



Mithibai College of Arts,
Chauhan Institute of Science &
Amrutben Jivanlal College of Commerce and Economics
(AUTONOMOUS)

Affiliated to University of Mumbai

NAAC Reaccredited 'A' Grade, CGPA : 3.57 (February 2016)

GRANTED UNDER RUSA FOR RESEARCH, INNOVATION AND QUALITY IMPROVEMENT
GRANTED UNDER FIST-DST & STAR COLLEGE SCHEME OF DBT, GOVERNMENT OF INDIA
BEST COLLEGE (2016-17), University of Mumbai



A
Project Report
On
Sales and Process Tracking Web-Application

Project Report submitted for
Bachelor of Computer Science

Submitted by
Yash Babel

Under the guidance of
Prof. Zahir Mulani

YEAR: 2019-2020



Mithibai College of Arts,
Chauhan Institute of Science &
Amrutben Jivanlal College of Commerce and Economics
(AUTONOMOUS)
Affiliated to University of Mumbai

NAAC Reaccredited 'A' Grade, CGPA : 3.57 (February 2016)
GRANTED UNDER RUSA FOR RESEARCH, INNOVATION AND QUALITY IMPROVEMENT
GRANTED UNDER FIST-DST & STAR COLLEGE SCHEME OF DBT, GOVERNMENT OF INDIA
BEST COLLEGE (2016-17), University of Mumbai



CERTIFICATE

This is to certify that the project titled **Gransells** is successfully done by

Mr. YASH BABEL in fulfilment of **Bachelor Of Computer Science** under the
University of Mumbai, Mumbai through the Mithibai College Of Arts,
Chauhan Institute Of Science & Amrutanben Jivanlal College Of Commerce &
Economics (Autonomous) Mumbai carried out by him under our guidance and
supervision.

Head Of Department

External Examiner

Internal Guide

Date:

Department of Computer Science

DECLARATION

This Web Application will help Granite Firms to market their product in a better way and will also help them to track internal processing of orders. This will help make tracking of firm's processes easy and possible for the Administrator. Customers can review the firm's portfolio which will help them select a better product and they can place the order over the application.

The Company will hence be able to market its products in a better way. The worker's data can be managed much efficiently through the application.

It is tried to develop the software with utmost care, but there may be some limitations, for the project is subjective to future enhancements as per the needs of the client.

Mr. Yash Babel

ACKNOWLEDGEMENT

I avail this opportunity to express my sincere and deep gratitude to many who are a factor in helping me gain the knowledge and experience during the project and throughout the course.

I have great pleasure in presenting this project. The completion of this project is not merely due to only my own efforts but also due to the guidance given by our professors.

I am thankful to our project guide Prof. Zaheer Mulani Sir & Prof. Jayshree Ravi Ma'am for their support. I also thank H.O.D. , Principal, Vice Principal and respected faculty members for their kind support and help throughout the entire course.

I am also highly indebted to our project team leader Mr Zaheer Mulani sir for his continuous support & guidance without which this project couldn't have been in reality.

Finally, I express my deep regards to all of those who stretch their helping in the execution of my project.

Index

Topic	Page Number
I. Preliminary Investigation	
i. Organizational Overview	7
ii. Description of the System	9
iii. Limitations of the present system	10
iv. Proposed system and its advantages	11
v. Feasibility Study	13
vi. Stakeholders	15
vii. Technologies used	15
II. System Analysis	
i. Fact Finding Techniques	16
ii. Event Table	19
iii. Use Case Diagram, Scenarios and Use case description	20
iv. ER Diagram	22
v. Class Diagram	23
vi. Activity Diagram	24
vii. State Diagram	26
viii. Sequence Diagram	27
III. System Design	
i. Converting ER Diagrams to Tables	30

ii. Deployment Diagram	32
IV. System Coding	
i. Menu Tree / Sitemap	33
ii. Code	34
iii. Output	70
V. Limitations	78
VI. Future Enhancements	79
VII. Reference and Bibliography	80

PRELIMINARY INVESTIGATION

ORGANISATIONAL OVERVIEW

The project is developed to handle the manufacturing and sales tasks for Granite Pvt Ltd which is a Granite slab manufacturing unit started at Shahjapur. It was established in 2009, since then it is continuously running with its full capacity.

It handles sales and manufacturing processes of Granites slabs and handle shipment of processed slabs to other cities and countries. Over the years the firm has established its name in this industry for the quality of their products they provide to the customer.

The firm performs processing of huge granite slabs. These slabs are ordered from other mining industries as and when required or when there is an order for that particular type. These slabs after arriving at the firm go through 5 stages of processing in the firm, these include Unloading, Machine cutting, Polishing, Packing and Loading.

The organisation is committed to provide customers with quality products and to assure that the firm is trying its best to cope with the technological advancements and with the help of this project the firm be able to market its product in a better way, customers won't have to visit firm to place order and it will also make it easy for the owners to track the internal processing of the orders.

DESCRIPTION OF SYSTEM

The factory units of Granite firms are located in the outskirts of cities, it becomes difficult for the Directors (Administrators) to visit daily and keep the track of Manufacturing and Sales processes.

Also, till today orders at most of the firms are placed traditionally i.e. on calls or visits. This Web Application will hence resolve the above problem making it easy and possible for the Administrator to track all the processes from home.

Customers can review the firm's portfolio which will help them select a better product and they can place the order over the application.

The Company will hence be able to market its products in a better way.

The Granite processing has certain steps(Block Loading, Machine cutting, Polishing, Packing, Loading), the manager will have the authority to update the stage of order and the Administrator will be able to track the same.

The worker's data can be managed much efficiently through the application.

Objectives

- A common login interface for the administrator, manager and customers
- Customer will be able to view the types of granite available at the firm at a glance
- Managing worker's data in much efficient way.
- Saving administrator's time as tracking becomes easy via the application.
- A better way to market products.

Most of the Granite firms, till today follow the traditional methods to manage their Sales and Manufacturing processes. This includes a lot of paperwork and direct human interactions.

The customers place orders to the firm by contacting the administrator. The detail regarding the color, type, dimensions, edging and polishing of the product are told to the administrator either on calls or text messages or on visits. These details are then informed to the manager to start the further processes. A separate transcribed document is maintained for the details regarding each and every order which is updated by the manager as and when the orders arrive.

The processing of Granite blocks involves different stages. Manager keeps the track of the stage the order has reached and inform administrator about the same. This tracking record is also maintained in documents with the respective information. Many visits by the administrator are followed to manage the orders and examine the documents.

Simultaneously, the information of workers and other staff members are also transcribed and maintained separately. This includes their personal details, identity proofs, shift timings, and salary details.

At the end of every month, stock calculation is done manually by the administrator and the manager to get the revenue of the firm and for sales analyses of the firm.

LIMITATIONS IN EXISTING SYSTEM

- Tracking becomes difficult task-
 - Manager has to keep on updating the administrator about the tracking stage of each order.
 - There is no way out the administrator can track it on its own, when not in firm.
- Fewer customers-
 - Only a few customers will know the varieties available with the firm, which can lead to less sales.
 - The company is not able to market its product in a better way.
- Managing data becomes difficult task-
 - To maintain all the data manually is tedious and time consuming as manager has to update each and every information regarding the orders, process and workers time to time.
- Time consuming and tedious for the administrator-
 - To examine the records and to keep the track of processing, the administrator has to keep visiting the firm and manage things.
 - As the firms are located on the outskirts of cities, visiting becomes tedious and time consuming for the administrator.
- Poor communication-
 - The poor communication between administrator and manager can lead to processing or sales issues.
 - Any misunderstanding in realizing the customer's needs can lead to amiss delivery, harming the image of the firm.
- Human and computational errors-
 - Data analysis becomes difficult when everything is done on paper, which may further lead to errors.
 - Stock calculation becomes challenging when done manually.
- Difficulty in location of files-
 - Due to large number of files, it becomes a challenging job to locate files and update them regularly.

PROPOSED SYSTEM

To overcome the drawbacks faced while using the manual system, an idea of a web application which can simulate each and every task is proposed.

The web application will have a common login interface for the customers, administrator and manager. The customers on their account will have a portfolio section which will showcase the products which are available with the firm for sale. The customers can go through it and place orders after specifying the required details.

The administrator and the manager will be able to see the order details. The orders placed by the customer will be accepted by the firm and the processing will start. The processing has different stages i.e. Block Loading, Machine cutting, Polishing, Packing and at last Delivery.

Every time the order passes one stage of processing, the manager will update it from his account and the administrator will be able to track it with help of elegant Interface designs. The administrator will also be able to manage and view worker's data. It will also have a section for the stock calculation in which after specifying the details, the calculated revenue amount can be displayed to the administrator.

The manager will have the authority to manage the orders as well as the worker's data from his account. As and when the order progresses, its stage it will be updated by the manager, making it easier for the administrator to track it from anywhere anytime. The workers data can be entered on the application and can be saved in database which helps in easily retrieving and managing it.

ADVANTAGES

- Tracking becomes easy-
 - Administrator do not have to visit the firm daily or contact the manager on regular basis, as tracking of each order can be done via the admins account.
- Customer base increases-
 - As the portal will be available on-line, marketing becomes easy for the firm.
 - The customer will be able to look at the varieties of products available which will help firm increase its sale.
- Managing data becomes simple-
 - Manager and administrator can handle the order data easily.
 - Worker's data can be managed easily.
- Saves time-
 - As tracking and managing data can be done through application, it saves time of manager, administrator and customer.
- No communication problem-
 - As everything happens over the application, communication is rare.
 - Order delivered will be up to the details provided by the customer.
- Less errors-
 - Chances of computational error reduces as calculations and management can be programmed.
- Common storage place-
 - All the data regarding the orders, workers can be saved and stored in databases which can ease in retrieving and managing it.

FEASIBILITY STUDY

Economic Feasibility

For any system if the expected benefits equal or exceed the expected costs, the system can be judged to be economically feasible. In economic feasibility, we analyse cost benefit in which expected costs and benefits are evaluated. Economic analysis is used for evaluating the effectiveness of the proposed system. The following costs have to be evaluated before coming to a conclusion.

- Development costs-
 - Development costs that are incurred during the development of the system are one-time investment
 - Wages.
 - Equipment used in development process
- Operating cost-
 - These are expenses required for the day to day running of the system. This includes the maintenance of the system. That can be in the form of maintaining the hardware or application programs a money paid to professionals responsible for running or maintaining the system
- Hardware and software cost-
 - It includes the cost of purchasing or leasing of computers and its peripherals. Software costs involve the cost of the required software.
- Personal Cost:
 - We have to evaluate the money spent on the people involved in the development of the system. These expenditures include other benefits such as health insurance, conveyance allowance, etc.
- Facility Cost-
 - Expenses incurred during the preparation of the physical site where the system will be operational should also be evaluated. These can be wiring, flooring, acoustics, lighting, and air conditioning.

If the total of the above costs come into the range of resources available with the team then the project is said to be economically feasible.

Technical Feasibility

The technologies used in the development of project are ASP DOT NET & MySql. These technologies are considered feasible and the team has prior knowledge of all the technologies.

And if not, then the needed versions can be bought with the available resources.

Operational Feasibility

The basic purpose of feasibility study is to determine whether the whole system of “Granite Sales and Process web application” is worth or not. Since the current system involves a lot of paperwork, human interactions, human efforts. There should be something to overcome these drawbacks. It was necessary to build a system which can simulate some tasks and reduce the human efforts.

The success of smooth functioning of this system depends upon the fact that most of task that were done with human efforts can be reduced. The data can be managed properly. The system should reduce the efforts put in by the administrator to manage the firm.

Secondly the prerequisites, to manage the system. The manager has to be trained to work with the system as he is one who is going to interact with the system most of the time. From managing workers data to managing orders, he is the one handling most of the resources. The customer section will be made more user friendly which will attract users and will boost the firm's popularity.

STAKEHOLDERS

- End users-
 - The customers of the firm.
- Project team-
 - The build team serves as the voice of reason that keeps the project within a manageable scope and develop it.
- Authorities-
 - This includes shareholders, and company owners.
 - Their inputs are considered before developing the system as they are the one investing in it.

TECHNOLOGIES

- Hardware specification
 - Processor: Intel Premium or More
 - RAM: 2 GB Ram or More
 - Hard Disk: PC with 30GB
- Software requirements
 - Operating System: Windows XP
 - Front-End: ASP.NET
- Servers
 - Database: MySQL
 - Client: Internet Explorer, Mozilla Firefox, Chrome etc

SYSTEM ANALYSIS

FACT FINDING TECHNIQUES

The formal process of using techniques such as interview, site visit, questionnaire and sampling of existing document is used to collect facts about system, requirements and preferences.

Observation of work environment

- In this technique, system analyst participates in the organization, studies the flow of documents, applies the existing system, and interacts with the users. Observation can be a useful technique when the system analyst has user point of view.
- So, I visited the firm for two days and observed how things are managed and also inspected the place where the registers were kept, it was a real mess. I saw the records for order, workers, and customers. To make a new entry for an order, to maintain the worker's data, to keep a track of each and every order, different registers are maintained and it is a difficult and effort intensive process. From site visit I had a good understanding of the functioning of the system. After this, we performed some personal interviews of customers, manager, administrator and other working staff. In the next section we'll look at these interviews.

Interview

- The manager was asked about the functioning of the firm. How the firm manages data. What is difficulty faced in managing it. How frequently u have to update the administrator about the order.
- And he explained one by one how the firm manages every task. The conclusion of the whole interview was that the firm needs an application over which it can manage its tasks. Transcribing of documents and then maintaining them is burdensome. These orders go through 5 stages of processing which are Block loading , Machine cutting, Polishing, Packing and Loading and to keep the administrator updated about the stages of order after each checkpoint is time consuming and strenuous.

- The web-application will have a proper tracking functionality over which the manager can keep on updating and administrator can track from his login. A decent functionality to maintain the data about the order, workers and customer was necessary.
- After manager, the administrator was interviewed and was asked whether he will readily adapt to the portal? Will he advise other firms to and the analyst came to know that it is difficult for the administrator to manage the order and keep a track on them. The manager and the administrator face the same problem. Visiting the firm that is approximately 100km from the city is not possible every day, so a functionality through which the administrator can keep track from anywhere is essential.
- The customer also needs a way out so that he doesn't have to visit the firm to place the order.

Sampling of existing documentation & forms

- The best way to analyse the existing system is to collect facts from existing documentation rather than from human sources.
- After understanding the user requirements, the analyst teams go through the existing documents that are maintained by the firm.
- This includes the registers maintained to
 - To manage workers data
 - To manage order data
 - To manage tracking details
 - To maintain attendance of workers
 - To manage customer data
- The user firmly says that he wants these task to be included in the web application as functionalities. The worker's data like their personal details, their shift timings, salary are maintained in the register. A different register is maintained to manage their attendance. To handle the order and its tracking stage, a register is maintained and it was observed that to make a new entry or to read or analyse it is difficult for the manager as well as administrator.

Modules

- **Customer**

- The customer will have a UI where he will be able to go through the portfolio of the firm i.e. is the varieties of Granites available
- After clicking on the specific product, he will get a screen having the description of the product.
- To place the order, the customer have to specify the type and then select the size for the blocks and place the order.
- A window showing customers details will be shown to user to confirm the details, after confirming a confirmation window will open.
- A user profile page will be there, which will help the customer to change his/her details and show them their order history.

- **Manage**

- After the customer places an order, the manager will have a screen to see if there is any other order
- Since the manager will update the tracking stage of the orders, he will have a screen from where he will be able to update it. This will be individually done for each order.
- Manager can view worker records
- A proper revenue calculation module will be there.

