ST. ANNE'S
COLLEGE OF ENGINEERING AND TECHNOLOGY

ANGUCHETTYPALAYAM, PANRUTI – 607 110

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

LAB MANUAL

CCS356-OBJECT ORIENTED SOFTWARE ENGINEERING

Regulation 2021

Year / Semester: III/VI

APR 20245– MAY 2025

CONTENTS

Ex.No : 1
Date :

PASSPORT AUTOMATION SYSTEM

AIM

To develop the Passport Automation System using Agro UML, visual basic and MS access.

PROBLEM ANALYSIS AND PROJECT PLAN

To simplify the process of applying passport, software has been created by designing through rational rose tool, using visual basic as a front end and Microsoft access as a back end. Initially the applicant login the passport automation system and submits his details. These details are stored in the database and verification process done by the passport administrator, regional administrator and police the passport is issued to the applicant.

PROBLEM STATEMENT

1. Passport Automation System is used in the effective dispatch of passport to all of the applicants. This system adopts a comprehensive approach to minimize the manual work and schedule resources, time in a cogent manner.

2. The core of the system is to get the online registration form (with details such as name, address etc.,) filled by the applicant whose testament is verified for its genuineness by the Passport Automation System with respect to the already existing information in the database.

3. This forms the first and foremost step in the processing of passport application. After the first round of verification done by the system, the information is in turn forwarded to the regional administrator's (Ministry of External Affairs) office.

4. The application is then processed manually based on the report given by the system, and any forfeiting identified can make the applicant liable to penalty as per the law.

5. The system forwards the necessary details to the police for its separate verification whose report is then presented to the administrator. After all the necessary criteria have been met, the original information is added to the database and the passport is sent to the applicant.

SOFTWARE REQUIREMENTS SPECIFICATION

INTRODUCTION

Passport Automation System is an interface between the Applicant and the Authority responsible for the Issue of Passport. It aims at improving the efficiency in the Issue of Passportand reduces the complexities involved in it to the maximum possible extent.

PURPOSE

If the entire process of 'Issue of Passport' is done in a manual manner then it would take several months for the passport to reach the applicant. Considering the fact that the number of applicants for passport is increasing every year, an Automated System becomes essential to meet the demand. So this system uses several programming and database techniques to elucidate the work involved in this process. As this is a matter of National Security, the system has been carefully verified and validated in order to satisfy it.

SCOPE

The System provides an online interface to the user where they can fill in their personal details. The authority concerned with the issue of passport can use this system to reduce his workload and process the application in a speedy manner. Provide a communication platform between the applicant and the administrator Transfer of data between the Passport Issuing Authority and the Local Police for verification of applicant's information.

DEFINITIONS, ACRONYMS AND THE ABBREVIATIONS

1. Administrator - Refers to the super user who is the Central Authority who has been vested with the privilege to manage the entire system. It can be any higher official in the Regional Passport Office of Ministry of External Affairs.
2. Applicant - One who wishes to obtain the Passport.
3. PAS - Refers to this Passport Automation System.

REFERENCES IEEE Software Requirement Specification format.

TECHNOLOGIES TO BE USED • Microsoft Visual Basic 6.0

TOOLS TO BE USED • Agro UML (for developing UML Patterns)

OVERVIEW

SRS includes two sections overall description and specific requirements - Overall description will describe major role of the system components and inter-connections. Specific requirements will describe roles & functions of the actors.

OVERALL DESCRIPTION

PRODUCT PERSPECTIVE

The PAS acts as an interface between the 'applicant' and the 'administrator'. This system tries to make the interface as simple as possible and at the same time not risking the security of data stored in. This minimizes the time duration in which the user receives the passport.

SOFTWARE INTERFACE

1. **Front End Client** - The applicant and Administrator online interface is built using Microsoft Visual Basic 6.0.
2. **Back End** – MS Access database

HARDWARE INTERFACE

The server is directly connected to the client systems. The client systems have access to the database in the server.

SYSTEM FUNCTIONS

1. Secure Registration of information by the Applicants.
2. Message box for Passport Application Status Display by the Administrator.
3. Administrator can generate reports from the information and is the only authorized personnel to add the eligible application information to the database.

USER CHARACTERISTICS

1. Applicant - They are the people who desires to obtain the passport and submit the information to the database.
2. Administrator - He has the certain privileges to add the passport status and to approve the issue of passport. He may contain a group of persons under him to verify the documents and give suggestion whether or not to approve the dispatch of passport.
3. Police - He is the person who upon receiving intimation from the PAS, perform a personal verification of the applicant and see if he has any criminal case against him before or at present. He has been vetoed with the power to decline an application by suggesting it to the Administrator if he finds any discrepancy with the applicant. He communicates via this PAS.

CONSTRAINTS

1. The applicants require a computer to submit their information.
2. Although the security is given high importance, there is always a chance of intrusion in the web world which requires constant monitoring.
3. The user has to be careful while submitting the information. Much care is required.

ASSUMPTIONS AND DEPENDENCIES

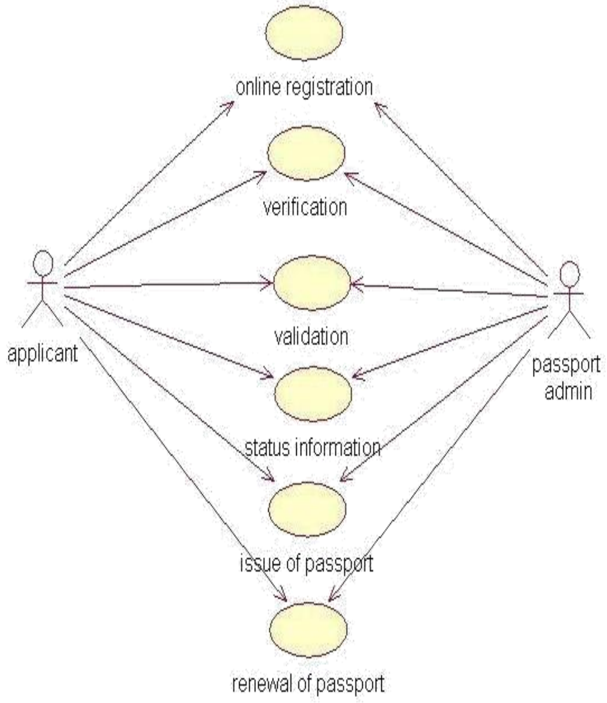
1. The Applicants and Administrator must have basic knowledge of computers and English Language.
2. The applicants may be required to scan the documents and send.

UML DIAGRAMS

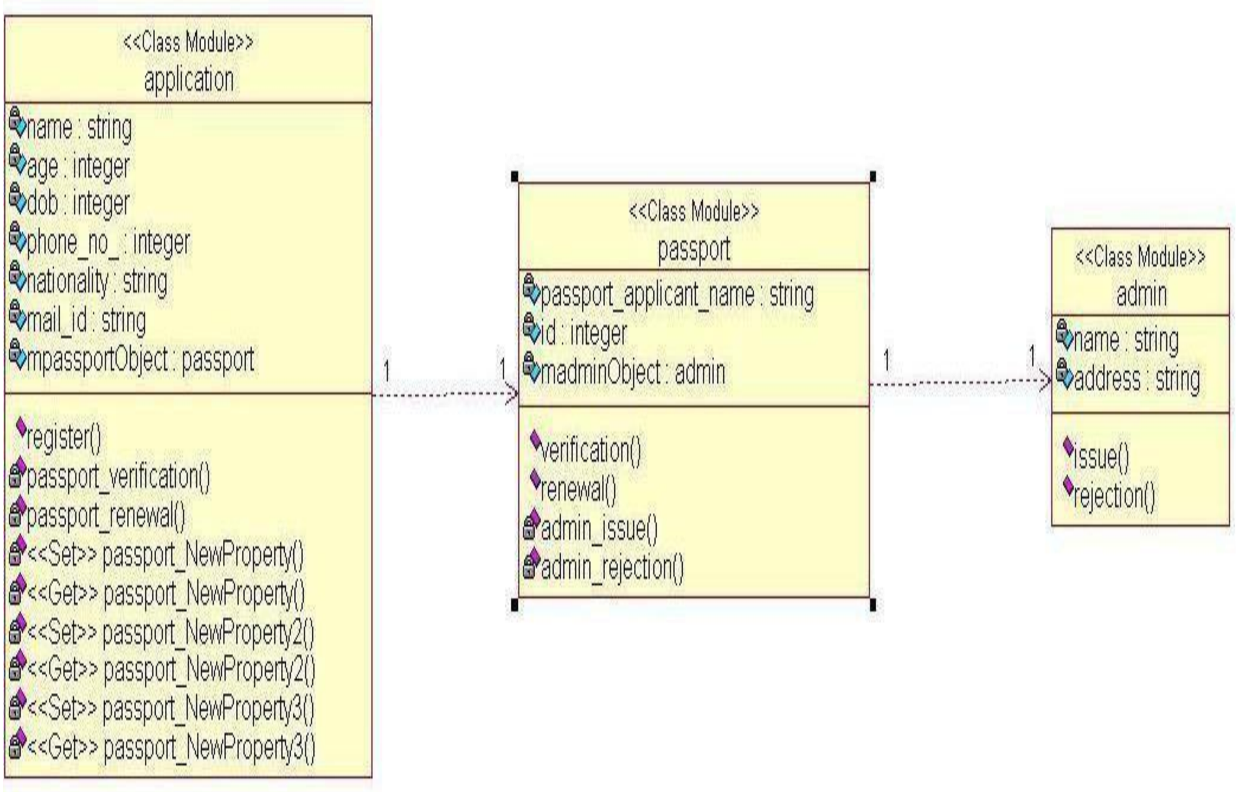
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

