

**Thilakarathna PC Center**

DISE 35 Batch

Name : M.P.R.L Pathirana

Course : DISE

Batch : 35

Reg No : GAM/A-009592

we like to thank our dear Sir Mr. **Nuwan Hirantha** for his help. They always guide us on the proper way and always help us to get better knowledge about Information Technology Planning.

We thank all of them who give us as their directions helped us to success this System.

# Table of Contents

[Table of Contents 3](#_Toc37586636)

[Table of figures 5](#_Toc37586637)

[Chapter 01 6](#_Toc37586638)

[Introduction 6](#_Toc37586639)

[The software Thilakarathna PC center has four main modules 6](#_Toc37586640)

[PROJECT AIMS AND OBJECTIVES 7](#_Toc37586641)

[BACKGROUND OF PROJECT 7](#_Toc37586642)

[OPERATION ENVIRONMENT 8](#_Toc37586643)

[CHAPTER 2 9](#_Toc37586644)

[Requirement Gathering 9](#_Toc37586645)

[Functional requirements of Thilakarathna PC center System 9](#_Toc37586646)

[Nonfunctional requirements of Thilakarathna PC center System 10](#_Toc37586647)

[Chapter 03 10](#_Toc37586648)

[System Analyzing 10](#_Toc37586649)

[Drawbacks of the manual system 11](#_Toc37586650)

[Proposed system 11](#_Toc37586651)

[Project materials 11](#_Toc37586652)

[Hardware tools 12](#_Toc37586653)

[Chapter 04 13](#_Toc37586654)

[System Design 13](#_Toc37586655)

[Software Development Model 13](#_Toc37586656)

[Use case Diagram 13](#_Toc37586657)

[Chapter 5 14](#_Toc37586658)

[Coding 14](#_Toc37586659)

[Login 14](#_Toc37586660)

[Home Interface 16](#_Toc37586661)

[Stock Interface 20](#_Toc37586662)

[Chapter 6 22](#_Toc37586663)

[Software testing 22](#_Toc37586664)

[Chapter 07 23](#_Toc37586665)

[Conclusion 23](#_Toc37586666)

# 

# Table of figures

[Figure 1 Visual studio 2015 11](#_Toc37586788)

[Figure 2 Xampp 12](#_Toc37586789)

[Figure 3 Microsoft word 12](#_Toc37586790)

[Figure 4 use case diagram 13](#_Toc37586791)

[Figure 5login 14](#_Toc37586792)

[Figure 6 Login Interface 14](#_Toc37586793)

[Figure 7 Home interface 16](#_Toc37586794)

[Figure 8 Inventory interface 20](#_Toc37586795)

[Figure 11 Gantt chart 23](#_Toc37586796)

# Chapter 01

## Introduction

The project titled Thilakarathna PC center is Management Software for monitoring and controlling the Inventory, Sell, Customers, Employee, and stock in a Computer shop. The project “Thilakarathna PC center " is developed in C#, which mainly focuses on basic operations in a shop like customer details, Employee details, adding new customer, adding new sell items, items repairing and inventory controlling

“Thilakarathna PC center " is a windows application written for 32-bit and 64-bit windows operating systems, designed to help users maintain and organize shop. Our software is easy to use for both beginners and advanced users. It features a familiar and well thought-out, an attractive user interface, combined with strong searching insertion and reporting capabilities.

### The software Thilakarathna PC center has four main modules

Customer details

Employee details

Stock

Inventory and Sale

### PROJECT AIMS AND OBJECTIVES

The project aims and objectives that will be achieved after completion of this project are discussed in this subchapter. The aims and objectives are as follows:

* **Manage Payments**
* **Register customers and employees**
* **Manage Membership**
* **Generate monthly reports**
* **Stock changing**
* **An inventory**
* **Sales management**

## BACKGROUND OF PROJECT

Thilakarathna PC center Management System is an application which refers to computer shop system which is generally medium in size. It is used by shop owners to manage the shop details using a computerized system where he/she can record various sales and details like adding members, creating pay rolls, sales reports, inventory details, stock system etc.

Member maintenance modules are also included in this system which would keep track of the members using the computer system and also a detailed description about the store contains. With this computerized system there will be no loss of goods record or member record which generally happens when a non-computerized system is used.

In addition, report module is also included in Thilakarathna PC center Management System. The owners are able to generate different kinds of reports like lists of members registered, list of inventories, issue and return reports.

All these modules are able to help shop owners to manage the Thilakarathna PC center system with more convenience and in a more efficient way as compared to Thilakarathna PC center systems which are not computerized.

## OPERATION ENVIRONMENT

|  |  |
| --- | --- |
| PROCESSOR | Intel Core Processor Or Better Performance |
| OPERATING SYSTEM | Windows 7, Windows 8, Windows 10 |
| MEMORY | 2 GB Ram or More |
| HARD DISK SPACE | Minimum 3 Gb For Database Usage For Future |
| DATABASE | Mysql |
|  |  |

# CHAPTER 2

## Requirement Gathering

Before start to build a new system the development team has to categorize requirements what are going to help and not. System analyst is the position who had responsible for that. Some of requirements are visible so it can find easily and some are well hidden. To gather most of these hidden requirements analyst needs some skills and techniques. So these techniques called by fact finding techniques.

There are various types of techniques business analyst can use to find facts. There are 5 main types,

1. Record searching
2. Questionnaires
3. Interviews
4. Sampling
5. Observation

### Functional requirements of Thilakarathna PC center System

* Administrator should able to create and manage customers and employees
* Administrator should able to search goods, customers etc.
* Administrator should able to view history of payment, sales and users.
* Administrator should able to manage payment.
* Administrators should able to generate monthly summery reports

### Nonfunctional requirements of Thilakarathna PC center System

1. Security Requirement

* Restrict and prevent unauthorized attempts to login.
* Check login validation and verification with username and passwords.

2. Performance Requirement

* The Database should able to store minimum 50,000 records.
* The system should exceed 99% uptime
* The system should have user-friendly interface.

4. Reliability Requirement

* The system provide alert to the administrator when errors occurred.

