

# **INVENTORY MANAGEMENT SYSTEM**

## **PROJECT REPORT**

Submitted in partial fulfilment of the

Requirements for the award of the Degree of

**BACHELOR OF SCIENCE (INFORMATION TECHNOLOGY)**

**By**

**Argha Arup Karmakar**

**Seat No: 3040463**

**Under the esteemed guidance of**

**Prof. Megha Patil**

**Designation: Lecturer**

**B.SC. IT (SEM V) EXAMINATION**



**DOMBIVILI SHIKSHAN PRASARAK MANDAL'S**  
**K V PENDHARKAR COLLEGE**  
**OF ARTS, SCIENCE AND COMMERCE,**  
DOMBIVILI, MAHARASHTRA  
ACCREDITED 'A' GRADE BY NAAC

**DEPARTMENT OF INFORMATION TECHNOLOGY**

*(Affiliated to university of Mumbai)*

**DOMBIVLI, 421203**

**MAHARASHTRA**

**YEAR - 2018-2019**

**DEPARTMENT OF INFORMATION TECHNOLOGY**

**K V PENDHARKAR COLLEGE**

*(Affiliated to University of Mumbai)*

**DOMBIVLI, 421203 MAHARASHTRA**

**2018-2019**

**DEPARTMENT OF INFORMATION TECHNOLOGY**



**DOMBIVILI SHIKSHAN PRASARAK MANDAL'S**  
**K V PENDHARKAR COLLEGE**  
**OF ARTS, SCIENCE AND COMMERCE,**  
**DOMBIVILI, MAHARASHTRA**  
**ACCREDITED 'A' GRADE BY NAAC**

**CERTIFICATE**

This is to certify that the project entitled , “ INVENTORY MANAGEMENT SYSTEM “ , is a bonfire work of ARGHA ARUP KARMAKAR ‘ bearing Seat No: 3040463 submitted in partial fulfilment of the requirement for the award of degree of BACHELOR OF SCIENCE , in INFORMATION TECHNOLOGY from University of Mumbai.

-----  
**Internal Guide**

-----  
**Coordinator**

-----  
**External Examiner**

**Date:**

**College seal**

## PROFORMA FOR THE APPROVAL PROJECT PROPOSAL

PNR No.: .....

Roll Number: 3040463

1. Name of the Student: ARGHA ARUP KARMAKAR

2. Title of the Project: Inventory Management System.

3. Name of the Guide: Prof. Megha Patil

4. Teaching experience of the guide:

5. Is this your first submission? Yes

Signature of the student

Signature of the Guide

Date.-----

Signature of the Coordinator

# **ABSTRACT**

To develop software that will help in the maintenance and management of any inventory system. This is an integrated system that covers different kinds of features like customer order, payment, billing, supplier details etc. Administrator is the user who logs in and manages the entire inventory. Report generation is done at every stage of sale as well as purchase.

The software developed takes care of both the sales and the storage details of goods. Customer orders are processed and goods are delivered to the customer. And when there is stock out, admin updates the inventory by ordering goods from different suppliers depending on the requirements.

The manual system involves paper work in the form of maintaining various files and manuals. Maintaining the critical information in the file, and manual is full of the risk and a tedious process. Including in the framework showing how to apply in the .NET progresses and skill as confidences grow. The application is designed in such a way that any further enhancements can be done with ease. New modules can be added to the existing system with less effort.

# ACKNOWLEDGMENT

This Project, being a maiden experience for us, has been a big confidence booster. Contributions of the following individuals / persons are noteworthy. I extend my sincere thanks to my teachers and friends who helped me in getting the details of the processes of the Inventory Management System. I would like to thank my classmates for their help in ASP.NET.

In this endeavor I acknowledge the valuable contribution and expert guidance of our internal project guide Prof. MEGHA PATIL of Department of Information Technology. Her frank suggestions for improvements & innovative ideas have inspired us.

I am greatly thankful to our Prof. MEGHA PATIL. For her continued and unending support. I thank Prof. RUPALI PATIL Head of Information Technology Department for granting me permission to work on this project.

We wish to express our gratitude to the staff of I.T. department for their cooperative & considerate approach. Last but not least, our sincere thanks to College who gives us a great opportunity to grasp the knowledge.

# DECLARATION

I hereby declared that the entitled “Inventory management system “done at Mumbai University. Has not been in any case duplicated to any submitted of to any other university for the award of the any degree. To the best of my knowledge other than me no one has submitted to any other university.

The project is done partial fulfilment of the requirement for the award to degree of **BACHELOR OF SCIENCE (INFORMATION OF TECHNOLOGY)** to be submitted as final semester project as part of the our curriculum.

ARGHA KARMAKAR

**MAIN REPORT**  
**OF**  
**INVENTORY MANAGFEMEDNT SYSTEM**

# INDEX

Sr.No.	Topic name	Page no.
1.	<b>Introduction of the Project.</b>  <input type="checkbox"/> Background <ul style="list-style-type: none"> <li>• Objective</li> <li>• Scope</li> <li>• Purpose</li> <li>• Applicability</li> <li>• Achievement</li> <li>• Organization of Report</li> </ul>	11-15
2.	<b>Survey of technology</b> <b>VISUAL STUDIO 2010</b> <b>MS SQL SERVER</b> <b>Advantages of MS SQL</b> <ul style="list-style-type: none"> <li>• Installation Is Streamlined</li> <li>• Security Features Are Better</li> <li>• Enhanced Performance</li> <li>• Lower Cost of Ownership</li> </ul>	16-18
3	<b>Requirement and Analyses</b> <b>3.1 &gt;Problem Definition</b> 1> Flexibility study 2> Technical flexibility 3> Cost flexibility 4> Market flexibility	19-24



	<b>5&gt; Operation flexibility</b>  <b>3.2 &gt; Requirement Specifications:</b> ❖ Software requirement: ❖ Hardware Requirements  <b>3.3&gt; Planning and Scheduling:</b>  <b>3.4 &gt; Software and Hardware Requirement:</b>  <b>3.5 &gt; Preliminary Product Description</b>	
<b>4.</b>	<b>Methodology</b> <b>Project Methodology</b>	<b>25-26</b>
<b>5.</b>	<b>System maintenance and evaluation.</b>  <input type="checkbox"/> Maintenance type.	<b>27</b>
<b>6.</b>	<b>Lifecycle of Project.</b>  <ul style="list-style-type: none"> <li>• Waterfall Model</li> <li>• Class Diagram</li> <li>• Use Case Diagram</li> <li>• Sequence Diagram</li> <li>• Activity Diagram</li> <li>• State Diagram</li> <li>• Database Design</li> </ul>	<b>28-38</b>

<b>7.</b>	<b>Implementation and Testing.</b>  <input type="checkbox"/> <b>Code Convention</b>	<b>39-91</b>
	<input type="checkbox"/> <b>Testing Approach</b> i.   Unit Testing ii.   Validation Testing iii.   System Testing  <input type="checkbox"/> <b>Test Cases</b>	<b>92-95</b>
<b>8.</b>	<b>Screen Layout</b>	<b>96-103</b>
<b>9.</b>	<b>Future Scope &amp; Conclusion.</b>	<b>104</b>
<b>10.</b>	<b>References.</b>  <input type="checkbox"/> <b>References</b>	<b>105</b>

# 1. INTRODUCTION

## BACKGROUND:

.Net is both a business strategy from Microsoft and its collection of programming support for what are known as web rather than your own computer for various services. Microsoft's goal is to provide individual and business users with a seamlessly interoperable and web-enabled interface for application and computing devices and to make computing activities increasingly web browser oriented. The .Net platform includes server, building block services, such as web-based data storage, and device software. It also includes passport, Microsoft's fill-on-the form only. Once identity verification services.

The .NET platform was designed to provide:

- 1) The ability to move the entire range of computing devices work together and to have user information automatically updates and synchronized on all them.
- 2) Increased interactive capability for web sites, enabled by greater use of XML rather than HTML.
- 3) A premium online subscription services, that will feature customized access and delivery of products and services to the user from a central starting point for the management of various application, such as e-mail, for example of software such as office .NET.
- 4) Centralized data storage, which will increase efficacy and ease of access to information, as well as synchronization of information among user and devices.
- 5) According to Bill Gates, Microsoft expects that .NET will have as significant an effect on the computing world as the introduction of windows.
- 6) The full release of .NET is expected to take several years to complete with intermittent release of products such as a personal security service and new version of windows and office that implement the .NET strategy coming on the market separately visual studio .NET is a development environment that is now available windows up support certain .NET capability.

## **OBJECTIVE:**

1. Hassle free management of goods.
2. Website is easily accessible to the dealer for confirming of order arrival.
3. Reduction in investment in inventory.
4. Proper and efficient use of raw materials
5. Privacy to the dealers.
6. No bottleneck in production.
7. Improvement in production and sales.
8. Bulk purchase of raw materials.
9. Maintenance of adequate inventories reduces the setup cost associated with each production run.
10. Economy in purchasing and reasonable price.
11. Provides security to the data by using login & password method.

## **SCOPE:**

The scope of an inventory system defines which needs it addresses, including valuing the inventory, measuring the change in inventory and planning for future inventory levels. The value of the inventory at the end of each period provides a basis for financial reporting on the balance sheet. Measuring the change in inventory allows the company to determine the cost of inventory sold during the period. Together, inventory values and level changes allow the company to plan for future inventory needs.

## **PURPOSE:**

All inventories, regardless of their type or volume can be considered as a blocked working capital of an enterprise in the form of raw materials or finished products. So, ideally it should be zero, i.e. the volume of stock demanded by the customers should be directly sourced from the manufacturers without any time lag. However, in the real world scenario this is not possible as there is always a time lag between the manufacturers manufacturing goods and the customers ordering for them. As such, proper arrangements need to be made for the proper maintenance of the inventory in an organization so that no material gets wasted, and the available space is properly utilized. Also, the optimum volume of stock need to be maintained at all times in the inventory, as both over & under stocking can have adverse effects on the business, and result in massive losses.

1. **Production Planning:** These days many manufacturers directly sell off their goods to the end-customers. For such companies, it is important that the sales forecast
2. made after careful analysis is passed over to the product planning department so that they can adjust the manufacturing process accordingly.
3. **Material Requirements Planning:** For manufacturing, an adequate quantity of raw materials is a must. With accurate sales forecast, manufacturing companies will be in a better position to analyze their raw material requirements, and hence, plan it better.
4. **Inventory Reduction:** As the volume of stock manufactured is quickly sourced out as sales, so the net volume of stock in the warehouse is less. The reduction in inventory means the company will have to spend lesser on its maintenance, and lesser storage space will be required.
5. **Efficient tracking of goods in the inventory:** This is undoubtedly the main objective of an inventory management system. It should be able to track all the goods entering the warehouse, and all the goods being shipped out upon sales. Moreover, it should also be able to track the movement of goods in the warehouse.
6. **Minimize the human workforce required for warehouse management:** As most of the tedious bookkeeping and good tracking work of a warehouse is done automatically by the inventory management system, the organization can efficiently manage the inventory with far lesser

employees. This translates into huge savings for the organization, especially for small and medium.

## **APPLICABILITY**

Manufacturers primarily use inventory management software to create work orders and bills of materials. This facilitates the manufacturing process by helping manufacturers efficiently assemble the tools and parts they need to perform specific tasks. For more complex manufacturing jobs, manufacturers can create multilevel work orders and bills of materials, which have a timeline of processes that need to happen in the proper order to build a final product. Other work orders that can be created using inventory management software include reverse work orders and auto work orders. Manufacturers also use inventory management software for tracking assets, receiving new inventory and additional tasks businesses in other industries use it.

## **ACHIEVEMENT:**

This Project, being a maiden experience for us, has been a big confidence booster. Contributions of the following individuals / persons are noteworthy. I extend my sincere thanks to my teachers and friends who helped me in getting the details of the processes of the Inventory Management. I would like to thank my classmates for their help in ASP.NET.

## **ORANIZATION OF REPORT:**

The purpose of this unit is to introduce you to some general aspect of reports writing in both the general function and structure of reports increases it is better to adapt report writing reports communicate information which has been complied as a result of research and analysis of data and old issues the type of information to be communicated in a report and who the information is for will a reports scope and style the purpose of report vary widely the purpose are to information to provide a basis for discussion and debate to sell

or persuade . According to user requirements the aim is to develop a desktop based Application system of improved facilities. The proposed system can over come all the limitation of the existing .

## 2. SURVEY OF TECHNOLOGY

### □ VISUAL STUDIO 2010:

- On 12 April 2010, Microsoft released Visual Studio 2010, codenamed Dev10 and .NET Framework 4.
- The Visual Studio 2010 IDE was redesigned which, according to Microsoft, clears the UI organization and "reduces clutter and complexity." The new IDE Bette on 12 April 2010, Microsoft released Visual Studio 2010, codenamed Dev10 and .NET Framework 4.
- The Visual Studio 2010 IDE was redesigned which, according to Microsoft, clears the UI organization and "reduces clutter and complexity." The new IDE better supports multiple document windows and floating tool windows, while offering better multi-monitor support. The IDE shell has been rewritten using the Windows Presentation Foundation (WPF), whereas the internals have been redesigned using Managed Extensibility Framework (MEF) that offers more extensibility points than previous versions of the IDE that enabled add-ins to modify the behavior of the IDE.
- Visual Studio 2010 comes with .NET Framework 4 and supports developing applications targeting Windows 7. It supports IBM DB2 and Oracle databases, in addition to Microsoft SQL Server. It has integrated support for developing Microsoft Silverlight applications, including an interactive designer. Visual Studio 2010 offers several tools to make parallel programming simpler: in addition to the Parallel Extensions for the .NET

Framework and the Parallel Patterns Library for native code, Visual Studio 2010 includes tools for debugging parallel applications. The new tools allow the visualization of parallel Tasks and their runtime stacks. Tools for profiling parallel applications can be used for visualization of thread wait times and thread migrations across processor cores. Intel and Microsoft have jointly pledged support for a new Concurrency Runtime in Visual Studio 2010[108] and Intel has launched parallelism support in Parallel Studio as an add-on for Visual Studio.

- Visual Studio 2010 features a new Help System replacing the MSDN Library viewer. The Help System is no longer based on Microsoft Help 2 and does not use Microsoft Document Explorer. Dynamic help containing links to related help topics based on where



the developer was in the IDE has been removed in the shipping product, but can be added back using a download from Microsoft.

- Visual Studio 2010 no longer supports development for Windows Mobile prior to Windows Phone 7. Visual Studio 2010 Service Pack 1 was released in March 2011.

## **MS SQL SERVER**

Microsoft SQL is a popular Relational Database Management System (RDBMS) developed by Microsoft. Being a database server, its primary function is to store and retrieve the data as and when requested by other software applications. Administering Microsoft SQL server database can help you optimize as well as maintain your server's performance while ensuring its recoverability and availability. In this blog post, we will explore the advantages and best practices associated with MS SQL.

### ☐ **Advantages of MS SQL**

#### ☐ **Installation Is Streamlined**

It can be installed via a setup wizard and the prerequisite updates are detected and downloaded by the installer automatically. The complexity of installing the software is minimized significantly because of automatic installation of updates. Other components such as analytical and database services can be installed separately afterward. Automatic updation also reduces maintenance costs quite significantly.

#### ☐ **Security Features Are Better**

SQL Server 2008 uses Policy-Based Management to detect security policies that are non-compliant. This feature allows only authorized personnel access to the database. Security audits and events can be written automatically to log files.

## ☐ **Enhanced Performance**

The MS SQL server has built-in transparent data compression feature along with encryption. Users don't need to modify programs in order to encrypt the data. The MS SQL server has access control coupled with efficient permission management tools. Further, it offers an enhanced performance when it comes to data collection.

## ☐ **Lower Cost of Ownership**

SQL server includes effective data management and data mining tools along with disk partitioning. Your server's optimum maintenance can be ensured by following effective data management practices. These practices also help you ensure the availability and recoverability of data

## 3. REQUIREMENT AND ANALYSES

### 3.1 Problem Definition

To develop software that will help in the maintenance and management of any inventory system. This is an integrated system that covers different kinds of features like customer order, payment, billing, supplier details etc. Administrator is the user who logs in and manages the entire inventory. Report generation is done at every stage of sale as well as purchase.

### FEASIBILITY:

Feasibility study is conducted once the problem is clearly understood. Feasibility study is a high-level capsule version of the entire system analysis and design process. The objective is to determine quickly at a minimum expense how to solve a problem. The purpose of feasibility is not to solve the problem but to determine if the problem is worth solving. The system has been tested for feasibility in the following points.

1. Technical Feasibility.
2. Cost Feasibility.
3. Market Feasibility.
4. Operational Feasibility.

**2.2.1. Technical Feasibility:** The project's hardware configuration requirement is very less which makes it technically feasible. Also, it is using a database made in SQLite which makes it easy to use as the software are easily available. It is completely made in Android Studio and thus easy for further developments and changes. The Language used to development is Java and XML which is very popular and easy to understand.

**2.2.2. Cost Feasibility:** The project is economical to develop, as all the technology used is open source. Also, software like SQLite and Android Studio is available freely on from the internet. Also, the hardware used is very standard without any special requirement needed.

**2.2.2. Market Feasibility:** As the project is an Android based application it is very feasible. As Android Smartphone being a very common device these days the applications use becomes very easy once a phone is configured.

**2.2.3. Operational Feasibility:** The project is operationally very feasible as it over comes all the problems faced by the old system. Also, it Digitalize the current system eliminating the usage of external medium like paper and pen.

### **3.2 Requirement Specifications:**

#### ☐ **Software requirement:**

- US 2010 (Also contains inbuilt SQL)
- Paradigm Structured imperative, object oriented, declarative.
- Appeared in: 2001 (Last revised 2008) ☐ Designed by: Microsoft corporation.
- Deepening Discipline: Dynamic, strong, both safe and unsafe nominative.
- Molar Implementations: .NET Framework, Mona
- Influenced: None
- SQL Server 2008:

#### ☐ **Hardware Requirements**

- PROCESSOR: WINDOWS 8.1, INTEL CORE I3
- RAM: 4GB
- HARDDISK: 1TB

1. If provides feedback to the customer: An SQL in the customer's assurance that the development organization under stance the issues of problems, to be solved and the software behavior necessary to address these problems.
- 2 The simples act of writing down, software requirement in a well. Designed formal organize information places border, around the problem.