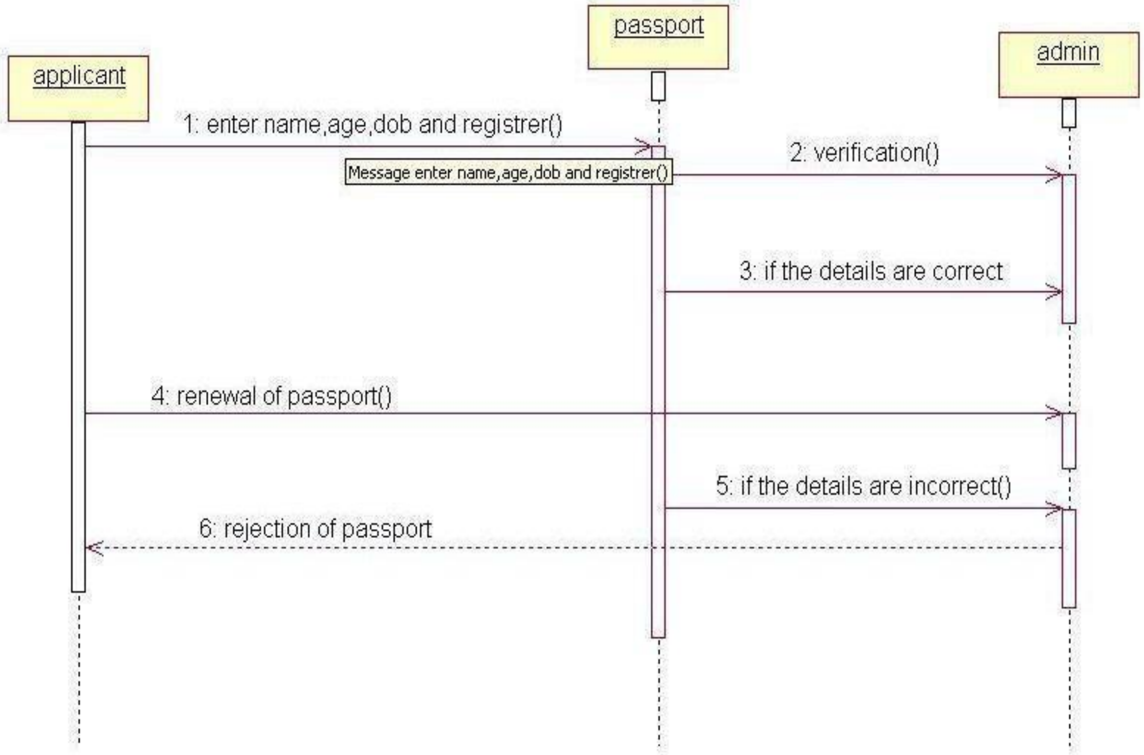
USE CASE DIAGRAM



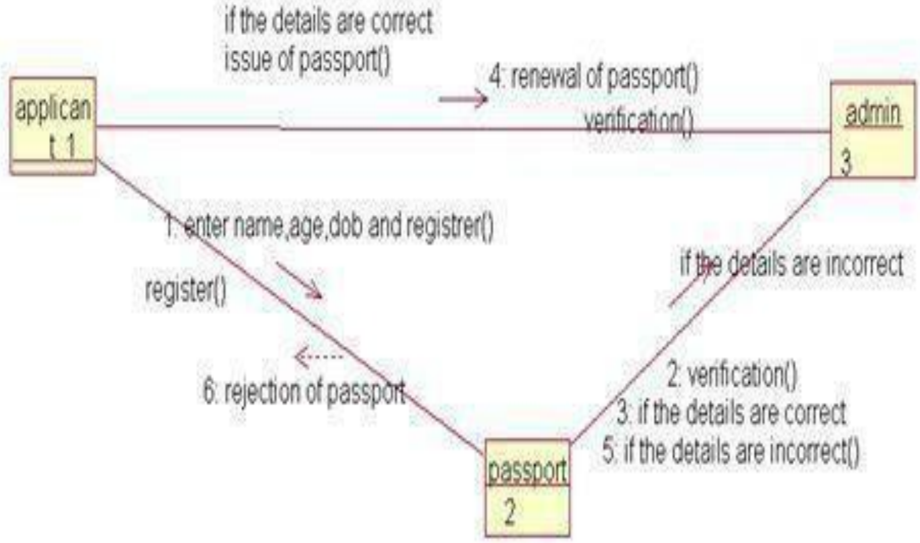
CLASS DIAGRAM



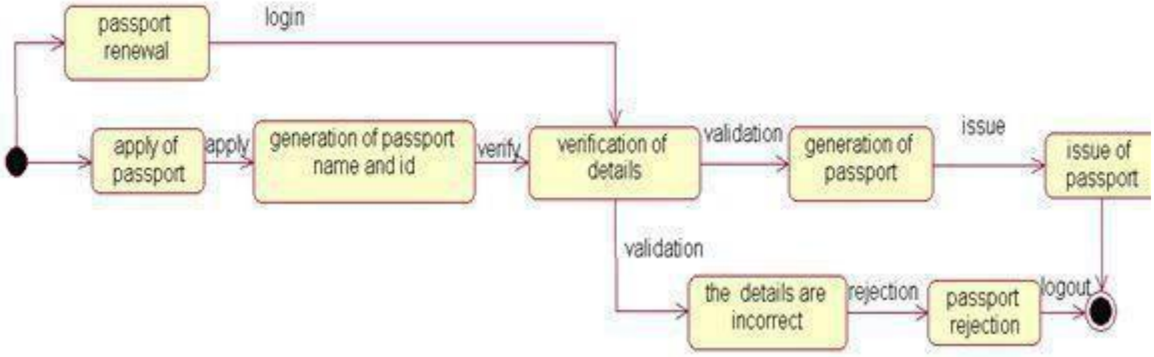
SEQUENCE DIAGRAM


passport
admin
applicant
1: enter name,age,dob and registrer()
Message enter name,age,dob and registrer()
2: verification()
3: if the details are correct
4: renewal of passport()
5: if the details are incorrect()
6: rejection of passport

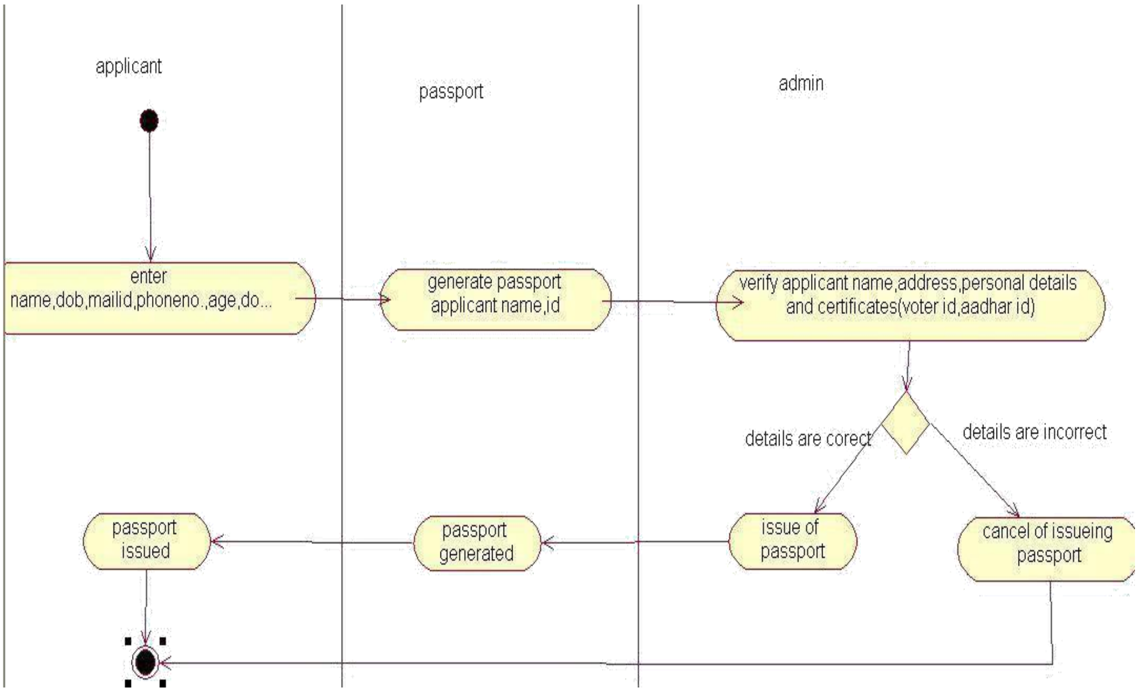
COLLABORATION DIAGRAM


if the details are correct
issue of passport()
4: renewal of passport()
verification()
applican t 1
admin 3
1: enter name,age,dob and registrer()
register()
if the details are incorrect
6: rejection of passport
2: verification()
3: if the details are correct
5: if the details are incorrect()
passport 2