# Chapter 03

## System Analyzing

In this chapter, we will discuss and analyze about the developing process of Thilakarathna PC center System including software requirement specification (SRS) and comparison between existing and proposed system. The functional and non-functional requirements are included in SRS part to provide complete description and overview of system requirement before the developing process is carried out. Besides that, existing VS proposed provides a view of how the proposed system will be more efficient than the existing one.

## 

### Drawbacks of the manual system

* Lot of human resources is required
* Large amount of clerical time is required
* The record maintainability is difficult
* Accessibility of accurate information from the past record is difficult

### Proposed system

This software can satisfy the customer by adding more security measures and make the interface more user friendly. It is important that this software is able to cover all things. This is software that can be identified as an all-in-one package.

## Project materials

Microsoft visual Studio 2015

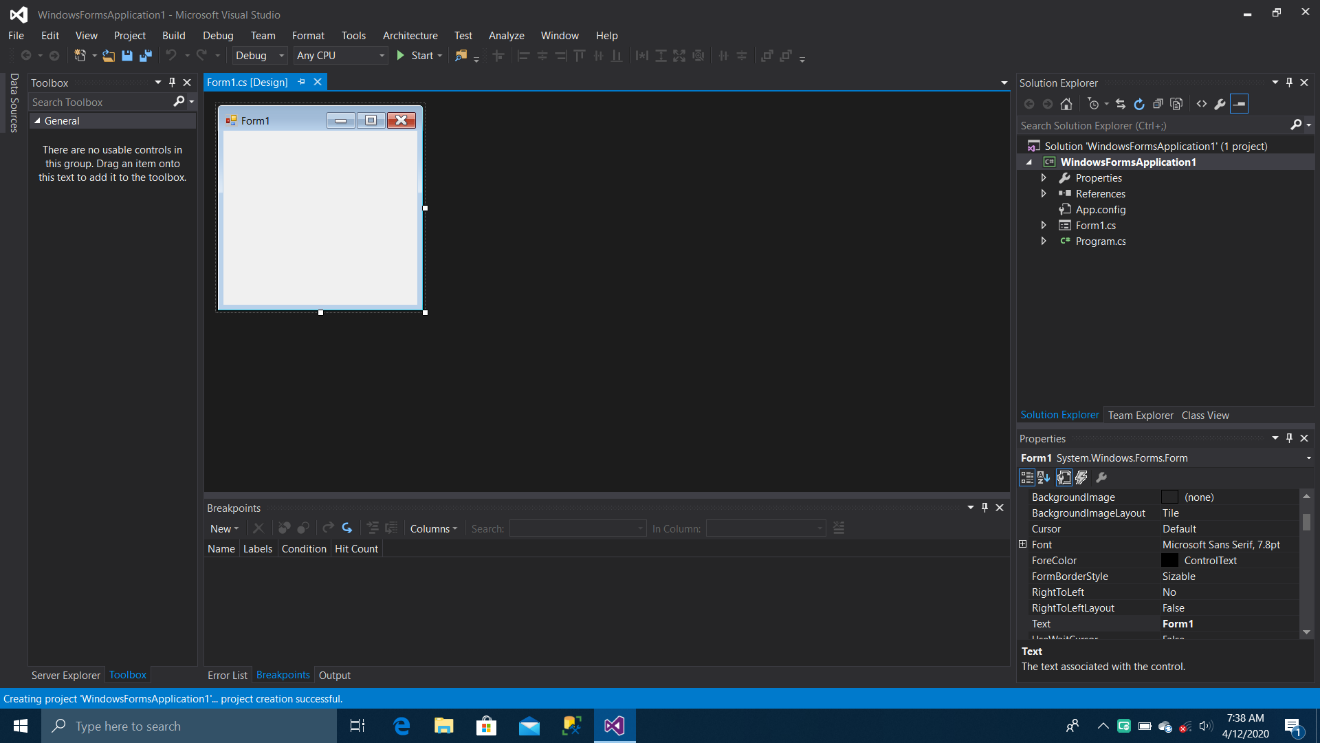


Figure 1 Visual studio 2015

Xampp

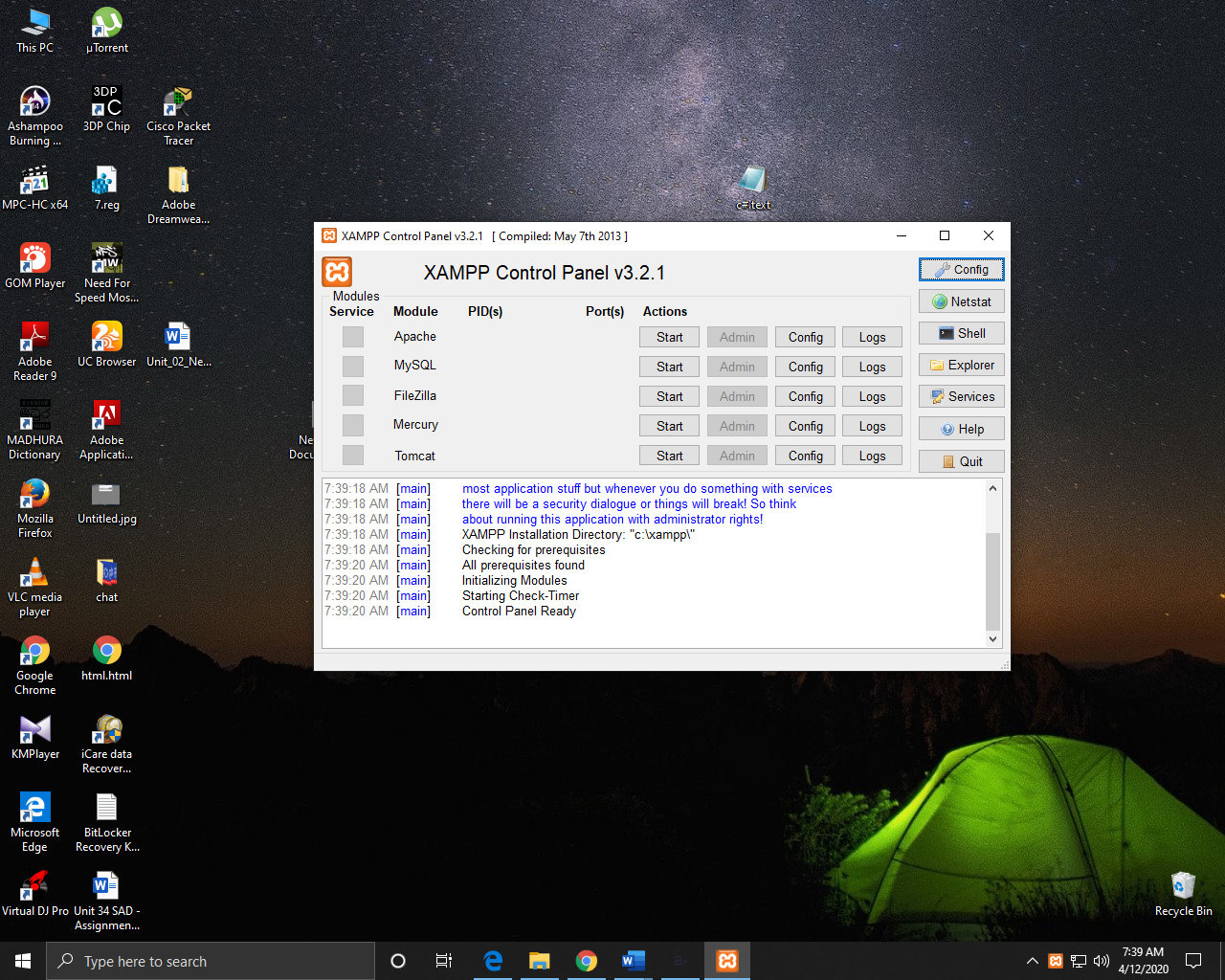


Figure 2 Xampp

Microsoft word for preparing documentation.

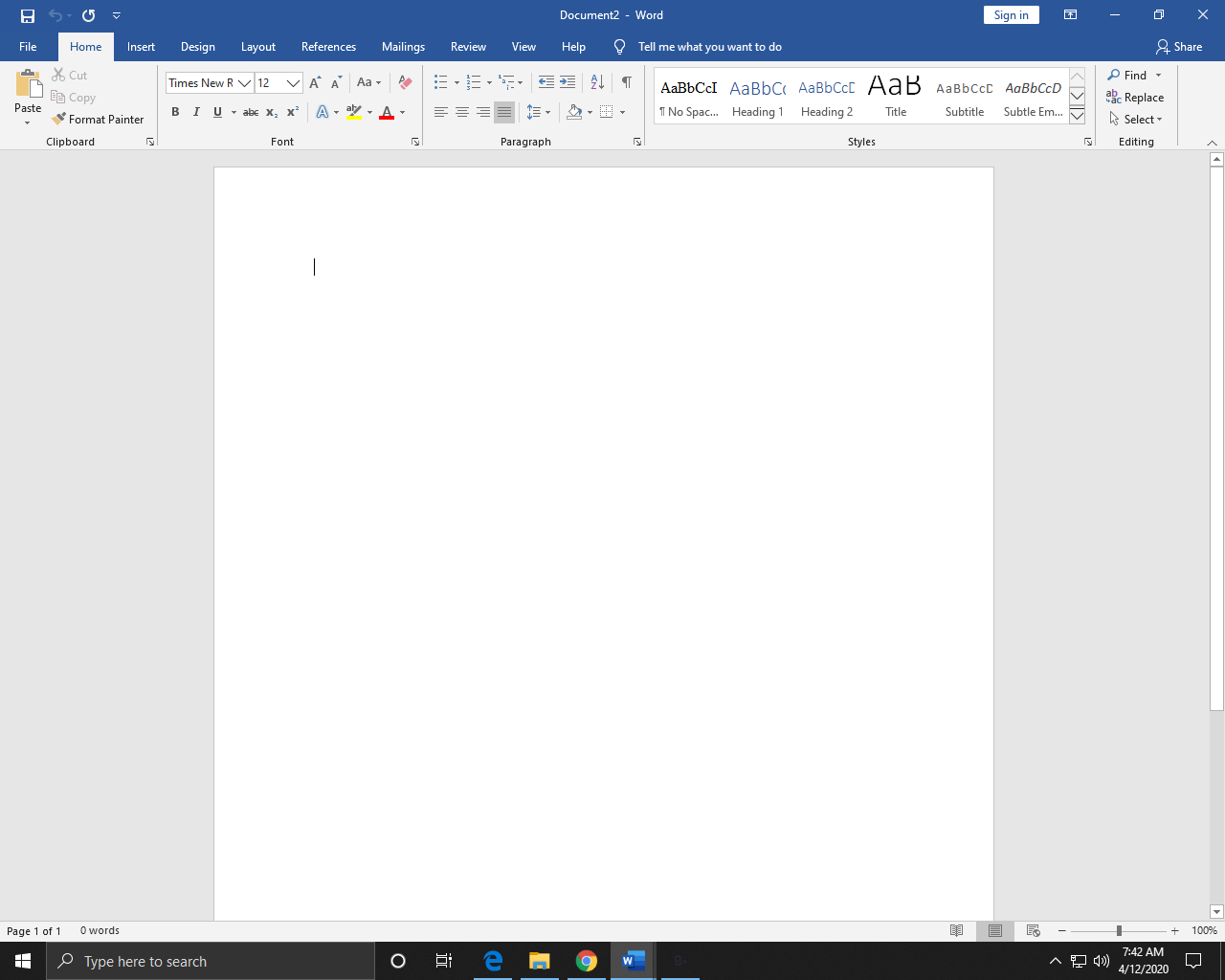


Figure 3 Microsoft word

## Hardware tools

* Computers
* Printer
* Pen drives
* CDs

# Chapter 04

## System Design

System design is the design of the system before creating a system. If a creates a system for the customer, they must submit these plans and get approval for it. The system developer has the ability to easily improve the system by designing the system. Most system development agencies have separate personnel for system design. System design is a very important task.