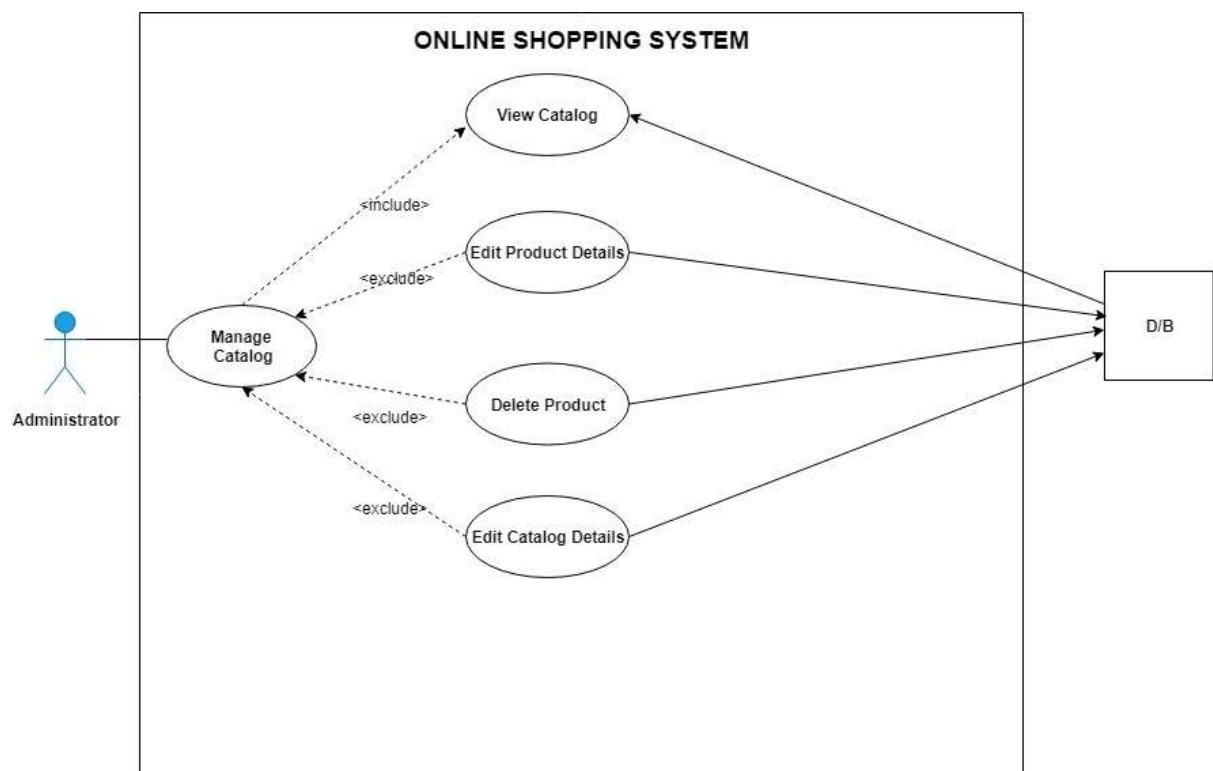
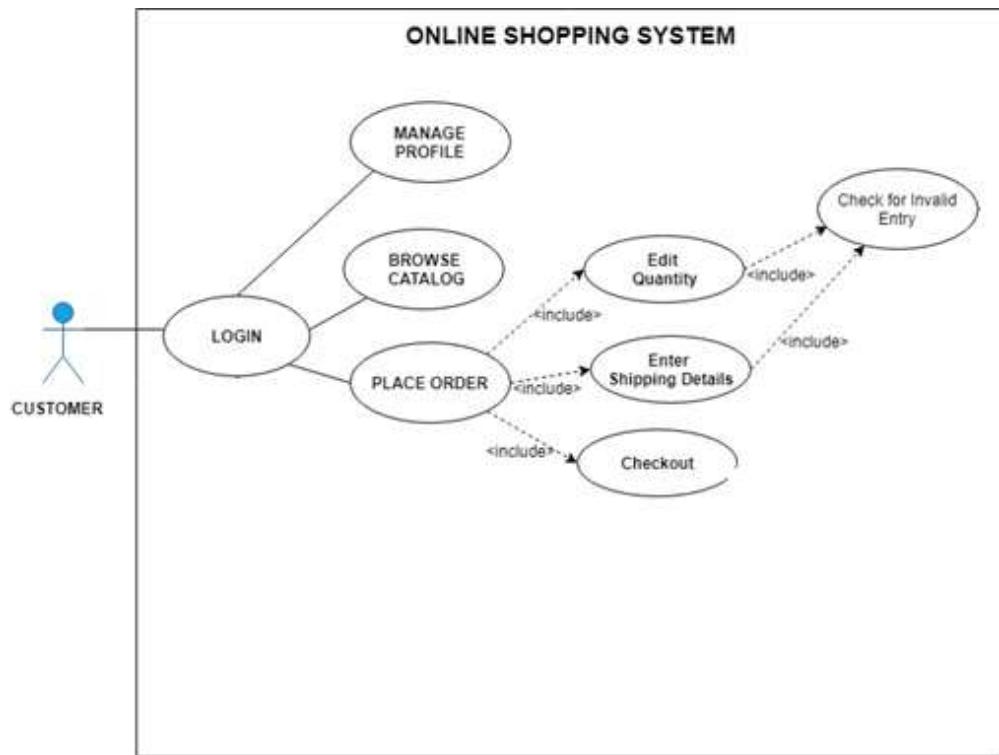
- **Director**

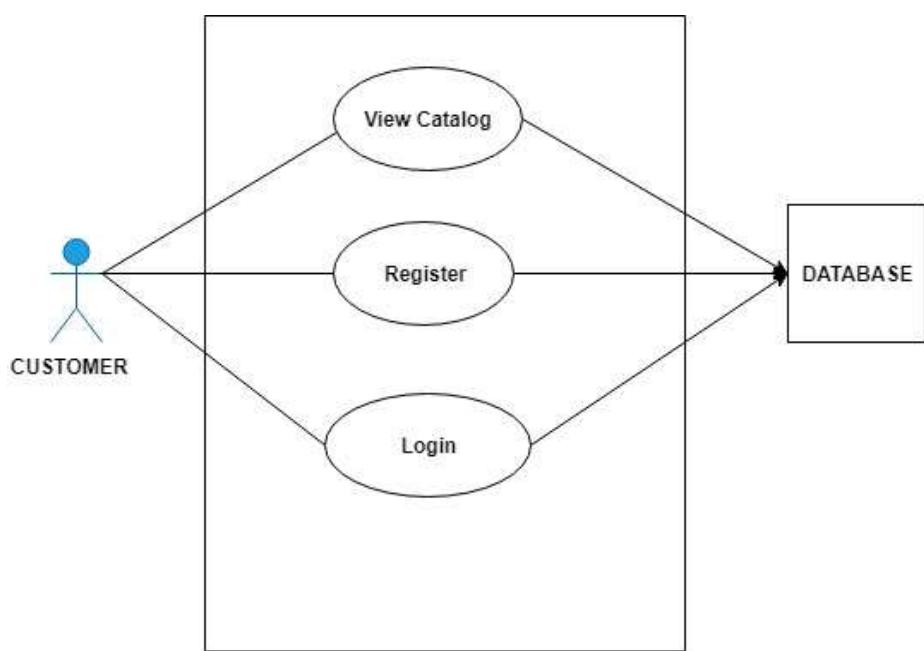
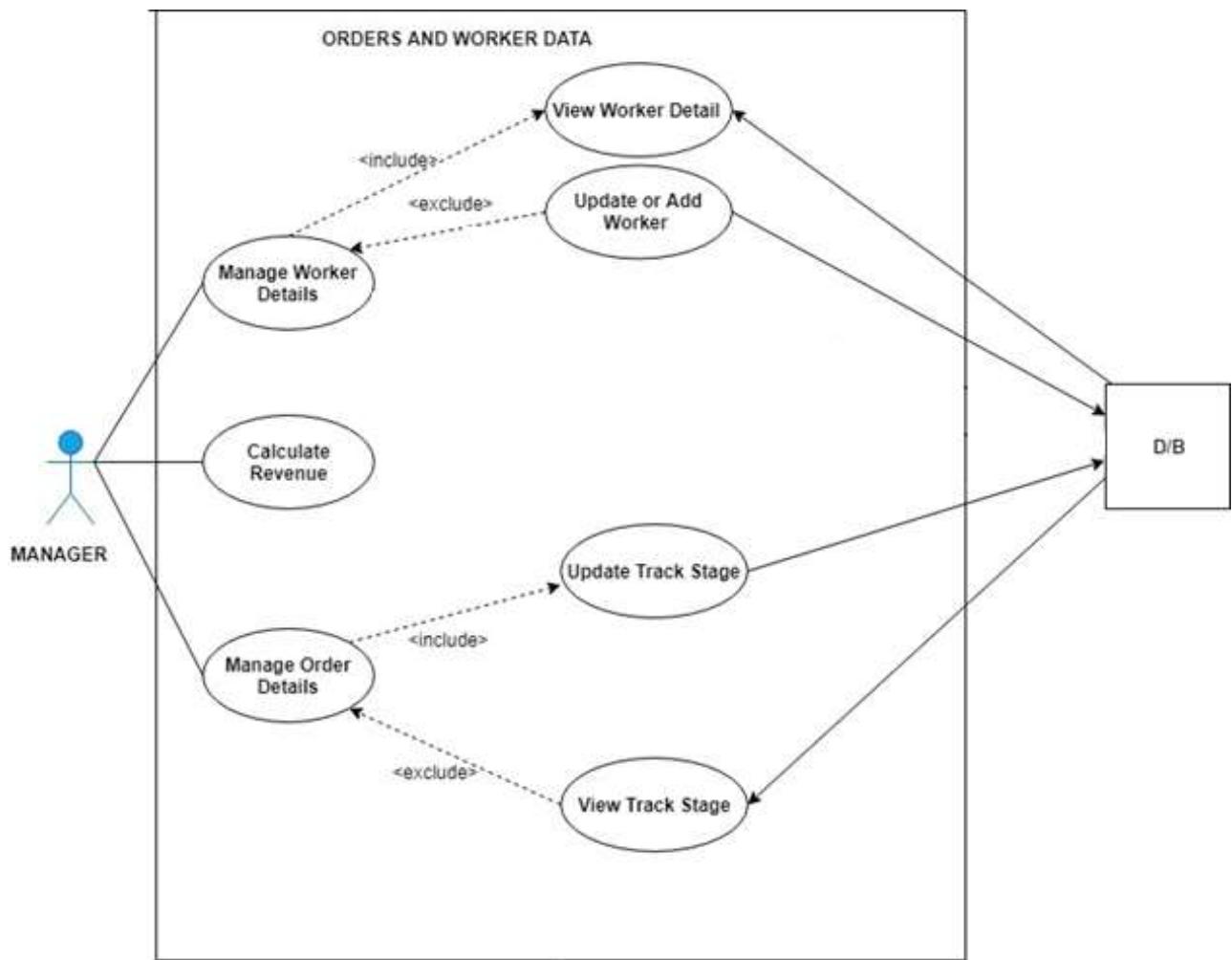
- A UI to manage and look at order and their details
- A screen that will be having a proper tracking UI to manage each stage of the order
- Same screen for the administrator to calculate the revenue
- To manage the workers details.

EVENT TABLE

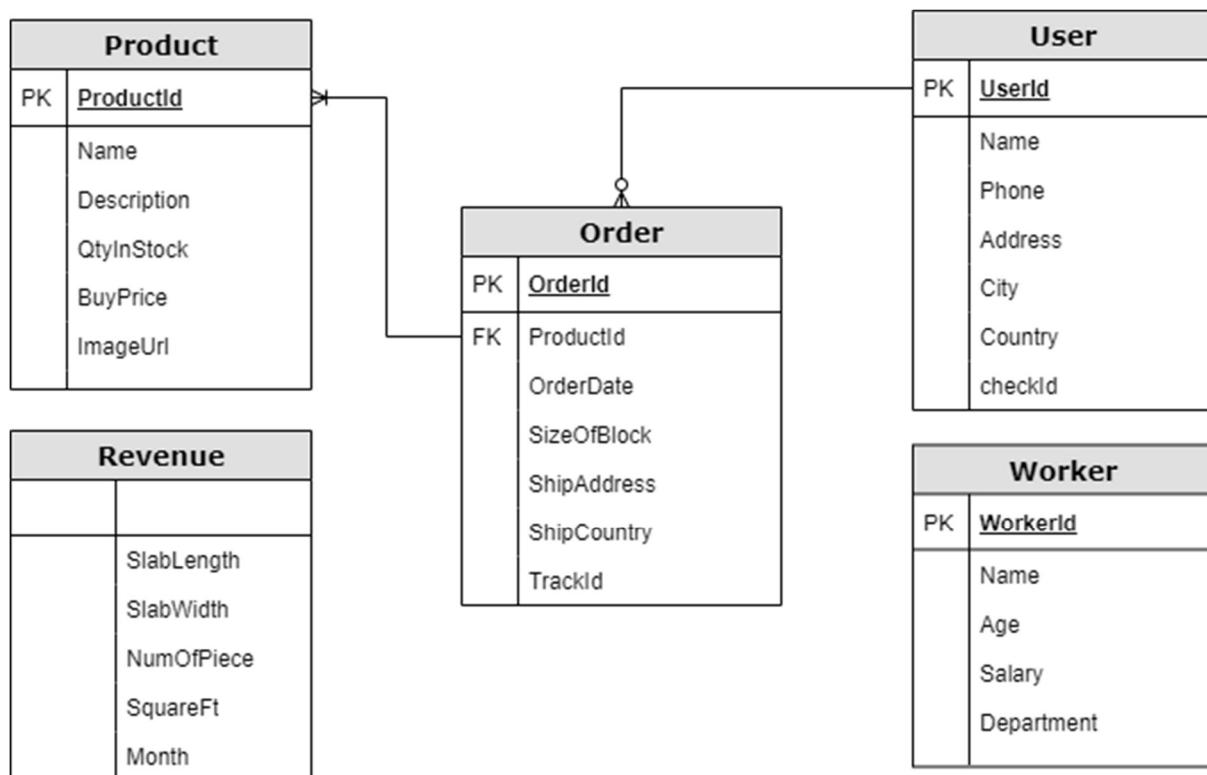
S no.	Event	Trigger	Source	Use Case Activity	Response	Destination
1	User enter login credentials	Opening the web-application and tapping on login option	User	Logs in the user to his/her account	Verifies or asks to Re-enter	Account
2	If the user is Customer, he goes through the catalog and firm's information	Customer taps on the Portfolio option or About option.	Customer	Opens the portfolio of the firm	-	Portfolio page
3	Customer selects a product to Buy it and enters desired qty and size	Customer taps on the Buy option of the particular product	Customer	A new page with all details shown to user	-	Confirmation page
4.	If the User is manger, he selects to check for any new order	Selecting the Order option	Manager	Opens the Order page and retrieve data from database	-	List of all order appears in a grid format
5.	If the manager selects to view/add the data of worker	Selecting Worker option.	Manager	Display the details of workers in grid format with an option to add	Data updated or unsuccessful	Worker's data page
6.	If the user is Director, he selects the order to view track stage	Selecting the order option	Director	Retrieve from database the details of each order	-	Page with list of all orders with their respective track stage