### **3.3 Planning and Scheduling:**

A Gantt chart is a graphical representation of the duration of tasks against the progression of time. A Gantt chart is a useful tool for planning and scheduling projects. A Gantt chart is helpful when monitoring a project's progress.

1. A Gantt chart allows you to assess how long a project should take. A Gantt chart lays out the order in which tasks need to be carried out. A Gantt chart helps manage the dependencies between tasks. A Gantt chart allows you to see immediately what should have been achieved at a point in time. A Gantt chart allows you to see how remedial action may bring the project back on course.
2. A Gantt chart is constructed with a horizontal axis representing the total time span of the project, broken down into increments (for example, days, weeks, or months) and a vertical axis representing the tasks that make up the project (for example, if the project is outfitting your computer with new software, the major tasks involved might be: conduct research, choose software, install software). Horizontal bars of varying lengths represent the sequences, timing, and time span for each task.
3. Using the same example, you would put "conduct research" at the top of the vertical axis and draw a bar on the graph that represents the amount of time you expect to spend on the research, and then enter the other tasks below the first one and representative bars at the points in time when you expect to undertake them. The bar spans may overlap, as, for example, you may conduct research and choose software during the same time span. As the project progresses, secondary bars, arrowheads, or darkened bars may be added to indicate completed tasks, or the portions of tasks that have been completed. A vertical line is used to represent the report date

Tasks	5 July	30 July	2 Aug	12 Sep	15 Sep	23 Oct	25 Oct	13 Feb	16 Feb	27 Feb	28 Feb
Requirement Gathering											
Planning											
Design											
Coding											
Testing											
Delivery											

### 3.4 Software and Hardware Requirement:

On 12 April 2010, Microsoft released Visual Studio 2010, codenamed Dev10 and .NET

Framework 4.0. The Visual Studio 2010 IDE was redesigned which, according to Microsoft, clears the UI organization and "reduces clutter and complexity." The new IDE better supports multiple document windows and floating tool windows, while offering better multi-monitor support.

- 4 The IDE shell has been rewritten using the Windows Presentation Foundation (WPF), whereas the internals have been redesigned using Managed Extensibility Framework (MEF) that offers more extensibility points than previous versions of the IDE that enabled add-ins to modify the behavior of the IDE.
- 5 Visual Studio 2010 comes with .NET Framework 4 and supports developing applications targeting Windows 7. It supports IBM DB2 and Oracle databases, in addition to Microsoft SQL Server. It has integrated support for developing Microsoft Silverlight applications, including an interactive designer. Visual Studio 2010 offers several tools to make parallel programming simpler: in addition to the Parallel Extensions for the .NET Framework and the Parallel Patterns Library for native code, Visual Studio 2010 includes tools for debugging parallel applications.
- 6 The new tools allow the visualization of parallel Tasks and their runtime stacks. Tools for profiling parallel applications can be used for visualization of thread wait-times and thread migrations across processor cores. Intel and Microsoft have jointly pledged

support for a new Concurrency Runtime in Visual Studio 2010[108] and Intel has launched parallelism support in Parallel Studio as an add-on for Visual Studio.

- 7 Visual Studio 2010 features a new Help System replacing the MSDN Library viewer. The Help System is no longer based on Microsoft Help 2 and does not use Microsoft Document Explorer. Dynamic help containing links to related help topics based on where the developer was in the IDE has been removed in the shipping product, but can be added back using a download from Microsoft.
- 8 Visual Studio 2010 no longer supports development for Windows Mobile prior to Windows Phone 7. Visual Studio 2010 Service Pack 1 was released in March 2011

i. **SQL:**

Computer programming language used for retrieving records or parts of records in databases and performing various calculations before displaying the results. SQL is particularly suitable for searching relational databases. It has a formal, powerful syntax and is able to accommodate logical operators. Its sentence-like structure resembles natural language except that its syntax is limited and fixed.

## ☐ **Hardware Requirements**

- PROCESSOR: WINDOWS 8.1, INTEL CORE I3
- RAM: 4GB
- HARDDISK: 1TB

### **3.5 Preliminary Product Description:**

1. The first step the system development life cycle is the preliminary are investigation to determine the feasibility of the system.
2. The purpose of the preliminary investigation is to evaluate project requests.it is not a design study dose note it include the collection of details to describe the business system in all respect.
3. It is the collecting of information that helps committee members to evaluate the merits of the project request and make an informed about the feasibility of the proposed project.

\* Analysis working on the preliminary investigation should accomplish the following objectives.

- Clarify and understand the project request of alternative approaches.
- Determine the size of the project.
- Access cost and benefits of alternative approaches.
- Determine the technical and operational feasibility of alternative approaches.



## 4. METHODOLOGY

A methodology is a model, which project managers employ for the design, planning, implementation and achievement of their project objectives. There are different project management methodologies to benefit different projects. For example, there is a specific methodology, which NASA uses to build a space station while the Navy employs a different methodology to build submarines. Hence, there are different project management methodologies that cater to the needs of different projects spanned across different business domains. Project Methodologies: Following are the most frequently used project management methodologies in the project management practice:

### **1. System/Information Engineering and Modeling:**

As software is always of a large system (or business), work begins by establishing the requirements for all system elements and then allocating some subset of these requirement to software. This system view is essential when the software must interface with other elements such as hardware, people and other resources.

### **2. Software Requirement Analysis:**

The requirements gathering process is intensified and focused specifically on software. To understand the nature of the program to build, the software engineer must understand the information domain for the software as well as required function, behaviour, performance and interface. Requirements for both the system and the software are documented and reviewed with customers.

It is very necessary to know about the requirements of the user before starting any project. Working on the present requirements of the user will be helpful in gaining popularity of your project.

**3. System Analysis and Design:** In this phase, the software development process, the software's overall structure and its nuances are defined. A software development model is thus created. Analysis and

design are very crucial in the whole development cycle. Much care is taken during this phase. The logical system of the product is developed in this phase.

**4. Code Generation:** This design must be translated into a machine-readable form. The code generation steps perform is this task. If designed is performed in a detailed manner. If the generator is good, it will allow you to change functionality, but that almost increases the complexity of the code generated.

## **5. Testing**

Once code has been generated, program testing begins. The testing process begins focuses on the logical internals of the software, ensuring that all the statements have been tested, and on the functional externals: that is, conducting tests to uncover errors and ensure that defined input will produce actual results that agree with required results.

- It confirms that all the declared functional requirements have been implemented.
- It gives confidence in the quality of the final product.
- It confirms that the application has no errors in the code.

## **6. Maintenance:**

The software will definitely undergo changes once it is delivered to the customer. There can be many reasons for this change to occur. Change could happen because of some unexpected input values into the system. In addition, the changes in the system could directly affect the software operations. The software should be developed to accommodate changes that could happen during the post implementation period.

- Maintenance has to be performed to repair existing faults in the software
- Software maintenance is important part of the software after its execution

## 5. SYSTEM MAINTAINANCE AND EVALUATION

System Maintenance / Enhancement Maintenance means restoring something to its original conditions. Enhancement means adding, modifying the code to support the changes in the user specification. System maintenance conforms the system to its original requirements and enhancement adds to system capability by incorporating new requirements. Thus, maintenance changes the existing system, enhancement adds features to the existing system, and development replaces the existing system. It is an important part of system development that includes the activities which corrects errors in system design and implementation, updates the documents, and tests the data.

- **Maintenance:** Types: System maintenance can be classified into three types
- **Corrective Maintenance** - Enables user to carry out the repairing and correcting leftover problems.
- **Adaptive Maintenance** - Enables user to replace the functions of the programs.

## 6. LIFECYCLE OF PROJECT

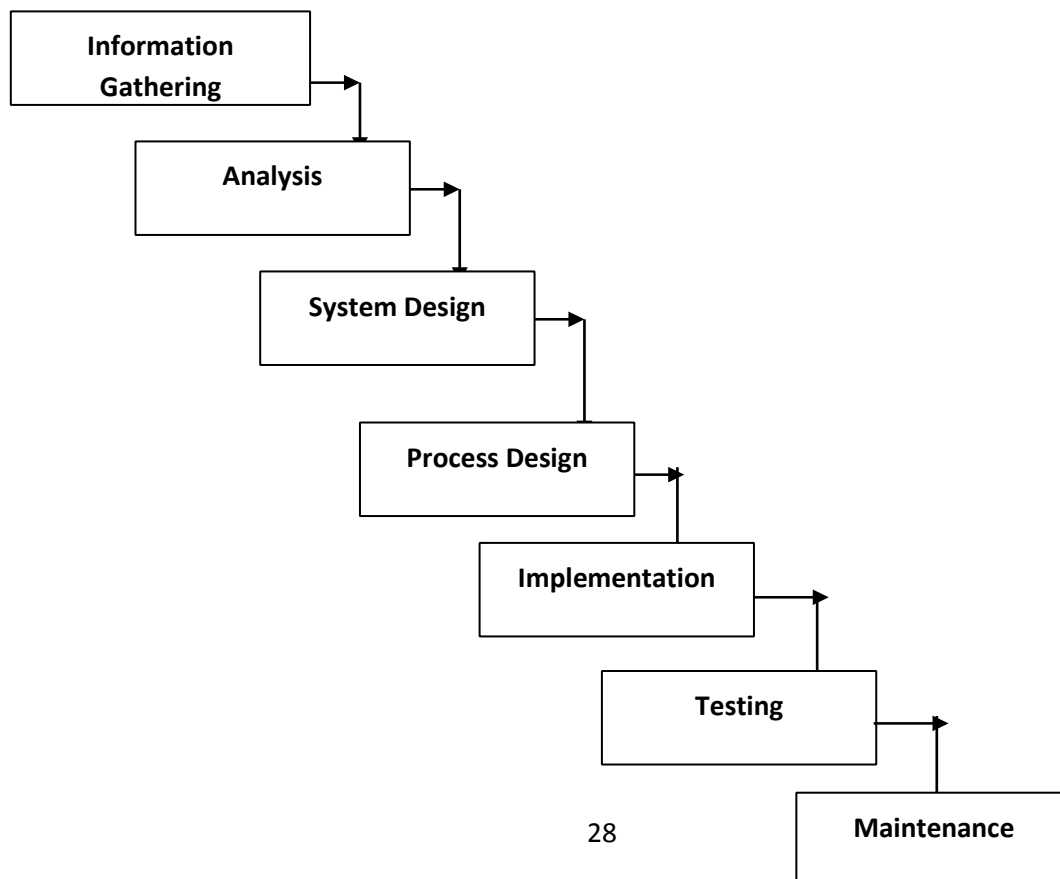
### Waterfall Model:

As waterfall flows from top to bottom, the system model shows the development process from the top to the bottom in steps. As water does not rise from lower level to a higher level, it is presumed that once a step in the model is over, it is not required to go back. Specifications are not required frequently, the minor changes can be taken care of through a maintenance process or through small design changes. In project, our system will behave like a waterfall model. We have to set goal along with some options that in case of any error problems; we shall make some changes in coding. Then came the analysis part, where we analyzed the various data and then information provided to us by the institute. Then we checked out various relations of data. Some data flow problems came so then we did what we could, some issues remained unsolved, but the agency accepted the package as whole. We then also created the database with tables. In each table we had primary keys, which also had been the foreign key for the other table.

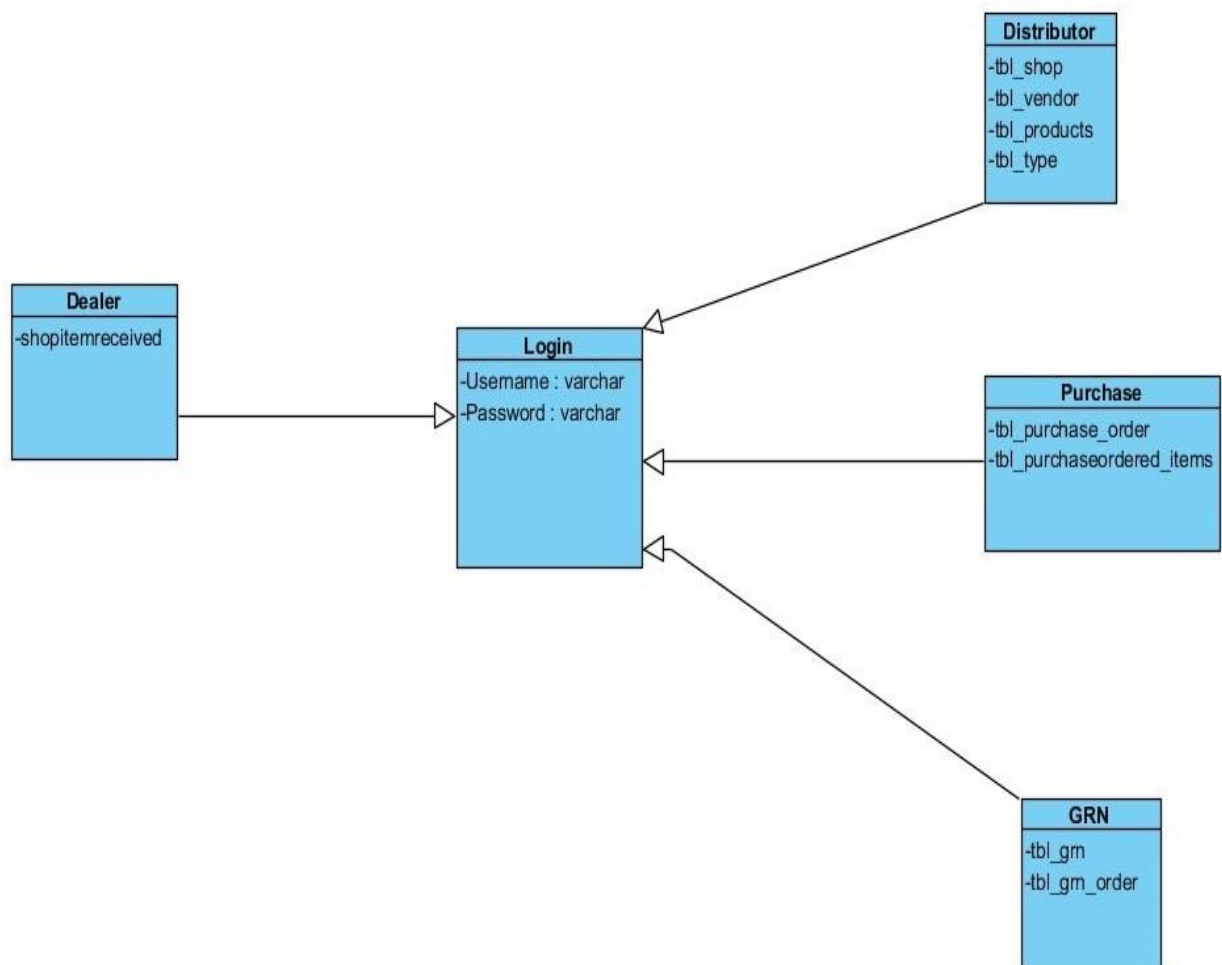
### 1. Class Diagram

Class diagram is static diagram describing the attributes and the constraints imposed on the system. It shows the collection of classes, interfaces, associations, collaboration, and constraint.

After designing part we prepared flow chart given.

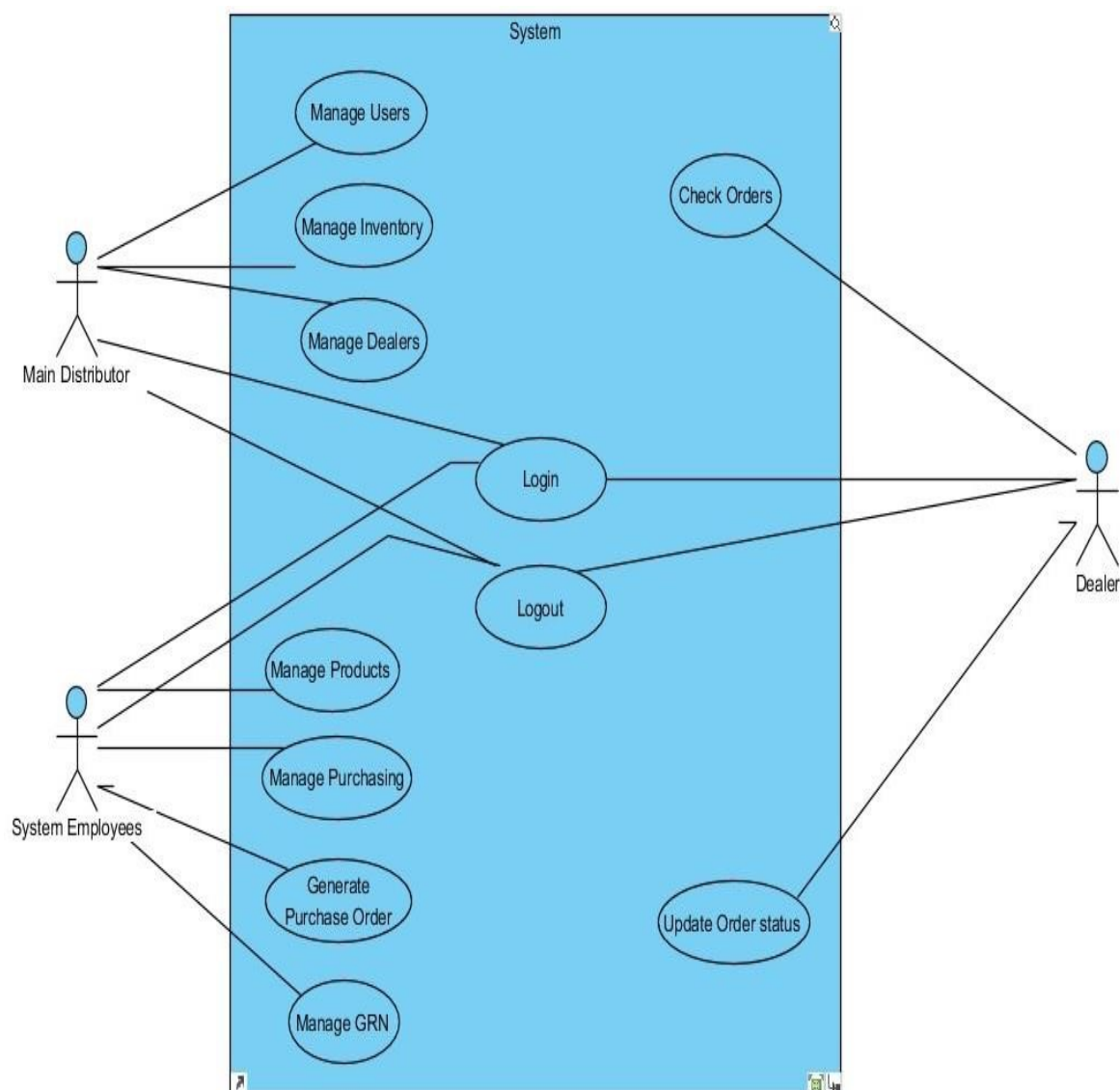


The including the main class as admin, customer order the system has to manage the customer order customer name generate the bill and give the update on particular software system .