## Software Development Model

Software development models the various processes or methodologies that are being selected for the development of a software project depending on the project‟s aims and goals. The selected model specifies the various stages of the process and the order in which they are carried out.

## Use case Diagram

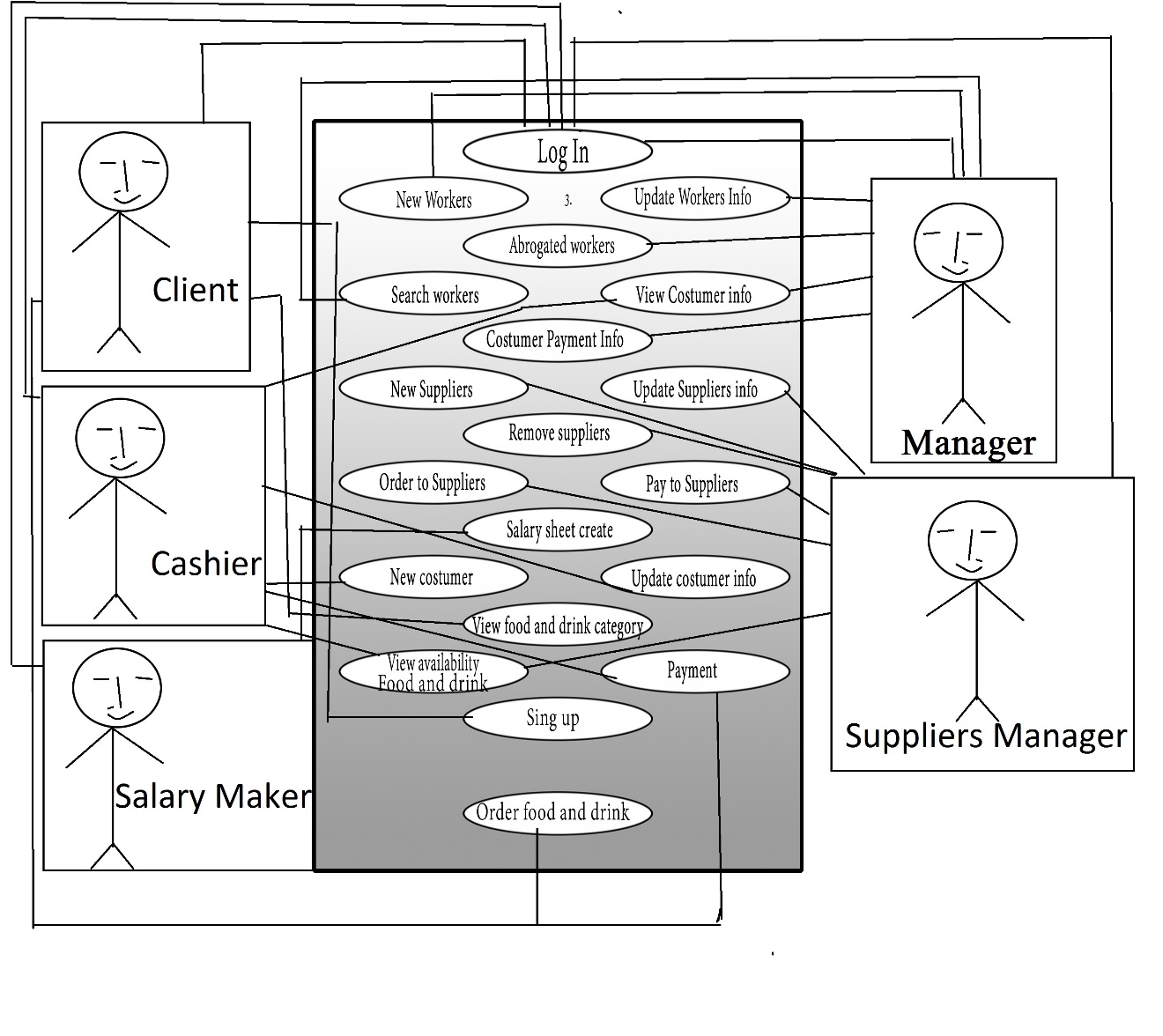


Figure 4 use case diagram

# Chapter 5

## Coding

Codes are the programming language used to develop the program.