USE CASE DIAGRAM

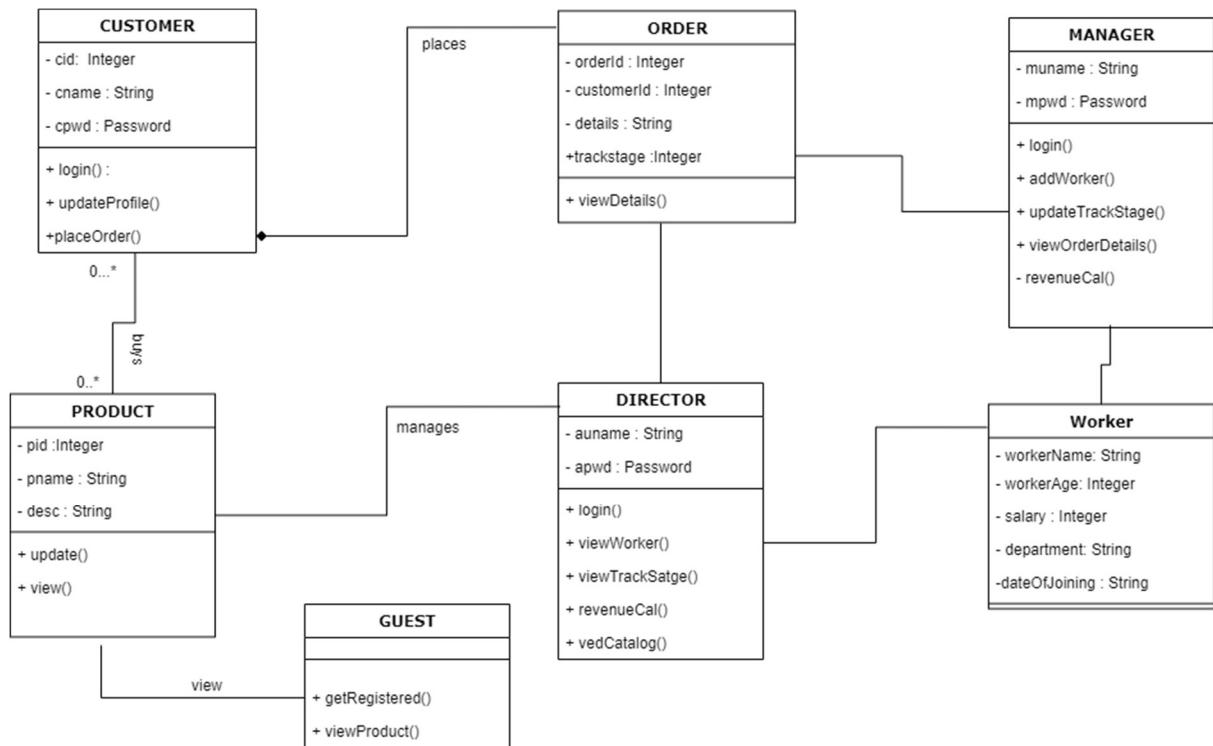




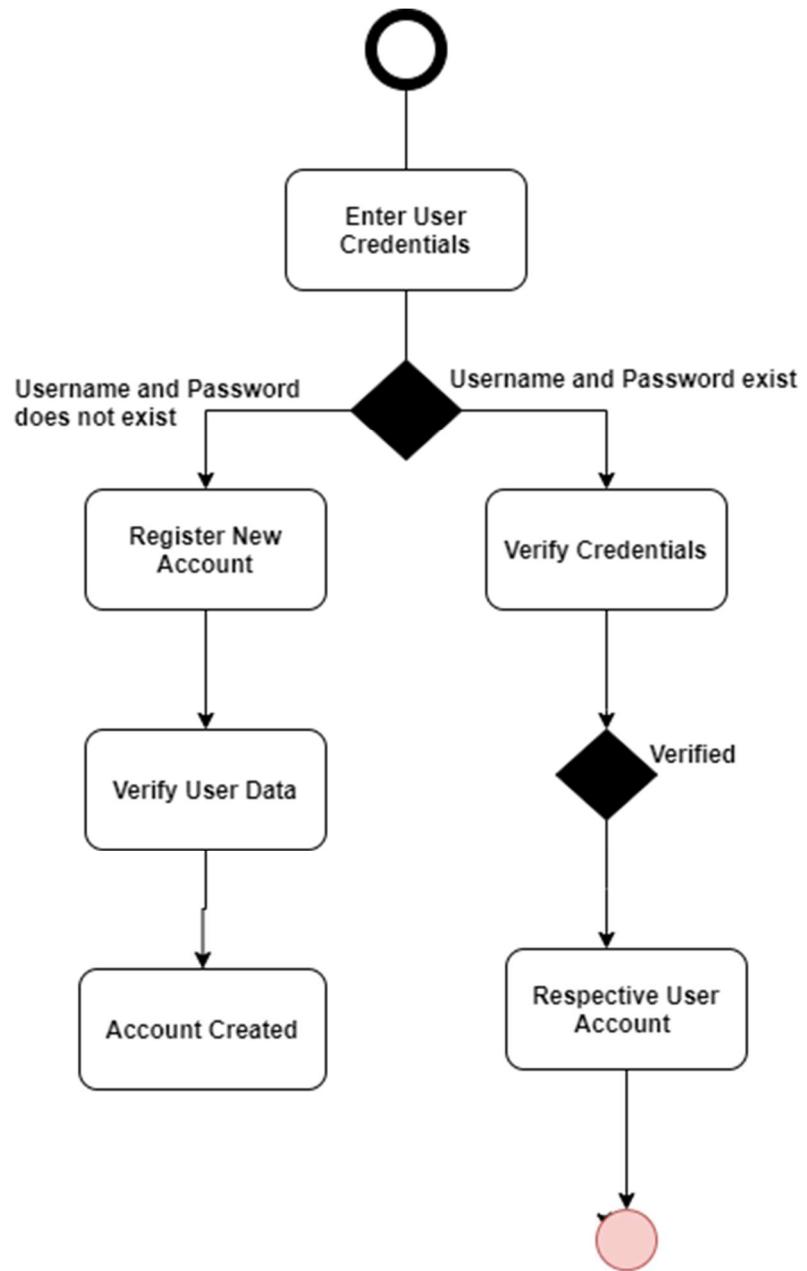
ENTITY RELATION DIAGRAM

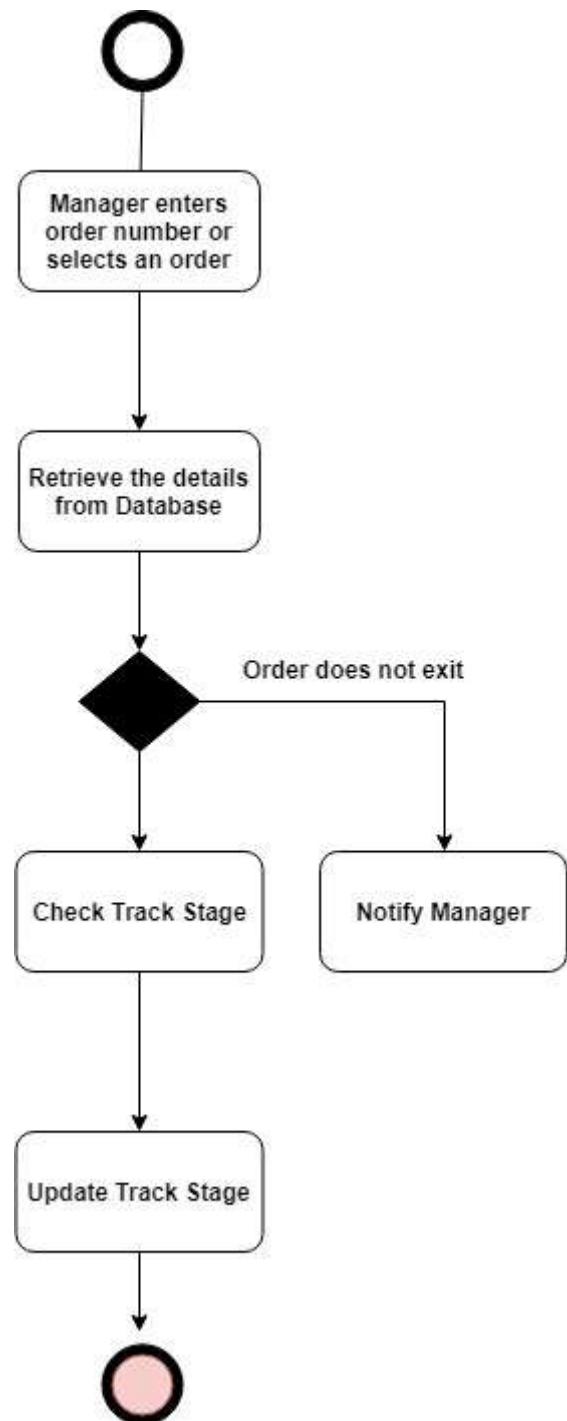


CLASS DIAGRAM

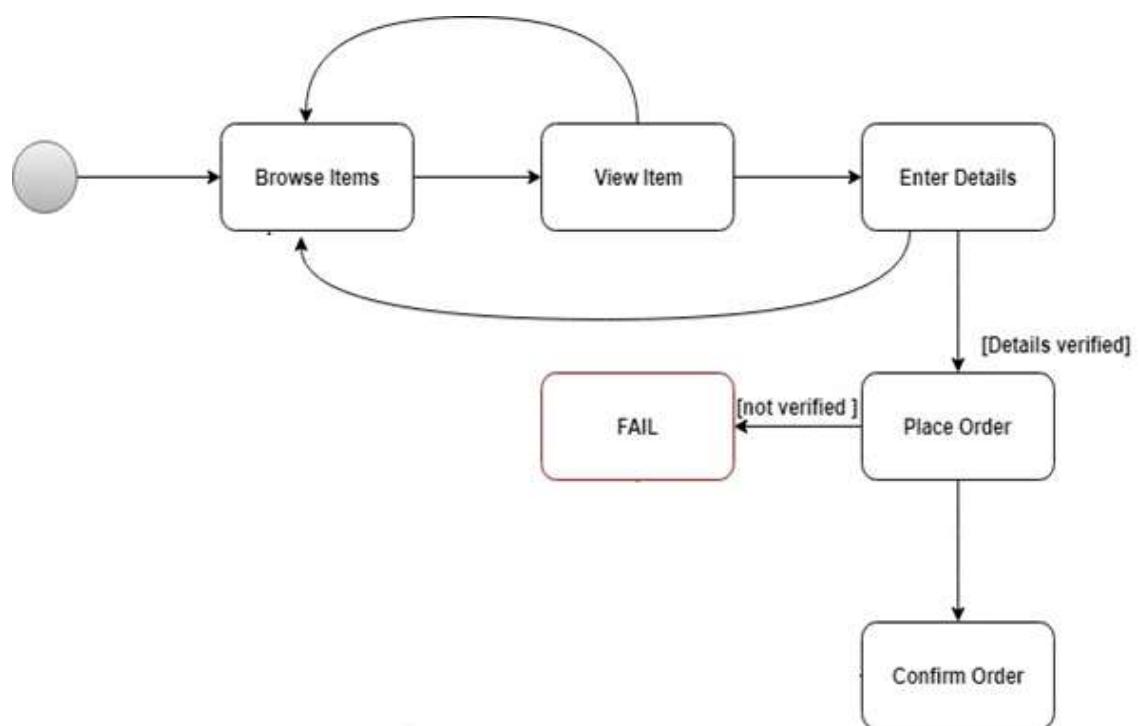
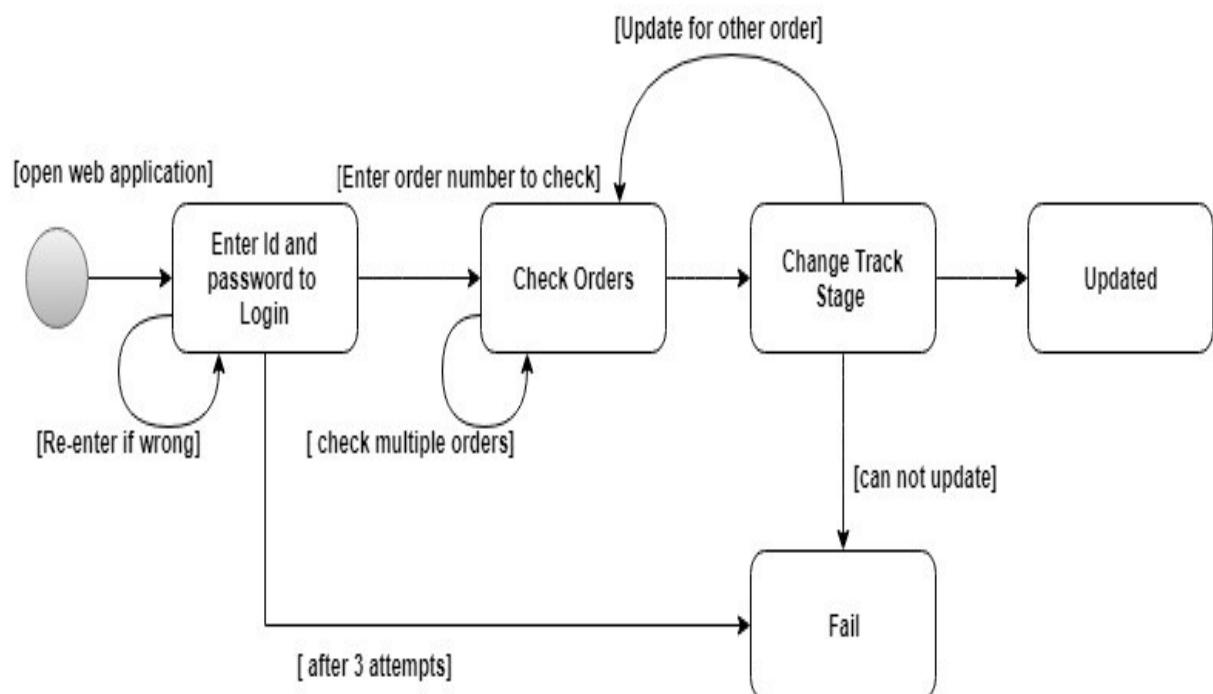


