected result</b>	<b>Pass/Fail</b>	<b>Comment</b>
1	tanvir	1234	Invalid		
2	tanvir		Password can't be blank		
3		000000	Username can't be blank		
4	Admin	Root	Successfully login to the application and redirected to the dashboard		
<b>Post-conditions:</b> Users including sr online delivery system will successfully login to the application.					



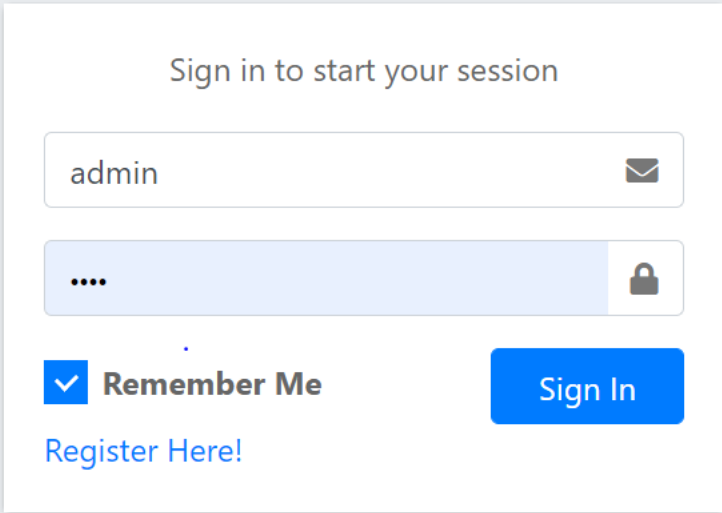
# **Chapter 6**

## **Requirement Analysis**

## 6.1 User Interface

First of all, after landing any application, interface of that software is appeared to the user. So, the importance of user interface is very high. For success of any software application, a good looking user interface plays a vital role.

### 6.1.1 This is my log in page



LOGIN

Sign in to start your session

admin

...

Remember Me


Sign In


[Register Here!](#)