C# coding for this program

### Login

Interface

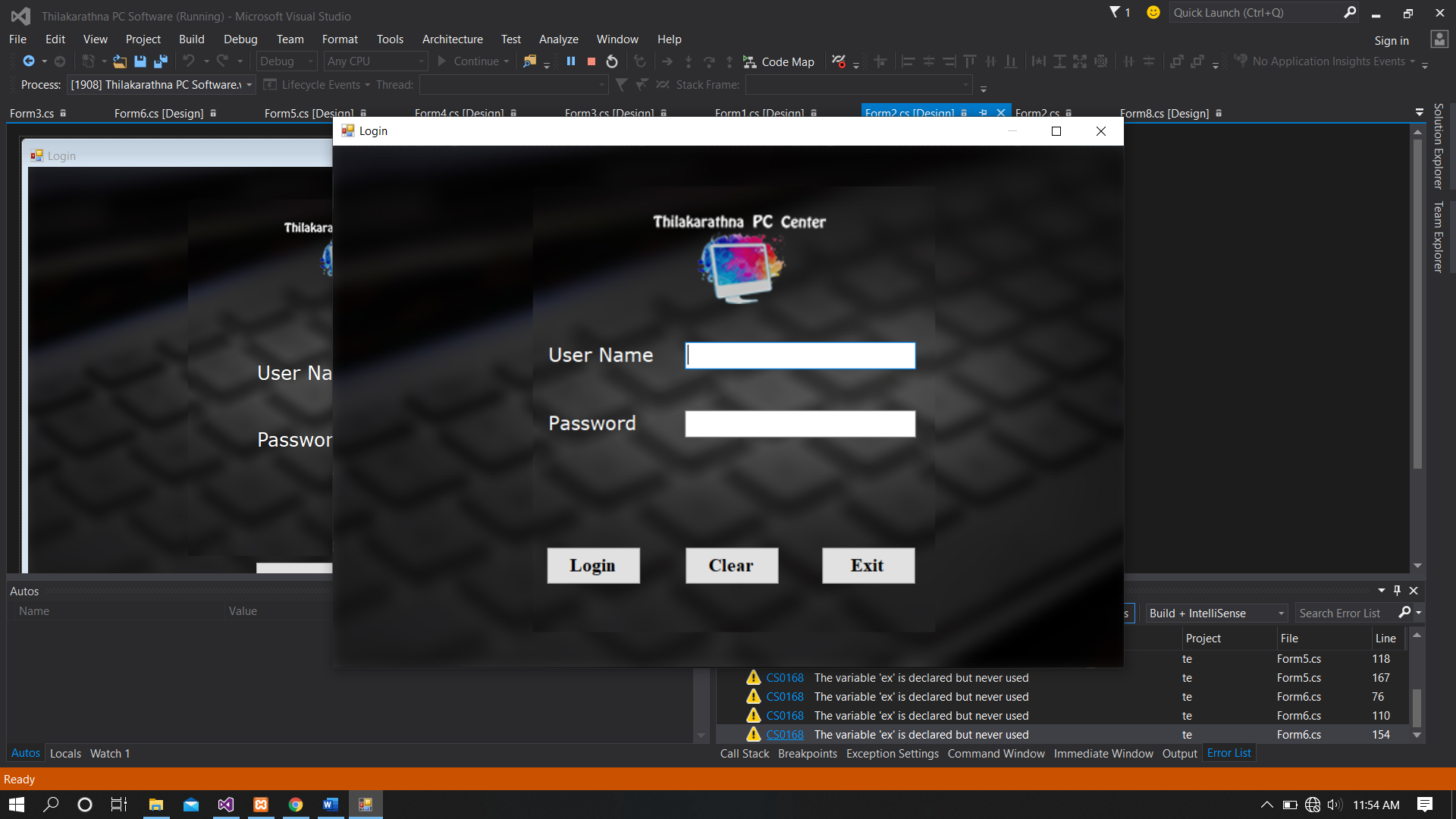


Figure 5login

Figure 6 Login Interface

Login C# code

|  |
| --- |
| string connectionstring = "server=localhost; userid=root; password=;database=thilakarathna\_pc\_center;";  MySqlConnection myconnection = new MySqlConnection(connectionstring);  string username = user\_name.Text;  string password = passw.Text;  string querystring = "SELECT \* FROM `user` WHERE `username` ='" + username + "' AND `password`='" + password + "';";  MySqlCommand mycommand = new MySqlCommand(querystring, myconnection);  myconnection.Open();  MySqlDataAdapter myadp = new MySqlDataAdapter(mycommand);  DataTable mytb = new DataTable();  myadp.Fill(mytb);  if (mytb.Rows.Count > 0)  {  string level = mytb.Rows[0][3].ToString();  if (level == "1")  {  main\_page c = new main\_page();  c.Show();  this.Hide();  }  else if (level == "2")  {  main\_page c = new main\_page();  c.Show();  this.Hide();  }  else if (level == "4")  {  main\_page c = new main\_page();  c.Show();  this.Hide();  }  else if (level == "3")  {  main\_page c = new main\_page();  c.Show();  this.Hide();  }  else  {  }  }  else  {  MessageBox.Show("Invalid Username or Password");  }  myconnection.Close(); |