ACTIVITY DIAGRAM

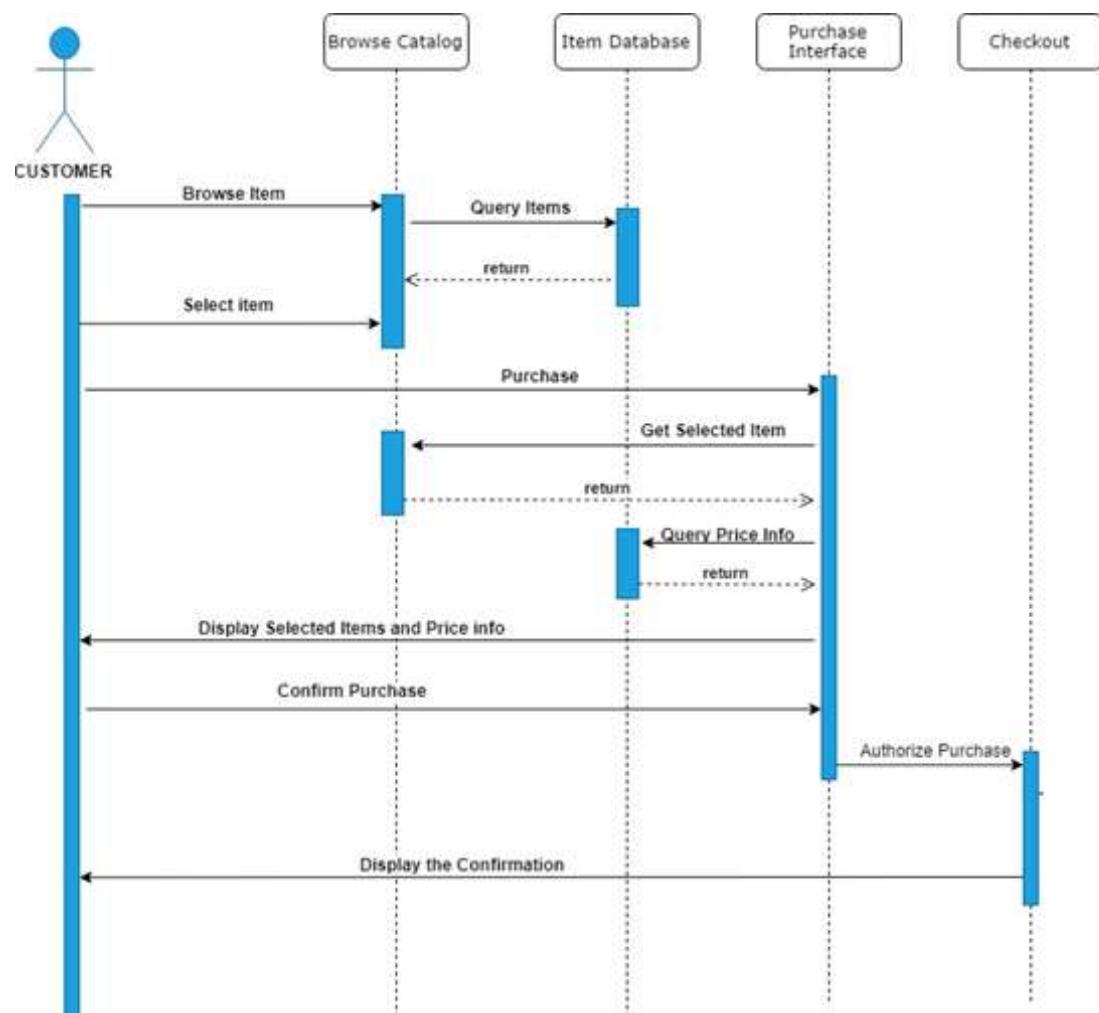


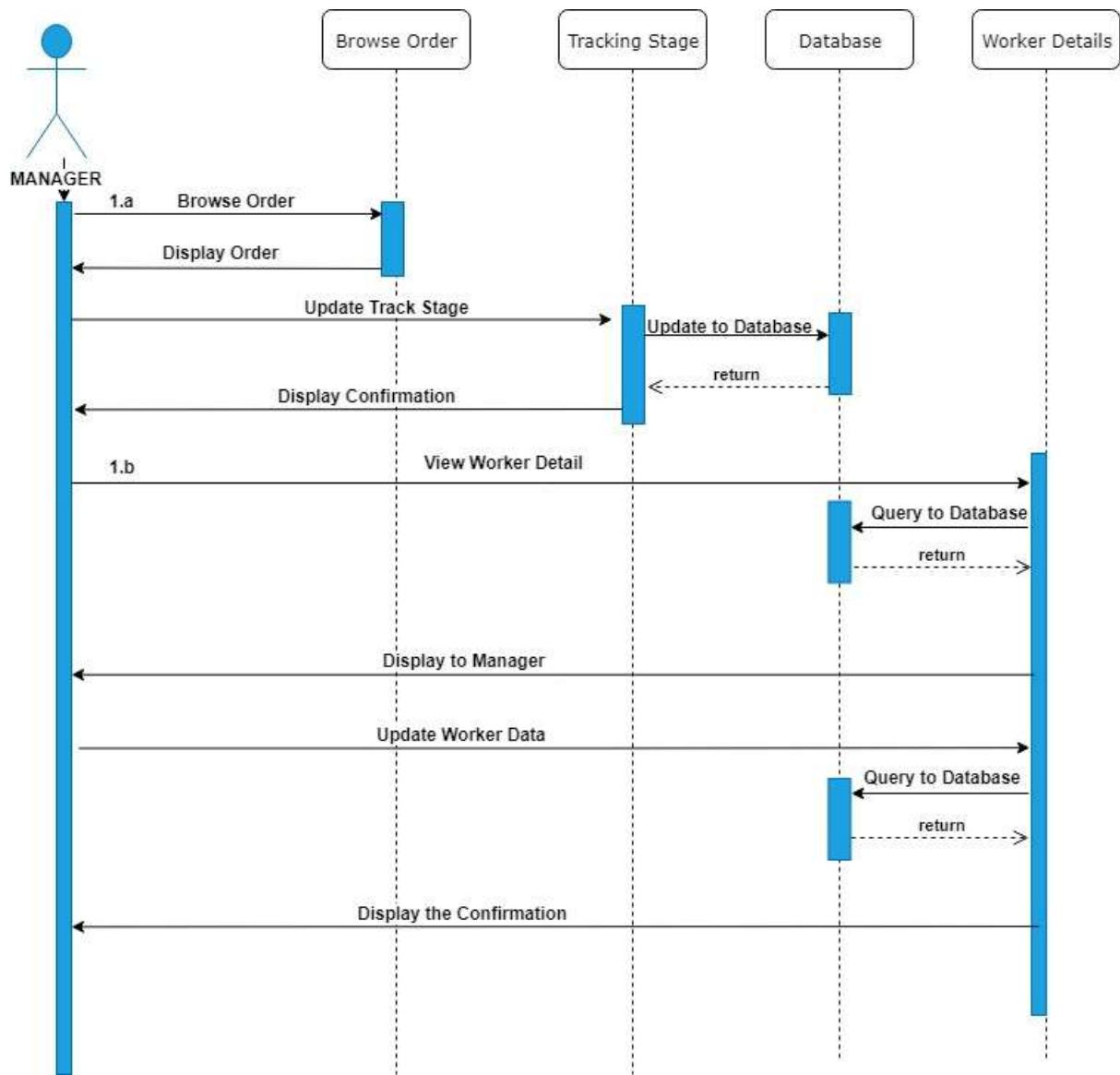


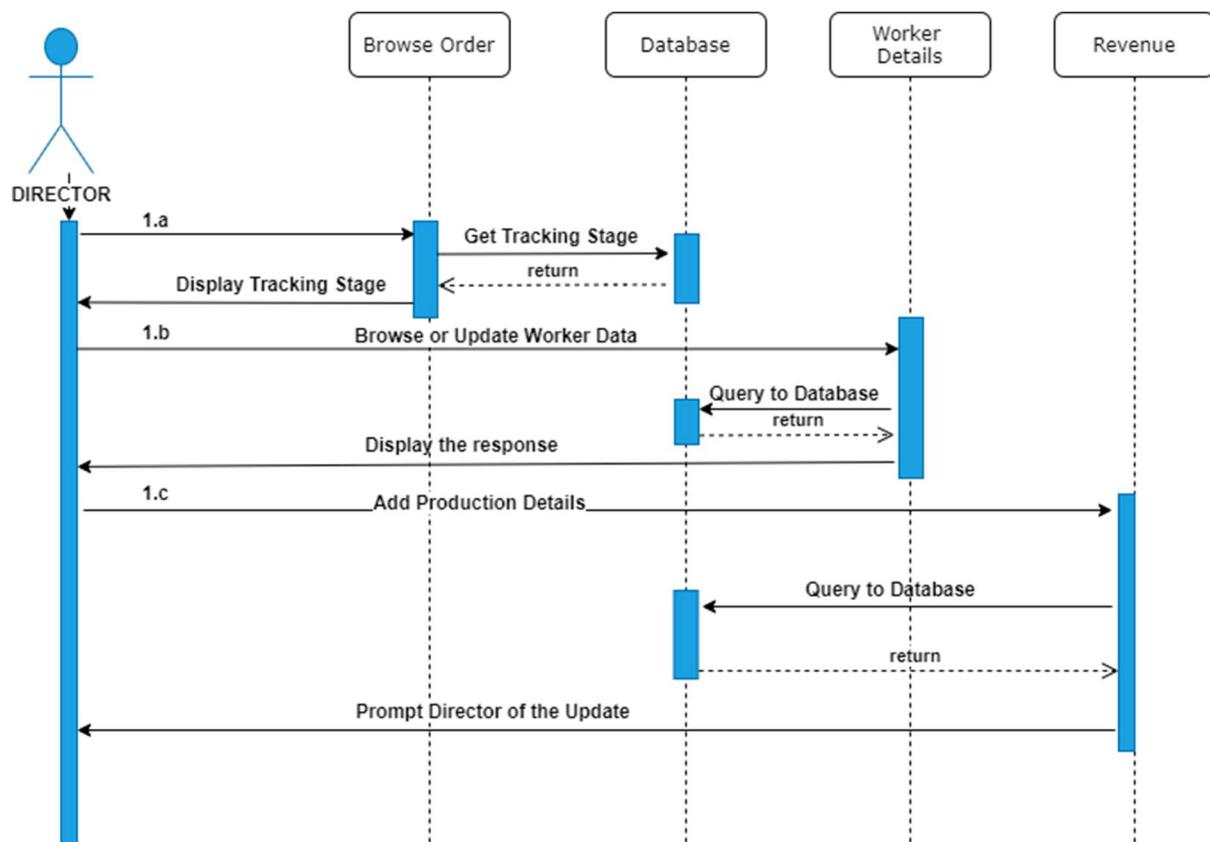
STATE DIAGRAM



SEQUENCE DIAGRAMS







SYSTEM DESIGN

CONVERTING ER DIAGRAM TO TABLES

- To maintain the users information.

```
mysql> desc login;
+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key  | Default | Extra        |
+-----+-----+-----+-----+-----+
| userId | int(11) | NO   | PRI   | NULL    | auto_increment
| password | varchar(50) | YES  |       | NULL    |
| userName | varchar(50) | YES  |       | NULL    |
| name | varchar(50) | YES  |       | NULL    |
| phone | mediumtext | YES  |       | NULL    |
| address | varchar(200) | YES  |       | NULL    |
| city | varchar(50) | YES  |       | NULL    |
| country | varchar(50) | YES  |       | NULL    |
| checkId | tinyint(1) | YES  |       | NULL    |
+-----+-----+-----+-----+-----+
9 rows in set (0.03 sec)
```

- To maintain the order details.

```
mysql> desc orders;
+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key  | Default | Extra        |
+-----+-----+-----+-----+-----+
| orderId | int(11) | NO   | PRI   | NULL    | auto_increment
| userId | int(11) | YES  | MUL   | NULL    |
| image | varchar(50) | YES  |       | NULL    |
| type | varchar(20) | YES  |       | NULL    |
| price | decimal(10,0) | YES  |       | NULL    |
| orderDate | varchar(20) | YES  |       | NULL    |
| size | varchar(20) | YES  |       | NULL    |
| ship_address | varchar(100) | YES  |       | NULL    |
| productId | int(11) | YES  | MUL   | NULL    |
| track_id | int(11) | YES  |       | NULL    |
+-----+-----+-----+-----+-----+
10 rows in set (0.00 sec)
```

- To maintain the product details.

```
mysql> desc product
-> ;
+-----+-----+-----+-----+-----+
| Field | Type   | Null | Key  | Default | Extra        |
+-----+-----+-----+-----+-----+
| productId | int(11) | NO   | PRI   | NULL    | auto_increment
| title | varchar(20) | YES  |       | NULL    |
| description | varchar(410) | YES  |       | NULL    |
| qtyInStock | int(11) | YES  |       | NULL    |
| tilePrice | int(11) | YES  |       | NULL    |
| slabPrice | int(11) | YES  |       | NULL    |
| image | varchar(200) | YES  |       | NULL    |
+-----+-----+-----+-----+-----+
7 rows in set (0.00 sec)
```

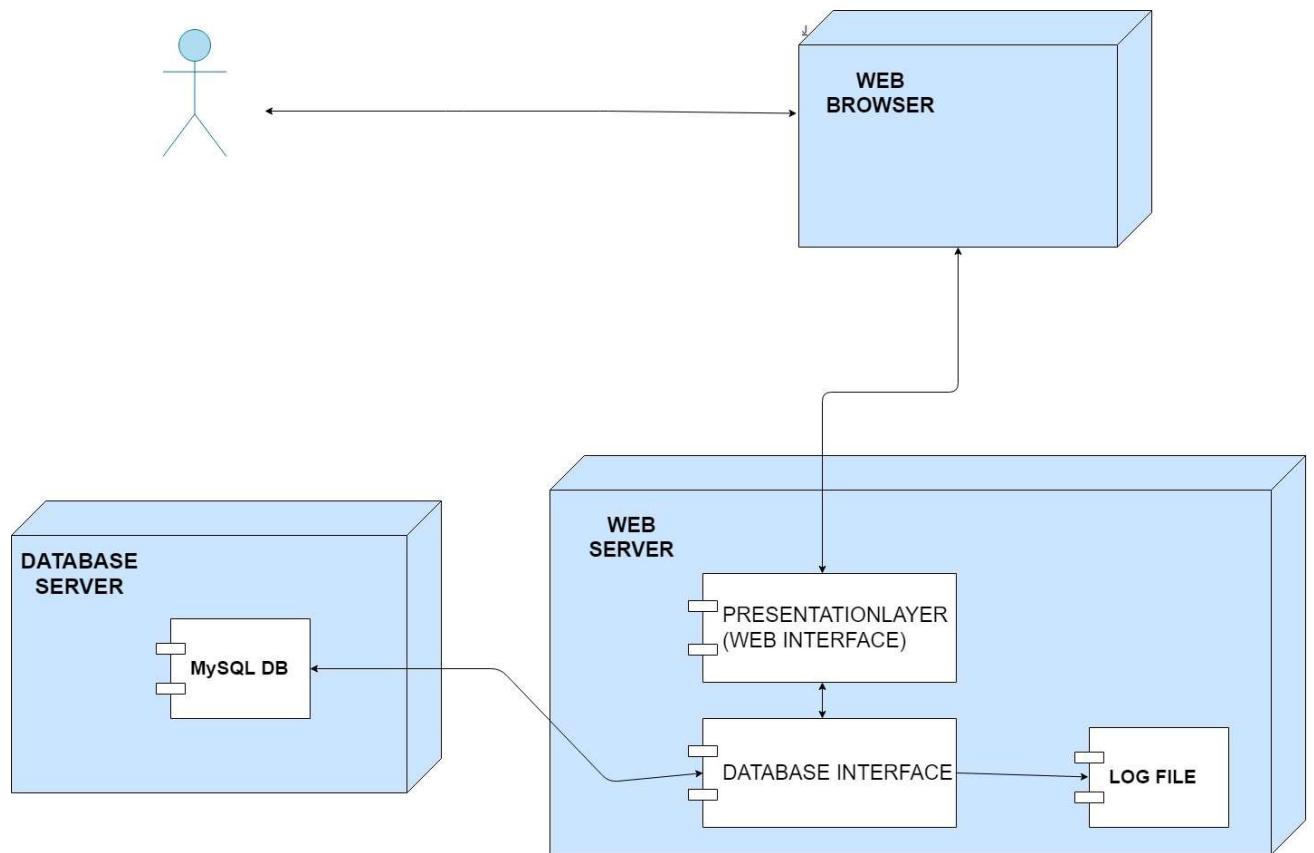
- To maintain the worker details.

```
mysql> desc worker;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| workerId | int(11) | NO | PRI | NULL | auto_increment |
| name | varchar(20) | YES | | NULL |
| age | int(11) | YES | | NULL |
| department | varchar(20) | YES | | NULL |
| phone | mediumtext | YES | | NULL |
| address | varchar(100) | YES | | NULL |
| city | varchar(20) | YES | | NULL |
| month_of_joining | varchar(20) | YES | | NULL |
| salary | mediumtext | YES | | NULL |
+-----+-----+-----+-----+-----+-----+
9 rows in set (0.01 sec)
```

- To maintain the revenue details.