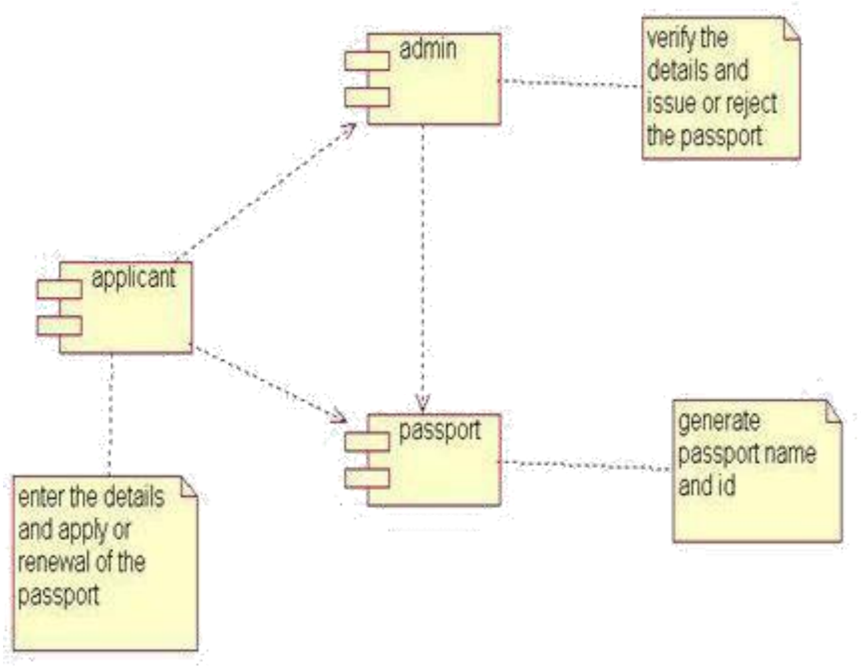
STATECHART DIAGRAM



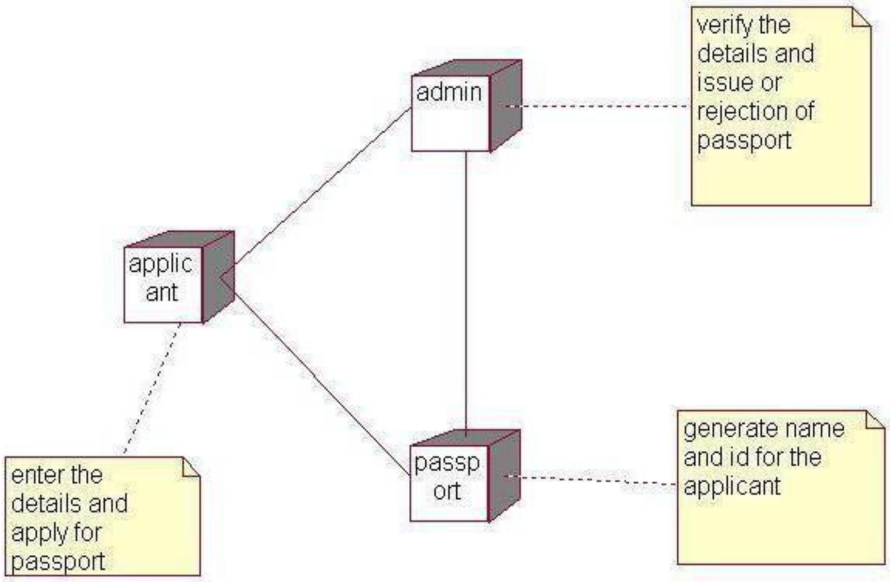
ACTIVITY DIAGRAM


applicant
passport
admin
enter name,dob,mailid,phoneno.,age,do...
generate passport applicant name,id
verify applicant name,address,personal details and certificates(voter id,aadhar id)
details are corect
details are incorrect
issue of passport
cancel of issueing passport
passport generated
passport issued

COMPONENT DIAGRAM

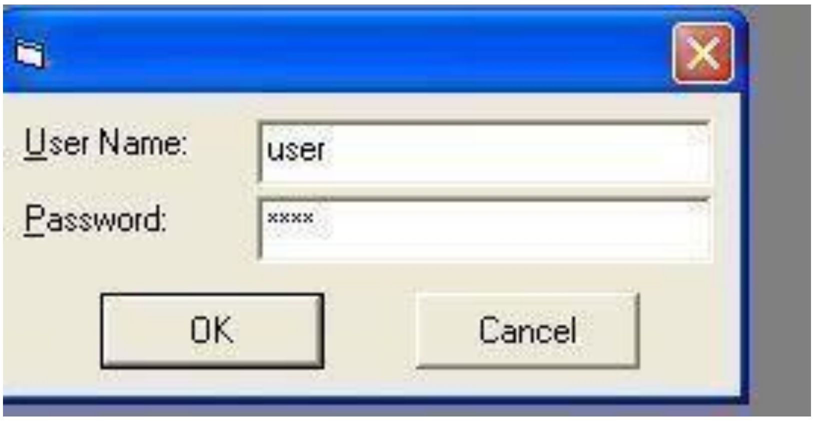

admin
verify the details and issue or reject the passport
applicant
passport
generate passport name and id
enter the details and apply or renewal of the passport

DEPLOYMENT DIAGRAM


admin
verify the details and issue or rejection of passport
applic ant
passp ort
generate name and id for the applicant
enter the details and apply for passport

FORMS:

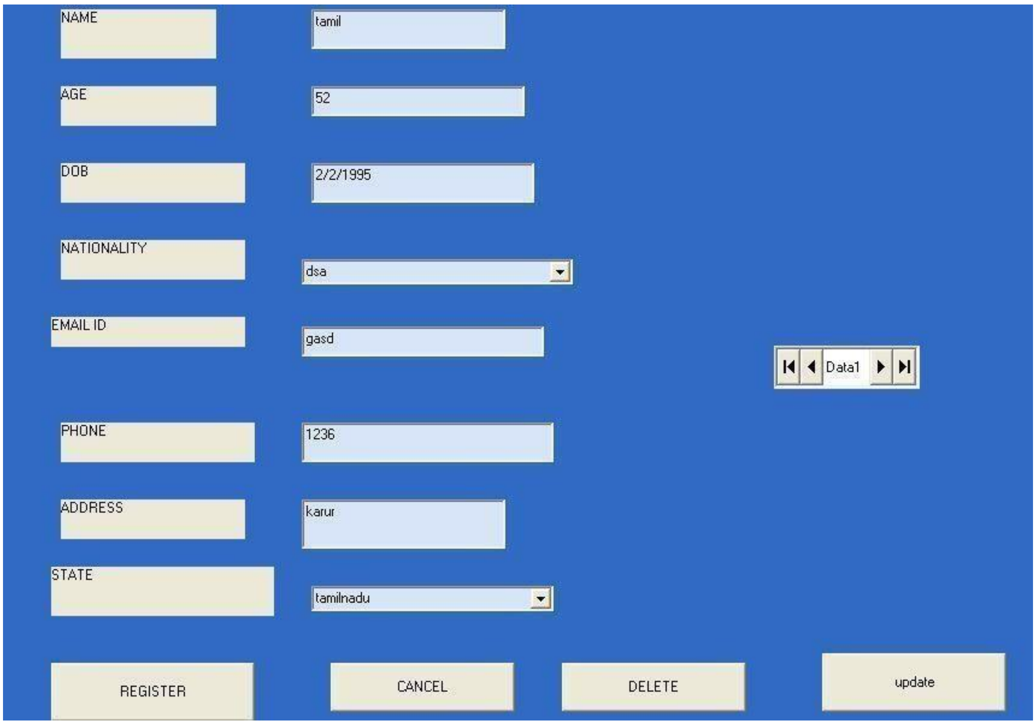
LOGIN FORM:



User Name: user

Password: ****

APPLICANT FORM



NAME tamil

AGE 52

DOB 2/2/1995

NATIONALITY dsa

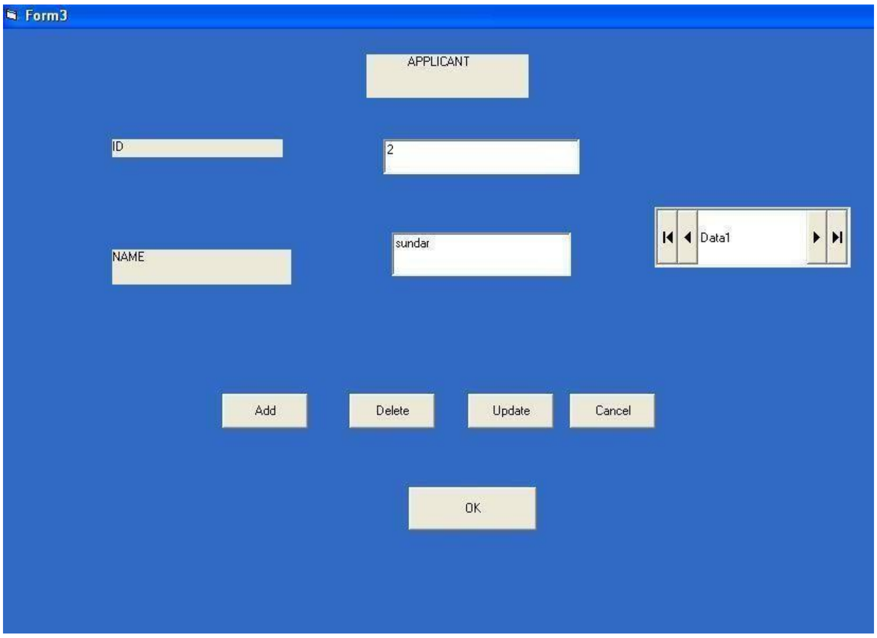
EMAIL ID gasd

PHONE 1236

ADDRESS karur

STATE tamilnadu

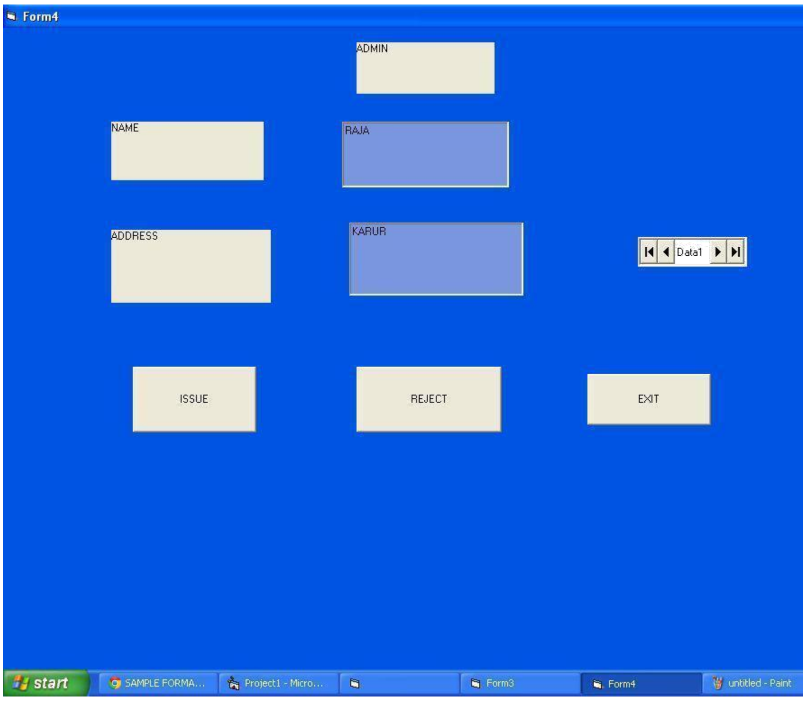
PASSPORT FORM:



ID 2

NAME sundar

ADMIN FORM:



NAME RAJA

ADDRESS KARUR

PROGRAM:

LOGIN FORM:

```
Private Sub cmdOK_Click()
'check for correct password
If txtPassword = "pass" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form2.Show

ElseIftxtPassword = "admin" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form4.Show
Else

MsgBox "Invalid Password, try again!", , "Login"
txtPassword.SetFocus
SendKeys "{Home}+{End}"
End If
End If
End Sub

Private Sub Form_Load()
End Sub
```

APPLICANT FORM:

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub
Private Sub Command2_Click()
Data1.Refresh
End
End Sub
Private Sub Command3_Click()
Data1.UpdateRecord
MsgBox "records are added successfully"
End Sub

Private Sub Command4_Click()
Data1.Recordset.Delete
End Sub
```

```
Private Sub go_Click()
Form3.Show
End Sub

Private Sub Command5_Click()
Form3.Show
End Sub

Private Sub Form_Load()
End Sub
```

APPLICANT FORM:

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub
Private Sub Command2_Click()
Data1.Refresh
End
End Sub

Private Sub Command3_Click()
Data1.UpdateRecord
MsgBox "records are added successfully"
End Sub

Private Sub Command4_Click()
Data1.Recordset.Delete
End Sub

Private Sub go_Click()
Form3.Show
End Sub

Private Sub Command5_Click()
Form3.Show
End Sub

Private Sub Form_Load()
End Sub
```

ADMIN FORM:

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub

Private Sub Command2_Click()
Data1.Refresh
End Sub

Private Sub Command3_Click()
frmLogin.Show
End Sub
```

CONCLUSION:

Thus the project for Passport Automation System was designed and codes are generated and then it was executed successfully.

Ex.No : 2

Date :

BOOK BANK MANAGEMENT

AIM

To develop a project of Book bank management system using Agro UML Software and to implement the software in Visual Basic.

PROBLEM ANALYSIS AND PROJECT DESIGN

The book bank management system is an software in which a member can register themselves and then he can borrow books from the book bank. It mainly concentrates on providing books for engineering students.

PROBLEM STATEMENT

The process of members registering and purchasing books from the book bank are described sequentially through following steps:

a. First the member registers himself if he was new to the book bank.

b. Old members will directly select old member button..

c. They select their corresponding year.

d. After selecting the year they fill the necessary details and select the book and he will be directed towards administrator

e. The administrator will verify the status and issue the book.

SOFTWARE REQUIREMENT SPECIFICATION

S.NO	CONTENTS
1.	INTRODUCTION
2.	OBJECTIVE
3.	OVERVIEW
4.	GLOSSARY
5.	PURPOSE
6.	SCOPE
7.	FUNCTIONALITY
8.	USABILITY
9.	PERFORMANCE
10.	RELIABILITY
11.	FUNCTIONAL REQUIREMENTS
12.	EXTERNAL INTERFACE REQUREMENTS

1. INTRODUCTION

This system would be used by members who are students of any college to check the availability of the books and borrow the books, and then the databases are updated. The purpose of this document is to analyze and elaborate on the high-level needs and features of the book bank management system**.** It also tells the usability, reliability defined in use case specification.