### Home Interface

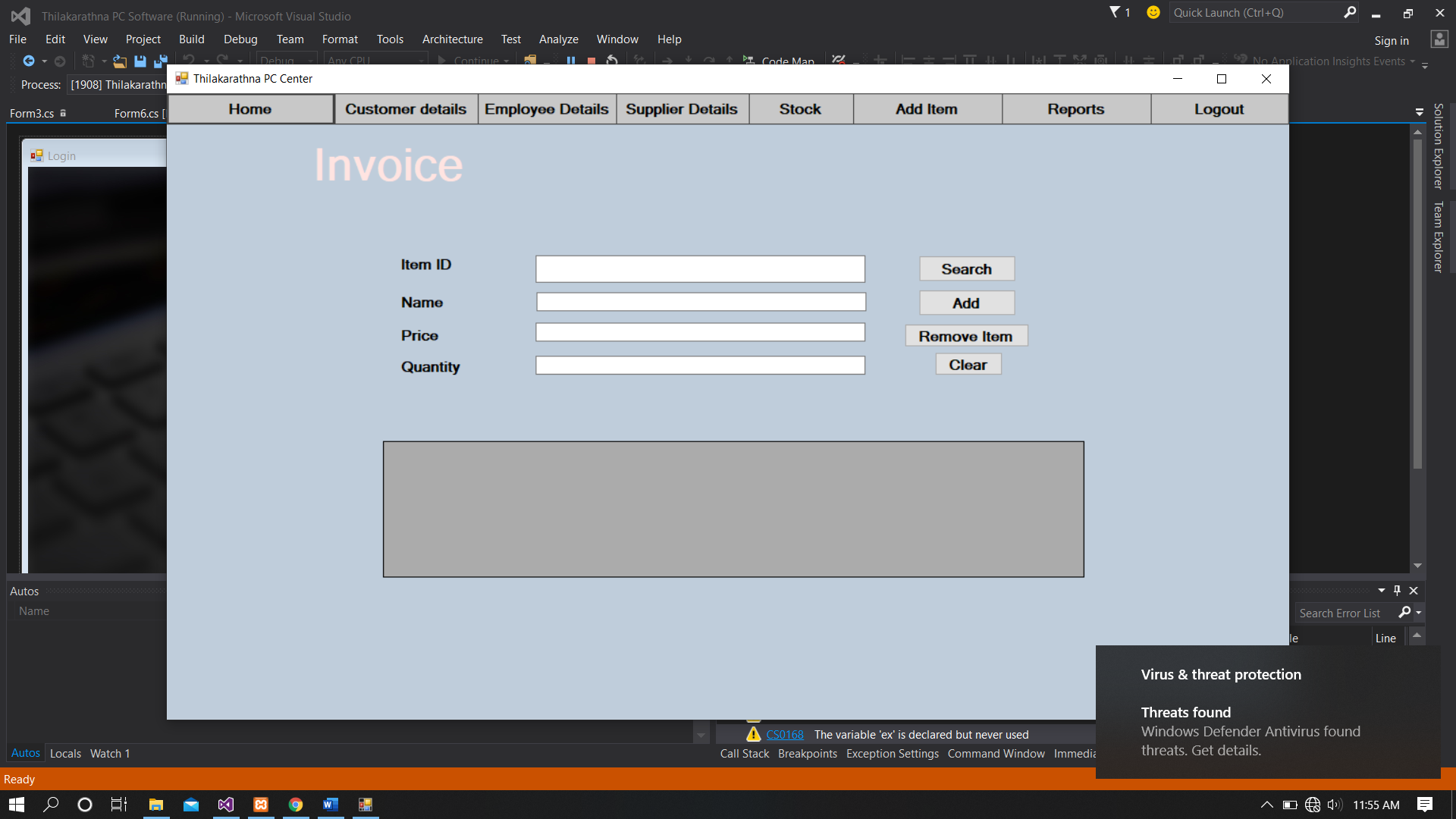
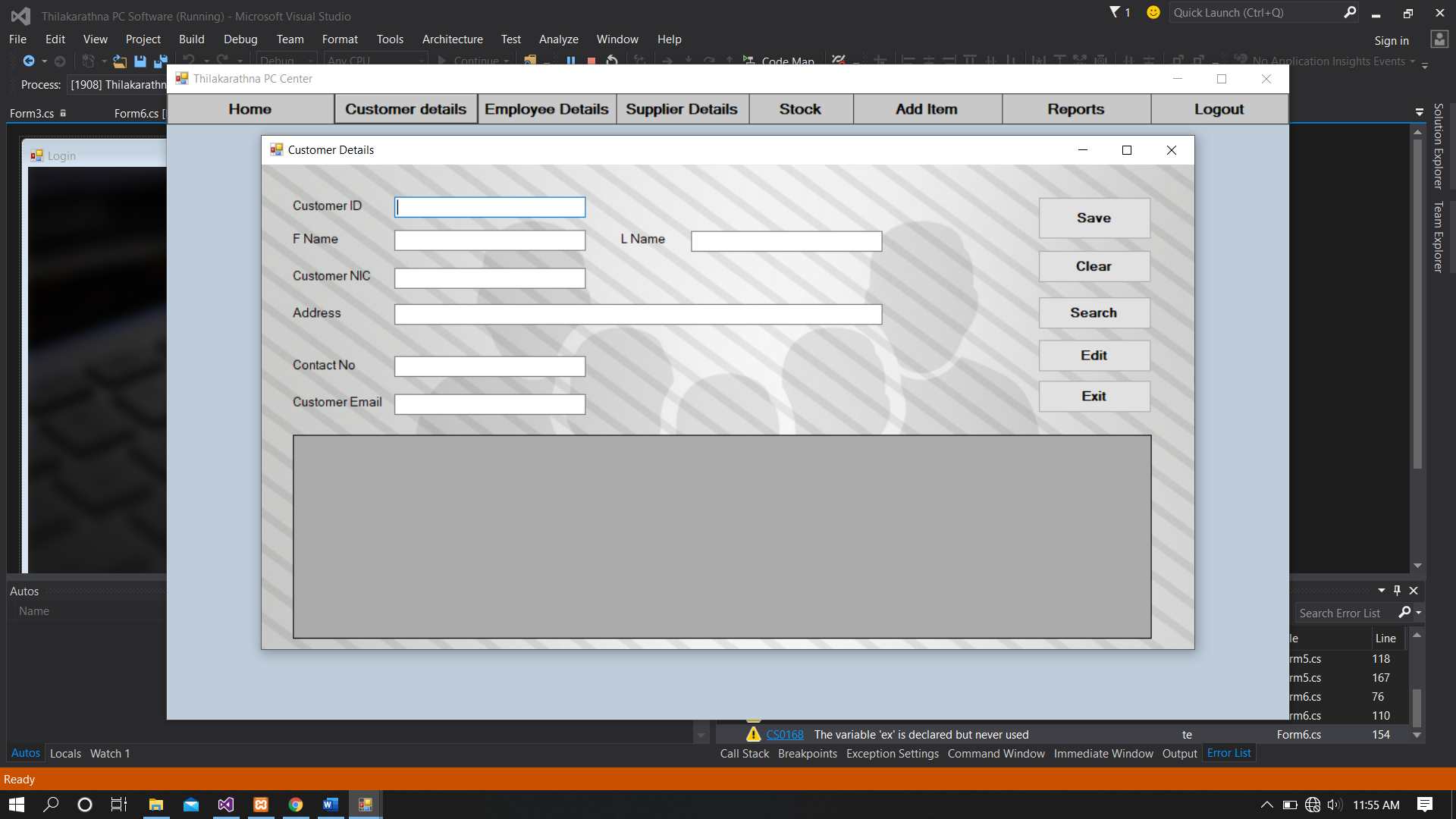


Figure 7 Home interface

Customer detail



Customer Save

|  |
| --- |
| try  {  string con = "server=localhost; username=root; password=;database=thilakarathna\_pc\_center;";  MySqlConnection myconnection = new MySqlConnection(con);  string id = text\_cus\_id.Text;  string fname = text\_cus\_fname.Text;  string lname = text\_cus\_lname.Text;  string nic = text\_cus\_nic.Text;  string address = text\_cus\_address.Text;  string tel = text\_cus\_tel.Text;  string email = text\_cus\_email.Text;  string querystring = "INSERT INTO `customer\_info`(`id`, `fname`, `lname`, `nic`, `address`, `tel`, `email`) VALUES ('"+id+"','"+fname+"','"+lname+"','"+nic+"','"+address+"','"+tel+"','"+email+"')";  MySqlCommand mycommand = new MySqlCommand(querystring, myconnection);  myconnection.Open();  mycommand.ExecuteNonQuery();  myconnection.Close();  // to add valus to data grid  MySqlConnection myconnection3 = new MySqlConnection(con);  string customerid = text\_cus\_id.Text;    string querystring3 = "SELECT \* FROM `customer\_info` WHERE `id`='" + customerid + "'";  MySqlCommand mycommand2 = new MySqlCommand(querystring3, myconnection3);  myconnection3.Open();  MySqlDataAdapter myadp1 = new MySqlDataAdapter(mycommand2);  DataTable mytb1 = new DataTable();  myadp1.Fill(mytb1);  BindingSource bSource1 = new BindingSource();  bSource1.DataSource = mytb1;  bSource1.DataSource = mytb1;  dataGridView1.DataSource = bSource1;  myconnection3.Close();  MessageBox.Show("data inserted successfully");  }  catch (Exception ex)  {  MessageBox.Show("error occured in inserting");  } |