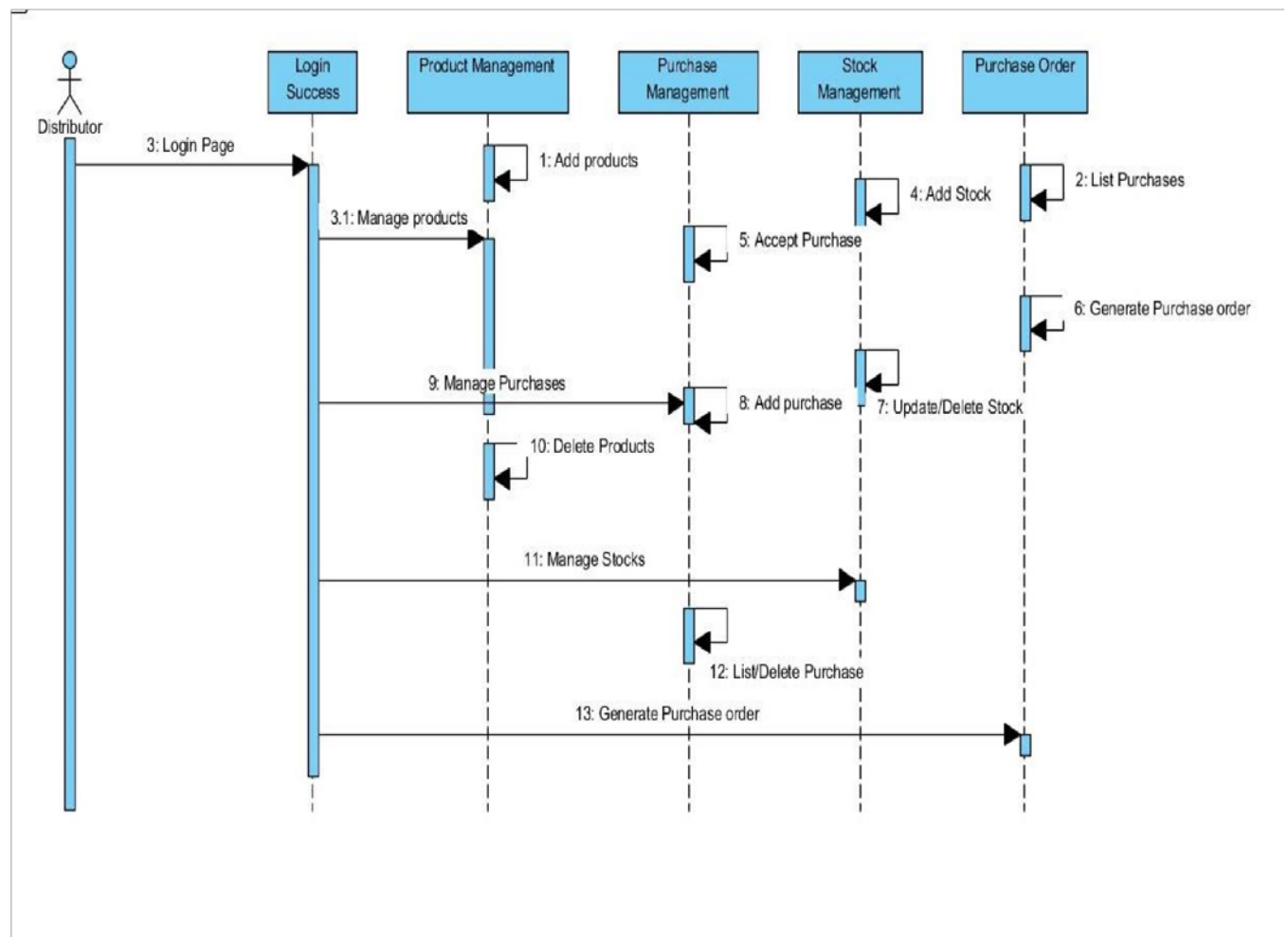
## 2.Use Case Diagram:

A **use case diagram** at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use case in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.



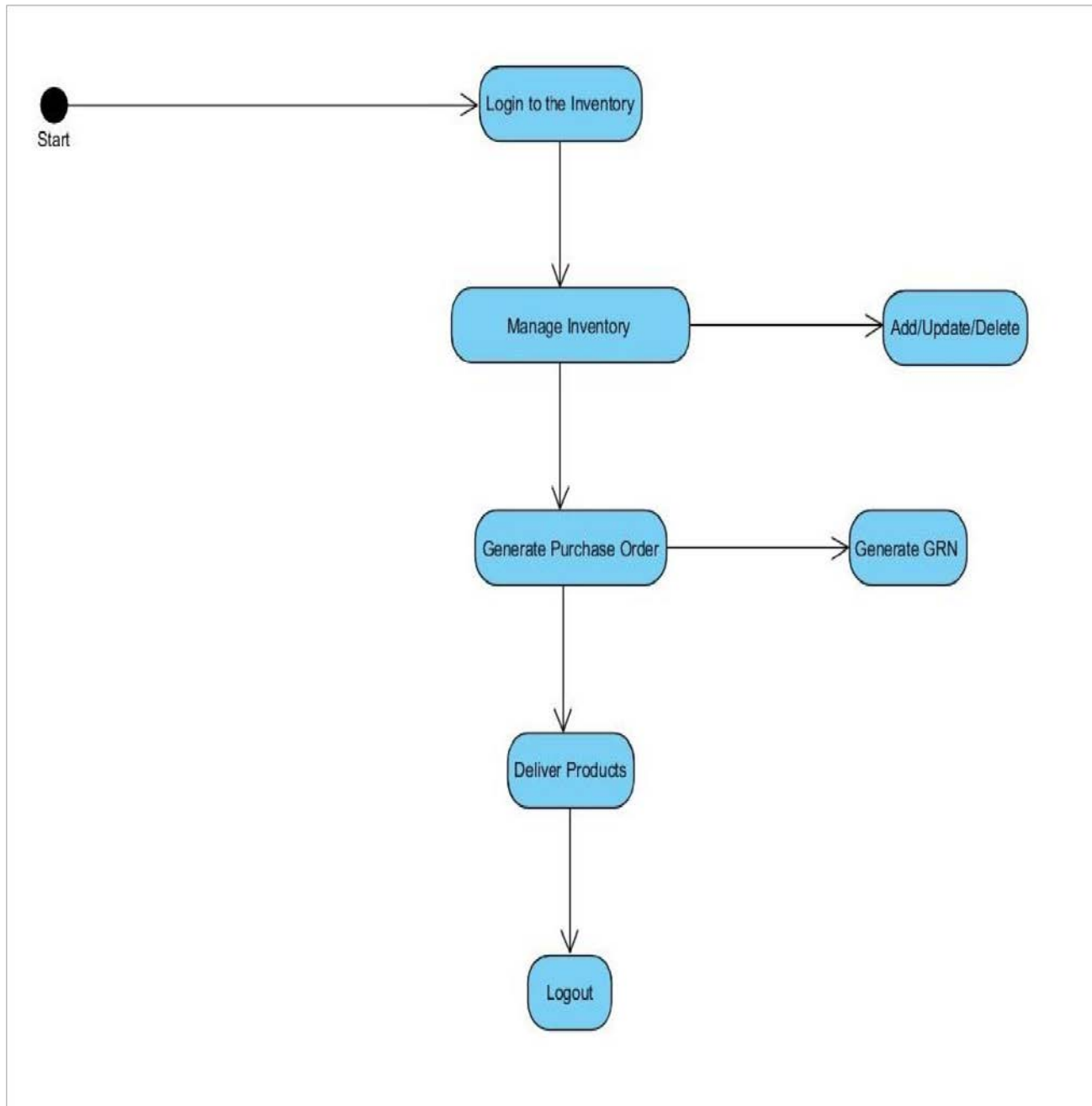
### 3.Sequence Diagram

A simple flowchart representing a process for dealing with a non-functioning lamp. A flowchart is a type of diagram that represents an algorithm, workflow or process. The flowchart shows the steps as boxes of various kinds, and their order by connecting the boxes with arrows. This diagrammatic representation illustrates a solution model to a given problem. Flowcharts are used in analyzing, designing, documenting or managing a process or program in various fields.



## 4. Activity Diagram

The graphical representations of work flow of stepwise activities and actions<sup>1</sup> with support for choice, iteration and concurrency. In the UML, activity diagrams are intended to model both computational and organizational processes (e:g: workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.





## 5. State Diagram

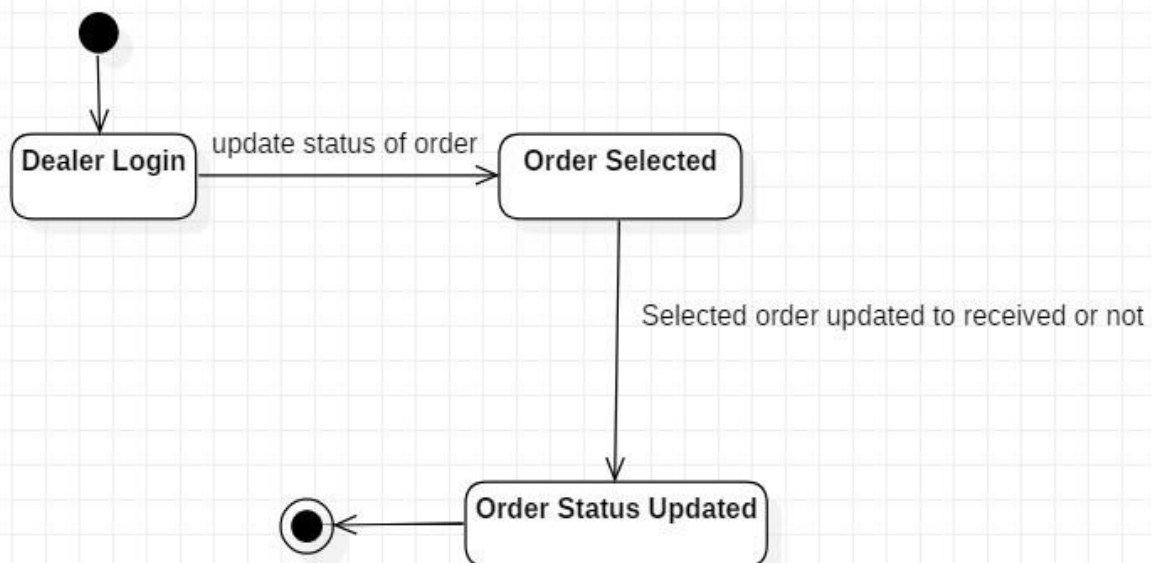
diagrams are very important for describing the states. States Statechart can be identified as the condition of objects when a particular event occurs.

Before drawing a State chart diagram we should clarify the following points – □Identify the important objects to be analyzed.

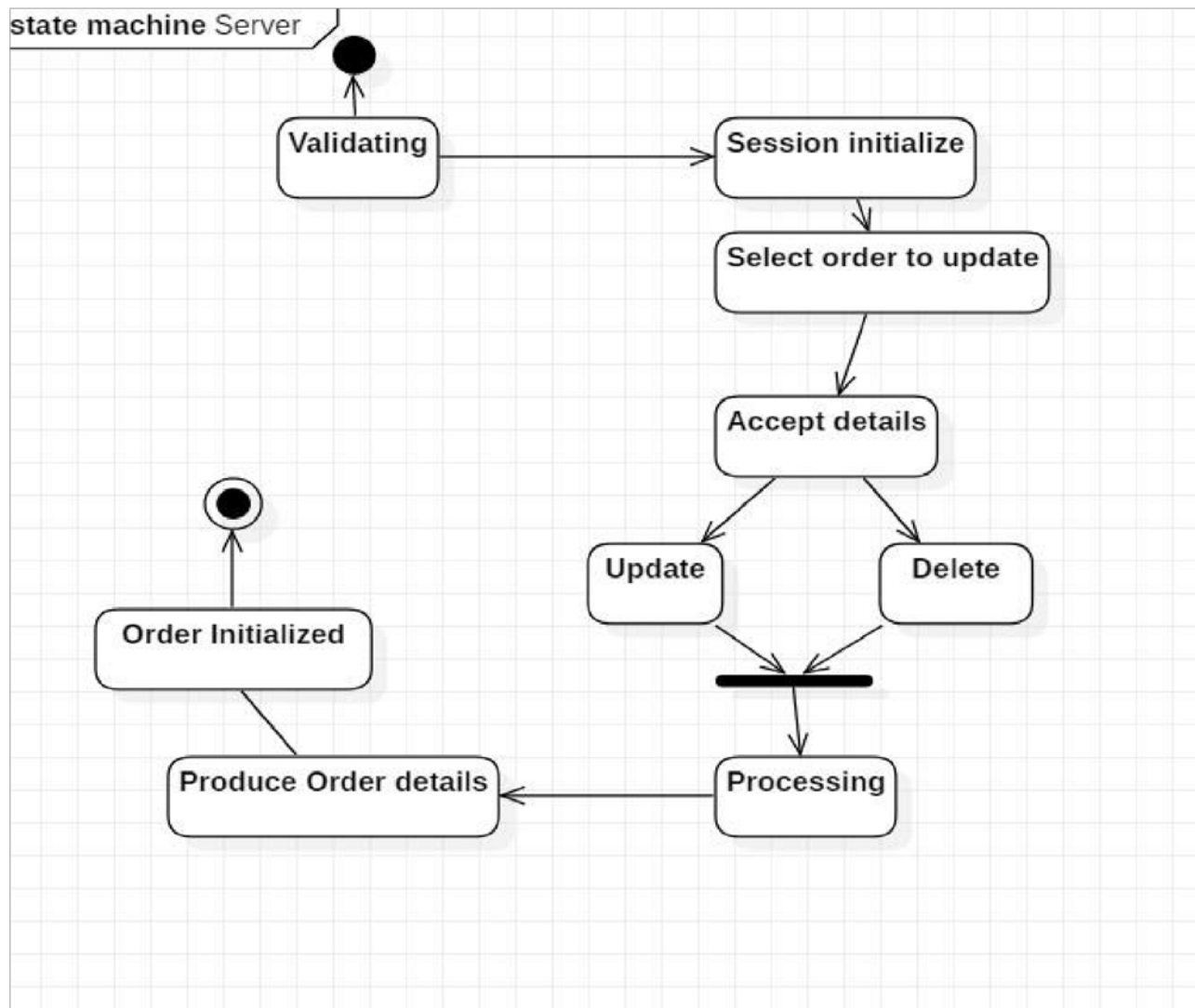
- Identify the states.
- Identify the events.

### 1.Dealer State:

state machine Dealer State



## 2.Server State:



## Database Design:

- User login:

LAPTOP-A9ETK1BG\...bo.tbl_userlogin			
	Column Name	Data Type	Allow Nulls
🔑	uid	numeric(18, 0)	<input type="checkbox"/>
	username	varchar(50)	<input checked="" type="checkbox"/>
	password	varchar(50)	<input checked="" type="checkbox"/>
	roleid	numeric(18, 0)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

- Product table:

LAPTOP-A9ETK1BG...dbo.tbl_products			
	Column Name	Data Type	Allow Nulls
🔑	product_id	numeric(18, 0)	<input type="checkbox"/>
	product_name	varchar(50)	<input checked="" type="checkbox"/>
	product_mrp	numeric(18, 0)	<input checked="" type="checkbox"/>
	product_vprice	numeric(18, 0)	<input checked="" type="checkbox"/>
	product_distprice	numeric(18, 0)	<input checked="" type="checkbox"/>
	product_type_id	numeric(18, 0)	<input checked="" type="checkbox"/>
	enable	varchar(50)	<input checked="" type="checkbox"/>
▶			<input type="checkbox"/>


- Shop table:

LAPTOP-A9ETK1BG\...ry - dbo.tbl_shop			
	Column Name	Data Type	Allow Nulls
▶	shop_id	numeric(18, 0)	<input type="checkbox"/>
	shop_name	varchar(50)	<input checked="" type="checkbox"/>
	shop_address	varchar(50)	<input checked="" type="checkbox"/>
	shop_phoneno	varchar(50)	<input checked="" type="checkbox"/>
	owners_name	varchar(50)	<input checked="" type="checkbox"/>
	username	varchar(50)	<input checked="" type="checkbox"/>
	password	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>


- Vendor table:

LAPTOP-A9ETK1BG\... dbo.tbl_vendors			
	Column Name	Data Type	Allow Nulls
▶	vendor_id	numeric(18, 0)	<input type="checkbox"/>
	vendor_name	varchar(50)	<input checked="" type="checkbox"/>
	vendor_address	varchar(50)	<input checked="" type="checkbox"/>
	vendor_phoneno	varchar(50)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>


- Product-Type table:

LAPTOP-A9ETK1BG\...ry - dbo.tbl_type			LAPTOP-A9ETK1
Column Name	Data Type	Allow Nulls	
 type_id	numeric(18, 0)	<input type="checkbox"/>	
type_name	varchar(50)	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	


- Purchase-Order table:

LAPTOP-A9ETK1B...aseordred_items			
Column Name	Data Type	Allow Nulls	
 id	numeric(18, 0)	<input type="checkbox"/>	
purchase_id	numeric(18, 0)	<input checked="" type="checkbox"/>	
product_id	numeric(18, 0)	<input checked="" type="checkbox"/>	
qty	numeric(18, 0)	<input checked="" type="checkbox"/>	
total_price	numeric(18, 0)	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

- GRN table:

LAPTOP-A9ETK1BG\...ry - dbo.tbl_grn			LAPTOP-A9ETK1BG
Column Name	Data Type	Allow Nulls	
 grn_id	numeric(18, 0)	<input type="checkbox"/>	
grn_qty	numeric(18, 0)	<input checked="" type="checkbox"/>	
total_price	numeric(18, 0)	<input checked="" type="checkbox"/>	
dated	varchar(50)	<input checked="" type="checkbox"/>	
purchase_id	numeric(18, 0)	<input checked="" type="checkbox"/>	
		<input type="checkbox"/>	

- Distribution table:

LAPTOP-A9ETK1BG\S...dbo.tbl_dist_list			
	Column Name	Data Type	Allow Nulls
	id	numeric(18, 0)	<input type="checkbox"/>
	product_id	numeric(18, 0)	<input checked="" type="checkbox"/>
	product_name	varchar(50)	<input checked="" type="checkbox"/>
	shop_id	numeric(18, 0)	<input checked="" type="checkbox"/>
	qty	numeric(18, 0)	<input checked="" type="checkbox"/>
	total_price	numeric(18, 0)	<input checked="" type="checkbox"/>
	dist_id	numeric(18, 0)	<input checked="" type="checkbox"/>
	received	numeric(18, 0)	<input checked="" type="checkbox"/>
			<input type="checkbox"/>

## 7. IMPLEMENTATION AND TESTING

### Coding Detail and code efficiency:

In the software, to validate the form there are many if else and try-catch statements are used. To view the data there are use of data grid view technology is used. For backup and restore the SQL query is use to create the .bak file and restore it with replace. There is also code for insertion, updation, deletion operations. The main or core part of software code is to perform database operation, Various Manipulation Operation, form validation, etc.