## 6.1.2 Registration for SR


### REGISTER


Register a new membership


Select ▼ 

Enter Name 

Enter Email Address 

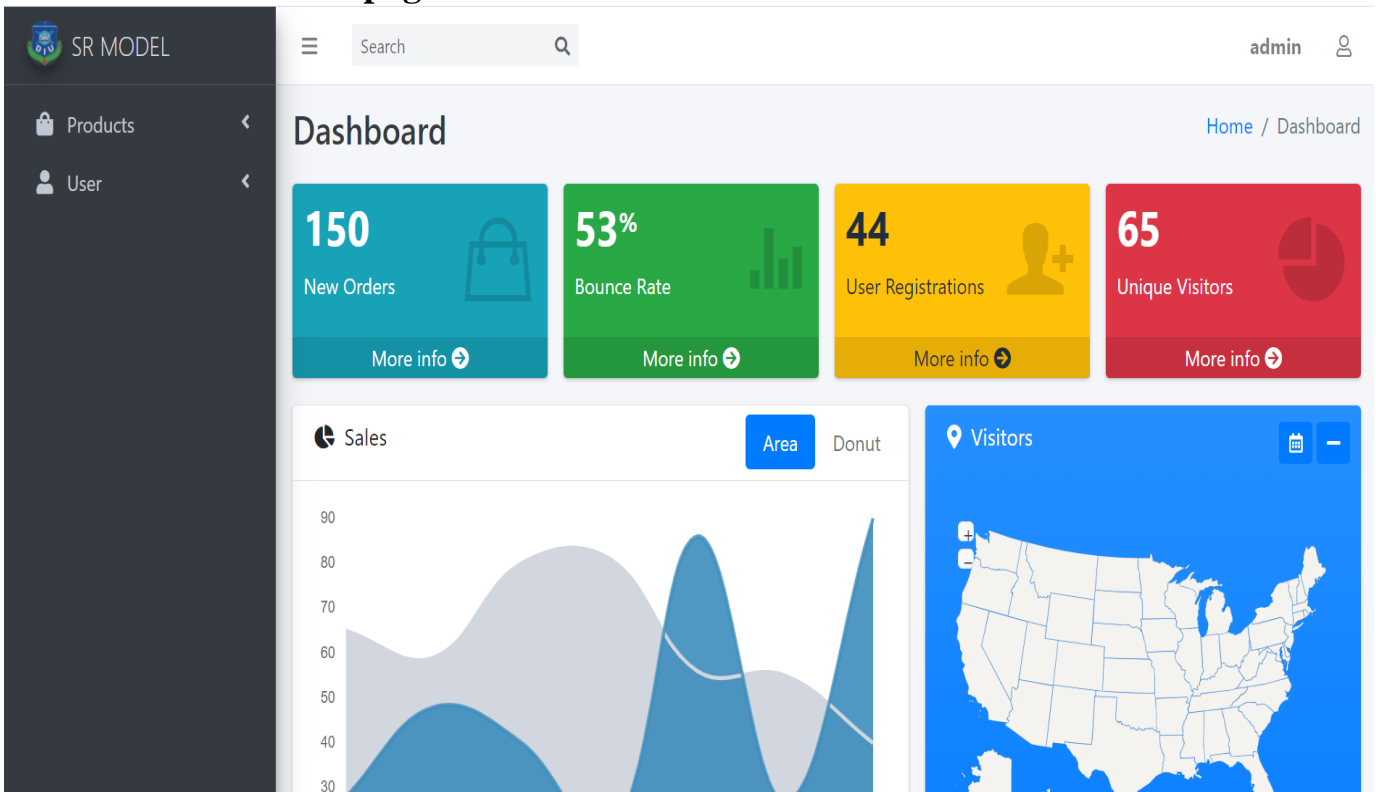
admin 

.... 

Confirm Password 

[I already have a membership](#) Register

## 6.1.3 Admin home page



## 6.1.4 Pending request for admin page

The screenshot shows the 'Pending User Request' page for an admin user. The page features a dark sidebar with 'SR MODEL', 'Products', and 'User' options. The main content area has a search bar and a user profile for 'admin'. Below the header is a table with two rows of pending requests.

User Name	For	Email	Action
Tanvir	SR	Tanvir@gmail.com	<a href="#">Approve</a>
Tanvir	Delivery Man	Tanvir@gmail.com	<a href="#">Approve</a>

## 6.1.5 SR home page

The screenshot shows the 'Dashboard' page for a user named 'Tanvir'. The page features a dark sidebar with 'SR MODEL', 'Products', and 'Salary' options. The main content area has a search bar and a user profile for 'Tanvir'. Below the header is a 'Dashboard' section with four key metrics: New Orders (150), Bounce Rate (53%), User Registrations (44), and Unique Visitors (65). Each metric has a 'More info' link. Below the metrics are two charts: 'Sales' (Area chart) and 'Visitors' (Map of the United States).

## 6.1.6 Product adding page

SR MODEL

admin

Search

### Product

Product ID

Product Name

Price

Create

## 6.1.7 Stock adding page

SR MODEL

admin

Search

### Product Stock

Product ID	Name	Price	Qty	
<input type="checkbox"/> 101	laptop	23000	<input type="text" value="0"/>	Add
<input type="checkbox"/> 102	pendrive	1000	<input type="text" value="0"/>	
<input type="checkbox"/> 103	keyboard	1500	<input type="text" value="0"/>	

## 6.1.8 List of product

### Product List

—

Product ID	Product Name	Price	Action
101	laptop	23000	<a href="#">Edit</a>   <a href="#">Delete</a>
102	pendrive	1000	<a href="#">Edit</a>   <a href="#">Delete</a>
103	keyboard	1500	<a href="#">Edit</a>   <a href="#">Delete</a>

Version 1.0.1

Copyright © 2018 DIU. All rights reserved.