Customer search

|  |
| --- |
| try  {  string connectionstring = "server=localhost; userid=root; password=;database=thilakarathna\_pc\_center;";  MySqlConnection myconnection = new MySqlConnection(connectionstring);  int id = Convert.ToInt32(text\_cus\_id.Text);  string querystring = "SELECT \* FROM `customer\_info` WHERE `id`=" + id + "";  MySqlCommand mycommand = new MySqlCommand(querystring, myconnection);  myconnection.Open();  MySqlDataAdapter myadp = new MySqlDataAdapter(mycommand);  DataTable mytb = new DataTable();  myadp.Fill(mytb);  text\_cus\_fname.Text = mytb.Rows[0][1].ToString();  text\_cus\_lname.Text = mytb.Rows[0][2].ToString();  text\_cus\_nic.Text = mytb.Rows[0][3].ToString();  text\_cus\_address.Text = mytb.Rows[0][4].ToString();  text\_cus\_tel.Text = mytb.Rows[0][5].ToString();  text\_cus\_email.Text = mytb.Rows[0][6].ToString();  myconnection.Close();  }  catch (Exception ex)  {  MessageBox.Show("An error");  } |

Customer Edit

|  |
| --- |
| try  {  string connectionstring = "server=localhost; userid=root; password=;database=thilakarathna\_pc\_center;";  MySqlConnection myconnection = new MySqlConnection(connectionstring);  int id = Convert.ToInt32(text\_cus\_id.Text);  string fname = text\_cus\_fname.Text;  string lname = text\_cus\_lname.Text;  string nic = text\_cus\_nic.Text;  string address = text\_cus\_address.Text;  string tel = text\_cus\_tel.Text;  string email = text\_cus\_email.Text;  string querystring = "UPDATE `customer\_info` SET `fname`='" + fname + "',`lname`='" + lname + "',`nic`='" + nic + "',`address`='" + address + "',`tel`='" + tel + "',`email`='" + email + "' WHERE `id`=" + id + "";  MySqlCommand mycommand = new MySqlCommand(querystring, myconnection);  myconnection.Open();  mycommand.ExecuteNonQuery();  myconnection.Close();  MessageBox.Show("data udtated successfully");  text\_cus\_id.Text = "";  text\_cus\_fname.Text = "";  text\_cus\_lname.Text = "";  text\_cus\_nic.Text = "";  text\_cus\_address.Text = "";  text\_cus\_tel.Text = "";  text\_cus\_email.Text = "";  }  catch (Exception ex)  {  MessageBox.Show("An error");  } |

Customer search

|  |
| --- |
| try  {  string connectionstring = "server=localhost; userid=root; password=;database=thilakarathna\_pc\_center;";  MySqlConnection myconnection = new MySqlConnection(connectionstring);  int id = Convert.ToInt32(text\_cus\_id.Text);  string querystring = "SELECT \* FROM `customer\_info` WHERE `id`=" + id + "";  MySqlCommand mycommand = new MySqlCommand(querystring, myconnection);  myconnection.Open();  MySqlDataAdapter myadp = new MySqlDataAdapter(mycommand);  DataTable mytb = new DataTable();  myadp.Fill(mytb);  text\_cus\_fname.Text = mytb.Rows[0][1].ToString();  text\_cus\_lname.Text = mytb.Rows[0][2].ToString();  text\_cus\_nic.Text = mytb.Rows[0][3].ToString();  text\_cus\_address.Text = mytb.Rows[0][4].ToString();  text\_cus\_tel.Text = mytb.Rows[0][5].ToString();  text\_cus\_email.Text = mytb.Rows[0][6].ToString();  myconnection.Close();  }  catch (Exception ex)  {  MessageBox.Show("An error");  } |