### Distributor login:

#### login.aspx

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="login.aspx.cs" Inherits="login" %>

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

<html xmlns="http://www.w3.org/1999/xhtml">
<head id="Head1" runat="server">

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

<style type="text/css">
body
{
    background-image:url("img/office.jpg");
    background-size:cover;
    background-attachment:fixed;
    background-repeat:no-repeat;
}
.jumbotron
{
    margin-top: 100px;
    box-shadow: 5px 5px 15px 2px black;
    border-radius: 5px;
    padding-bottom: 20px;
    background: light-grey;

}
.title
{
```

```

        background : blue;
        padding :10px;
        text-align:center;
        color:white;
        border-radius:100px 100px 100px 100px;
    }

</style>
</head>
<body>
    <form id="form1" runat="server">
        <div>
            <div class="container">
                <div class="row">
                    <div class="col-md-4"></div>
                    <div class="col-md-4">
                        <div class="jumbotron ">
                            <form>
                                <div class="form-group">
                                    <h2 class="title">Login Now </h2>
                                    <p id="profile-name" class="profile-name-card" ></p>

                                    <asp:TextBox ID="TextBox1" runat="server" CssClass="form-control"
placeholder="Username"></asp:TextBox>

                                </div>
                                <div class="form-group">

                                    <asp:TextBox ID="TextBox2" runat="server" CssClass="form-control"
placeholder="Password" TextMode="Password"></asp:TextBox>

                                </div>

                                <asp:Button ID="Button1" runat="server" Text="Login" CssClass="btn btn-block btn-
primary" onclick="Button1_Click" />
                                <br />
                            </form>
                        </div>
                    </div>
                    <div class="col-md-4"></div>
                </div>
            </div>
        </form>
    </body>
</html>

```



## login.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class login : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        string output = d.getSingleRow("select * from tbl_userlogin where username='" + TextBox1.Text + "'
and password='" + TextBox2.Text + "'");
        if (output.Length == 0)
        {
            Response.Write("<script type='text/javascript'>alert('LOGIN FAILED')</script>");
            // Response.Redirect("login.aspx");
        }
        else
        {
            string[] alldata = output.Split(':');
            if (alldata[1].Equals(TextBox1.Text) && alldata[2].Equals(TextBox2.Text))
            {
                Response.Write("<script type='text/javascript'>alert('VALID')</script>");
                //Response.Write("valid");
                Session["uid"] = alldata[0];
                Session["uname"] = alldata[1];
                Session["role"] = d.getSingleRow("select role_name from tbl_role where role_id=" +
alldata[3]).Split(':')[0];
                if (Session["role"].Equals("ADMIN"))
                {
                    Response.Redirect("admin_addproduct.aspx");
                }
                else if (Session["role"].Equals("PURCHASE"))
                {
                    Response.Redirect("purchase_purchase_order.aspx");
                }
                else if (Session["role"].Equals("GRN"))
                {
                    Response.Redirect("generategrn.aspx");
                }
                else if (Session["role"].Equals("DISTRIBUTER"))
                {
                    Response.Redirect("prod_distri.aspx");
                }
            }
        }
    }
}
```

```

    }
    else
    {
        //Response.Write("INVALID DETAILS");
        Response.Write("<script type='text/javascript'>alert('INVALID DETAILS')</script>");
    }
}
}
}
}

```

### Web.config

```

<?xml version="1.0"?>
<!--
  For more information on how to configure your ASP.NET application, please visit
  http://go.microsoft.com/fwlink/?LinkId=169433
-->
<configuration>
  <connectionStrings>
    <add name="invconn" connectionString="Data Source=.\\sqlexpress;Initial
Catalog=argha_inventory;Integrated Security=True" providerName="System.Data.SqlClient"/>
  </connectionStrings>
  <system.web>
    <compilation debug="true" targetFramework="4.0"/>
  </system.web>
</configuration>

```

## Db.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Data.SqlClient;
using System.Web.Configuration;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Net;
using System.Net.Mail;

public class db
{
    SqlConnection con = new
SqlConnection(WebConfigurationManager.ConnectionStrings["invconn"].ConnectionString);
    SqlCommand cmd = new SqlCommand();

    public db()
    {

    }

    public void openConn()
    {
        if (con.State == System.Data.ConnectionState.Open)
        {
            con.Close();
            con.Open();
        }
        else
        {
            con.Open();
        }
    }

    public void insertData(string query)
    {
        try
        {
            cmd.Dispose();
```

```

        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        cmd.ExecuteNonQuery();
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
    }
    finally
    {
        con.Close();
    }
}

```

```

public void deleteData(string query)
{
    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        cmd.ExecuteNonQuery();
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
    }
    finally
    {
        con.Close();
    }
}

```

```

public void updateData(string query)
{
    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        cmd.ExecuteNonQuery();
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
    }
    finally
    {

```

```

        con.Close();
    }
}

public string latestId(string table, string column)
{
    string id = "error";
    try
    {
        string query = "select max(" + column + ") + 1 from " + table + "";
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        if (cmd.ExecuteScalar().ToString().Equals(""))
        {
            id = "1";
        }
        else
        {
            id = cmd.ExecuteScalar().ToString();
        }
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
    }
    finally
    {
        con.Close();
    }
    return id;
}

public Dictionary<string, string> fillDropdown(string table, string valueCol, string displayCol)
{
    Dictionary<string, string> list = new Dictionary<string, string>();

    string query = "select " + valueCol + "," + displayCol + " from " + table + "";

    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        SqlDataReader dr = cmd.ExecuteReader();
        while (dr.Read())
        {

```

```

        list.Add(dr[0].ToString(), dr[1].ToString());
    }
}
catch (Exception x)
{
    Console.WriteLine(x);
}
finally
{
    con.Close();
}
return list;
}

public Dictionary<string, string> fillDropdown(string table, string valueCol, string displayCol, string
compareCol, string compareValueId)
{
    Dictionary<string, string> list = new Dictionary<string, string>();

    string query = "select " + valueCol + "," + displayCol + " from " + table + " where " + compareCol + " = "
+ compareValueId;

    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        SqlDataReader dr = cmd.ExecuteReader();
        while (dr.Read())
        {
            list.Add(dr[0].ToString(), dr[1].ToString());
        }
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
    }
    finally
    {
        con.Close();
    }
    return list;
}

public string checkCompareDropdown(string tableName, string column1, string valueColumn1, string
column2, string valueColumn2)
{
    //returns fail if no record found and returns pass if record exist
    string flag = "fail";

```

```

try
{
    cmd.Dispose();
    cmd.Connection = con;
    cmd.CommandText = "select * from " + tableName + " where " + column1 + "=" + valueColumn1 + "
and " + column2 + "=" + valueColumn2 + """;
    openConn();
    SqlDataReader dr = cmd.ExecuteReader();
    if (dr.HasRows)
    {
        dr.Read();
        flag = "pass";
    }
}
catch (Exception x)
{
    Console.WriteLine(x);
}
finally
{
    con.Close();
}
return flag;
}

```

```

public string comparevalidateuser(string tableName, string columnName, string
valueUserNameToCompare)
{
    //returns fail:null:null if no user found and returns pass:id:role if user exist
    string flag = "fail:null:null";
    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = "select * from " + tableName + " where " + columnName + "=" +
valueUserNameToCompare + """;
        openConn();
        SqlDataReader dr = cmd.ExecuteReader();
        if (dr.HasRows)
        {
            dr.Read();
            flag = "pass:" + dr[0].ToString() + ":" + dr[1].ToString();
        }
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
    }
    finally
    {
        con.Close();
    }
}

```

```

    }
    return flag;
}

public DataSet selectData(string query)
{
    DataSet dsdata = new DataSet();

    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        SqlDataAdapter da = new SqlDataAdapter(cmd);
        da.Fill(dsdata);
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
    }
    finally
    {
        con.Close();
    }
    return dsdata;
}

public Boolean checkAvailablity(string TableName, String ColumnName, String ValueToCompare)
{
    Boolean flag = true;
    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = "select " + ColumnName + " from " + TableName + " where " + ColumnName +
" = '" + ValueToCompare + "'";
        openConn();
        SqlDataReader dr = cmd.ExecuteReader();
        if (dr.HasRows)
        {
            flag = true;
        }
        else
        {
            flag = false;
        }
    }
}

```



```

        catch (Exception x)
        {
            Console.WriteLine(x);
        }
        finally
        {
            con.Close();
        }

        return flag;
    }

    public string sliderDetails(string table, string ImageCol, string DescCol, string TitleCol, string GroupCol,
    string hlink, string grpvalue)
    {
        // if tourTypeId is = something give id else pass is not null
        string html = "";
        try
        {
            string query = "select adv_id , " + ImageCol + " , " + TitleCol + " , " + DescCol + " , " + hlink + " from " + table + " where " + GroupCol + " = " + grpvalue + """;
            //html = query;
            cmd.Dispose();
            cmd.Connection = con;
            cmd.CommandText = query;
            openConn();
            SqlDataReader dr = cmd.ExecuteReader();
            if (dr.HasRows)
            {
                while (dr.Read())
                {
                    string title = "";
                    title = "<b>" + dr[2].ToString() + "</b><br><i>" + dr[3].ToString() + "</i><br><a href='" + dr[4].ToString() + ">Read More</a>";
                    html += "<li><img src='images/' + dr[1].ToString() + " title=\"'" + title + "\"/></li>";
                }
            }
        }
        catch (Exception x)
        {
            Console.WriteLine(x);
        }
        finally
        {
            con.Close();
        }
        return html;
    }
}

```

```

public string VsliderDetails(string table, string headCol, string DescCol, string LinkCol)
{
    // if tourTypeId is = something give id else pass is not null
    string html = "";
    try
    {
        string query = "select " + headCol + "," + DescCol + "," + LinkCol + " from " + table + "";
        html = query;
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        SqlDataReader dr = cmd.ExecuteReader();
        if (dr.HasRows)
        {
            string title = "";
            while (dr.Read())
            {
                title += "<li><div class=\"vscrollHeight\"><b>" + dr[0] + "</b><br><br>" +
                    "<i>" + dr[1] + "</i><br><br><a href=\"" + dr[2] + "\">Read More</a><hr></div></li>";
            }
            html = title;
        }
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
        html = x.ToString();
    }
    finally
    {
        con.Close();
    }
    return html;
}

public string getSingleRow(string query)
{
    string ret = "";
    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        SqlDataReader dr = cmd.ExecuteReader();
        if (dr.HasRows)
        {
            dr.Read();

```

```

        for (int i = 0; i < dr.FieldCount; i++)
        {
            ret += dr[i].ToString() + ":";
        }
    }
}
catch (Exception x)
{
    Console.WriteLine(x);
}
finally
{
    con.Close();
}
return ret;
}

public string GetSingleRow(string query)
{
    string ret = "";
    try
    {
        cmd.Dispose();
        cmd.Connection = con;
        cmd.CommandText = query;
        openConn();
        SqlDataReader dr = cmd.ExecuteReader();
        if (dr.HasRows)
        {
            dr.Read();
            for (int i = 0; i < dr.FieldCount; i++)
            {
                ret += dr[i].ToString() + ":";
            }
        }
    }
    catch (Exception x)
    {
        Console.WriteLine(x);
    }
    finally
    {
        con.Close();
    }
    return ret;
}

public void sendmail(string rec, string subject, string body)
{

```

```
// before updating the code for yours gmail username and password perform one step in gmail account
// https://www.google.com/settings/security/lesssecureapps
// Access for less secure apps select --> Turn on
```

```
// also add namespaces in db.cs file and add this method in db.cs file
```

```
//using System.Net;
//using System.Net.Mail;
```

```
string un = "queryresolver123@gmail.com";
string pas = "query@123";
MailMessage message = new MailMessage();
message.From = new MailAddress(un);
    message.To.Add(rec);
    message.Subject = subject;
    message.IsBodyHtml = true;
message.Body = body;
```

```
var client = new SmtpClient("smtp.gmail.com", 587)
{
    Credentials = new NetworkCredential(un, pas),
    EnableSsl = true,
    //DeliveryMethod = SmtpDeliveryMethod.PickupDirectoryFromIis
};

client.Send(message);
//Console.WriteLine("Sent");
//Console.ReadLine();
}
```

```
}
```

## admin\_addproduct.aspx

```
<% @ Page Title="" Language="C#" MasterPageFile="~/admin.master" AutoEventWireup="true"
CodeFile="admin_addproduct.aspx.cs" Inherits="addproduct" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
<link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
```

```
<div class="row">
  <div class="col-md-3"></div>
  <div class="col-md-6">
    <div class="panel-group">
      <div class="panel panel-primary">
        <div class="panel-heading"><center><b>Add Product</b></center></div>
        <div class="panel-body">
          <table class="table">
            <tr><td><b>PRODUCT ID</b></td><td>
              <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
            </td></tr>
            <tr><td><b>PRODUCT NAME</b></td><td>
              <asp:TextBox ID="TextBox1" runat="server"
                CssClass="form-control" Width="216px"></asp:TextBox>
              <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                ControlToValidate="TextBox1" ErrorMessage="*" ForeColor="Red"
                Font-Bold="True"></asp:RequiredFieldValidator>
            </td></tr>
            <tr><td>
              <b>PRODUCT MRP</b></td><td>
              <asp:TextBox ID="TextBox2" runat="server" TextMode="Number"
                CssClass="form-control"></asp:TextBox>
            </td></tr>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

</td><td>
  <asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
    ControlToValidate="TextBox2" ErrorMessage="*" ForeColor="Red"
    Font-Bold="True"></asp:RequiredFieldValidator>
</td></tr>
<tr><td>
  <b>PRODUCT&nbsp;&#39;S PRICE</b></td><td>
    <asp:TextBox ID="TextBox3" runat="server" TextMode="Number"
      CssClass="form-control"></asp:TextBox>
  </td><td>
    <asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server"
      ControlToValidate="TextBox3" ErrorMessage="*" ForeColor="Red"
      Font-Bold="True"></asp:RequiredFieldValidator>
  </td></tr>
<tr><td>
  <b>PRODUCT&nbsp;&#39;S&nbsp;&nbsp;&nbsp;PRICE</b></td><td>
    <asp:TextBox ID="TextBox4" runat="server" TextMode="Number"
      CssClass="form-control"></asp:TextBox>
  </td><td>
    <asp:RequiredFieldValidator ID="RequiredFieldValidator4" runat="server"
      ControlToValidate="TextBox4" ErrorMessage="*" ForeColor="Red"
      Font-Bold="True"></asp:RequiredFieldValidator>
  </td></tr>
<tr><td>
  <b>PRODUCT TYPE</b></td><td>
    <asp:DropDownList ID="DropDownList1" runat="server"
      DataTextField="type_name"
      DataValueField="type_id"
      CssClass="form-control" DataSourceID="SqlDataSource2">
    </asp:DropDownList>

    <asp:SqlDataSource ID="SqlDataSource2" runat="server"
      ConnectionString="<%$ ConnectionStrings:invconn %>"
      SelectCommand="SELECT * FROM [tbl_type]"></asp:SqlDataSource>

  </td><td>
    <asp:RequiredFieldValidator ID="RequiredFieldValidator5" runat="server"
      ControlToValidate="DropDownList1" ErrorMessage="*" ForeColor="Red"
      Font-Bold="True"></asp:RequiredFieldValidator>
  </td></tr>
<tr><td>
  <b>&nbsp;&nbsp;&nbsp;ENABLE</b></td>

  <td>
    <asp:DropDownList ID="DropDownList4" runat="server" CssClass="form-control-sm dropdown-
toggle-split rounded">
      <asp:ListItem Value="Y">ENABLE</asp:ListItem>
      <asp:ListItem Value="N">DISABLE</asp:ListItem>
    </asp:DropDownList>
  </td>
</tr>

```

[illegible]