## 6.1.9 Salary adding page

 SR MODEL

[Products](#) <

[User](#) <

### Salary

userName

month

basicSalary

taDa

others

Create

## 6.1.10 salary list

SR MODEL

Search

Tanvir

### Salary List

username	Month	Basic	TA/DA	Other	Total
tanvir	5000	3000	2000	1000	6000
mr.x	7000	5000	3000	1000	9000

## 6.1.1 take order

SR MODEL

Search

Tanvir

### Order

Order Number: 1

Order Date: 15-09-2020

Product ID	Name	Price	Stock	Quantity	
<input type="checkbox"/> 101	laptop	23000	-2	<input type="text" value="0"/>	Add
<input type="checkbox"/> 102	pendrive	1000	-3	<input type="text" value="0"/>	
<input type="checkbox"/> 103	keyboard	1500	-5	<input type="text" value="0"/>	

# **Chapter 7**

## **Conclusion**



## 7.1 GitHub Link

## 7.2 Project Summary

I have started to develop this project from January. From the very starting of development of this project, it demands hard working, patients, persistency to meet the requirements of stakeholders. After that I have proposed the design then started to work.

Database plays a vital role for any application software. And so this is why, I have designed the database diagram having tables with proper relationship. After that, I have prepared the user interface and take their approval to continue to the next part. It is to be said that, the interface of my application is very simple and easy to understand. After completing that, I have started to write the core functionality of the project.

Developing project is not end of all tasks actually. There are some other important tasks to perform. And that is testing. It is also known as quality assurance also. Almost at every software company there have a quality assurance team. Their main responsibility is to find the loop holes or vulnerability of software. If there any bug remains before handover to the stakeholders, there is a change to ruin the whole project. So testing plan is very important. And after developing the project, I have tried to assure the quality of this project.

## 7.3 Limitations

For developing this project, I have faced some limitations. Now I will describe those in brief.

- **Sending reports by location:** In my application, there is not having any GPS system by which one can contract with others very easily and find location. End user need to find best product on manual way.

## 7.4 Obstacles and Achievements

I believe that if there are not any obstacles to develop a project, then there doesn't have any challenges. Because we know, challenges give us the opportunity to prove ourselves. Obstacles, challenges and achievements are like a path to the success. Before starting this project, I didn't know the actual flow of software development life cycle. By developing this project, I have learnt to know how to have a row requirement from clients. After that I have learnt system analysis, database design and many things. My supervisor helps me a lot from the very beginning of the development of this project. I am very much lucky to have such kind of supervisor like her. She helped me lot.

There are some other obstacles and achievements also that I will describe below.

- **Lack of Actor's Engagement:** There are different types of stakeholders in my full web application. And each actors uses different functionality.
- **Scope Change:** Sometimes, some features need to be changed or modified. Then I need to follow reverse engineering process and again designed to meet the new requirements. It also made me frustrated sometimes.

## **7.5 Future Scope**

I have learnt a lot throughout the whole development stage of this project. For making this project developed, I have also meets some young entrepreneurs and enthusiasts also. I am very much thankful to all of them as their idea and discussion gave me some opportunities to make my project complete. It will help me to work with similar type project in future also.

## 7.6 References

I have gained some knowledge from some platforms. Obviously I will mention those references. For making my project successful those resources help me a lot. Not I will mention the names below.

- [www.google.com](http://www.google.com)
- [www.youtube.com](http://www.youtube.com)
- [www.github.com](http://www.github.com)
- [www.wikipedia.com](http://www.wikipedia.com)
- [www.php.net](http://www.php.net)
- [www.mysql.com](http://www.mysql.com)
- [www.mysql.com/products/workbench](http://www.mysql.com/products/workbench)
- [www.getbootstrap.com](http://www.getbootstrap.com)
- [www.w3schools.com](http://www.w3schools.com)
- [www.jquery.com](http://www.jquery.com)
- [www.sublimetext.com/3](http://www.sublimetext.com/3)
- [www.stakeoverflow.com](http://www.stakeoverflow.com)
- [www.notepad++.com](http://www.notepad++.com)
- <https://draw.io/>