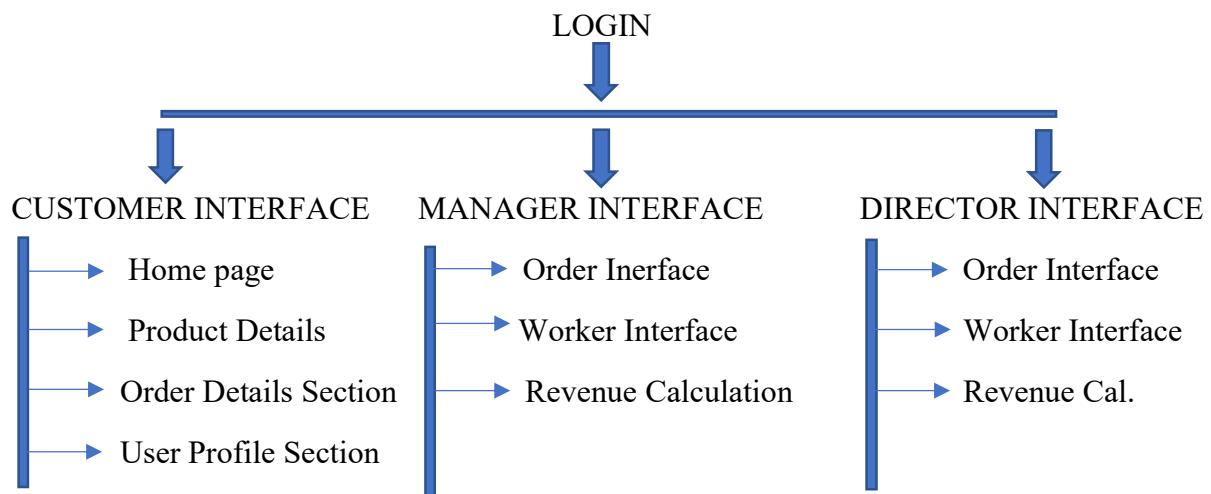
```
mysql> desc revenue;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| slab_length | int(11) | YES | | NULL |
| slab_width | int(11) | YES | | NULL |
| num_of_peice | int(11) | YES | | NULL |
| sq_ft | mediumtext | YES | | NULL |
| month | varchar(20) | YES | | NULL |
| year | int(11) | YES | | NULL |
+-----+-----+-----+-----+-----+
6 rows in set (0.00 sec)
```

DEPLOYMENT DIAGRAM



SYSTEM CODING

SITE MAP



Default.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
namespace WebApplication7
{
    public partial class _Default : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["check_login"]!=null)
            {
                Response.Redirect("BuyNow.aspx");
            }
            if (Session["customerId"] != null)
            {
                hpMyProfile.Visible = true;
                LinkButton1.Visible = true;
                hpLogin.Visible = false;
            }
        }
        protected void LinkButton1_Click(object sender, EventArgs e)
        {
            Session.Clear();
            Response.Redirect("Login.aspx");
        }

        protected void imgProduct2_Click(object sender, ImageClickEventArgs e)
        {
            Session["imgId"] = "imgProduct2";
            Response.Redirect("Product.aspx");
        }

        protected void imgProduct1_Click(object sender, ImageClickEventArgs e)
        {
            Session["imgId"] = "imgProduct1";
            Response.Redirect("Product.aspx");
        }

        protected void imgProduct3_Click(object sender, ImageClickEventArgs e)
        {
            Session["imgId"] = "imgProduct3";
            Response.Redirect("Product.aspx");
        }

        protected void imgProduct4_Click(object sender, ImageClickEventArgs e)
        {
            Session["imgId"] = "imgProduct4";
            Response.Redirect("Product.aspx");
        }
    }
}
```

Login.aspx.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;
using MySql.Data.MySqlClient;
using System.Security.Cryptography;
using System.Text;

namespace WebApplication7
{
    public partial class WebForm2 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["customerId"] != null)
            {
                Session["customerId"] = null;
            }
        }

        protected void btLogin_Click(object sender, EventArgs e)
        {
            DataTable dt = new DataTable();
            string username = tbUserName.Text.ToLower();
            string password = encryption(tbPassword.Text);

            string query = "select * from login where userName=" + "'" + username + "'";
            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();

                using (MySqlCommand cmd = new MySqlCommand(query, conn))
                {
                    using (MySqlDataAdapter sda = new MySqlDataAdapter(cmd))
                    {

                        sda.Fill(dt);
                    }
                }
            }

            if (dt.Rows.Count == 0)
            {
                lbCheck.Text = "Invalid Username or Password";
                lbCheck.Visible = true;
            }
            else
            {
                if (username == "manager" && password ==
dt.Rows[0]["password"].ToString())
                {
                    Session["checkManager"] = 1;
```

```

        Response.Redirect("Manager_Order.aspx");
    }
    else if (username == "admin" && password ==
dt.Rows[0]["password"].ToString())
{
    Session["checkAdmin"] = 1;
    Response.Redirect("Admin_Order.aspx");
}

else if (password == dt.Rows[0]["password"].ToString())
{
    lbCheck.Visible = false;
    Session["customerId"] = dt.Rows[0]["userId"].ToString();
    Response.Redirect("Default.aspx");

}
else
{
    lbCheck.Visible = true;
    lbCheck.Text = "Invalid Username or Password";
}
}

public string encryption(String password)
{
    byte[] data = UTF8Encoding.UTF8.GetBytes(password);
    using (MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider())
    {
        byte[] keys =
md5.ComputeHash(UTF8Encoding.UTF8.GetBytes("KeyForGransells"));
        using (TripleDESCryptoServiceProvider tripDes = new
TripleDESCryptoServiceProvider() { Key = keys, Mode = CipherMode.ECB, Padding =
PaddingMode.PKCS7 })
        {
            ICryptoTransform transform = tripDes.CreateEncryptor();
            byte[] results = transform.TransformFinalBlock(data, 0,
data.Length);
            String pass = Convert.ToBase64String(results, 0, results.Length);
            return pass;
        }
    }
}
protected void btContinue_Click(object sender, EventArgs e)
{
    btContinue.Visible = false;
    lbPass.Visible = true;
    tbPassword.Visible = true;
    btLogin.Visible = true;
    hlForPass.Visible = true;

}

protected void btNewAcc_Click(object sender, EventArgs e)
{
    Response.Redirect("RegForm.aspx");
}

}

```

RegForm.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using MySql.Data.MySqlClient;
using System.Data;
using System.Security.Cryptography;
using System.Text;

namespace WebApplication7
{
    public partial class RegForm : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void Button1_Click(object sender, EventArgs e)
        {
            if (Page.IsValid)
            {

                string name = tbFirstName.Text + " " + tbLastName.Text;
                string email = tbEmail.Text;
                string password = encryption(tbConfirmPassword.Text);
                long phone = Convert.ToInt64(tbPhoneNumber.Text.ToString());
                string address = tbAddress1.Text + ", " + tbAddress2.Text;
                string city = tbCity.Text;
                string country = tbCountry.Text;
                string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";

                string Query = "Insert into
login(password,userName,name,phone,address,city,country,checkId) values('" + password
+ "','" + email + "','" + name + "','" + phone + "','" + address + "','" + city +
"','" + country + "','" + 3 + "')";
                MySqlConnection MyConn2 = new MySqlConnection(str);
                MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
                MySqlDataReader MyReader2;
                MyConn2.Open();
                MyReader2 = MyCommand2.ExecuteReader();
                // MessageBox.Show("Save Data");
                while (MyReader2.Read())
                {
                }
                MyConn2.Close();
                Response.Redirect("Login.aspx");
            }
        }

        public string encryption(String password)
        {
            byte[] data = UTF8Encoding.UTF8.GetBytes(password);
            using (MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider())
            {
```

```

        byte[] keys =
    md5.ComputeHash(UTF8Encoding.UTF8.GetBytes("KeyForGransells"));
        using (TripleDESCryptoServiceProvider tripDes = new
    TripleDESCryptoServiceProvider() { Key = keys, Mode = CipherMode.ECB, Padding =
    PaddingMode.PKCS7 })
    {
        ICryptoTransform transform = tripDes.CreateEncryptor();
        byte[] results = transform.TransformFinalBlock(data, 0,
    data.Length);
        String pass = Convert.ToBase64String(results, 0, results.Length);
        return pass;
    }
}
protected void CustomValidator1_ServerValidate(object source,
ServerValidateEventArgs args)
{
    DataTable dt = new DataTable();
    string username = tbEmail.Text.ToLower();
    string password = tbPassword.Text;

    string query = "select * from login ";
    string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
    using (MySqlConnection conn = new MySqlConnection(str))
    {

        conn.Open();

        using (MySqlCommand cmd = new MySqlCommand(query, conn))
        {
            using (MySqlDataAdapter sda = new MySqlDataAdapter(cmd))
            {

                sda.Fill(dt);
            }
        }
    }
    int flag = 0;
    for (int i = 0; i < dt.Rows.Count; i++)
    {
        if (username == dt.Rows[i]["userName"].ToString())
        {
            args.IsValid = false;
            break;
        }
        else
        {
            flag = 1;
        }
    }
    if (flag == 1)
    {
        args.IsValid = true;
    }
}
}}

```

ForgetPassword.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Net;
using System.Net.Mail;
using System.Data;
using MySql.Data.MySqlClient;
using System.Security.Cryptography;
using System.Text;

namespace WebApplication7
{
    public partial class ForgotPassword : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {

        }

        protected void btLogin_Click(object sender, EventArgs e)
        {
            DataTable dt = new DataTable();
            string username = tbUserName.Text.ToLower();

            string query = "select * from login where userName=" + "'" + username +
""";
            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();

                using (MySqlCommand cmd = new MySqlCommand(query, conn))
                {
                    using (MySqlDataAdapter sda = new MySqlDataAdapter(cmd))
                    {

                        sda.Fill(dt);
                    }
                }
                if (dt.Rows.Count != 0 && tbUserName.Text!="manager" &&
tbUserName.Text!="admin")
                {
                    lbCheck.Visible = true;
                    MailMessage Msg = new MailMessage();
                    // Sender e-mail address.
                    Msg.From = new MailAddress("yashbabel99@gmail.com");
                    // Recipient e-mail address.
                    String dec = dt.Rows[0]["Password"].ToString();
                    string pass = decrypt(dec);
                    Msg.To.Add(tbUserName.Text);
                    Msg.Subject = "Your Password Details";
                }
            }
        }
    }
}
```

```

        Msg.Body = "Hi, <br/>Please check your Login  

Detailss<br/><br/>Your Username: " + dt.Rows[0]["UserName"] + "<br/><br/>Your  

Password: " + pass + "<br/><br/>";
        Msg.IsBodyHtml = true;
        // your remote SMTP server IP.
        SmtpClient smtp = new SmtpClient("smtp.gmail.com", 587);
        smtp.Host = "smtp.gmail.com";
        smtp.Port = 587;
        smtp.UseDefaultCredentials = false;
        smtp.Credentials = new
System.Net.NetworkCredential("yashbabel99@gmail.com", "babel6867");
        smtp.EnableSsl = true;

        smtp.Send(Msg);
        //Msg = null;
        lbCheck.Text = "Your Password Details Sent to your mail";
        lbCheck.Visible = true;
        // Clear the textbox values
        tbUserName.Text = "";
    }
    else
    {
        lbCheck.Text = "Please enter a valid Email address";
        lbCheck.Visible = true;
    }
}
private static string decrypt(string pass)
{
    byte[] data = Convert.FromBase64String(pass); // decrypt the incrypted
text
    using (MD5CryptoServiceProvider md5 = new MD5CryptoServiceProvider())
    {
        byte[] keys =
md5.ComputeHash(UTF8Encoding.UTF8.GetBytes("KeyForGransells"));
        using (TripleDESCryptoServiceProvider tripDes = new
TripleDESCryptoServiceProvider() { Key = keys, Mode = CipherMode.ECB, Padding =
PaddingMode.PKCS7 })
        {
            ICryptoTransform transform = tripDes.CreateDecryptor();
            byte[] results = transform.TransformFinalBlock(data, 0,
data.Length);
            string result = UTF8Encoding.UTF8.GetString(results);
            return result;
        }
    }
}
}
}

```

Product.aspx.cs

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

namespace WebApplication7
{
    public partial class Product : System.Web.UI.Page
    {
        DataTable dt = new DataTable();
        double sw;
        double sl;
        int inp;
        protected void Page_Load(object sender, EventArgs e)
        {

            if (Session["customerId"] != null)
            {
                hpMyProfile.Visible = true;
                LinkButton1.Visible = true;
                hpLogin.Visible = false;
            }

            string productId = Session["imgId"].ToString();
            if (productId == "imgProduct1")
            {
                productId = "1";
            }
            else if (productId == "imgProduct2")
            {
                productId = "2";
            }
            else if (productId == "imgProduct3")
            {
                productId = "3";
            }
            else
            {
                productId = "4";
            }

            string query = "select * from product where productId=" + productId;
            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            using (MySqlConnection conn = new MySqlConnection(str))
```

```

        {

            conn.Open();
            using (MySqlCommand cmd = new MySqlCommand(query, conn))
            {

                using (MySqlDataAdapter sda = new MySqlDataAdapter(cmd))
                {

                    sda.Fill(dt);

                }
            }

        }
        if (!Page.IsPostBack)
        {
            Label2.Text = dt.Rows[0]["title"].ToString();
            Label3.Text = dt.Rows[0]["description"].ToString();
            Image1.ImageUrl = dt.Rows[0]["image"].ToString();

        }

    }
protected void LinkButton1_Click(object sender, EventArgs e)
{
    Session.Clear();
    Response.Redirect("Login.aspx");
}
protected void ddComapanyFilter_SelectedIndexChanged(object sender, EventArgs e)
{
    if (DropDownList1.SelectedValue == "0")//All
    {
        lbPpsf.Visible = false;
        lbSelect.Visible = false;
        lbPrice.Visible = false;
        tilesPrice.Visible = false;
        TextBox1.Visible = false;
        TotalPrice.Visible = false;
        slabPriceHeight.Visible = false;
        SlabPriceWidth.Visible = false;
        lbrs.Visible = false;
        lbPeicesReq.Visible = false;

    }
    else if (DropDownList1.SelectedValue == "1")//Flagged
    {
        lbPpsf.Visible = true;
        lbrs.Visible = false;
        lbPeicesReq.Visible = false;
        lbSelect.Visible = false;
        lbPrice.Visible = false;
        slabPriceHeight.SelectedIndex = 0;
        SlabPriceWidth.SelectedIndex = 0;

        TextBox1.Visible = false;
        TotalPrice.Visible = false;

        tilesPrice.Text = dt.Rows[0]["slabPrice"].ToString();
        tilesPrice.Visible = true;
    }
}

```

```

        slabPriceHeight.Visible = true;
        SlabPriceWidth.Visible = true;
        lbSelect.Visible = true;

    }

    else if (DropDownList1.SelectedValue == "2")//New
    {

        //code
        lbPpsf.Visible = true;
        lbrs.Visible = false;
        lbSelect.Visible = false;
        lbPrice.Visible = false;
        TextBox1.Text = "";
        slabPriceHeight.Visible = false;
        SlabPriceWidth.Visible = false;
        TotalPrice.Visible = false;
        tilesPrice.Visible = true;
        tilesPrice.Text = dt.Rows[0]["tilePrice"].ToString();
        lbPeicesReq.Visible = true;
        TextBox1.Visible = true;

    }
}

protected void TextBox1_TextChanged(object sender, EventArgs e)
{
    try
    {
        inp = int.Parse(TextBox1.Text);
        int per = Convert.ToInt32(dt.Rows[0]["tilePrice"]);
        int cal = inp * per;
        TotalPrice.Text = cal.ToString();
        TotalPrice.Visible = true;
        lbPrice.Visible = true;
        lbrs.Visible = true;
        RegularExpressionValidator1.Visible = false;
    }
    catch
    {
        RegularExpressionValidator1.Visible = true;
    }

}

protected void slabPriceHeight_SelectedIndexChanged(object sender, EventArgs e)
{
    if (slabPriceHeight.SelectedValue != "0" && SlabPriceWidth.SelectedValue
!= "0")
    {

        double sl = 0;
        double sw = 0;
    }
}

```

```

        sl = Convert.ToDouble(slabPriceHeight.SelectedValue);
        sw = Convert.ToDouble(SlabPriceWidth.SelectedValue);
        TotalPrice.Visible = true;
        lbPrice.Visible = true;
        ltrs.Visible = true;
        TotalPrice.Text = (sl * sw *
Convert.ToInt32(dt.Rows[0]["slabPrice"])).ToString();
    }
    else
    {
        lbPrice.Visible = false;
        ltrs.Visible = false;
        TotalPrice.Visible=false;
    }