```

</Columns>
</asp:GridView>
<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%%$ ConnectionStrings:invconn %>"
    DeleteCommand="DELETE FROM [tbl_products] WHERE [product_id] = @product_id"
    InsertCommand="INSERT INTO [tbl_products] ([product_id], [product_name], [product_mrp],
[product_vprice], [product_distprice], [product_type_id], [enable]) VALUES (@product_id, @product_name,
@product_mrp, @product_vprice, @product_distprice, @product_type_id, @enable)"
    SelectCommand="SELECT * FROM [tbl_products]"
    UpdateCommand="UPDATE [tbl_products] SET [product_name] = @product_name, [product_mrp]
= @product_mrp, [product_vprice] = @product_vprice, [product_distprice] = @product_distprice,
[product_type_id] = @product_type_id, [enable] = @enable WHERE [product_id] = @product_id">
    <DeleteParameters>
        <asp:Parameter Name="product_id" Type="Decimal" />
    </DeleteParameters>
    <InsertParameters>
        <asp:Parameter Name="product_id" Type="Decimal" />
        <asp:Parameter Name="product_name" Type="String" />
        <asp:Parameter Name="product_mrp" Type="Decimal" />
        <asp:Parameter Name="product_vprice" Type="Decimal" />
        <asp:Parameter Name="product_distprice" Type="Decimal" />
        <asp:Parameter Name="product_type_id" Type="Decimal" />
        <asp:Parameter Name="enable" Type="String" />
    </InsertParameters>
    <UpdateParameters>
        <asp:Parameter Name="product_name" Type="String" />
        <asp:Parameter Name="product_mrp" Type="Decimal" />
        <asp:Parameter Name="product_vprice" Type="Decimal" />
        <asp:Parameter Name="product_distprice" Type="Decimal" />
        <asp:Parameter Name="product_type_id" Type="Decimal" />
        <asp:Parameter Name="enable" Type="String" />
        <asp:Parameter Name="product_id" Type="Decimal" />
    </UpdateParameters>
</asp:SqlDataSource>

<br />
</div>
</div>
</div>
</div>
</div>
</div>

```



## admin\_addproduct.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class addproduct : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = d.latestId("tbl_products", "product_id");
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        d.insertData("insert into tbl_products values(" + d.latestId("tbl_products", "product_id") + "," +
        TextBox1.Text + "," + TextBox2.Text + "," + TextBox3.Text + "," + TextBox4.Text + "," +
        DropDownList1.SelectedValue + "," + DropDownList4.SelectedValue + ")");
        Response.Redirect("admin_addproducts.aspx");
    }
}
```

## admin\_addrole.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/admin.master" AutoEventWireup="true"
CodeFile="admin_addrole.aspx.cs" Inherits="img_admin_addrole" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
<link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<div class="row">
  <div class="col-md-3"></div>
  <div class="col-md-6">
    <div class="panel-group">
      <div class="panel panel-primary">
        <div class="panel-heading"><center><b>Add Role</b></center></div>
        <div class="panel-body">
          <table class="table">
            <tr><td><b>Role ID</b></td>
              <td class="style1">
                <asp:Label ID="Label2" runat="server" Text="Label"></asp:Label>
              </td></tr>
            <tr><td><b>Role Name</b></td>
              <td class="style1">
                <asp:TextBox ID="TextBox1" runat="server" CssClass="form-control"></asp:TextBox>
                <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                  ControlToValidate="TextBox1" ErrorMessage="*" Font-Bold="True"
                  ForeColor="Red"></asp:RequiredFieldValidator>
              </td></tr>
            <tr><td>&nbsp;</td><td class="style1">
                <asp:Button ID="Button1" runat="server" Text="ADD"
                  CssClass="btn btn-primary btn-block" onclick="Button1_Click" />
              </td></tr>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>
```

```
<div class="col-md-3">
</div>
<br />
</div>
&nbsp;<br />
<div class="row">
<div class="col-md-3"></div>
<div class="col-md-6">
```

```

<div class="panel-group">
<div class="panel panel-primary">
  <div class="panel-heading"><center><b>Show Roles</b></center></div>
  <div class="panel-body">
    <asp:GridView ID="GridView1" runat="server" CssClass="table table-bordered table-striped"
AutoGenerateColumns="False"
    DataKeyNames="role_id" DataSourceID="SqlDataSource1">
      <Columns>
        <asp:CommandField ShowEditButton="True" />
        <asp:BoundField DataField="role_id" HeaderText="role_id" ReadOnly="True"
SortExpression="role_id" />
        <asp:BoundField DataField="role_name" HeaderText="role_name"
SortExpression="role_name" />
      </Columns>
    </asp:GridView>
    <asp:SqlDataSource ID="SqlDataSource1" runat="server"
ConnectionString="<%$ ConnectionStrings:invconn %>"
DeleteCommand="DELETE FROM [tbl_role] WHERE [role_id] = @role_id"
InsertCommand="INSERT INTO [tbl_role] ([role_id], [role_name]) VALUES (@role_id,
@role_name)"
SelectCommand="SELECT * FROM [tbl_role]"
UpdateCommand="UPDATE [tbl_role] SET [role_name] = @role_name WHERE [role_id] =
@role_id">
      <DeleteParameters>
        <asp:Parameter Name="role_id" Type="Decimal" />
      </DeleteParameters>
      <InsertParameters>
        <asp:Parameter Name="role_id" Type="Decimal" />
        <asp:Parameter Name="role_name" Type="String" />
      </InsertParameters>
      <UpdateParameters>
        <asp:Parameter Name="role_name" Type="String" />
        <asp:Parameter Name="role_id" Type="Decimal" />
      </UpdateParameters>
    </asp:SqlDataSource>
    <br />
  </div>
</div>
</div>
</div>
</div>
<div class="col-md-3"></div>
</asp:Content>

```

## **admin\_addrole.cs**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class img_admin_addrole : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {
        Label2.Text = d.latestId("tbl_role", "role_id");
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        d.insertData("insert into tbl_role values(" + d.latestId("tbl_role", "role_id") + ",'" + TextBox1.Text + "'");
        Response.Redirect("admin_addrole.aspx");
    }
}
```

## admin\_addshops.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/admin.master" AutoEventWireup="true"
CodeFile="admin_addshops.aspx.cs" Inherits="addshops" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
<link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
```

```
<div class="row">
  <div class="col-md-3"></div>
  <div class="col-md-6">
    <div class="panel-group">
      <div class="panel panel-primary">
        <div class="panel-heading"><center><b>Add Shop</b></center></div>
        <div class="panel-body">
          <table style="width:100%;" class="table">
            <tr>
              <td>
                <b>SHOPID</b> </td>
              <td>
                <asp:Label ID="Label1" runat="server" Text="Label" ></asp:Label>
              </td>
              <td>
                &nbsp;</td>
            </tr>
            <tr>
              <td><b>SHOP NAME</b>
              </td>
              <td>
                <asp:TextBox ID="TextBox1" runat="server" CssClass="form-control"></asp:TextBox>
              </td>
              <td>
                <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                  ControlToValidate="TextBox1" ErrorMessage="*" Font-Bold="True"
                  ForeColor="Red"></asp:RequiredFieldValidator>
              </td>
            </tr>
            <tr>
              <td><b>SHOP ADDRESS</b>
              </td>
              <td>
                <asp:TextBox ID="TextBox2" runat="server" Height="116px" TextMode="MultiLine"
                  CssClass="form-control"></asp:TextBox>
              </td>
            </tr>
```

```

        <td>
            <asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
                ControlToValidate="TextBox2" ErrorMessage="*" Font-Bold="True"
ForeColor="Red"></asp:RequiredFieldValidator>
        </td>
    </tr>
    <tr>
        <td><b>SHOP PHONE NO</b>
        </td>
        <td>
            <asp:TextBox ID="TextBox3" runat="server" TextMode="Number"
                CssClass="form-control"></asp:TextBox>
        </td>
        <td>
            <asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server"
                ControlToValidate="TextBox3" ErrorMessage="*" Font-Bold="True"
ForeColor="Red"></asp:RequiredFieldValidator>
        </td>
    </tr>
    <tr>
        <td><b>OWNER NAME</b>
        </td>
        <td>
            <asp:TextBox ID="TextBox4" runat="server" CssClass="form-control"></asp:TextBox>
        </td>
        <td>
            <asp:RequiredFieldValidator ID="RequiredFieldValidator4" runat="server"
                ControlToValidate="TextBox4" ErrorMessage="*" Font-Bold="True"
ForeColor="Red"></asp:RequiredFieldValidator>
        </td>
    </tr>
    <tr>
        <td><b>USERNAME</b>
        </td>
        <td>
            <asp:TextBox ID="TextBox5" runat="server" CssClass="form-control"></asp:TextBox>
        </td>
        <td>
            <asp:RequiredFieldValidator ID="RequiredFieldValidator5" runat="server"
                ControlToValidate="TextBox5" ErrorMessage="*" Font-Bold="True"
ForeColor="Red"></asp:RequiredFieldValidator>
        </td>
    </tr>
    <tr>
        <td><b>PASSWORD</b>
        </td>
        <td>
            <asp:TextBox ID="TextBox6" runat="server" CssClass="form-control"
                TextMode="Password" ></asp:TextBox>
        </td>
        <td>
    
```

[illegible]

```

        <asp:BoundField DataField="password" HeaderText="password"
            SortExpression="password" />
    </Columns>
</asp:GridView>

<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%$ ConnectionStrings:invconn %>"
    DeleteCommand="DELETE FROM [tbl_shop] WHERE [shop_id] = @shop_id"
    InsertCommand="INSERT INTO [tbl_shop] ([shop_id], [shop_name], [shop_address],
[shop_phoneno], [owners_name], [username], [password]) VALUES (@shop_id, @shop_name,
@shop_address, @shop_phoneno, @owners_name, @username, @password)"
    SelectCommand="SELECT * FROM [tbl_shop]"
    UpdateCommand="UPDATE [tbl_shop] SET [shop_name] = @shop_name, [shop_address] =
@shop_address, [shop_phoneno] = @shop_phoneno, [owners_name] = @owners_name, [username] =
@username, [password] = @password WHERE [shop_id] = @shop_id">
    <DeleteParameters>
        <asp:Parameter Name="shop_id" Type="Decimal" />
    </DeleteParameters>
    <InsertParameters>
        <asp:Parameter Name="shop_id" Type="Decimal" />
        <asp:Parameter Name="shop_name" Type="String" />
        <asp:Parameter Name="shop_address" Type="String" />
        <asp:Parameter Name="shop_phoneno" Type="Decimal" />
        <asp:Parameter Name="owners_name" Type="String" />
        <asp:Parameter Name="username" Type="String" />
        <asp:Parameter Name="password" Type="String" />
    </InsertParameters>
    <UpdateParameters>
        <asp:Parameter Name="shop_name" Type="String" />
        <asp:Parameter Name="shop_address" Type="String" />
        <asp:Parameter Name="shop_phoneno" Type="Decimal" />
        <asp:Parameter Name="owners_name" Type="String" />
        <asp:Parameter Name="username" Type="String" />
        <asp:Parameter Name="password" Type="String" />
        <asp:Parameter Name="shop_id" Type="Decimal" />
    </UpdateParameters>
</asp:SqlDataSource>

    <br />
</div>
</div>
</div>
</div>
</div>
<div class="col-md-3"></div>

</div>
</asp:Content>

```



### **admin\_addshops.cs**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class addshops : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = d.latestId("tbl_shop", "shop_id");
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        if (d.checkAvailablity("tbl_shop", "shop_phoneno", TextBox3.Text))
        {
            Response.Write("Shop already added!");
        }
        else
        {
            d.insertData("insert into tbl_shop values(" + d.latestId("tbl_shop", "shop_id") + "," + TextBox1.Text +
            "," + TextBox2.Text + "," + TextBox3.Text + "," + TextBox4.Text + "," + TextBox5.Text + "," +
            TextBox6.Text + ")");
            Response.Redirect("admin_addshop.aspx");
        }
    }
}
```

## admin\_addtype.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/admin.master" AutoEventWireup="true"
CodeFile="admin_addtype.aspx.cs" Inherits="admin_addtype" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
<link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<div class="row">
  <div class="col-md-3"></div>
  <div class="col-md-6">
    <div class="panel-group">
      <div class="panel panel-primary">
        <div class="panel-heading"><center><b>Add Type</b></center></div>
        <div class="panel-body">
          <table class="table">
            <tr><td><b>TYPE ID</b></td><td>
              <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label>
            </td><td></td></tr>
            <tr><td><b>TYPE NAME</b></td><td>
              <asp:TextBox ID="TextBox1" runat="server"
                CssClass="form-control"></asp:TextBox>
              <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                ControlToValidate="TextBox1" ErrorMessage="*" Font-Bold="True"
                ForeColor="Red"></asp:RequiredFieldValidator>
            </td><td></td></tr>
            <tr><td>&nbsp;</td><td>
              <asp:Button ID="Button1" runat="server" Text="ADD"
                CssClass=" btn btn-primary btn-block" onclick="Button1_Click" />
            </td><td>&nbsp;</td></tr>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>

<div class="col-md-3">
</div>
<br />
```

```
<div class="row">
<div class="col-md-3"></div>
<div class="col-md-6">
  <div class="panel-group">
    <div class="panel panel-primary">
```

```

<div class="panel-heading"><center><b>Show Type</b></center></div>
<div class="panel-body">

<asp:GridView ID="GridView1" runat="server"
    CssClass="table table-bordered table-striped" AutoGenerateColumns="False"
    DataKeyNames="type_id" DataSourceID="SqlDataSource1">
    <Columns>
        <asp:CommandField ShowEditButton="True" />
        <asp:BoundField DataField="type_id" HeaderText="type_id" ReadOnly="True"
            SortExpression="type_id" />
        <asp:BoundField DataField="type_name" HeaderText="type_name"
            SortExpression="type_name" />
    </Columns>
</asp:GridView>


<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%= $ConnectionStrings:invconn %>"
    DeleteCommand="DELETE FROM [tbl_type] WHERE [type_id] = @type_id"
    InsertCommand="INSERT INTO [tbl_type] ([type_id], [type_name]) VALUES (@type_id,
@type_name)"
    SelectCommand="SELECT * FROM [tbl_type]"
    UpdateCommand="UPDATE [tbl_type] SET [type_name] = @type_name WHERE [type_id] =
@type_id">
    <DeleteParameters>
        <asp:Parameter Name="type_id" Type="Decimal" />
    </DeleteParameters>
    <InsertParameters>
        <asp:Parameter Name="type_id" Type="Decimal" />
        <asp:Parameter Name="type_name" Type="String" />
    </InsertParameters>
    <UpdateParameters>
        <asp:Parameter Name="type_name" Type="String" />
        <asp:Parameter Name="type_id" Type="Decimal" />
    </UpdateParameters>
</asp:SqlDataSource>


<br />
</div>
</div>
</div>
</div>
<div class="col-md-3"></div>

</asp:Content>

```

### **admin\_addtype.cs**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class admin_addtype : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = d.latestId("tbl_type", "type_id");
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        d.insertData("insert into tbl_type values(" + d.latestId("tbl_type", "type_id") + ", '" + TextBox1.Text + "')");
        Response.Redirect("admin_addtype.aspx");
    }
}
```

## admin\_addvendor.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/admin.master" AutoEventWireup="true"
CodeFile="admin_addvendor.aspx.cs" Inherits="admin_addvendor" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
<link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<div class="row">
  <div class="col-md-3"></div>
  <div class="col-md-6">
    <div class="panel-group">
      <div class="panel panel-primary">
        <div class="panel-heading"><center><b>Add Vendor</b></center></div>
        <div class="panel-body">
          <table style="width:100%;" class="table">
            <tr>
              <td>
                <b>VENDOR ID</b></td>
              <td>
                <asp:Label ID="Label1" runat="server" Text="Label" CssClass="form-control"></asp:Label>
              </td>
            <tr>
              <td>
                &nbsp;</td>
            </tr>
            <tr>
              <td>
                <b>VENDOR NAME</b></td>
              <td>
                <asp:TextBox ID="TextBox1" runat="server"
                  TextMode="SingleLine" CssClass="form-control"></asp:TextBox>
              </td>
              <td>
                <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                  ControlToValidate="TextBox1" ErrorMessage="*" Font-Bold="True"
                  ForeColor="Red"></asp:RequiredFieldValidator>
              </td>
            </tr>
            <tr>
              <td>
                <b>VENDOR ADDRESS</b>
              </td>
              <td>
                <asp:TextBox ID="TextBox2" runat="server" Height="72px" TextMode="MultiLine"
                  Width="208px" CssClass="form-control" ></asp:TextBox>
              </td>
              <td>
                <asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
                  ControlToValidate="TextBox2" ErrorMessage="*" Font-Bold="True"
                  ForeColor="Red"></asp:RequiredFieldValidator>
              </td>
            </tr>
          </table>
        </div>
      </div>
    </div>
  </div>
</div>
```

```

<tr>
  <td><b>VENDOR PHONE NO</b>
  </td>
  <td>
    <asp:TextBox ID="TextBox3" runat="server" CssClass="form-control"
      TextMode="Number"></asp:TextBox>
  </td>
  <td>
    <asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server"
      ControlToValidate="TextBox3" ErrorMessage="*" Font-Bold="True"
      ForeColor="Red"></asp:RequiredFieldValidator>
  </td>
</tr>
<tr>
  <td>
    &nbsp;  </td>
  <td>
    <asp:Button ID="Button1" runat="server" onclick="Button1_Click" Text="ADD" CssClass="btn
    btn-primary btn-block" />
  </td>
  <td>
    &nbsp;  </td>
</tr>
</table>
</div>
</div>
</div>
</div>

```

```

<div class="col-md-3">
</div>
<br />
</div>
&nbsp;  <br />

```

```

<div class="row">
<div class="col-md-3"></div>
<div class="col-md-6">
  <div class="panel-group">
    <div class="panel panel-primary">
      <div class="panel-heading"><center><b>Show Vendors</b></center></div>
      <div class="panel-body">

```

```

<asp:GridView ID="GridView1" runat="server"
  CssClass="table table-bordered table-striped" AutoGenerateColumns="False"
  DataKeyNames="vendor_id" DataSourceID="SqlDataSource1">
  <Columns>
    <asp:CommandField ShowEditButton="True" />
    <asp:BoundField DataField="vendor_id" HeaderText="Ven_Id" ReadOnly="True"

```