### Stock Interface

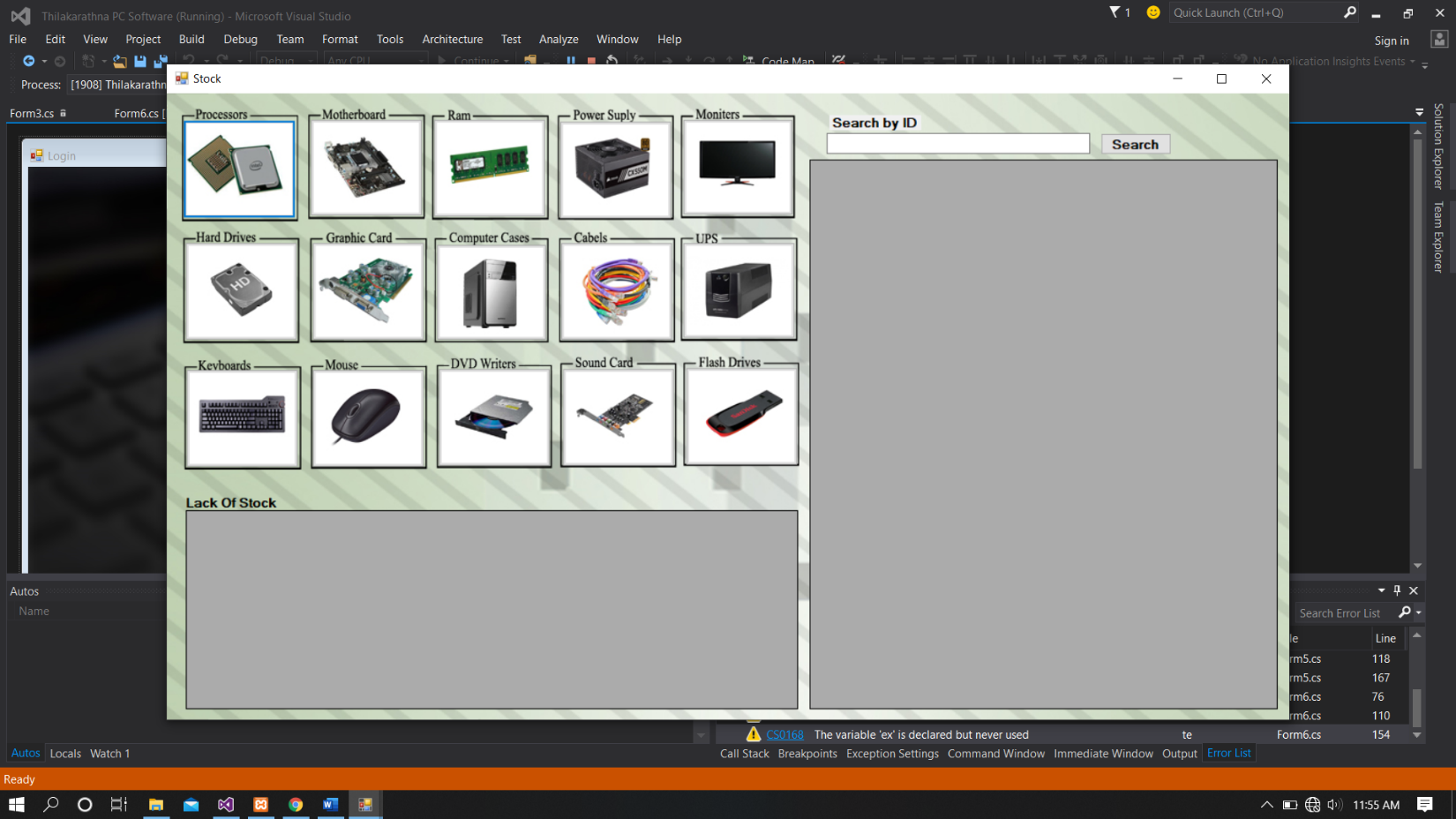
****

Figure 8 Inventory interface

New inventory item adds C# code

|  |
| --- |
| string con = "server=localhost; username=root; password=;database=thilakarathna\_pc\_center;";  MySqlConnection myconnection = new MySqlConnection(con);  string productid = item\_id.Text;  string productname = item\_name.Text;  string description = item\_description.Text;  string quantity = item\_quantity.Text;  string price = item\_price.Text;  string querystring = " INSERT INTO `additem`(`productid`, `productname`, `description`, `quantity`, `price`) VALUES ('" + productid + "','" + productname + "','" + description + "','" + quantity + "','" + price + "')";  MySqlCommand mycommand = new MySqlCommand(querystring, myconnection);  myconnection.Open();  mycommand.ExecuteNonQuery();  myconnection.Close();  // to add valus to data grid    MySqlConnection myconnection3 = new MySqlConnection(con);  string id = item\_id.Text;  string querystring3 = "SELECT \* FROM `additem` WHERE `productid`='" + id + "'";  MySqlCommand mycommand2 = new MySqlCommand(querystring3, myconnection3);  myconnection3.Open();  MySqlDataAdapter myadp1 = new MySqlDataAdapter(mycommand2);  DataTable mytb1 = new DataTable();  myadp1.Fill(mytb1);  BindingSource bSource1 = new BindingSource();  bSource1.DataSource = mytb1;  bSource1.DataSource = mytb1;  dataGridView1.DataSource = bSource1;  myconnection3.Close();  MessageBox.Show("data inserted successfully"); |

# Chapter 6

## Software testing

According to this testing model it will help to identify the errors, bugs in the system. In some situation, it would help to figure out the data processing paths and the method, basically this testing method has categorized in unit testing, integration testing, system testing and acceptance testing.

Login Form testing

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test case | Description | Expected result | Actual result | Pass/fail | Remarks |
| 01 | Run software | Loading login form | Loading login form | pass |  |
| 02 | Incorrect username or password | Get error message | Get error message | Pass |  |
| 03 | Correct password | Access next form | Access next form | pass |  |

shop management form

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test case | Description | Expected result | Actual result | Pass/fail | Remarks |
| 04 | New supplier | Successful insert supplier details | Successful insert supplier details | Pass |  |
| 05 | Add/update/ delete inventory item | Successful adding/ deleting/ updating | Successful adding/ deleting/ updating | Pass |  |
| 06 | Sales report | Accessing correct sales report | Accessing correct sales report | Pass |  |

# Chapter 07

## Conclusion

This project has been successfully completed and I am satisfied with it. This project can cover all the processes of an organization. It was able to cover all intents and purposes as planned.

Gantt Chart

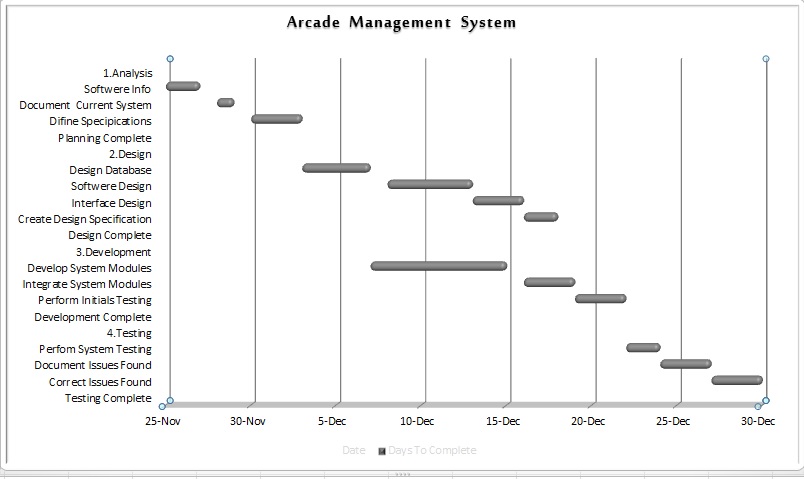


Figure 11 Gantt chart