}

public void SlabPriceWidth_SelectedIndexChanged(object sender, EventArgs e)
{
    if (slabPriceHeight.SelectedValue != "0" && SlabPriceWidth.SelectedValue
!= "0")
    {
        double sw = 0;
        double sl = 0;

        sw = Convert.ToDouble(SlabPriceWidth.SelectedValue);
        sl = Convert.ToDouble(slabPriceHeight.SelectedValue);
        TotalPrice.Visible = true;
        ltrs.Visible = true;
        lbPrice.Visible = true;
        TotalPrice.Text = (sl*sw *
Convert.ToInt32(dt.Rows[0]["slabPrice"])).ToString();
    }
    else
    {
        lbPrice.Visible = false;
        ltrs.Visible = false;
        TotalPrice.Visible = false;
    }
}

protected void Button2_Click(object sender, EventArgs e)
{
    string productId = Session["imgId"].ToString();
    if (productId == "imgProduct1")
    {
        productId = "1";
    }
    else if (productId == "imgProduct2")
    {
        productId = "2";
    }
    else if (productId == "imgProduct3")
    {
        productId = "3";
    }
    else
    {
}

```

```
        productId = "4";
    }
    if (DropDownList1.SelectedValue == "1")
    {
        Session["slbH"] = slabPriceHeight.SelectedItem.ToString();
        Session["slbW"] = SlabPriceWidth.SelectedItem.ToString();
        Session["check_slab_tile"] = "SLAB";
    }
    else if (DropDownList1.SelectedValue == "2")
    {
        Session["tile_size"] = TextBox1.Text;
        Session["check_slab_tile"] = "TILE";
    }
    Session["pid_BuyNow"] = productId;
    Session["price"] = TotalPrice.Text;
    if (Session["customerId"] == null)
    {
        Session["check_login"] = "1";
        Response.Redirect("Login.aspx");
    }
    else
    {
        Response.Redirect("BuyNow.aspx");
    }
}

}

}

}
```

BuyNow.aspx.cs

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using MySql.Data.MySqlClient;
using System.Data;
using System.Net.Mail;

namespace WebApplication7
{
    public partial class WebForm1 : System.Web.UI.Page
    {

        DataTable dt1 = new DataTable();
        DataTable dt2 = new DataTable();
        string image_url, size, address;
        protected void Page_Load(object sender, EventArgs e)
        {

            Session["check_login"] = null;
            if (Session["customerId"] != null)
            {
                hpMyProfile.Visible = true;
                LinkButton1.Visible = true;
                hpLogin.Visible = false;
            }
            else
            {
                Response.Redirect("Default.aspx");
            }

            int productIdForImage = Convert.ToInt32(Session["pid_BuyNow"].ToString());
            string query = "select * from product where productId=" +
productIdForImage;
            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();
                using (MySqlCommand cmd1 = new MySqlCommand(query, conn))
                {
                    using (MySqlDataAdapter sda1 = new MySqlDataAdapter(cmd1))
                    {

                        sda1.Fill(dt1);
                    }
                }
            }
            string query2 = "select * from login where userId=" +
Convert.ToInt16(Session["customerId"]);
            using (MySqlConnection conn = new MySqlConnection(str))
            {
```

```

        conn.Open();
        using (MySqlCommand cmd2 = new MySqlCommand(query2, conn))
        {
            using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
            {

                sda2.Fill(dt2);

            }
        }

    }
image_url = dt1.Rows[0]["image"].ToString();
imgOrderImage.ImageUrl = image_url;
tbpProductTitle.Text = dt1.Rows[0]["title"].ToString();
tbNameOfCustomer.Text = dt2.Rows[0]["name"].ToString().ToUpper();
Session["email"] = dt2.Rows[0]["userName"].ToString();
address = dt2.Rows[0]["address"].ToString() + ", " +
dt2.Rows[0]["city"].ToString() + ", " + dt2.Rows[0]["country"].ToString();
tbShipAddress.Text = address;
tbPrice.Text = Session["price"].ToString();
tbSlabOrTile.Text = Session["check_slab_tile"].ToString();
if (Session["slbH"] != null && Session["slbW"]!= null)
{
    size=Session["slbH"].ToString() + " X " + Session["slbW"].ToString();
    tbSize.Text = size;

}
else if (Session["tile_size"] != null)
{
    size=Session["tile_size"].ToString();
    tbSize.Text = size;
}

}

protected void LinkButton1_Click(object sender, EventArgs e)
{
    Session.Clear();
    Response.Redirect("Login.aspx");
}

protected void btnPayment_Click(object sender, EventArgs e)
{
    string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";

    string Query = "Insert into
orders(userId,image,type,price,orderDate,size,ship_address,productId,track_id)
values('" + Convert.ToInt16(Session["customerId"]) + "','" + image_url + "','" +
Session["check_slab_tile"].ToString() + "','" + Session["price"].ToString() + "','" +
System.DateTime.Now.ToShortDateString() + "','" + size + "','" + address + "','" +
Convert.ToInt32(Session["pid_BuyNow"]) + "','" + "+0+ "')";;
    MySqlConnection MyConn2 = new MySqlConnection(str);
    MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
    MySqlDataReader MyReader2;
    MyConn2.Open();
    MyReader2 = MyCommand2.ExecuteReader();
    Response.Write("<script>alert('Order Placed');</script>");
}

```

```

        while (MyReader2.Read())
    {
    }
    MyConn2.Close();
    confEmail();
    Response.Redirect("Confirm.aspx");

}
public void confEmail()
{
    try
    {
        MailMessage Msg = new MailMessage();
        // Sender e-mail address.
        Msg.From = new MailAddress("yashbabel99@gmail.com");
        // Recipient e-mail address.

        Msg.To.Add(Session["email"].ToString());
        Msg.Subject = "Greetings from Gransells. Your order is placed";
        Msg.Body = "Hi, <br/>Thanks for an placing order with
Gransells.<br/>Your order details are:<br/>Type: " +
Session["check_slab_tile"].ToString() + "<br/>Size: " + tbSize.Text + " <br/>Please
issue a check on the name 'Granites Pvt. Ltd.' of ₹ " + Session["price"] + " to
confirm your order. You will shortly receive a call from us.";
        Msg.IsBodyHtml = true;
        // your remote SMTP server IP.
        SmtpClient smtp = new SmtpClient("smtp.gmail.com", 587);
        smtp.Host = "smtp.gmail.com";
        smtp.Port = 587;
        smtp.UseDefaultCredentials = false;
        smtp.Credentials = new
System.Net.NetworkCredential("yashbabel99@gmail.com", "babel6867");
        smtp.EnableSsl = true;

        smtp.Send(Msg);
    }
    catch
    {
        Response.Redirect("Confirm.aspx");
    }
}
}

```

Confirm.aspx.cs

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

namespace WebApplication7
{
    public partial class WebForm3 : System.Web.UI.Page
    {
        protected void Page_Load(object sender, EventArgs e)
        {
            if(Session["customerId"]==null)
            {
                Response.Redirect("Default.aspx");
            }
        }

        protected void btContinueShop_Click(object sender, EventArgs e)
        {
            Response.Redirect("Default.aspx");
        }
    }
}
```

UserProfile.aspx.cs

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

namespace WebApplication7
{
    public partial class UserProfile : System.Web.UI.Page
    {
        string newFirstName=null,newUserName=null, newAddress=null, newCity=null,
newCountry=null;
        long newPhone = 0;
        protected void Page_Load(object sender, EventArgs e)
        {
            DataTable dt3 = new DataTable();
            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            string query3 = "select image,type,size,price,orderDate,ship_address  from
orders where userId=" + Convert.ToInt16(Session["customerId"]);
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();
                using (MySqlCommand cmd2 = new MySqlCommand(query3, conn))
                {
                    using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
                    {

                        sda2.Fill(dt3);
                        GridView1.DataSource = dt3;
                        //GridView1.Columns.RemoveAt(1);
                        GridView1.GridLines = GridLines.None;
                        GridView1.CellPadding = 50;
                        GridView1.DataBind();

                    }
                }
                conn.Close();
            }
            if (!Page.IsPostBack)
            {

                DataTable dt = new DataTable();
                if (Session["customerId"] != null)
                {
                    hpMyProfile.Visible = true;
                    LinkButton1.Visible = true;
                    hpLogin.Visible = false;
                }

                string query2 = "select * from login where userId=" +
Convert.ToInt16(Session["customerId"]);
                using (MySqlConnection conn = new MySqlConnection(str))
                {
```

```

        conn.Open();
        using (MySqlCommand cmd2 = new MySqlCommand(query2, conn))
        {
            using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
            {

                sda2.Fill(dt);

            }
        }
        conn.Close();
    }

    tbFirstName.Text = dt.Rows[0]["name"].ToString();

    tbPhoneNumber.Text = dt.Rows[0]["phone"].ToString();
    tbAddress1.Text = dt.Rows[0]["address"].ToString();
    tbCity.Text = dt.Rows[0]["city"].ToString();
    tbCountry.Text = dt.Rows[0]["country"].ToString();
}
}

protected string FormatImageUrl(string url)
{
    if (url != null && url.Length > 0)

        return ("~/" + url);

    else return null;
}

protected void LinkButton1_Click(object sender, EventArgs e)
{
    Session.Clear();
    Response.Redirect("Login.aspx");
}

protected void tbFirstName_TextChanged(object sender, EventArgs e)
{
    newFirstName = tbFirstName.Text;
}

protected void tbPhoneNumber_TextChanged(object sender, EventArgs e)
{
    newPhone = Convert.ToInt64(tbPhoneNumber.Text.ToString());
}

protected void tbCity_TextChanged(object sender, EventArgs e)
{
    newCity = tbCity.Text;
}

protected void tbAddress1_TextChanged(object sender, EventArgs e)
{
    newAddress = tbAddress1.Text;
}

```

```

protected void tbCountry_TextChanged(object sender, EventArgs e)
{
    newCountry = tbCountry.Text;
}

protected void btSaveChanges_Click(object sender, EventArgs e)
{
    string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
    MySqlConnection MyConn2 = new MySqlConnection(str);
    if (newFirstName != null)
    {
        string Query = "update login set name='" + newFirstName + "' where
userId=" + Convert.ToInt16(Session["customerId"]);
        MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
        MyConn2.Open();
        MyCommand2.ExecuteNonQuery();
        MyConn2.Close();
    }

    if (newPhone != 0)
    {
        string Query = "update login set phone='"
+ newPhone + "' where
userId=" + Convert.ToInt16(Session["customerId"]);
        MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
        MyConn2.Open();
        MyCommand2.ExecuteNonQuery();
        MyConn2.Close();
    }
    if (newAddress != null)
    {
        string Query = "update login set address='"
+ newAddress + "' where
userId=" + Convert.ToInt16(Session["customerId"]);
        MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
        MyConn2.Open();
        MyCommand2.ExecuteNonQuery();
        MyConn2.Close();
    }
    if (newCity != null)
    {
        string Query = "update login set city='"
+ newCity + "' where userId="
+ Convert.ToInt16(Session["customerId"]);
        MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
        MyConn2.Open();
        MyCommand2.ExecuteNonQuery();
        MyConn2.Close();
    }
    if (newCountry != null)
    {
        string Query = "update login set country='"
+ newCountry + "' where
userId=" + Convert.ToInt16(Session["customerId"]);
        MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
        MyConn2.Open();
        MyCommand2.ExecuteNonQuery();
        MyConn2.Close();
    }
}
}

```

Manager_Order.aspx.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;
using MySql.Data.MySqlClient;

namespace WebApplication7
{
    public partial class WebForm4 : System.Web.UI.Page
    {
        DataTable dt3 = new DataTable();

        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["checkManager"] == null)
            {
                Response.Redirect("Login.aspx");
            }
            if (!Page.IsPostBack)
            {
                track.Visible = false;
                updateTrackStage.Visible = false;
                bind();
            }
        }

        public void bind()
        {
            // DataTable dt3 = new DataTable();

            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            string query3 = "select
orderId,image,type,size,price,orderDate,ship_address,track_id  from orders ";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();
                using (MySqlCommand cmd2 = new MySqlCommand(query3, conn))
                {
                    using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
                    {

                        sda2.Fill(dt3);
                        GridView1.DataSource = dt3;
                        Session.Add("bindOrder", dt3);
                        //GridView1.Columns.RemoveAt(1);
                        GridView1.GridLines = GridLines.None;
                        GridView1.CellPadding = 50;
                        GridView1.DataBind();
                    }
                }
            }
        }
    }
}
```

```

        conn.Close();

    }
}

protected void LinkButton1_Click(object sender, EventArgs e)
{
    Session.Clear();
    Response.Redirect("Login.aspx");
}

protected void Tracking(object sender, GridViewCommandEventArgs e)
{
    btShowAll.Visible = true;
    int index = Convert.ToInt32(e.CommandArgument);
    Session["index"] = index;
    track.Visible = true;
    updateTrackStage.Visible = true;

    dt3 = (DataTable)Session["bindOrder"];

    int orderID;
    orderID = Convert.ToInt16(dt3.Rows[index]["orderId"]);
    Session["OrderId"] = orderID;
    int count=GridView1.Rows.Count;
    for (int i = 0; i < count; i++)
    {
        if (i == index)
            continue;
        else
            GridView1.Rows[i].Visible = false;
    }
    changeTrackStage();
}

public void changeTrackStage()
{
    dt3 = new DataTable();
    string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
    string query3 = "select
orderId,image,type,size,price,orderDate,ship_address,track_id  from orders ";
    using (MySqlConnection conn = new MySqlConnection(str))
    {

        conn.Open();
        using (MySqlCommand cmd2 = new MySqlCommand(query3, conn))
        {
            using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
            {

                sda2.Fill(dt3);

            }
        }
    }
}

```

```

        conn.Close();

    }
    int index = Convert.ToInt32(Session["index"]);
    if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 0)
    {
        placed.Attributes["class"] = "progtrckr-done";
        unloading.Attributes["class"] = "progtrckr-todo";
        hammering.Attributes["class"] = "progtrckr-todo";
        machineCutting.Attributes["class"] = "progtrckr-todo";
        polishing.Attributes["class"] = "progtrckr-todo";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 1)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-todo";
        machineCutting.Attributes["class"] = "progtrckr-todo";
        polishing.Attributes["class"] = "progtrckr-todo";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 2)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-todo";
        polishing.Attributes["class"] = "progtrckr-todo";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 3)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-done";
        polishing.Attributes["class"] = "progtrckr-todo";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 4)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-done";
        polishing.Attributes["class"] = "progtrckr-done";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 5)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-done";
        polishing.Attributes["class"] = "progtrckr-done";
        packing.Attributes["class"] = "progtrckr-done";
    }

```


Manager_Worker.aspx.cs

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

namespace WebApplication7
{
    public partial class Manager_Worker : System.Web.UI.Page
    {

        DataTable dt3 = new DataTable();
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["checkManager"] == null)
            {
                Response.Redirect("Login.aspx");
            }

            if (!Page.IsPostBack)
            {

                bind();
            }
        }
        public void bind()
        {
            // DataTable dt3 = new DataTable();

            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            string query3 = "select name,department,phone,city,month_of_joining,salary
from worker";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();
                using (MySqlCommand cmd2 = new MySqlCommand(query3, conn))
                {
                    using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
                    {

                        sda2.Fill(dt3);
                        GridView1.DataSource = dt3;
                        Session.Add("bindOrder", dt3);
                        //GridView1.Columns.RemoveAt(1);
                        GridView1.GridLines = GridLines.None;
                        GridView1.CellPadding = 50;
                        GridView1.DataBind();

                    }
                }
                conn.Close();
            }
        }
    }
}
```

```

protected void LinkButton1_Click(object sender, EventArgs e)
{
    Session.Clear();
    Response.Redirect("Login.aspx");
}

protected void btAddRecord_Click(object sender, EventArgs e)
{

    if (btAddRecord.Text == "Add Records")
    {
        divAddRecords.Visible = true;
        btAddRecord.Text = "Back";
    }
    else
    {
        divAddRecords.Visible = false;
        btAddRecord.Text = "Add Records";
    }
}

protected void btSaveChanges_Click(object sender, EventArgs e)
{
    string name = tbName.Text;
    int age = Convert.ToInt16(tbAge.Text);
    string dept = tbDept.Text;
    long number = Convert.ToInt64(tbPhoneNumber.Text);
    string address = tbAddress.Text;
    string city = tbCity.Text;
    string date = Calendar1.SelectedDate.ToShortDateString();
    long salary = Convert.ToInt64(tbSalary.Text);

    string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";

    string Query = "Insert into
worker(name,age,department,phone,address,city,month_of_joining,salary) values('" +
name + "','" + age + "','" + dept + "','" + number + "','" + address + "','" + city +
"','" + date + "','" + salary + "')";

    MySqlConnection MyConn2 = new MySqlConnection(str);
    MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
    MySqlDataReader MyReader2;
    MyConn2.Open();
    MyReader2 = MyCommand2.ExecuteReader();
    MyConn2.Close();
    btAddRecord.Text = "Add Records";
    divAddRecords.Visible = false;
    bind();
    Response.Redirect(Request.Url.AbsoluteUri);
}
protected void Calendar1_DayRender(object sender, DayRenderEventArgs e)
{
    if (e.Day.Date > DateTime.Now.Date)
    {
        e.Day.IsSelectable = false;
    }
}

}