```

        SortExpression="vendor_id" />
        <asp:BoundField DataField="vendor_name" HeaderText="Ven_Name"
            SortExpression="vendor_name" />
        <asp:BoundField DataField="vendor_address" HeaderText="Ven_Addr"
            SortExpression="vendor_address" />
        <asp:BoundField DataField="vendor_phoneno" HeaderText="Vend_PhNo"
            SortExpression="vendor_phoneno" />
    </Columns>
</asp:GridView>

```

```

<asp:SqlDataSource ID="SqlDataSource1" runat="server"
    ConnectionString="<%$ ConnectionStrings:invconn %>"
    DeleteCommand="DELETE FROM [tbl_vendor] WHERE [vendor_id] = @vendor_id"
    InsertCommand="INSERT INTO [tbl_vendor] ([vendor_id], [vendor_name], [vendor_address],
[vendor_phoneno]) VALUES (@vendor_id, @vendor_name, @vendor_address, @vendor_phoneno)"
    SelectCommand="SELECT * FROM [tbl_vendor]"
    UpdateCommand="UPDATE [tbl_vendor] SET [vendor_name] = @vendor_name, [vendor_address]
= @vendor_address, [vendor_phoneno] = @vendor_phoneno WHERE [vendor_id] = @vendor_id">
    <DeleteParameters>
        <asp:Parameter Name="vendor_id" Type="Decimal" />
    </DeleteParameters>
    <InsertParameters>
        <asp:Parameter Name="vendor_id" Type="Decimal" />
        <asp:Parameter Name="vendor_name" Type="String" />
        <asp:Parameter Name="vendor_address" Type="String" />
        <asp:Parameter Name="vendor_phoneno" Type="Decimal" />
    </InsertParameters>
    <UpdateParameters>
        <asp:Parameter Name="vendor_name" Type="String" />
        <asp:Parameter Name="vendor_address" Type="String" />
        <asp:Parameter Name="vendor_phoneno" Type="Decimal" />
        <asp:Parameter Name="vendor_id" Type="Decimal" />
    </UpdateParameters>
</asp:SqlDataSource>

```

```

<br />
</div>

```

```

</div>
</div>
</div>

```

```

</div>
<div class="col-md-3"></div>

```

```

</asp:Content>

```

### **admin\_addvendor.cs**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class admin_addvendor : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {
        Label1.Text = d.latestId("tbl_vendor", "vendor_id");
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        if (d.checkAvailablity("tbl_vendor", "vendor_phoneno", TextBox3.Text))
        {
        }
        else
        {
            d.insertData("insert into tbl_vendor values(" + d.latestId("tbl_vendor", "vendor_id") + ", " +
            TextBox1.Text + ", " + TextBox2.Text + ", " + TextBox3.Text + ") ");
            Response.Redirect("addvendor.aspx");
        }
    }
}
```



## generategrn.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/grn.master" AutoEventWireup="true"
CodeFile="generategrn.aspx.cs" Inherits="generategrn" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
    <link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
    <div class="row">
        <div class="col-md-3"></div>
        <div class="col-md-6">
            <div class="panel-group">
                <div class="panel panel-primary">
                    <div class="panel-heading">GRN</div>
                    <div class="panel-body">
                        <table class="table">
                            <tr><td><b>Select GRN</b></td><td>
                                <asp:DropDownList ID="DropDownList1"
                                    runat="server" AutoPostBack="True"
                                    DataSourceID="SqlDataSource3"
                                    DataValueField="purchase_id" CssClass="form-control" DataTextField="Expr1">
                                </asp:DropDownList>

                                <asp:SqlDataSource ID="SqlDataSource3" runat="server"
                                    ConnectionString="<%$ ConnectionStrings:invconn %>"
                                    SelectCommand="SELECT CONVERT (varchar(30), purchase_id) + ' :: ' + dated AS Expr1,
purchase_id FROM tbl_purchase_order WHERE (purchase_id NOT IN (SELECT purchase_id FROM
tbl_grn))SELECT CONVERT (varchar(30), purchase_id) + ' :: ' + dated AS Expr1, purchase_id FROM
tbl_purchase_order WHERE (purchase_id NOT IN (SELECT purchase_id FROM tbl_grn))">
                                </asp:SqlDataSource>

                                </td><td>
                                    <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                                        ControlToValidate="DropDownList1" ErrorMessage="*" Font-Bold="True"
                                        ForeColor="Red"></asp:RequiredFieldValidator>
                                </td></tr>

                            <tr><td><td>
                                <asp:Button ID="Button1" runat="server" CssClass=" btn btn-primary btn-block"
                                    Text="Create Grn" onclick="Button1_Click1" /></td></td></tr>
                        </table>

                    </div>
                </div>
            </div>
        </div>
    </div>
</div>
```

```

<div class="col-md-3">
</div>
<br />
<div class="row">
<div class="col-md-3"></div>
<div class="col-md-6">
<div class="panel-group">
<div class="panel panel-primary">
<div class="panel-heading"><center><b>Show GRN</b></center></div>
<div class="panel-body">

<asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
  CssClass="table" DataKeyNames="product_id" DataSourceID="SqlDataSource2">
  <Columns>
    <asp:BoundField DataField="product_id" HeaderText="product_id" ReadOnly="True"
      SortExpression="product_id" />
    <asp:BoundField DataField="product_name" HeaderText="product_name"
      SortExpression="product_name" />
    <asp:BoundField DataField="product_mrp" HeaderText="product_mrp"
      SortExpression="product_mrp" />
    <asp:BoundField DataField="product_vprice" HeaderText="product_vprice"
      SortExpression="product_vprice" />
    <asp:BoundField DataField="product_distprice" HeaderText="product_distprice"
      SortExpression="product_distprice" />
    <asp:TemplateField HeaderText="qty" SortExpression="qty">
      <EditItemTemplate>
        <asp:TextBox ID="TextBox1" runat="server" Text="<%# Bind("qty") %>"></asp:TextBox>
      </EditItemTemplate>
      <ItemTemplate>
        <asp:TextBox ID="TextBox2" runat="server" AutoPostBack="True"
          ontextchanged="TextBox2_TextChanged1" Text="<%# Bind("qty") %>"></asp:TextBox>
      </ItemTemplate>
    </asp:TemplateField>
    <asp:BoundField DataField="total_price" HeaderText="total_price"
      SortExpression="total_price" />
  </Columns>
</asp:GridView>

<br />
<asp:SqlDataSource ID="SqlDataSource2" runat="server"
  ConnectionString="<%= $ ConnectionStrings:invconn %>"
  SelectCommand="SELECT tbl_products.product_name, tbl_products.product_mrp,
tbl_products.product_vprice, tbl_puchaseordered_items.qty, tbl_products.product_distprice,
tbl_puchaseordered_items.total_price, tbl_products.product_id FROM tbl_products INNER JOIN
tbl_puchaseordered_items ON tbl_products.product_id = tbl_puchaseordered_items.product_id WHERE
(tbl_puchaseordered_items.purchase_id = @par)">
  <SelectParameters>
    <asp:ControlParameter ControlID="DropDownList1" Name="par"
      PropertyName="SelectedValue" />
  </SelectParameters>
</asp:SqlDataSource>

```

```

        </div>
    </div>
</div>

</div>

</div>

</asp:Content>

```

#### generategrn.cs

```

using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class generategrn : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void TextBox2_TextChanged1(object sender, EventArgs e)
    {
        for (int i = 0; i < GridView1.Rows.Count; i++)
        {
            int qty = Convert.ToInt32(((TextBox)GridView1.Rows[i].FindControl("TextBox2")).Text);
            int vprice = Convert.ToInt32(GridView1.Rows[i].Cells[3].Text);
            int tot = qty * vprice;
            GridView1.Rows[i].Cells[6].Text = tot.ToString();
        }
    }
    protected void Button1_Click1(object sender, EventArgs e)
    {
        int totqty = 0;
        int totalprice = 0;
        for (int i = 0; i < GridView1.Rows.Count; i++)
        {
            int qty = Convert.ToInt32(((TextBox)GridView1.Rows[i].FindControl("TextBox2")).Text);

```

```

    int vprice = Convert.ToInt32(GridView1.Rows[i].Cells[3].Text);
    int tot = qty * vprice;
    totqty += qty;
    totalprice += tot;
}
string grnid = d.latestId("tbl_grn","grn_id");
string qrygrn = "insert into tbl_grn values(" + grnid + "," + totqty + "," + totalprice + "," +
DateTime.Now.ToString() + "," + DropDownList1.SelectedValue + ")";
d.insertData(qrygrn);
//-----
for (int i = 0; i < GridView1.Rows.Count; i++)
{
    int qty = Convert.ToInt32(((TextBox)GridView1.Rows[i].FindControl("TextBox2")).Text);
    int vprice = Convert.ToInt32(GridView1.Rows[i].Cells[3].Text);
    int pid = Convert.ToInt32(GridView1.Rows[i].Cells[0].Text);
    string qrygrnorder = "insert into tbl_grn_order values(" + d.latestId("tbl_grn_order", "id") + "," + pid +
"," + qty + "," + (qty * vprice) + "," + grnid + ")";
    d.insertData(qrygrnorder);
}
Response.Redirect("generategrn.aspx");
}
}

```

## prod\_distri.aspx

```
<%@ Page Title="" Language="C#" MasterPageFile="~/prod_distri.master" AutoEventWireup="true"
CodeFile="prod_distri.aspx.cs" Inherits="prod_distri" %>

<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
<link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<div class="row">
    <div class="col-md-3"></div>
    <div class="col-md-6">
        <div class="panel-group">
            <div class="panel panel-primary">
                <div class="panel-heading"><center><b>Distribution</b></center></div>
                <div class="panel-body">
                    <table class="table">
                        <tr><td><b>Select Shop</b></td><td><asp:DropDownList ID="DropDownList1" runat="server"
                            DataSourceID="SqlDataSource1" DataTextField="shop_name"
                            DataValueField="shop_id" CssClass="form-control">
                        </asp:DropDownList></td><td>
                            <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                                ControlToValidate="DropDownList1" ErrorMessage="*" Font-Bold="True"
                                ForeColor="Red"></asp:RequiredFieldValidator>
                        </td></tr>
                        <asp:SqlDataSource ID="SqlDataSource1" runat="server"
                            ConnectionString="<%"$ ConnectionStrings:invconn %>"
                            SelectCommand="SELECT [shop_id], [shop_name] FROM [tbl_shop]">
                        </asp:SqlDataSource>
                        <tr><td><b>Select Product</b></td><td><asp:DropDownList ID="DropDownList2" runat="server"
                            DataSourceID="SqlDataSource2" DataTextField="Expr1" DataValueField="product_id"
                            CssClass="form-control">
                        </asp:DropDownList></td><td>
                            <asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
                                ControlToValidate="DropDownList2" ErrorMessage="*" Font-Bold="True"
                                ForeColor="Red"></asp:RequiredFieldValidator>
                        </td></tr>
                        <asp:SqlDataSource ID="SqlDataSource2" runat="server"
                            ConnectionString="<%"$ ConnectionStrings:invconn %>"
                            SelectCommand="SELECT product_id, product_name + ':' + CONVERT (VARCHAR(30),
                                product_distprice) AS Expr1 FROM tbl_products">
                        </asp:SqlDataSource>
                        <tr><td><b>Enter Quantity</b></td><td><asp:TextBox ID="TextBox1" runat="server"
                            CssClass="form-control" TextMode="Number"></asp:TextBox></td><td>
                            <asp:RequiredFieldValidator ID="RequiredFieldValidator3" runat="server"
                                ControlToValidate="TextBox1" ErrorMessage="*" Font-Bold="True"
                                ForeColor="Red"></asp:RequiredFieldValidator>
                        </td></tr>
                        <tr><td><td><asp:Button ID="Button1" runat="server" CssClass="btn btn-primary btn-block"
                            Text="ADD TO SHOP" onclick="Button1_Click" /></td></td></tr>
                    </table>
                    <asp:GridView ID="GridView1" runat="server">
```

```

        </asp:GridView>
        <br />
        <tr><td><td><asp:Button ID="Button2" runat="server" CssClass="btn btn-primary btn-block"
            Text="Update Distribution" onclick="Button2_Click" /></td></td></tr>

    </div>
</div>
</div>
</div>
</div>

<div class="row">
<div class="col-md-3"></div>
<div class="col-md-6">
    <div class="panel-group">
    <div class="panel panel-primary">
        <div class="panel-heading"><center><b>Show Product Distribution</b></center></div>
        <div class="panel-body">
            <asp:GridView ID="GridView2" runat="server"
                CssClass="table table-bordered table-striped" AutoGenerateColumns="False"
                DataKeyNames="id" DataSourceID="SqlDataSource3">
                <Columns>
                    <asp:BoundField DataField="id" HeaderText="id" ReadOnly="True"
                        SortExpression="id" />
                    <asp:BoundField DataField="product_id" HeaderText="prod_id"
                        SortExpression="product_id" />
                    <asp:BoundField DataField="product_name" HeaderText="prod_name"
                        SortExpression="product_name" />
                    <asp:BoundField DataField="shop_id" HeaderText="shop_id"
                        SortExpression="shop_id" />
                    <asp:BoundField DataField="qty" HeaderText="qty" SortExpression="qty" />
                    <asp:BoundField DataField="total_price" HeaderText="total_price"
                        SortExpression="total_price" />
                    <asp:BoundField DataField="dist_id" HeaderText="dist_id"
                        SortExpression="dist_id" />
                    <asp:BoundField DataField="received" HeaderText="received"
                        SortExpression="received" />
                </Columns>
            </asp:GridView>
            <asp:SqlDataSource ID="SqlDataSource3" runat="server"
                ConnectionString="<%$ ConnectionStrings:invconn %>"
                DeleteCommand="DELETE FROM [tbl_dist_list] WHERE [id] = @id"
                InsertCommand="INSERT INTO [tbl_dist_list] ([id], [product_id], [product_name], [shop_id], [qty],
[total_price], [dist_id], [received]) VALUES (@id, @product_id, @product_name, @shop_id, @qty,
@total_price, @dist_id, @received)"
                SelectCommand="SELECT * FROM [tbl_dist_list]"
                UpdateCommand="UPDATE [tbl_dist_list] SET [product_id] = @product_id, [product_name] =
@product_name, [shop_id] = @shop_id, [qty] = @qty, [total_price] = @total_price, [dist_id] = @dist_id,
[received] = @received WHERE [id] = @id">
                <DeleteParameters>

```

```

    <asp:Parameter Name="id" Type="Decimal" />
</DeleteParameters>
<InsertParameters>
    <asp:Parameter Name="id" Type="Decimal" />
    <asp:Parameter Name="product_id" Type="Decimal" />
    <asp:Parameter Name="product_name" Type="String" />
    <asp:Parameter Name="shop_id" Type="Decimal" />
    <asp:Parameter Name="qty" Type="Decimal" />
    <asp:Parameter Name="total_price" Type="Decimal" />
    <asp:Parameter Name="dist_id" Type="Decimal" />
    <asp:Parameter Name="received" Type="Decimal" />
</InsertParameters>
<UpdateParameters>
    <asp:Parameter Name="product_id" Type="Decimal" />
    <asp:Parameter Name="product_name" Type="String" />
    <asp:Parameter Name="shop_id" Type="Decimal" />
    <asp:Parameter Name="qty" Type="Decimal" />
    <asp:Parameter Name="total_price" Type="Decimal" />
    <asp:Parameter Name="dist_id" Type="Decimal" />
    <asp:Parameter Name="received" Type="Decimal" />
    <asp:Parameter Name="id" Type="Decimal" />
</UpdateParameters>
</asp:SqlDataSource>
<br />
</div>
</div>
</div>

</div>
<div class="col-md-3"></div>

```

```

</asp:Content>

```

## prod\_distri.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;

public partial class prod_distri : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        if (ViewState["CurrentTable"] != null || GridView1.Rows.Count != 0)
        {
            DataTable dtCurrentTable = (DataTable)ViewState["CurrentTable"];
            DataRow drCurrentRow = null;
            drCurrentRow = dtCurrentTable.NewRow();
            drCurrentRow["SHOP ID"] = DropDownList1.SelectedValue;
            drCurrentRow["PRODUCT ID"] = DropDownList2.SelectedValue;
            drCurrentRow["PRODUCT NAME"] = DropDownList2.SelectedItem.Text.Split(':')[0];
            drCurrentRow["DISTRIBUTION PRICE"] = DropDownList2.SelectedItem.Text.Split(':')[1];
            drCurrentRow["QTY"] = TextBox1.Text;
            drCurrentRow["TOTAL PRICE"] = (Convert.ToInt32(DropDownList2.SelectedItem.Text.Split(':')[1])
* Convert.ToInt32(TextBox1.Text));
            dtCurrentTable.Rows.Add(drCurrentRow);
            ViewState["CurrentTable"] = dtCurrentTable;
            GridView1.DataSource = dtCurrentTable;
            GridView1.DataBind();
        }
        else
        {
            DataTable dt = new DataTable();
            DataRow dr = null;
            dt.Columns.Add(new DataColumn("SHOP ID", typeof(string)));
            dt.Columns.Add(new DataColumn("PRODUCT ID", typeof(string)));
            dt.Columns.Add(new DataColumn("PRODUCT NAME", typeof(string)));
            dt.Columns.Add(new DataColumn("DISTRIBUTION PRICE", typeof(string)));
            dt.Columns.Add(new DataColumn("QTY", typeof(string)));
            dt.Columns.Add(new DataColumn("TOTAL PRICE", typeof(string)));
            dr = dt.NewRow();
            dr["SHOP ID"] = DropDownList1.SelectedValue;
            dr["PRODUCT ID"] = DropDownList2.SelectedValue;
            dr["PRODUCT NAME"] = DropDownList2.SelectedItem.Text.Split(':')[0];
            dr["DISTRIBUTION PRICE"] = DropDownList2.SelectedItem.Text.Split(':')[1];
            dr["QTY"] = TextBox1.Text;
```



```

        dr["TOTAL PRICE"] = (Convert.ToInt32(DropDownList2.SelectedItem.Text.Split(':')[1]) *
Convert.ToInt32(TextBox1.Text));
        dt.Rows.Add(dr);
        //dr = dt.NewRow();
        //Store the DataTable in ViewState
        ViewState["CurrentTable"] = dt;
        GridView1.DataSource = dt;
        GridView1.DataBind();
    }
}

public string totalqtyingrid()
{
    int tot = 0;
    for (int i = 0; i < GridView1.Rows.Count; i++)
    {
        tot += Convert.ToInt32(GridView1.Rows[i].Cells[4].Text);
    }
    return tot.ToString();
}

public string totalpriceingrid()
{
    int tot = 0;
    for (int i = 0; i < GridView1.Rows.Count; i++)
    {
        tot += Convert.ToInt32(GridView1.Rows[i].Cells[5].Text);
    }
    return tot.ToString();
}

protected void Button2_Click(object sender, EventArgs e)
{
    string did = d.latestId("tbl_dist_list", "dist_id");
    d.insertData("insert into tbl_dist_list values(" + did + "," + totalqtyingrid() + "," + totalpriceingrid() + "," +
+ DateTime.Now.ToString() + ")");
    for (int i = 0; i < GridView1.Rows.Count; i++)
    {
        d.insertData("insert into tbl_dist_list values(" + d.latestId("tbl_dist_list", "id") + "," +
DropDownList2.SelectedValue + "," + DropDownList2.SelectedItem.Text.Split(':')[0] + "," +
DropDownList1.SelectedValue + "," + GridView1.Rows[i].Cells[4].Text + "," +
GridView1.Rows[i].Cells[5].Text + "," + did + ",0)");
    }
    Response.Redirect("prod_distri.aspx");
}
}