2. OBJECTIVE

The main objective of the system are was to design an online book-bank monitoring system to enable a central monitoring mechanism of the book-bank be more faster and less error prone. Apart from this

a. To help the students acquire the right books for the syllabus at the right time.

b. To ensure availability of basic textbooks to students against limited funds and To develop students ability to handle property loaned to them.

3. OVERVIEW

The overview of this project is to design a tool for book bank so that it can be used by any book banks to lend their books as well as colleges.

4. GLOSSARY

TERMS	**DESCRIPTION**
MEMBER	The one who registers himself and purchase books from the bank.
DATABASE	Database is used to store the details of members and books.
ADMINISTRATOR	The one who verifies the availability of book and issue them
USER	Member
SOFTWARE REQUIREMENT SPECIFICATION	This software specification documents full set of features and function for online recruitment system that is performed in company website.

5. PURPOSE

The purpose of the book bank management system is to reduce the manual intervention.

6. SCOPE

The scope of this book bank management system is to act as a tool for book bank administrator for quick reference, availability of the books.

7. FUNCTIONALITY

Many members will be waiting to take the book from the book bank at a single day. To serve all the members.

8. USABILITY

User interface makes the Recruitment system to be efficient. That is the system will help the member to register easily and helps them to get their books easily. The system should be user friendly.