```

Revenue.aspx.cs

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

namespace WebApplication7
{
    public partial class Revenue : System.Web.UI.Page
    {

        public string month;

        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["checkManager"] == null)
            {
                Response.Redirect("Login.aspx");
            }
            bind();
        }
        public void bind()
        {
            DataTable dt3 = new DataTable();

            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            string query3 = "select month,slab_length,slab_width,num_of_peice,sq_ft
from revenue";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();
                using ( MySqlCommand cmd2 = new MySqlCommand(query3, conn))
                {
                    using ( MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
                    {

                        sda2.Fill(dt3);
                        GridView1.DataSource = dt3;
                        Session.Add("prod", dt3);
                        GridView1.GridLines = GridLines.None;
                        GridView1.CellPadding = 50;
                        GridView1.DataBind();

                    }
                }
                conn.Close();
            }
        }
        protected void LinkButton1_Click(object sender, EventArgs e)
        {
            Session.Clear();
            Response.Redirect("Login.aspx");
        }
    }
}
```

```

        }

    protected void btCalculate_Click(object sender, EventArgs e)
    {
        lbprod.Visible = true;
        lbSquareFeet.Visible = true;
        long production,slh,slb,nop;
        slh = Convert.ToInt64(tbSlabLength.Text);
        slb = Convert.ToInt64(tbSlabWidth.Text);
        nop = Convert.ToInt64(tbNOP.Text);
        production = slh * slb * nop;
        lbSquareFeet.Text = "₹ " + production.ToString() + " /-";
        btSave.Visible = true;
    }

    protected void btShowRecords_Click(object sender, EventArgs e)
    {
        lbcheck.Visible = false;
        ddlMonth.Visible = false;
        lbemop.Visible = false;
        divcal.Visible = false;
        lbemop.Visible = false;
        ddlMonth.Visible = false;

        divshow.Visible = true;
    }

    protected void btAddRecords_Click(object sender, EventArgs e)
    {
        divshow.Visible = false;

        ddlMonth.Visible = true;
        lbemop.Visible = true;
    }

    protected void btSave_Click(object sender, EventArgs e)
    {

        long production, slh, slb, nop;
        slh = Convert.ToInt64(tbSlabLength.Text);
        slb = Convert.ToInt64(tbSlabWidth.Text);
        nop = Convert.ToInt64(tbNOP.Text);
        production = slh * slb * nop;
        string mon = Session["month"].ToString();

        string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";

        string Query = "Insert into
revenue(slab_length,slab_width,num_of_peice,sq_ft,month,year) values('" + slh + "','" +
+ slb + "','" + nop + "','" + production + "','" + mon + "','" + 2020 + "');";
        MySqlConnection MyConn2 = new MySqlConnection(str);
        MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
        MySqlDataReader MyReader2;
        MyConn2.Open();
        MyReader2 = MyCommand2.ExecuteReader();
        MyConn2.Close();
    }

```

```

        bind();
        divcal.Visible = false;
        lbemop.Visible = false;
        ddlMonth.Visible = false;
        lbprod.Visible = false;
        lbSquareFeet.Visible = false;
        btSave.Visible = false;

    }

    protected void ddlMonth_SelectedIndexChanged1(object sender, EventArgs e)
    {
        if (ddlMonth.SelectedValue == "select")
        {
            lbcheck.Visible = false;
            divcal.Visible = false;

        }
        else
        {

            Session["month"] = ddlMonth.SelectedValue;
            divcal.Visible = false;
            string mon = Session["month"].ToString();
            int i;
            DataTable dt3 = new DataTable();
            dt3 = (DataTable)Session["prod"];
            if (dt3.Rows.Count == 0)
            {
                divcal.Visible = true;

            }
            else
            {

                for (i = 0; i < dt3.Rows.Count; i++)
                {

                    if (dt3.Rows[i]["month"].ToString() == mon)
                    {
                        lbcheck.Visible = true;
                        divcal.Visible = false;
                        break;
                    }
                    else
                    {
                        lbcheck.Visible = false;

                        divcal.Visible = true;
                    }
                }
            }
        }
    }
}

```

Admin_Order.aspx.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;
using MySql.Data.MySqlClient;

namespace WebApplication7
{
    public partial class Admin_Order : System.Web.UI.Page
    {
        DataTable dt3 = new DataTable();

        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["checkAdmin"] == null)
            {
                Response.Redirect("Login.aspx");
            }
            if (!Page.IsPostBack)
            {
                track.Visible = false;
                bind();
            }
        }

        public void bind()
        {
            // DataTable dt3 = new DataTable();

            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            string query3 = "select
orderId,image,type,size,price,orderDate,ship_address,track_id  from orders ";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();
                using (MySqlCommand cmd2 = new MySqlCommand(query3, conn))
                {
                    using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
                    {

                        sda2.Fill(dt3);
                        GridView1.DataSource = dt3;
                        Session.Add("bindOrder", dt3);
                        GridView1.GridLines = GridLines.None;
                        GridView1.CellPadding = 50;
                        GridView1.DataBind();
                        GridView2.DataSource = dt3;
                        GridView2.GridLines = GridLines.None;
                        GridView2.CellPadding = 50;
                        GridView2.DataBind();
                        GridView2.Visible = false;
                    }
                }
            }
        }
    }
}
```

```

        }
    }
    conn.Close();

}
}
protected void Tracking(object sender, GridViewCommandEventArgs e)
{
    int index = Convert.ToInt32(e.CommandArgument);
    int count = GridView1.Rows.Count;

Session["index"] = index;
track.Visible = true;

dt3 = (DataTable)Session["bindOrder"];

int orderID;
orderID = Convert.ToInt16(dt3.Rows[index]["orderId"]);
Session["OrderID"] = orderID;
GridView2.Visible = true;
for (int i = 0; i < count; i++)
{
    if (i > index)
    {
        GridView1.Rows[i].Visible = false;
        GridView2.Rows[i].Visible = true;
    }
    else if (i <= index)
    {
        GridView2.Rows[i].Visible = false;
    }
}
btShowAll.Visible = true;
changeTrackStage();
}
public void changeTrackStage()
{
    dt3 = new DataTable();
    string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
    string query3 = "select
orderId,image,type,size,price,orderDate,ship_address,track_id  from orders ";
    using (MySqlConnection conn = new MySqlConnection(str))
    {

        conn.Open();
        using (MySqlCommand cmd2 = new MySqlCommand(query3, conn))
        {
            using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
            {

                sda2.Fill(dt3);

            }
        }
    }
}

```

```

        conn.Close();

    }
    int index = Convert.ToInt32(Session["index"]);
    if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 0)
    {
        placed.Attributes["class"] = "progtrckr-done";
        unloading.Attributes["class"] = "progtrckr-todo";
        hammering.Attributes["class"] = "progtrckr-todo";
        machineCutting.Attributes["class"] = "progtrckr-todo";
        polishing.Attributes["class"] = "progtrckr-todo";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 1)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-todo";
        machineCutting.Attributes["class"] = "progtrckr-todo";
        polishing.Attributes["class"] = "progtrckr-todo";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 2)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-todo";
        polishing.Attributes["class"] = "progtrckr-todo";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 3)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-done";
        polishing.Attributes["class"] = "progtrckr-todo";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 4)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-done";
        polishing.Attributes["class"] = "progtrckr-done";
        packing.Attributes["class"] = "progtrckr-todo";
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 5)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-done";
        polishing.Attributes["class"] = "progtrckr-done";
        packing.Attributes["class"] = "progtrckr-done";
    }

```

```
        loaded.Attributes["class"] = "progtrckr-todo";

    }
    else if (Convert.ToInt16(dt3.Rows[index]["track_id"]) == 6)
    {
        unloading.Attributes["class"] = "progtrckr-done";
        hammering.Attributes["class"] = "progtrckr-done";
        machineCutting.Attributes["class"] = "progtrckr-done";
        polishing.Attributes["class"] = "progtrckr-done";
        packing.Attributes["class"] = "progtrckr-done";
        loaded.Attributes["class"] = "progtrckr-done";
    }
}
protected void LinkButton1_Click(object sender, EventArgs e)
{
    Session.Clear();
    Response.Redirect("Login.aspx");
}
protected void btShowAll_Click(object sender, EventArgs e)
{
    track.Visible = false;
    int count = GridView1.Rows.Count;
    for (int i = 0; i < count; i++)
    {
        GridView1.Rows[i].Visible = true;
    }
    GridView2.Visible = false;
    btShowAll.Visible = false;
}
}
```

Admin_Worker.aspx.cs

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

namespace WebApplication7
{
    public partial class Admin_worker : System.Web.UI.Page
    {
        DataTable dt3 = new DataTable();
        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["checkAdmin"] == null)
            {
                Response.Redirect("Login.aspx");
            }
            if (!Page.IsPostBack)
            {

                bind();
            }
        }
        public void bind()
        {
            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            string query3 = "select name,department,phone,city,month_of_joining,salary
from worker";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();
                using ( MySqlCommand cmd2 = new MySqlCommand(query3, conn))
                {
                    using ( MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
                    {
                        sda2.Fill(dt3);
                        GridView1.DataSource = dt3;
                        Session.Add("bindOrder", dt3);
                        //GridView1.Columns.RemoveAt(1);
                        GridView1.GridLines = GridLines.None;
                        GridView1.CellPadding = 50;
                        GridView1.DataBind();
                    }
                }
                conn.Close();
            }
        }
        protected void LinkButton1_Click(object sender, EventArgs e)
        {
            Session.Clear();
            Response.Redirect("Login.aspx");
        }
    }
}
```

Admin_Revenue.aspx.cs

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

namespace WebApplication7
{
    public partial class Admin_revenue : System.Web.UI.Page
    {
        public string month;

        protected void Page_Load(object sender, EventArgs e)
        {
            if (Session["checkAdmin"] == null)
            {
                Response.Redirect("Login.aspx");
            }
            bind();
        }
        public void bind()
        {
            DataTable dt3 = new DataTable();

            string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";
            string query3 = "select month,slab_length,slab_width,num_of_peice,sq_ft
from revenue";
            using (MySqlConnection conn = new MySqlConnection(str))
            {

                conn.Open();
                using (MySqlCommand cmd2 = new MySqlCommand(query3, conn))
                {
                    using (MySqlDataAdapter sda2 = new MySqlDataAdapter(cmd2))
                    {

                        sda2.Fill(dt3);
                        GridView1.DataSource = dt3;
                        Session.Add("prod", dt3);
                        GridView1.GridLines = GridLines.None;
                        GridView1.CellPadding = 50;
                        GridView1.DataBind();

                    }
                }
                conn.Close();
            }
        }
        protected void LinkButton1_Click(object sender, EventArgs e)
        {
            Session.Clear();
            Response.Redirect("Login.aspx");
        }
    }
}
```

```

protected void btCalculate_Click(object sender, EventArgs e)
{
    lbprod.Visible = true;
    lbSquareFeet.Visible = true;
    long production, slh, slb, nop;
    slh = Convert.ToInt64(tbSlabLength.Text);
    slb = Convert.ToInt64(tbSlabWidth.Text);
    nop = Convert.ToInt64(tbNOP.Text);
    production = slh * slb * nop;
    lbSquareFeet.Text = "₹ " + production.ToString() + " /-";
    btSave.Visible = true;
}

protected void btShowRecords_Click(object sender, EventArgs e)
{
    lbcheck.Visible = false;
    ddlMonth.Visible = false;
    lbemop.Visible = false;
    divcal.Visible = false;
    lbemop.Visible = false;
    ddlMonth.Visible = false;

    divshow.Visible = true;
}

protected void btAddRecords_Click(object sender, EventArgs e)
{
    divshow.Visible = false;

    ddlMonth.Visible = true;
    lbemop.Visible = true;

}

protected void btSave_Click(object sender, EventArgs e)
{
    long production, slh, slb, nop;
    slh = Convert.ToInt64(tbSlabLength.Text);
    slb = Convert.ToInt64(tbSlabWidth.Text);
    nop = Convert.ToInt64(tbNOP.Text);
    production = slh * slb * nop;
    string mon = Session["month"].ToString();

    string str = @"Data Source=localhost;
Database=gransells;CharSet=utf8;AllowUserVariables=True; User ID=root;
Password=PASSWORD";

    string Query = "Insert into
revenue(slab_length,slab_width,num_of_peice,sq_ft,month,year) values('" + slh + "','" +
+ slb + "','" + nop + "','" + production + "','" + mon + "','" + 2020 + "');";
    MySqlConnection MyConn2 = new MySqlConnection(str);
    MySqlCommand MyCommand2 = new MySqlCommand(Query, MyConn2);
    MySqlDataReader MyReader2;
    MyConn2.Open();
    MyReader2 = MyCommand2.ExecuteReader();
    MyConn2.Close();
    bind();
}

```

```

        divcal.Visible = false;
        lbemop.Visible = false;
        ddlMonth.Visible = false;
        lbprod.Visible = false;
        lbSquareFeet.Visible = false;
        btSave.Visible = false;

    }

protected void ddlMonth_SelectedIndexChanged1(object sender, EventArgs e)
{
    if (ddlMonth.SelectedValue == "select")
    {
        lbcheck.Visible = false;
        divcal.Visible = false;

    }
    else
    {

        Session["month"] = ddlMonth.SelectedValue;
        divcal.Visible = false;
        string mon = Session["month"].ToString();
        int i;
        DataTable dt3 = new DataTable();
        dt3 = (DataTable)Session["prod"];
        if (dt3.Rows.Count == 0)
        {
            divcal.Visible = true;

        }
        else
        {

            for (i = 0; i < dt3.Rows.Count; i++)
            {

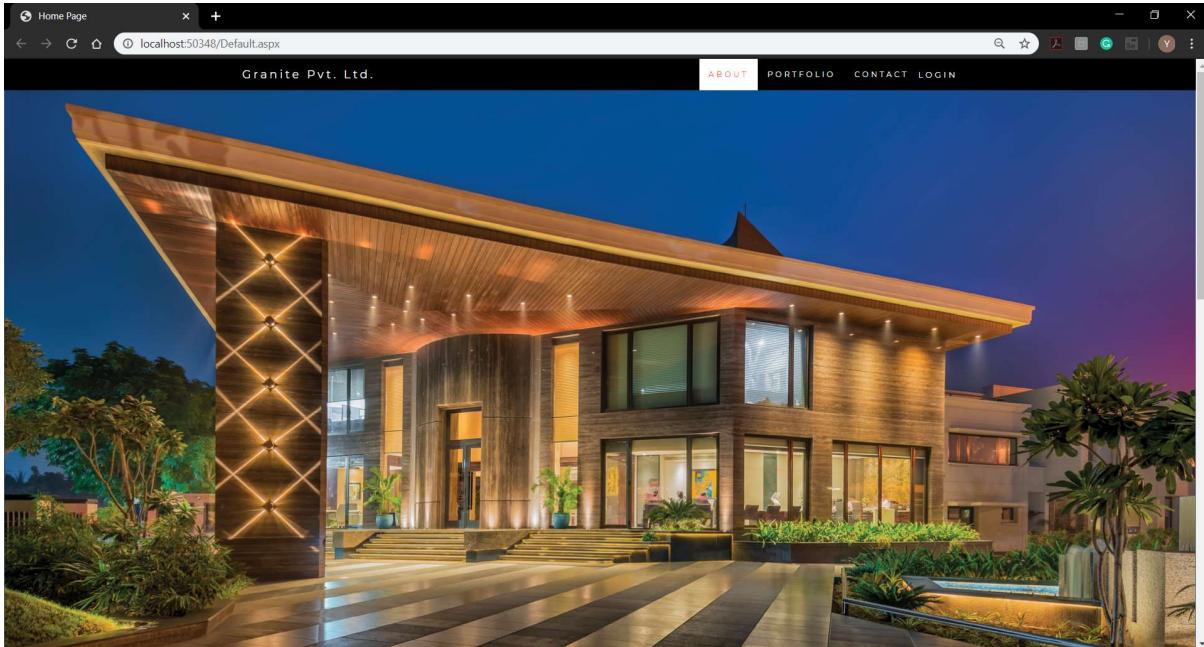
                if (dt3.Rows[i]["month"].ToString() == mon)
                {
                    lbcheck.Visible = true;
                    divcal.Visible = false;
                    break;
                }
                else
                {
                    lbcheck.Visible = false;

                    divcal.Visible = true;
                }
            }
        }
    }
}
}