```

## **purchase\_purchase\_order.aspx**

```
<%@ Page Title="" Language="C#" MasterPageFile="~/purchase.master" AutoEventWireup="true"
CodeFile="purchase_purchase_order.aspx.cs" Inherits="purchase_purchase_order" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
<link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
<style type="text/css">
    .style1
    {
        width: 420px;
    }
</style>
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
<div class="row">
    <div class="col-md-3"></div>
    <div class="col-md-6">
        <div class="panel-group">
            <div class="panel panel-primary">
                <div class="panel-heading">Product</div>
                <div class="panel-body">
                    <table class="table">
                        <tr><td><b>Select Product</b></td>
                            <td class="style1">
                                <asp:DropDownList ID="DropDownList1" runat="server" CssClass="form-control"
                                    DataSourceID="SqlDataSource1" DataTextField="Proname"
                                    DataValueField="product_id">
                                </asp:DropDownList>
                                <td>
                                    <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                                        ControlToValidate="DropDownList1" ErrorMessage="*" Font-Bold="True"
                                        ForeColor="Red"></asp:RequiredFieldValidator>
                                </td>
                                <asp:SqlDataSource ID="SqlDataSource1" runat="server"
                                    ConnectionString="<%$ ConnectionStrings:invconn %>"
                                    SelectCommand="SELECT product_id, product_name+'.'+convert(varchar(10), product_vprice) as
Proname FROM tbl_products">
                                </asp:SqlDataSource>
                            </td></tr>
                        <tr><td><b>Enter Quantity</b></td>
                            <td class="style1">
                                <asp:TextBox ID="TextBox1" runat="server" CssClass="form-control"
                                    TextMode="Number"></asp:TextBox>
                                <td>
                                    <asp:RequiredFieldValidator ID="RequiredFieldValidator2" runat="server"
                                        ControlToValidate="TextBox1" ErrorMessage="*" Font-Bold="True"
                                        ForeColor="Red"></asp:RequiredFieldValidator>
                                </td></tr>
                        <tr><td>&nbsp;</td><td class="style1">
                            <asp:Button ID="Button1" runat="server" Text="ADD" CssClass="btn btn-primary btn-block"
```

```

        onclick="Button1_Click" /></td></tr>
</table>
</div>
</div>
</div>
</div>

<div class="col-md-3">
</div>
<br />
</div>
&nbsp;<br />
<div class="row">
<div class="col-md-3"></div>
<div class="col-md-6">
<div class="panel-group">
<div class="panel panel-primary">
<div class="panel-heading">Show Purchase</div>
<div class="panel-body">
<asp:GridView ID="GridView1" runat="server">
</asp:GridView>
<br />

Total: <asp:Label ID="Label1" runat="server" Text="Label"></asp:Label><br />
<asp:Button ID="Button2" runat="server" Text="GENERATE PO" CssClass="btn btn-primary btn-
block"
        onclick="Button2_Click" />
</div>
</div>
</div>
</div>

</div>
<div class="col-md-3"></div>

</asp:Content>

```

## **purchase\_purchase\_order.cs**

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;

public partial class purchase_purchase_order : System.Web.UI.Page
{
    db d=new db();
    protected void Page_Load(object sender, EventArgs e)
    {
    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        int pre = 0;
        for (int i = 0; i < GridView1.Rows.Count; i++)
        {
            if (GridView1.Rows[i].Cells[0].Text.Equals(DropDownList1.SelectedValue))
            {
                pre = 1;
                break;
            }
        }

        if (pre == 0)
        {
            if (ViewState["CurrentTable"] != null || GridView1.Rows.Count != 0)
            {
                DataTable dtCurrentTable = (DataTable)ViewState["CurrentTable"];
                DataRow drCurrentRow = null;
                drCurrentRow = dtCurrentTable.NewRow();
                drCurrentRow["productid"] = DropDownList1.SelectedValue;
                drCurrentRow["productname"] = DropDownList1.SelectedItem.Text.Split(':')[0];
                drCurrentRow["price"] = DropDownList1.SelectedItem.Text.Split(':')[1];
                drCurrentRow["qty"] = TextBox1.Text;
                drCurrentRow["total"] = (Convert.ToInt32(DropDownList1.SelectedItem.Text.Split(':')[1]) *
                Convert.ToInt32(TextBox1.Text));
                dtCurrentTable.Rows.Add(drCurrentRow);
                ViewState["CurrentTable"] = dtCurrentTable;
                GridView1.DataSource = dtCurrentTable;
                GridView1.DataBind();
            }
            else
            {
                DataTable dt = new DataTable();
                DataRow dr = null;
                dt.Columns.Add(new DataColumn("productid", typeof(string)));
                dt.Columns.Add(new DataColumn("productname", typeof(string)));
            }
        }
    }
}
```

```

        dt.Columns.Add(new DataColumn("price", typeof(string)));
        dt.Columns.Add(new DataColumn("qty", typeof(string)));
        dt.Columns.Add(new DataColumn("total", typeof(string)));
        dr = dt.NewRow();
        dr["productid"] = DropDownList1.SelectedValue;
        dr["productname"] = DropDownList1.SelectedItem.Text.Split(':')[0];
        dr["price"] = DropDownList1.SelectedItem.Text.Split(':')[1];
        dr["qty"] = TextBox1.Text;
        dr["total"] = (Convert.ToInt32(DropDownList1.SelectedItem.Text.Split(':')[1]) *
Convert.ToInt32(TextBox1.Text));
        dt.Rows.Add(dr);
        //dr = dt.NewRow();
        //Store the DataTable in ViewState
        ViewState["CurrentTable"] = dt;
        GridView1.DataSource = dt;
        GridView1.DataBind();
    }
}
Label1.Text = totalinggrid();
}

public string totalinggrid()
{
    int tot = 0;
    for (int i = 0; i < GridView1.Rows.Count; i++)
    {
        tot += Convert.ToInt32(GridView1.Rows[i].Cells[4].Text);
    }
    return tot.ToString();
}

public string totalqtyingrid()
{
    int tot = 0;
    for (int i = 0; i < GridView1.Rows.Count; i++)
    {
        tot += Convert.ToInt32(GridView1.Rows[i].Cells[3].Text);
    }
    return tot.ToString();
}

protected void Button2_Click(object sender, EventArgs e)
{
    //----- tbl_purchase_order
    string lid = d.latestId("tbl_purchase_order", "purchase_id");
    d.insertData("insert into tbl_purchase_order values(" + lid + "," + totalqtyingrid() + "," + Label1.Text + "," +
DateTime.Now.ToString() + ")");
    //----- tbl_purchaseordered_items
    for (int i = 0; i < GridView1.Rows.Count; i++)
    {
        d.insertData("insert into tbl_puchaseordered_items values(" + d.latestId("tbl_puchaseordered_items",
"id") + "," + lid + "," + GridView1.Rows[i].Cells[0].Text + "," + GridView1.Rows[i].Cells[3].Text + "," +
GridView1.Rows[i].Cells[4].Text + ")");
    }
}

```

```

    }
    Response.Redirect("purchase_purchase_order.aspx");
}
}

```

## Shop login:

### shop\_login.aspx

```

<% @ Page Language="C#" AutoEventWireup="true" CodeFile="shop_login.aspx.cs" Inherits="shop_login"
%>

```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">

```

```

<html xmlns="http://www.w3.org/1999/xhtml">
<head id="Head1" runat="server">

```

```

<link rel="stylesheet" href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>
<script src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

```

```

<style type="text/css">

```

```

body

```

```

{
    background-image:url("img/office.jpg");
    background-size:cover;
    background-attachment:fixed;
    background-repeat:no-repeat;
}

```

```

.jumbotron

```

```

{
    margin-top: 100px;
    box-shadow: 5px 5px 15px 2px black;
    border-radius: 5px;
    padding-bottom: 20px;
    background: light-grey;
}

```

```

}

```

```

.title

```

```

{
    background : blue;
    padding :10px;
}

```

```

text-align:center;
color:white;
border-radius:100px 100px 100px 100px;
}

</style>
</head>
<body>
<form id="form1" runat="server">
<div>
<div class="container">
<div class="row">
<div class="col-md-4"></div>
<div class="col-md-4">
<div class="jumbotron">
<form>
<div class="form-group">
<h2 class="title">Login Now </h2>
<p id="profile-name" class="profile-name-card"></p>

<asp:TextBox ID="TextBox1" runat="server" CssClass="form-control"
placeholder="Username"></asp:TextBox>

</div>
<div class="form-group">

<asp:TextBox ID="TextBox2" runat="server" CssClass="form-control"
placeholder="Password" TextMode="Password"></asp:TextBox>

</div>
<div class="form-group">
<asp:CheckBox ID="CheckBox1" runat="server" Text="Remember Me?" ForeColor="black"
/>

</div>
<asp:Button ID="Button1" runat="server" Text="Login"
CssClass="btn btn-block btn-primary" onclick="Button1_Click" />
<br />
</form>
</div>
</div>
<div class="col-md-4"></div>
</div>
</div>
</div>
</form>
</body>
</html>

```

## shop\_login.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;

public partial class shop_login : System.Web.UI.Page
{
    db d = new db();
    protected void Page_Load(object sender, EventArgs e)
    {

    }
    protected void Button1_Click(object sender, EventArgs e)
    {
        string qry = d.getSingleRow("select * from tbl_shop where username='" + TextBox1.Text + "' and
password='" + TextBox2.Text + "'");
        if (qry.Length > 0)
        {
            string[] ar = qry.Split(':');
            Session["shopid"] = ar[0];
            Response.Redirect("shopitemreceived.aspx");
        }
    }
}
```



## shopitemreceived.aspx

```
<% @ Page Title="" Language="C#" MasterPageFile="~/shop.master" AutoEventWireup="true"
CodeFile="shopitemreceived.aspx.cs" Inherits="shopitemreceived" %>
```

```
<asp:Content ID="Content1" ContentPlaceHolderID="head" Runat="Server">
    <link href="vendor/bootstrap/css/panel1.css" rel="stylesheet" type="text/css" />
</asp:Content>
<asp:Content ID="Content2" ContentPlaceHolderID="ContentPlaceHolder1" Runat="Server">
    <div class="row">
        <div class="col-md-3"></div>
        <div class="col-md-6">
            <div class="panel-group">
                <div class="panel panel-primary">
                    <div class="panel-heading">Received Items</div>
                    <div class="panel-body">
                        <table class="table">
                            <tr><td><b>Select</b></td><td>
                                <asp:DropDownList ID="DropDownList1" runat="server" AutoPostBack="True"
                                    DataSourceID="SqlDataSource1"
                                    DataValueField="dist_id" Width="170px" CssClass="form-control" DataTextField="dist_id">
                                </asp:DropDownList></td><td>
                                    <asp:RequiredFieldValidator ID="RequiredFieldValidator1" runat="server"
                                        ControlToValidate="DropDownList1" ErrorMessage="*" Font-Bold="True"
                                        ForeColor="Red"></asp:RequiredFieldValidator>
                                </td></tr>
                        </table>
                        <asp:SqlDataSource ID="SqlDataSource1" runat="server"
                            ConnectionString="<%$ ConnectionStrings:invconn %>"
                            SelectCommand="SELECT dist_id FROM tbl_dist_list WHERE (received = 0) AND (shop_id =
@shopid)">
                            <SelectParameters>
                                <asp:SessionParameter Name="shopid" SessionField="shopid" />
                            </SelectParameters>
                        </asp:SqlDataSource>
                        <br />
                        <asp:GridView ID="GridView1" runat="server" AutoGenerateColumns="False"
                            DataSourceID="SqlDataSource3" CssClass="table table-bordered table-striped"
                            DataKeyNames="id">
                            <Columns>
                                <asp:CommandField ShowEditButton="True" />
                                <asp:BoundField DataField="id" HeaderText="id" ReadOnly="True"
                                    SortExpression="id" />
                                <asp:TemplateField HeaderText="prod_id" SortExpression="product_id">
                                    <EditItemTemplate>
                                        <asp:Label ID="Label6" runat="server" Text="<%# Bind("product_id") %>"></asp:Label>
                                    </EditItemTemplate>
                                    <ItemTemplate>
                                        <asp:Label ID="Label1" runat="server" Text="<%# Bind("product_id") %>"></asp:Label>
                                    </ItemTemplate>
                                </asp:TemplateField>
                            </Columns>
                        </asp:GridView>
                    </div>
                </div>
            </div>
        </div>
    </div>
</asp:Content>
```

```

</asp:TemplateField>
<asp:TemplateField HeaderText="prod_name" SortExpression="product_name">
  <EditItemTemplate>
    <asp:Label ID="Label7" runat="server" Text='<%# Bind("product_name") %>'></asp:Label>
  </EditItemTemplate>
  <ItemTemplate>
    <asp:Label ID="Label2" runat="server" Text='<%# Bind("product_name") %>'></asp:Label>
  </ItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="qty" SortExpression="qty">
  <EditItemTemplate>
    <asp:Label ID="Label8" runat="server" Text='<%# Bind("qty") %>'></asp:Label>
  </EditItemTemplate>
  <ItemTemplate>
    <asp:Label ID="Label3" runat="server" Text='<%# Bind("qty") %>'></asp:Label>
  </ItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="total_price" SortExpression="total_price">
  <EditItemTemplate>
    <asp:Label ID="Label9" runat="server" Text='<%# Bind("total_price") %>'></asp:Label>
  </EditItemTemplate>
  <ItemTemplate>
    <asp:Label ID="Label4" runat="server" Text='<%# Bind("total_price") %>'></asp:Label>
  </ItemTemplate>
</asp:TemplateField>
<asp:TemplateField HeaderText="received" SortExpression="received">
  <EditItemTemplate>
    <asp:DropDownList ID="DropDownList2" runat="server"
      SelectedValue='<%# Bind("received") %>'>
      <asp:ListItem Value="0">NOT RECEIVED</asp:ListItem>
      <asp:ListItem Value="1">RECEIVED</asp:ListItem>
    </asp:DropDownList>
  </EditItemTemplate>
  <ItemTemplate>
    <asp:Label ID="Label5" runat="server" Text='<%# Bind("received") %>'></asp:Label>
  </ItemTemplate>
</asp:TemplateField>
</Columns>
</asp:GridView>
<asp:SqlDataSource ID="SqlDataSource3" runat="server"
  ConnectionString='<%= $ ConnectionStrings:invconn %>'

  SelectCommand="SELECT [id], [product_id], [product_name], [qty], [total_price], [received] FROM
[tbl_dist_list] WHERE ([dist_id] = @dist_id)"
  DeleteCommand="DELETE FROM [tbl_dist_list] WHERE [id] = @id"
  InsertCommand="INSERT INTO [tbl_dist_list] ([id], [product_id], [product_name], [qty],
[total_price], [received]) VALUES (@id, @product_id, @product_name, @qty, @total_price, @received)"
  UpdateCommand="UPDATE [tbl_dist_list] SET [product_id] = @product_id, [product_name] =
@product_name, [qty] = @qty, [total_price] = @total_price, [received] = @received WHERE [id] = @id">
  <DeleteParameters>
    <asp:Parameter Name="id" Type="Decimal" />

```

```

</DeleteParameters>
<InsertParameters>
  <asp:Parameter Name="id" Type="Decimal" />
  <asp:Parameter Name="product_id" Type="Decimal" />
  <asp:Parameter Name="product_name" Type="String" />
  <asp:Parameter Name="qty" Type="Decimal" />
  <asp:Parameter Name="total_price" Type="Decimal" />
  <asp:Parameter Name="received" Type="Decimal" />
</InsertParameters>
<SelectParameters>
  <asp:ControlParameter ControlID="DropDownList1" DefaultValue="0" Name="dist_id"
    PropertyName="SelectedValue" Type="Decimal" />
</SelectParameters>
<UpdateParameters>
  <asp:Parameter Name="product_id" Type="Decimal" />
  <asp:Parameter Name="product_name" Type="String" />
  <asp:Parameter Name="qty" Type="Decimal" />
  <asp:Parameter Name="total_price" Type="Decimal" />
  <asp:Parameter Name="received" Type="Decimal" />
  <asp:Parameter Name="id" Type="Decimal" />
</UpdateParameters>
</asp:SqlDataSource>
<tr><td><td> </td></td></tr>