```

OUTPUT

Default.aspx

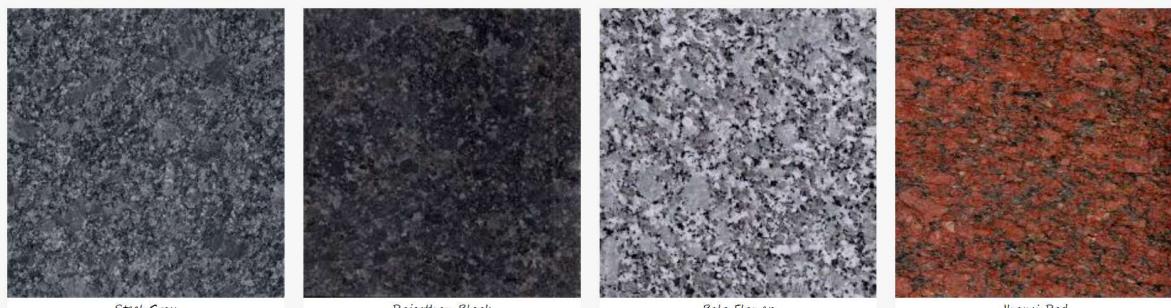


ABOUT COMPANY PAGE

Granite Pvt. Ltd. is a Granite slab manufacturing unit started at Shahjehapura. It was established in 2009. Since then it is continuously running with its full capacity. Initially it started with one block Cutter, three Polishing machines, a Jib Type Crane and Edge cutting. It increased its capacity in 2013 by introducing second Block Cutter and then installed a automatic polishing machine to increase its polishing production as well as quality of granites.



PRODUCTS



Steel Gray

Rajasthan Black

Bala Flower

Jhansi Red

WHAT OUR CUSTOMERS SAY

"THEY PROVIDE THE BEST POLISHED GRANITES!"

CONTACT

Shahjehapura, India

+91 9876789877

granitespvt@gmail.com

RegForm.aspx

The screenshot shows a registration form titled "Sign Up". It contains the following fields:

- First name: (empty)
- Last name: (empty)
- E-mail: (empty)
- Password: (empty)
At least 1 Lowercase & 1 Uppercase letter, 1 Special Character, 8 characters and 1 digit
- Confirm Password: (empty)
- Phone number: (empty)
- Address: (empty)
- City: (empty)
- Country: (empty)

A green "Sign Up" button is located at the bottom right of the form.

The screenshot shows the same registration form as above, but with validation errors displayed in red text below each field:

- First name: Yash6
Name can not have characters other than Alphabets
- Last name: babel
Invalid Email Structure
- E-mail: babelyash.yb@gmail.com.0vom
Invalid Email Structure
- Password:
At least 1 Lowercase & 1 Uppercase letter, 1 Special Character, 8 characters and 1 digit
Password not as per specification
- Confirm Password:
Password do not match
- Phone number: 6756435
Phone Number can only have 10 digits
- Address: B-72 Subhash Nagar, Shastri Nagar
- City: Jaipur?
Invalid City name
- Country: Maharashtra

A green "Sign Up" button is located at the bottom right of the form.

Login.aspx

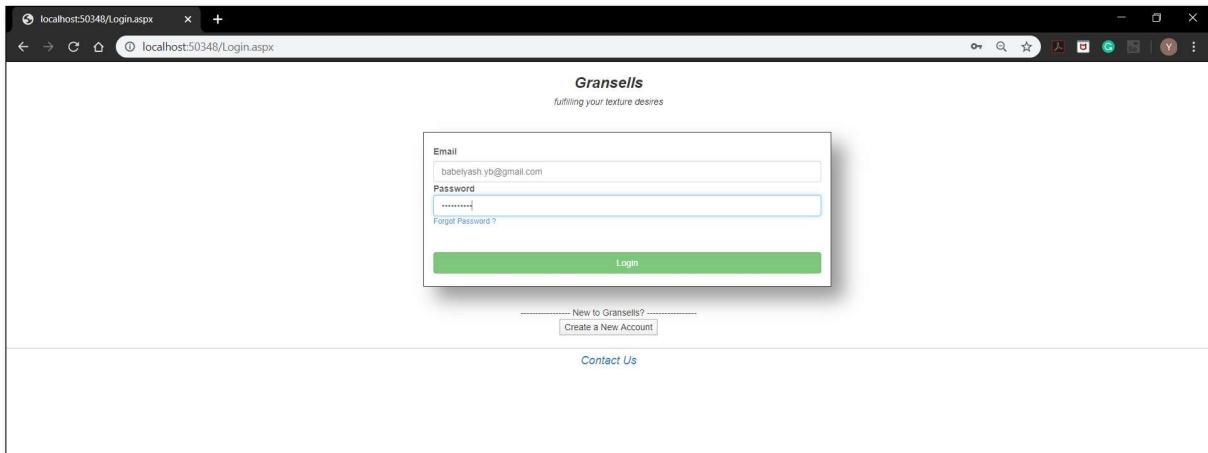
The screenshot shows a login form for "Gransells" with the tagline "fulfilling your texture desires". The form has a single field:

Email: babelyash.yb@gmail.com

Below the form is a green "Continue" button.

At the bottom of the page, there is a link for new users: "New to Gransells? Create a New Account".

A "Contact Us" link is also present at the bottom.



Product.aspx

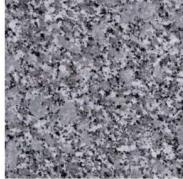
A screenshot of the Product.aspx page for "Bala Flower". The top navigation bar includes "Granite Pvt. Ltd.", "HOME", "MY PROFILE", and "LOGOUT". On the left is a large image of Bala Flower granite. To the right, the product name "BALA FLOWER" is displayed in bold. A detailed description follows: "Bala Flower is a kind of white granite quarried in China. This stone is especially good for Exterior - Interior wall and floor applications, countertops, mosaic, fountains, pool and wall capping and other design projects. Bala Flower can be processed into Polished, Sawn Cut, Sanded, Rockfaced, Sandblasted, Tumbled and so on." Below the description are dropdown menus for "Select for Slab or Tile" (set to "Slab"), "Price per Square Feet" (set to "125"), and "Select height & width" (with dropdowns for height and width). A "BUY NOW" button is at the bottom.

A screenshot of the Product.aspx page for "Bala Flower" showing a different set of dimensions. The "Select height & width" dropdowns show "6.5" and "2.75" respectively. The "Price" field shows "₹ 2234.375". The rest of the page content is identical to the first screenshot, including the product image, name, description, and "BUY NOW" button.

BuyNow.aspx

Granite Pvt. Ltd.

HOME MY PROFILE LOGOUT



Bala Flower

SLAB

Size: 6.5 X 2.75

Price: 2234.375

Deliver to: YASH BABEL

Shipping Address : B-72 Subhash Nagar, Shastri Nagar, Jaipur, India

CONFIRM ORDER

Email confirmation

Greetings from Gransells. Your order is placed ➤

yashbabel99@gmail.com to me ▾

Hi,
Thanks for placing an order with Gransells.
Your order details are:
Type: SLAB
Size: 6 X 2.25
Please issue a check on the name 'Granites Pvt. Ltd.' of ₹ 2227.5 to confirm your order. You will shortly receive a call from us.

Fri, Feb 14, 12:20 PM (2 days ago) ⭐ 🌐 ⋮

Reply Forward

UserProfile.aspx

Granite Pvt. Ltd.

HOME MY PROFILE LOGOUT

Personal info

First name:	Yash Babel
Mobile No.:	8107777220
Address:	B-72 Subhash Nagar, Shastri Nagar
City:	Jaipur
Country:	India

My Orders

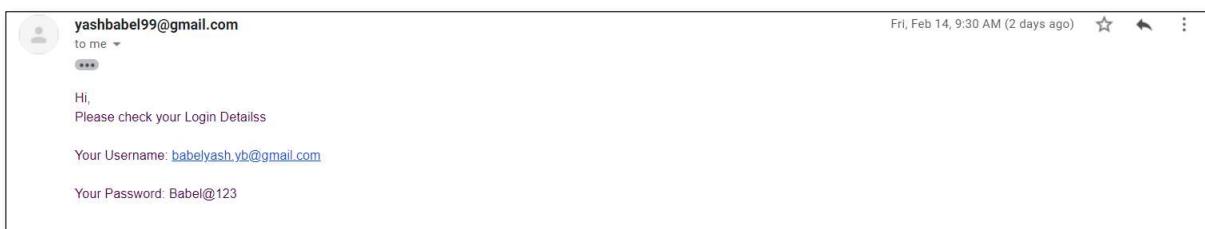
TYPE	SIZE	PRICE	DATE	SHIPPING ADDRESS
SLAB	6 X 2.25	2228	14-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India
SLAB	6.5 X 2.75	2234	16-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India
SLAB	6.5 X 2.75	2234	16-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India

* For order or cancellation related queries kindly contact firm *

ForgetPassword.aspx

A screenshot of a web browser window titled "localhost:50348/ForgetPassword". The page contains a form with a single input field labeled "Email" containing the value "babelyash.yb@gmail.com". Below the input field is a green rectangular button with the text "Send Email".

Email of Password



ManagerOrder.aspx

Granite Pvt. Ltd.						ORDERS	WORKERS	PRODUCTION	LOGOUT
	TYPE	SIZE	PRICE	DATE	SHIPPING ADDRESS				
	SLAB	6 X 2.25	2228	14-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India				
	SLAB	6 X 2.25	1148	15-02-2020	Mithibai College, Vile Parle, Mumbai, India				
	TILE	67	3886	15-02-2020	Mithibai College, Vile Parle, Mumbai, India				
	SLAB	6.5 X 2.75	2234	16-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India				
	SLAB	6.5 X 2.75	2234	16-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India				

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

TYPE	SIZE	PRICE	DATE	SHIPPING ADDRESS
SLAB	6 X 2.25	1148	15-02-2020	Mithibai College, Vile Parle, Mumbai, India.

Show All | Update Track Stage | Select

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

TYPE	SIZE	PRICE	DATE	SHIPPING ADDRESS
SLAB	6 X 2.25	1148	15-02-2020	Mithibai College, Vile Parle, Mumbai, India.

Show All | Update Track Stage | Polished

Manager_Worker.aspx

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Name	Department	Number	City	Joining Date	Salary
yash	loading packing	7878787878 8107777220	Jaipur Mumbai	12-02-2020 13-02-2020	13000

Add Records

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Name	Department	Number	City	Joining Date	Salary
yash	loading packing	7878787878 8107777220	Jaipur Mumbai	12-02-2020 13-02-2020	13000

Back

Name:

Age:

Department:

Mobile No.:

Address:

City:

Joining Date:

Salary:

Save Changes

Revenue.aspx

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Show Production Add Production

Month	Slab Length	Slab Width	Number of Piece	Production
January	378	678	1200	307540000
February	378	678	12078	3095398152

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Show Production Add Production

Enter Month of Production: January ▾

Month already exist

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Show Production Add Production

Enter Month of Production: March ▾

Slab Length:

Slab Width:

Number of Piece:

Calculate

Admin_Order.aspx

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

TYPE	SIZE	PRICE	DATE	SHIPPING ADDRESS
SLAB	6 X 2.25	2228	14-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India
SLAB	6 X 2.25	1148	15-02-2020	Mithibai College, Vile Parle, Mumbai, India
TILE	67	3886	15-02-2020	Mithibai College, Vile Parle, Mumbai, India

Order Placed Unloaded Hammering Cutting Polished Packed Loaded

Done

SLAB	6.5 X 2.75	2234	16-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India
SLAB	6.5 X 2.75	2234	16-02-2020	B-72 Subhash Nagar, Shastri Nagar, Jaipur, India

Admin_Worker.aspx

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Name vikram gehlot yash	Department leading packing	Number 7876787878 8107777220	City Jaipur Mumbai	Joining Date 12-02-2020 13-02-2020	Salary 13000 13000
-------------------------------	----------------------------------	------------------------------------	--------------------------	--	--------------------------

Revenue.aspx

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Show Production Add Production

Month	Slab Length	Slab Width	Number of Piece	Production
January	378	678	1200	307540800
February	378	678	12078	3095398152

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Show Production Add Production

Enter Month of Production: January ▾
Month already exist

Granite Pvt. Ltd.

ORDERS WORKERS PRODUCTION LOGOUT

Show Production Add Production

Enter Month of Production: March ▾

Slab Length:

Slab Width:

Number of Piece:

Calculate

LIMITATIONS

- **Payment:-**

- After placing an order, since there is no payment facility, the customer has to issue a check, which is totally customers choice. If he fails to issue the check the order details will be there as long as the administrator deletes it.

- **Order Cancellation:-**

- If after placing order the customer want to cancel the order, he has to contact firm which is again a limitation of the system.
- The firm also doesn't have authority to cancel the order. any changes have to done by database administrator.

- **Password**

- When the user selects the “Forget Password” option, the module sends the user a mail with his/her password which is considered security issue.
- Also, there is no option for the user to change his/her password.
- There is same login section for admin and manager, which can again let someone do operations from there login if he has their credentials.

FUTURE ENHANCEMENT

The project will constitute all the features that are listed above and will further be updated as and when needed to comply with growing technology advancements. Some of the features that will be added in updates are :-

- **SHOPPING CART:-**

- Shopping cart feature for customers will be added. Since there may be several products customers want to order in one go. This feature is kept for further updates

- **PAYMENT :-**

- The orders now are confirmed only when after placing the order, customer issues a check and the same is confirmed by the staff at the firm. But later a proper payment facility will be there which will automatically confirm the payment

- **TRYING COMBINATIONS OF GRANITES:-**

- As while buying there may be a conflict in buyer's mind whether this type will go with some colors or not, so to ease this task the user while buying can check creating a virtual environment and trying different combinations

- **DIFFERENT MANAGER LOGIN:-**

- For this system there will be just one manager updating the track stage of orders. But in future updates, stage of the order can be updated by the manager handling that task in firm. This will relieve the burden off from a single person.

- **MANAGE ATTENDENCE OF WORKERS :-**

- This will be a feature which will be added for manager and director to maintain the attendance for workers, which will be month-wise and further salary can be calculated on that basis.
 - Further this attendance recording can be done from biometric system which will record the attendance day wise.

- **TRACKING FOR CUSTOMERS**

- Customers will be able to view the status of their order like shipped, where it has reached and other details

REFERENCES & BIBLIOGRAPHY

- www.projectmanager.com
- www.simplimearn.com
- <https://tallyfy.com/uml-diagram/>
- www.stackoverflow.com
- www.w3schools.com
- www.rkmarbles.com
- Beginning ASP.NET 4.5 in C# by Matthew McDonald