</div>
</div>
</div>
</div>
</div>

<div class="col-md-3">
</div>
<br />
</asp:Content>

```

## **Testing Approach**

### **Unit Testing-**

This is the testing process which we can do manually because in this testing program is a tested individually using dummy record to see whether that program produce satisfied output as the company and validation also. We have tested each and every unit and have validated the required ones. If the user (admin) forgets to fill a field in login page the user will get a validation error. If the values are correct he will be redirected to main GUI page.

### **Validation Testing-**

In validation testing we have provided validations for login page, shop page, product page, vendor page, user add page, shopkeeper page, We have used required field validation in login page. In pages we have used error provider for each pages like user add, shop add, product add, grn add etc.. In the every pages I use error provider as required field validator, range validator, contact number validator etc. It provides the full proper input-output working of the whole system.

Validation Testing ensures that product actually meets the client's needs. It can also be defined as to demonstrate that the product fulfills the its intended use when deployed on appropriate environment.

### **Integration Testing:**

In Integration we test all our inventory database, user database, login & its purchases, all System modules and vital coding entirely as an integrated work, after unit testing integration testing provides the conformity to the best level of security measures and policies. We deal it as the all module proper working as a whole project. It helps to the validation and verification analysis.

### **System Testing:**

We have tested the system on high end and low end computers. We have come up with the minimum system requirements to run the software smoothly. The minimum requirements are as follows:

- 2 GB RAM
- Windows 7 or higher
- Intel i3 or higher
- 2 GB free HDD space

- **Test Cases:**

In this software the security depends upon the password authentication of the visual studio software is installed and the login page of the software. If a person tries to access then he firstly needs to crack the password. If he is able to do so then he can easily make changes and get the details of the students.

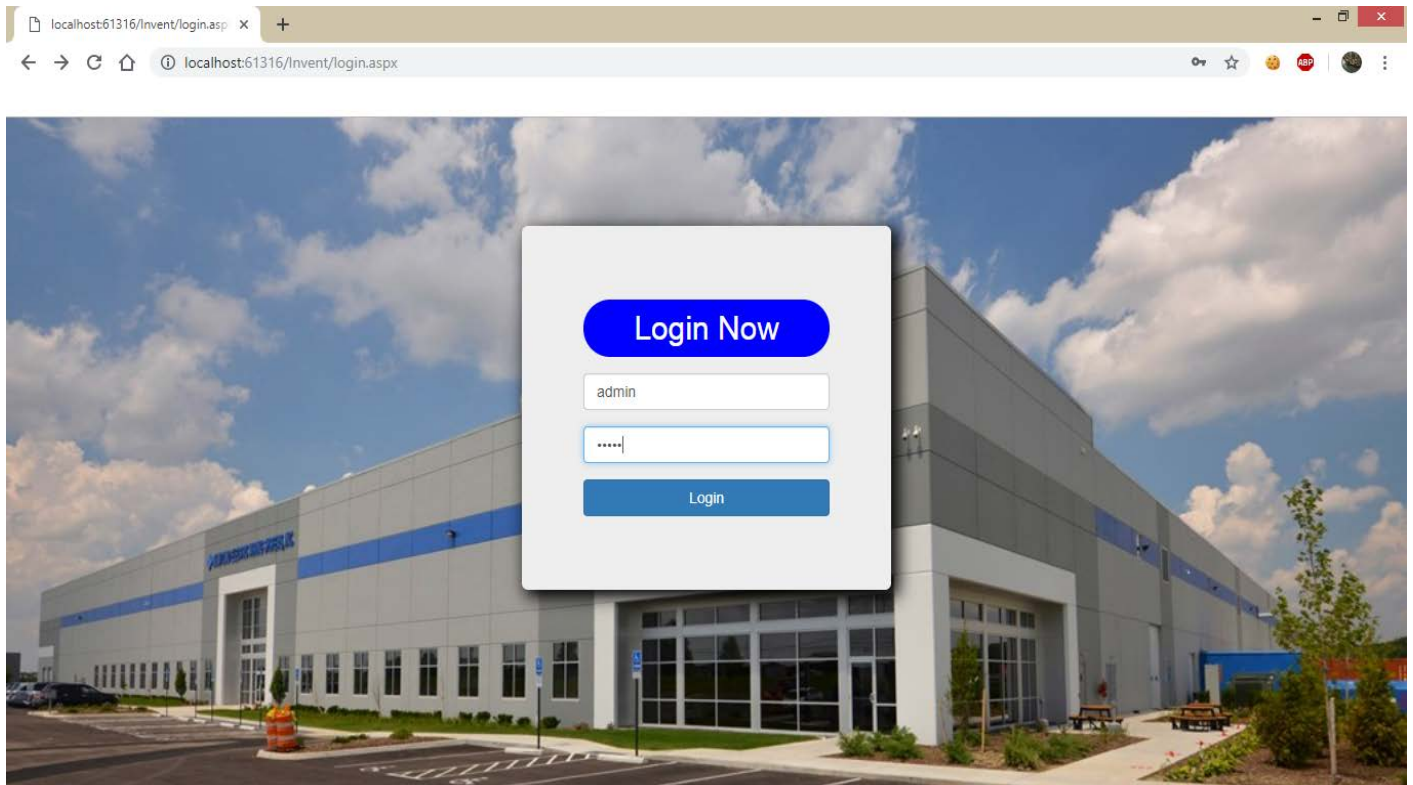
<b>Sr. no</b>	<b>Action</b>	<b>Input</b>	<b>Expected output</b>	<b>Actual Output</b>	<b>Test result</b>	<b>Test comment</b>
<b>1.</b>	<b>Enter correct username and password</b>	<b>Username:</b> admin  <b>Password:</b> *****	<b>Go to admin page</b>	<b>Admin Page</b>	<b>Pass</b>	<b>Admin Page will display</b>
<b>2.</b>	<b>If username and password are incorrect</b>	<b>Username:</b> abc  <b>Password:</b> *****	<b>“Login failed”</b>	<b>“Invalid User”</b>	<b>Pass</b>	<b>Invalid username and password</b>
<b>3.</b>	<b>Enter Add product or shop id is correct</b>	<b>Enter Product id/shop id</b>	<b>“product/shop added successfully”</b>	<b>“product/shop Details inserted”</b>	<b>Pass</b>	<b>Successful</b>

<b>4.</b>	<b>If entered product MRP is in character Format</b>	<b>Enter product MRP</b>	<b>“Invalid Product MRP”</b>	<b>“Invalid product mrp”</b>	<b>Pass</b>	<b>successful</b>
<b>5.</b>	<b>If entered distributor price is character format</b>	<b>Enter distributor price</b>	<b>“Invalid entry”</b>	<b>“Invalid entry”</b>	<b>Pass</b>	<b>successful</b>
<b>6.</b>	<b>If entered vendor or shop Phone No is character format</b>	<b>Enter Phone No</b>	<b>“Invalid Phone No”</b>	<b>“Invalid Phone No”</b>	<b>Pass</b>	<b>successful</b>
<b>7.</b>	<b>If entered Phone no length is less than 10 or should not blank</b>	<b>Enter Phone no.</b>	<b>“Invalid Phone no”</b>	<b>“Invalid Phone no”</b>	<b>Pass</b>	<b>successful</b>

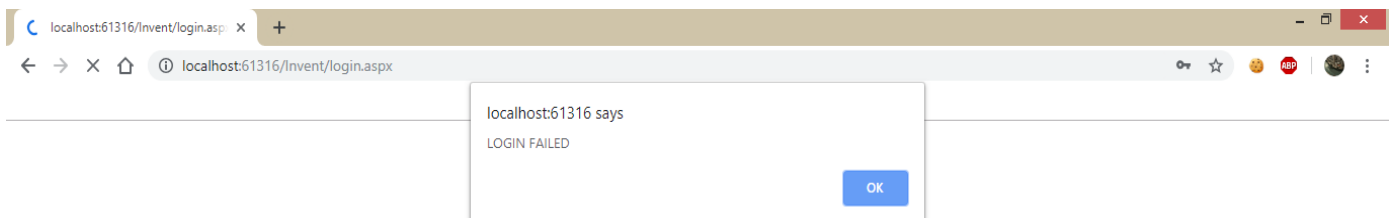
<b>8.</b>	<b>If Entered vendors email id Does not match</b>	<b>Enter email id</b>	<b>“email id not valid ”</b>	<b>“email id not valid”</b>	<b>Pass</b>	<b>successful</b>
<b>9.</b>	<b>Fill details of the vendor</b>	<b>Details of the vendor</b>	<b>“Vendor is added successfully”</b>	<b>“Vendor is added successfully”</b>	<b>Pass</b>	<b>successful</b>
<b>10.</b>	<b>Fill details of the shop</b>	<b>Details of the customer</b>	<b>“Customer is added successfully”</b>	<b>“Customer is added successfully”</b>	<b>Pass</b>	<b>successful</b>

## 8. SCREEN LAYOUT WITH TEST DATA & TEST RESULTS

When we enter Valid Data, it shows navigates to Admin Page



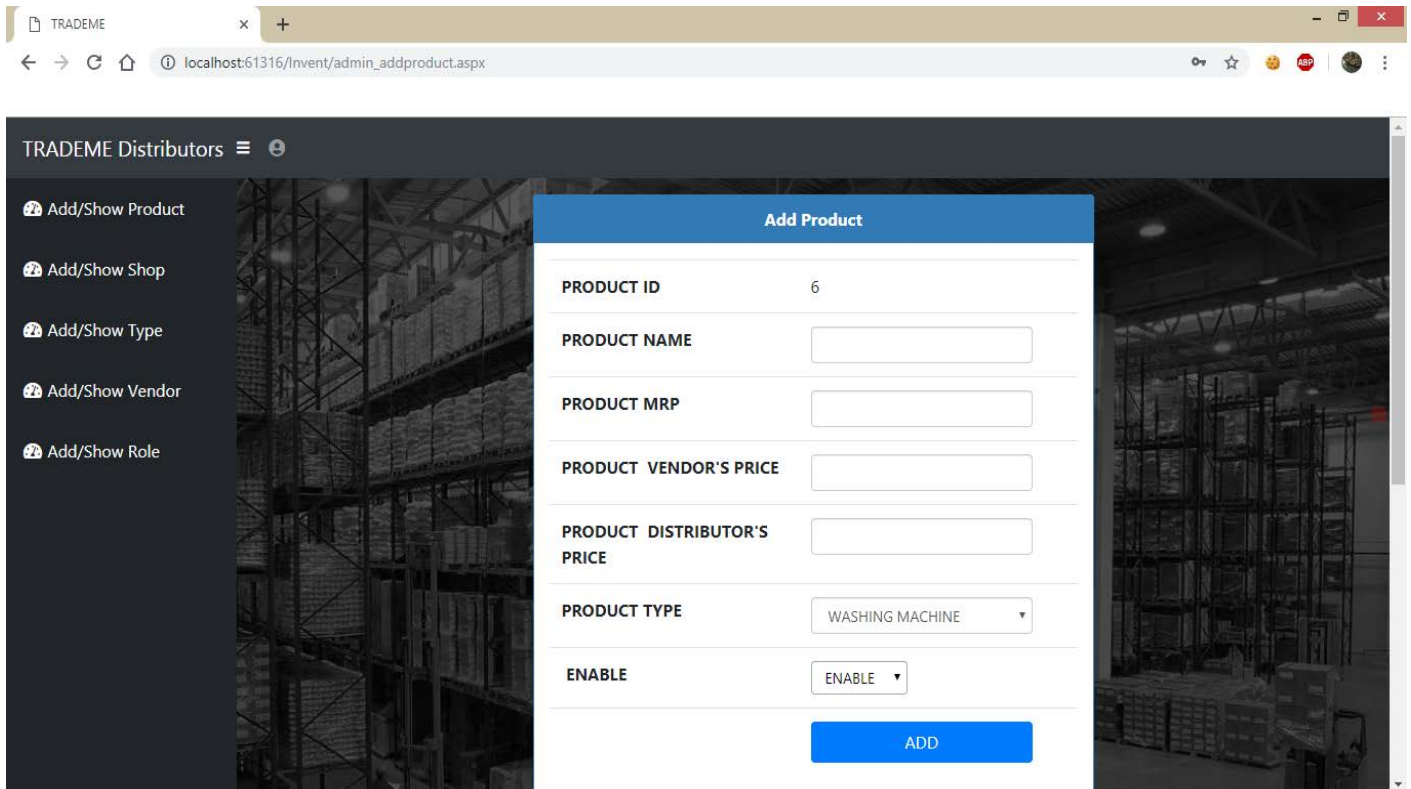
When username and password is invalid then it shoes invalid user message





After Login Displaying the Admin Page will look like this:

## ADD PRODUCT:



The screenshot shows a web browser window with the URL `localhost:61316/Invent/admin_addproduct.aspx`. The page title is "TRADEME Distributors". On the left, there is a sidebar menu with the following items: "Add/Show Product", "Add/Show Shop", "Add/Show Type", "Add/Show Vendor", and "Add/Show Role". The main content area displays the "Add Product" form. The form has the following fields:

Add Product	
PRODUCT ID	6
PRODUCT NAME	<input type="text"/>
PRODUCT MRP	<input type="text"/>
PRODUCT VENDOR'S PRICE	<input type="text"/>
PRODUCT DISTRIBUTOR'S PRICE	<input type="text"/>
PRODUCT TYPE	WASHING MACHINE ▼
ENABLE	ENABLE ▼
<input type="button" value="ADD"/>	

TRADEME

localhost:61316/invent/admin\_addproduct.aspx

ADD

Show Products

	Prod_ID	Product_Name	Product_MRP	Product_Vprice	Product_distprice	Product_Type_id	Enable
<a href="#">Edit</a>	2	DELL INSPIRON	40000	30000	36000	2	Y
<a href="#">Edit</a>	3	SAMSUNG AUTO	25000	15000	20000	1	Y
<a href="#">Edit</a>	4	ONIDA	35000	28000	31000	4	Y
<a href="#">Edit</a>	5	HP PRINTER	15000	10000	13000	2	Y

Copyright © TRADEME 2019

ADD SHOP:

TRADEME

localhost:61316/Invent/admin\_addshops.aspx

TRADEME Distributors

Add/Show Product

Add/Show Shop

Add/Show Type

Add/Show Vendor

Add/Show Role

Add Shop

SHOPID

6

SHOP NAME

SHOP ADDRESS

SHOP PHONE NO

OWNER NAME

USERNAME

PASSWORD

ADD TYPE:

TRADEME

localhost:61316/Invent/admin\_addtype.aspx

TRADEME Distributors

Add/Show Product

Add/Show Shop

Add/Show Type

Add/Show Vendor

Add/Show Role

Add Type

TYPE ID

7

TYPE NAME

ADD

Show Type

	type_id	type_name
<a href="#">Edit</a>	1	WASHING MACHINE
<a href="#">Edit</a>	2	COMPUTERS
<a href="#">Edit</a>	3	FRIDGE

ADD VENDOR:

TRADEME Distributors

- Add/Show Product
- Add/Show Shop
- Add/Show Type
- Add/Show Vendor
- Add/Show Role

**Add Vendor**

VENDOR ID

2

VENDOR NAME

VENDOR ADDRESS

VENDOR PHONE NO

ADD

**Show Vendors**

--	--	--	--	--

TRADEME

- Add/Show Vendor
- Add/Show Role

VENDOR ADDRESS

VENDOR PHONE NO

ADD

**Show Vendors**

	Ven_Id	Ven_Name	Ven_Addr	Vend_PhNo
Edit	1	AAA	BHIWANDI	8794561253

Copyright © TRADEME 2019

## ADD ROLE:

TRADEME Distributors

- Add/Show Product
- Add/Show Shop
- Add/Show Type
- Add/Show Vendor
- Add/Show Role

### Add Role

Role ID: 6

Role Name:

ADD

### Show Roles

	role_id	role_name
Edit	1	ADMIN
Edit	2	PURCHASE
Edit	3	GRN

## GENERATE GRN:

TRADEME Distributors

### GRN

Select GRN:

Create Grn

### Show GRN

Copyright © TRADEME 2019



## PURCHASE ORDER:

TRADEME Distributors

Product

Select Product: DELL INSPIRON:30000

Enter Quantity: 14

ADD

Show Purchase

productid	productname	price	qty	total
2	DELL INSPIRON	30000	14	420000

Total: 420000

GENERATE PO

## PRODUCT DISTRIBUTION:

TRADEME Distributors

### Distribution

Select Shop: SHOP 1

Select Product: DELL INSPIRON:36000

Enter Quantity: 10

ADD TO SHOP

SHOP ID	PRODUCT ID	PRODUCT NAME	DISTRIBUTION PRICE	QTY	TOTAL PRICE
1	2	DELL INSPIRON	36000	10	360000

Update Distribution

### Show Product Distribution

id	prod_id	prod_name	shop_id	qty	total_price	dist_id	received
----	---------	-----------	---------	-----	-------------	---------	----------

## SHOP ITEM RECEIVED:

TRADEME Distributors

### Received Items

Select Distribution ID: 1

	id	prod_id	prod_name	qty	total_price	received
Edit	1	2	DELL INSPIRON	10	360000	0
Edit	2	2	DELL INSPIRON	15	540000	0

Copyright © TRADEME 2019

## **9. FUTURE SCOPE & CONCLUSION**

### **Future scope of the project**

My project “INVENTORY MANAGEMENT SYSTEM” will be able to implement in future after making some changes and modifications as we make our project at a very low level. So the modification that can be done in our project is to add one major change which can be done in this project is that to add the snaps of the product of which the record is entered. This will result in total identification of the given product. Similarly various modifications can be done to enhance the usability of the given project as suitable for user’s requirement.

### **Conclusion:**

From this project we can conclude that if this program is very useful in inventory management as it provide more convenience than the manual work. It provides easy methods to manage the load of work easily for the users. It is much fast and more efficient as the data once entered can be modified and accessed easily. The program can be used per the requirement of the user as it is very easy to understand.

Benefit of inventory management system for shop day to day activity are list the product that still available in database, update the database whether by reducing the number of stocks available in database, it also kept the GRN details of product.



## 10. REFERENCES

- <http://www.youtube.com>
- <https://www.c-sharpcorner.com>
- <https://stackoverflow.com/>
- <https://www.tutorialspoint.com/c#/> and associated YouTube videos for C#-tutorial.