9. PERFORMANCE

It describes the capability of the system to perform the recruitment process of the applicant without any error and performing it efficiently.

10. RELIABILITY

The book bank management system should be able to serve the applicant with correct information and day-to-day update of information.

11. FUNCTIONAL REQUIREMENTS

Functional requirements are those refer to the functionality of the system. That is the services that are provided to the member who borrows book.

12. EXTERNAL INTERFACE REQUIREMENTS

SOFTWARE REQUIREMENTS

1. **Front end:** Agro UML.
2. **Back end:** visual basic 6.0.

HARDWARE REQUIREMENTS

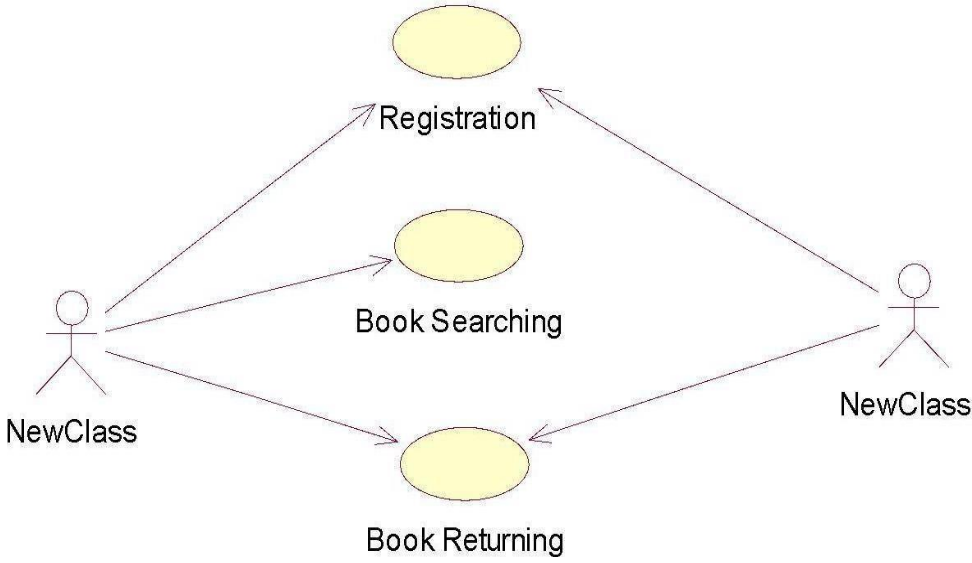
1. **Processor:** Pentium 4.
2. **RAM : 256** MB
3. **Operating system :** Microsoft windows XP.
4. **Free disk space : 1**GB

UML DIAGRAMS:

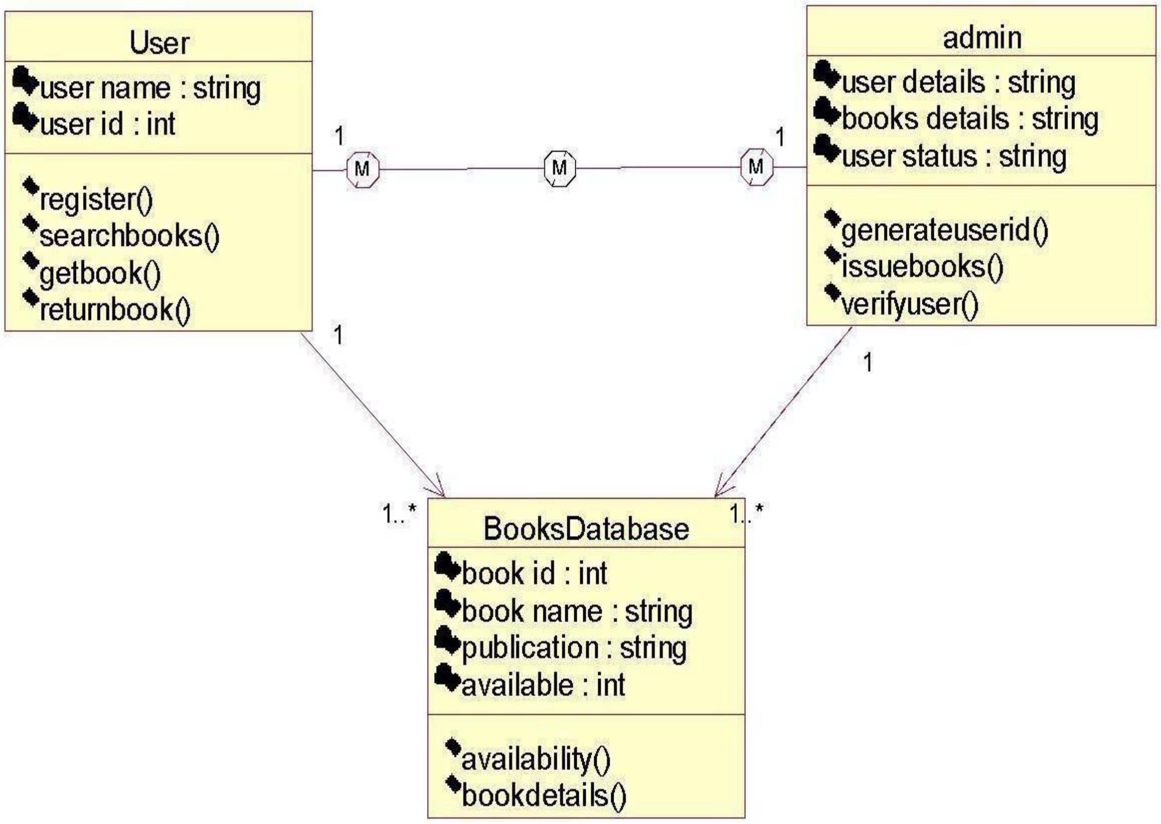
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

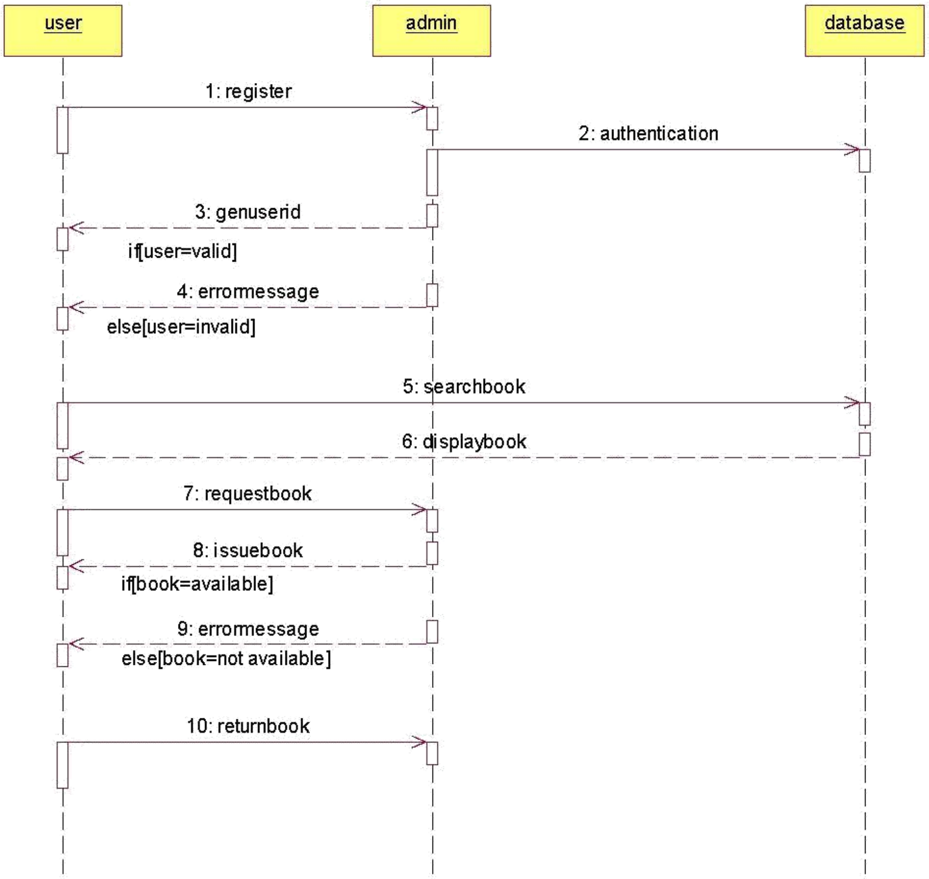
USE CASE DIAGRAM



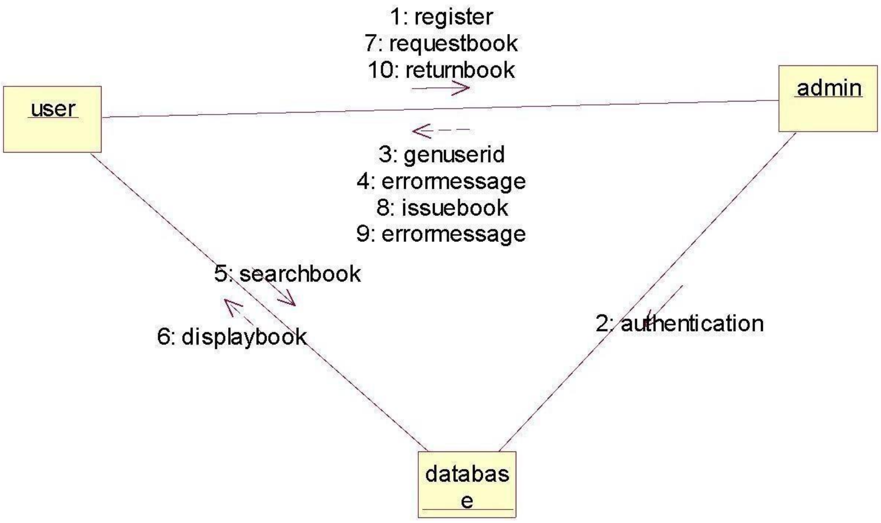
CLASS DIAGRAM



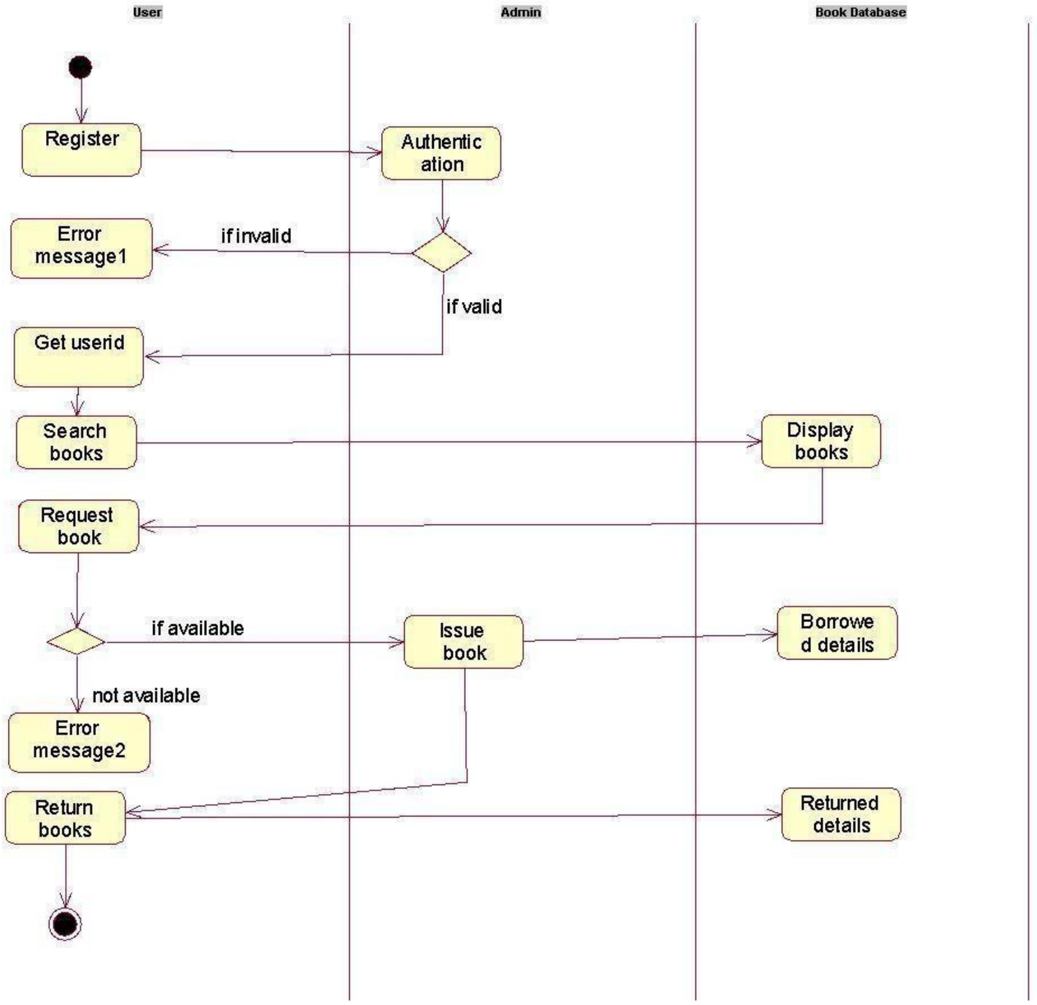
SEQUENCE DIAGRAM



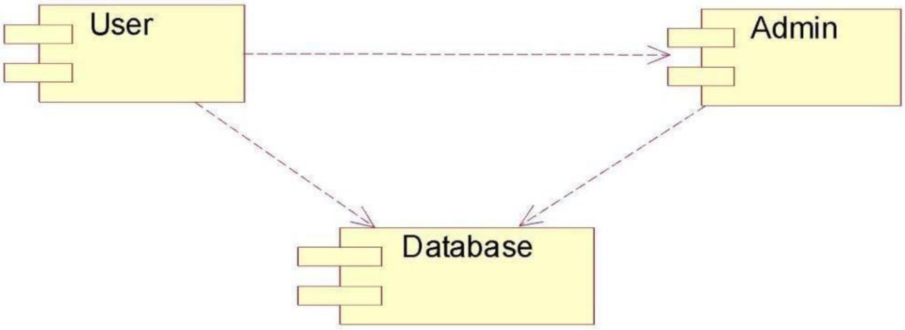
COLLABORATION DIAGRAM



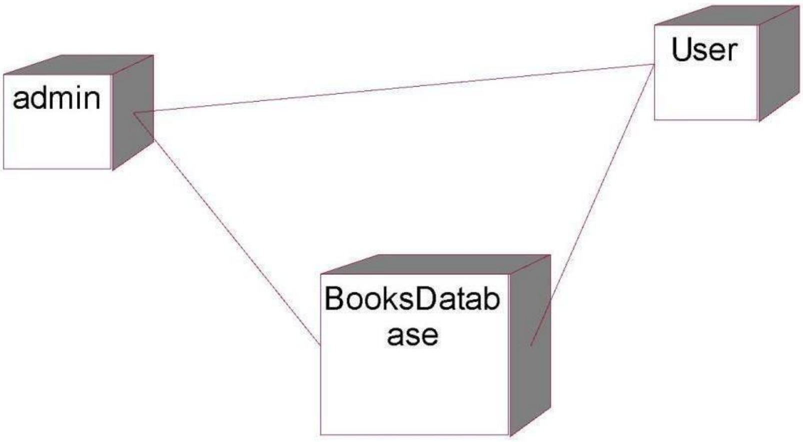
ACTIVITY DIAGRAM



COMPONENT DIAGRAM

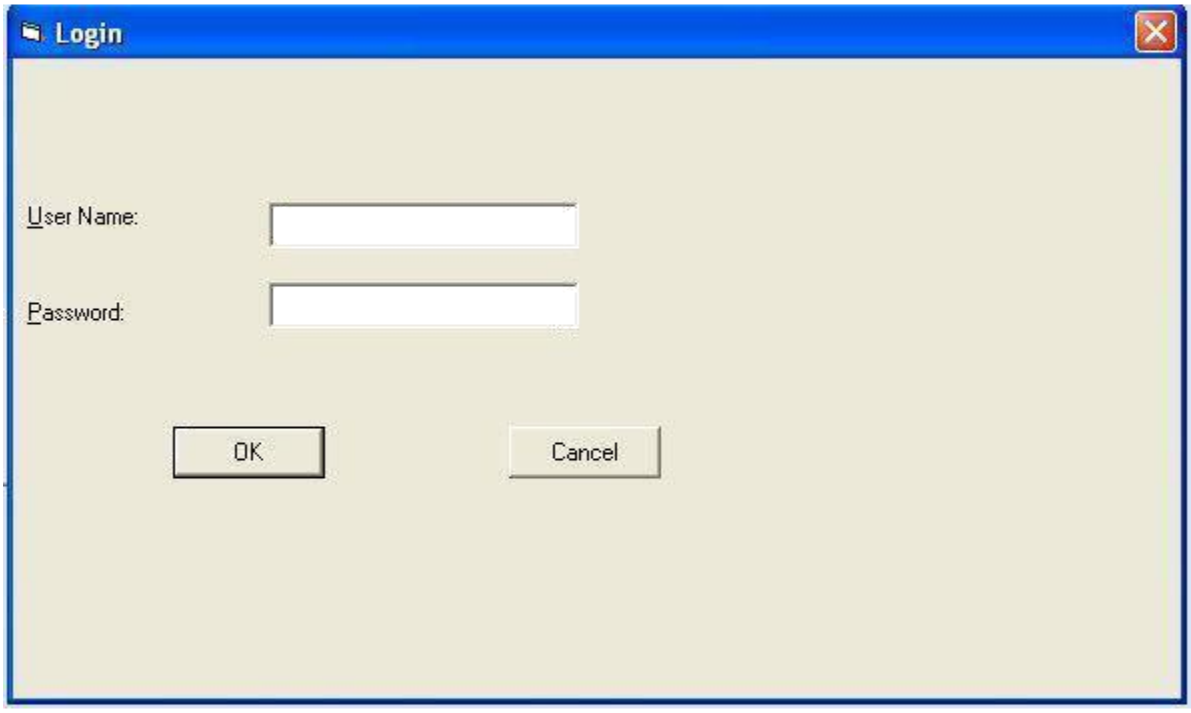


DEPLOYMENT DIAGRAM

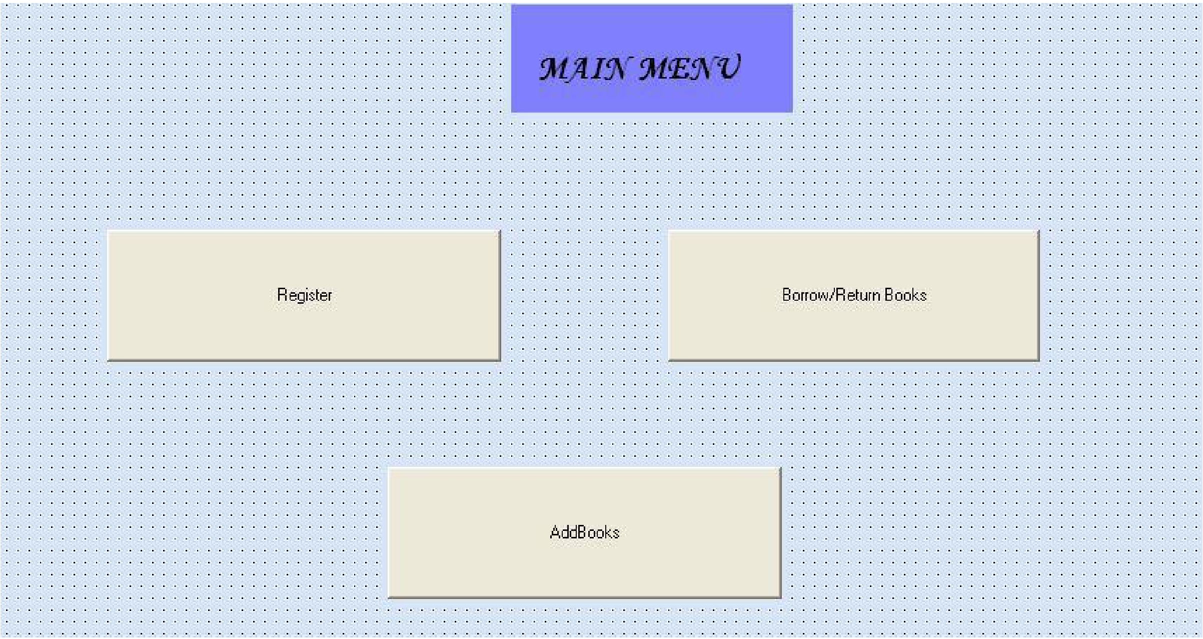


IMPLEMENTATION

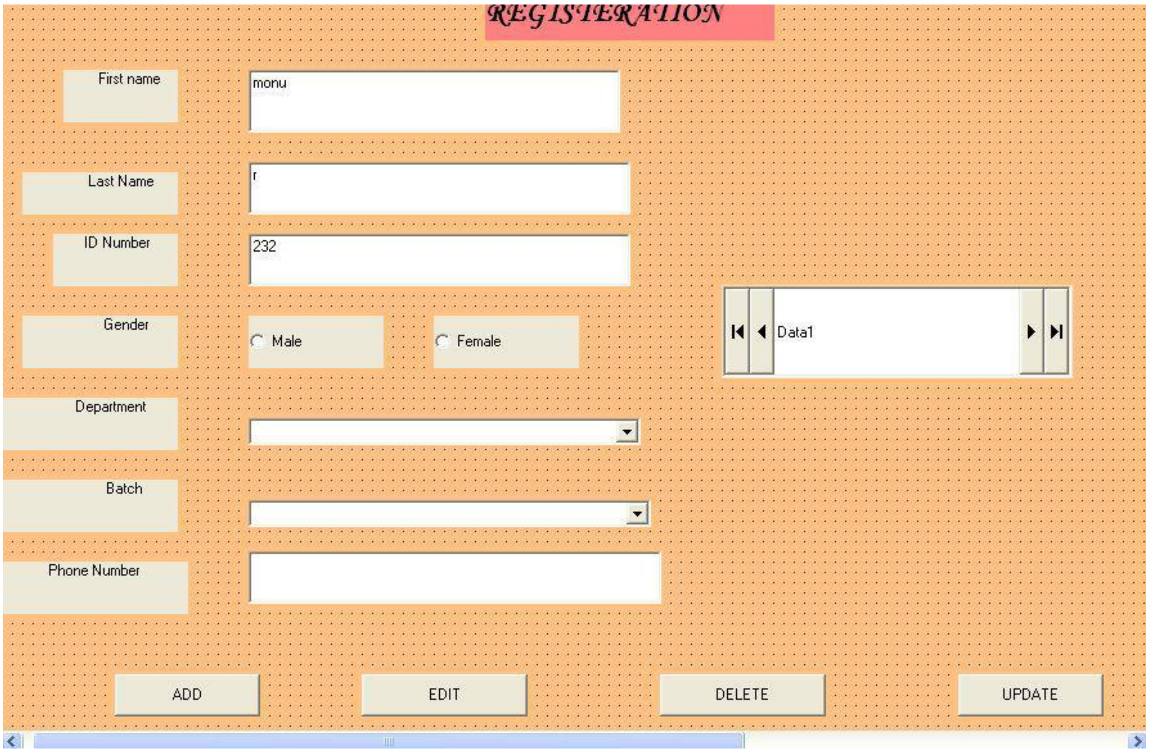
FORM 1:



FORM 2:



FORM 3:


REGISTERATION

First name: monu

Last Name: r

ID Number: 232

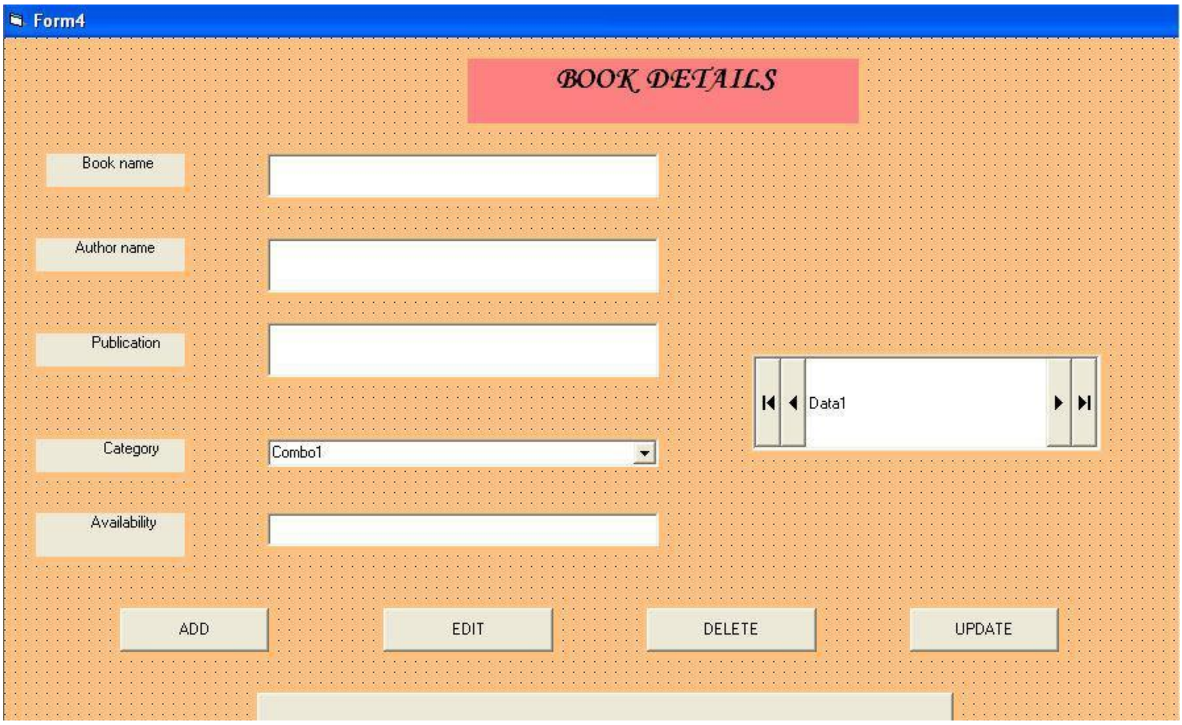
Gender: ○ Male ○ Female

Department:

Batch:

Phone Number:

FORM 4:



PROGRAM:

FORM1:

```
Private Sub Command1_Click()
Form2.Show
End Sub
Private Sub Command2_Click()
frmLogin.Show
End Sub


Private Sub Command4_Click()
frmLogin.Show
End Sub
```

FORM2:

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub

Private Sub Command2_Click()
Data1.Recordset.Edit
End Sub
Private Sub Command3_Click()
```

```
Data1.Recordset.Delete
MsgBox "the records are deleted successfully !!!"
End Sub

Private Sub Command4_Click()
Data1.Recordset.Update
MsgBox "the records are updated successfully"
End Sub

Private Sub Command5_Click()
Form1.Show
End Sub
```

FORM3:

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub

Private Sub Command2_Click()
Data1.Recordset.Edit
End Sub

Private Sub Command3_Click()
Data1.Recordset.Delete
End Sub

Private Sub Command4_Click()
Data1.Recordset.Update
End Sub

Private Sub Text2_Change()
End Sub


Private Sub Command6_Click()
Form1.Show
End Sub
```

CONCLUSION:

Thus the project for Book bank management system was designed and codes are generated and then it was executed successfully.

Ex.No:3
Date:

EXAM REGISTRATION SYSTEM

AIM

To develop a project Exam Registration using Agro UML Software and to implement the software in Visual Basic.

PROBLEM ANALYSIS AND PROJECT PLANNING

The Exam Registration is an application in which applicant can register themselvesfor the exam. The details of the students who have registered for the examination will be stored in a database and will be maintained. The registered details can then be verified for any fraudulent or duplication and can be removed if found so. The database which is verified can be used to issue hall tickets and other necessary materials to the eligible students.

PROBLEM STATEMENT

The process of students accessing the registration application and applying for the examination by filling out the form with proper details and then the authorities verify those details given for truth and correctness are sequenced through steps

a. The students access exam registration application.
b. They fill out the form with correct and eligible details. They complete the payment process.
c. The authorities verify or check the details.
d. After all verification the exam registration database is finalized.

SOFTWARE REQUIREMENT SPECIFICATION

S.NO	CONTENTS
1.	INTRODUCTION
2.	OBJECTIVE
3.	OVERVIEW
4.	GLOSSARY
5.	PURPOSE
6.	SCOPE
7.	FUNCTIONALITY
8.	USABILITY
9.	PERFORMANCE
10.	RELIABILITY
11.	FUNCTIONAL REQUIREMENTS
12.	EXTERNAL INTERFACE REQUREMENTS

1. INTRODUCTION

Exam Registration application is an interface between the Student and the Authority responsible for the Exams. It aims at improving the efficiency in the registration of exams and reduces the complexities involved in it to the maximum possible extent.

2. OBJECTIVE

The main objective of Exam Registration System is to make applicants register themselves and apply for the exam. Exam Registration System provides easy interface to all the users to apply for the exam easily.

3. OVERVIEW

The overview of the project is to design an exam registration tool for the registration process which makes the work easy for the applicant as well as the Authorities of Exam. Authorities of the exam can keep track of and maintain the database of the registered applicants for the exams.

4. GLOSSARY

TERMS	DESCRIPTION
APPLICANT OR STUDENT DATABASE	Applicant can register himself by filling out the registration form and finally paying the payment for attending the exam. Database is used to maintain and store the details of registered applicants.
SOFTWARE REQUIREMENT SPECIFICATION	This software specification documents full set of features and function for online recruitment system that is performed in company website.

5. PURPOSE

The purpose of exam registration system is to register for the exam in an easier way and to maintain the registered details in an effective manner.

6. SCOPE

The scope of this Exam Registration process is to provide an easy interface to the applicants where they can fill their details and the authorities maintain those details in an easy and effective way.

7. FUNCTIONALITY

The main functionality of registration system is to make the registration and database for it to be maintained in an efficient manner.

8. USABILITY

User interface makes the Exam Registration system to be efficient. That is the system will help the applicant to register easily and helps the authorities to maintain details effectively. The system should be user friendly.

9. PERFORMANCE

It describes the capability of the system to perform the registration process of the applicant without any error and performing it efficiently.

10. RELIABILITY

The Exam Registration system should be able to serve the applicant with correct information and day-to-day update of information.

11. FUNCTIONAL REQUIREMENTS

Functional requirements are those refer to the functionality of the system. That is the services that are provided to the applicant who apply for the Exam.

12. EXTERNAL INTERFACE REQUIREMANTS

SOFTWARE REQUIREMENTS

1. **Front end:** Agro UML.
2. **Back end:** visual basic 6.0.

HARDWARE REQUIREMENTS

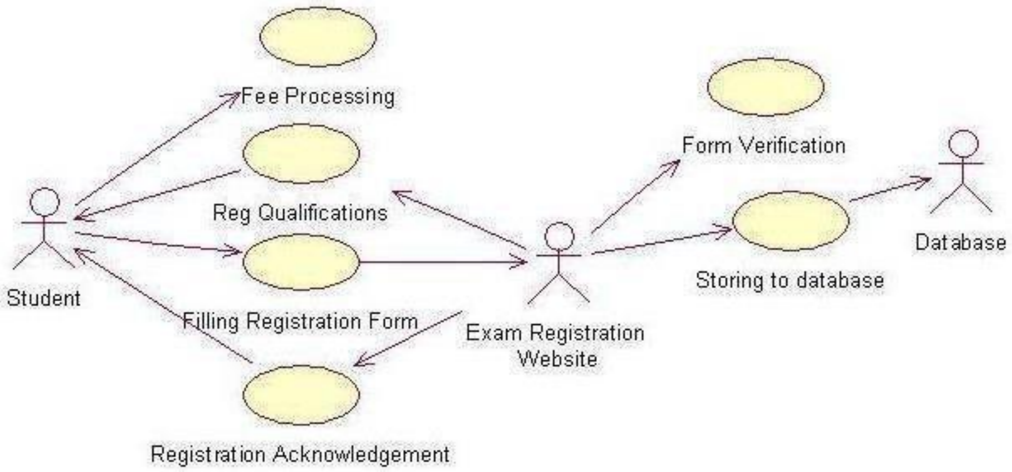
1. **Processor :** Pentium 4.
2. **RAM :** 256MB
3. **Operating system :** Microsoft windows XP.
4. **Free disk space : 1**GB

UML DIAGRAMS

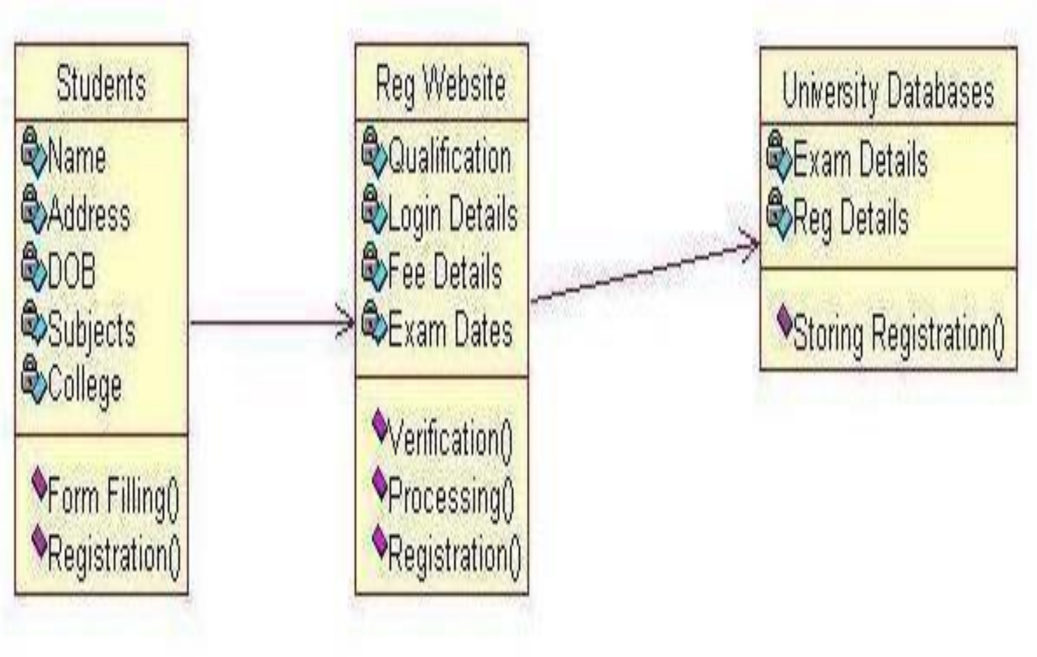
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

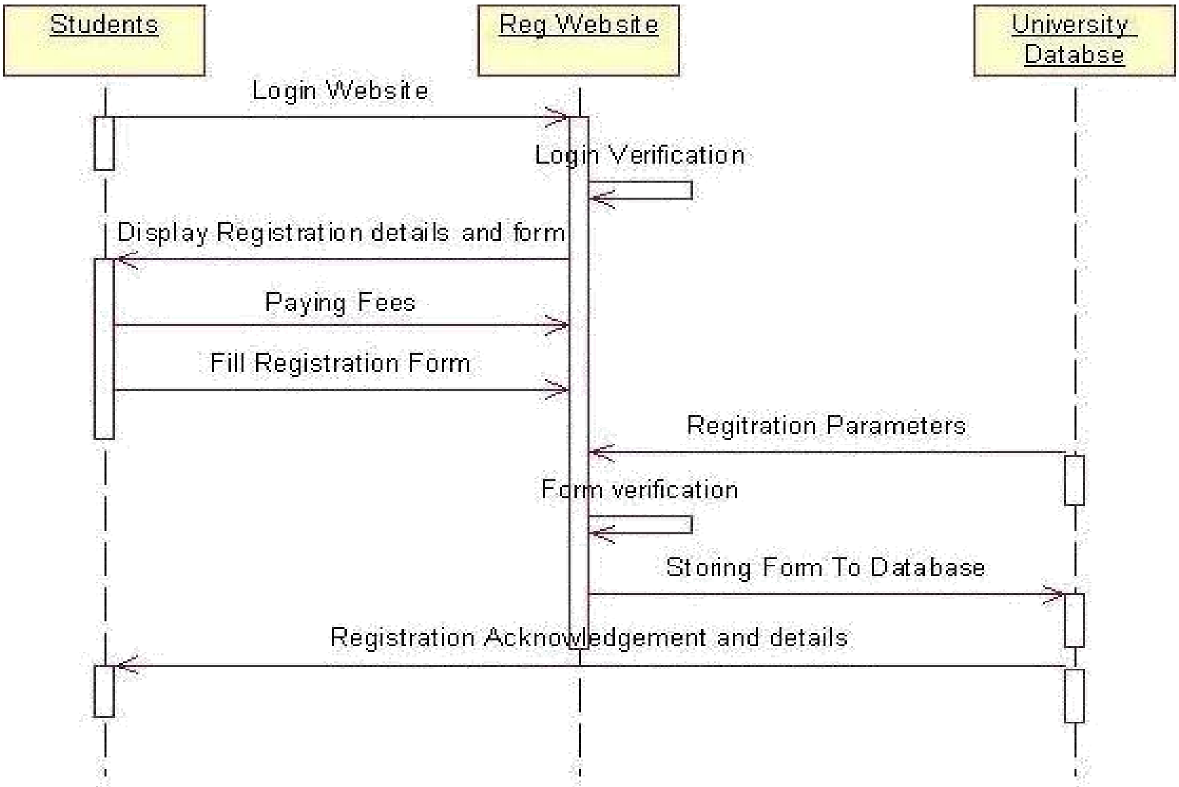
USE CASE DIAGRAM



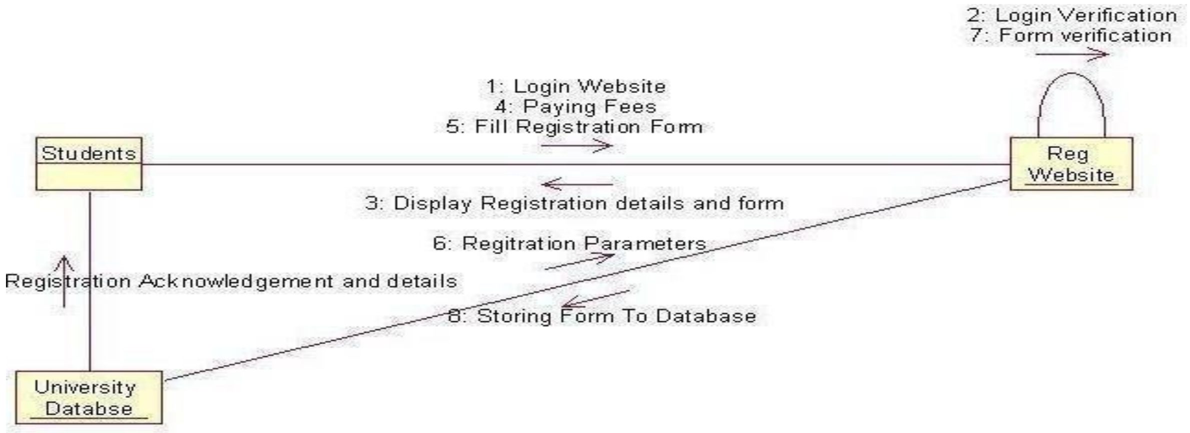
CLASS DIAGRAM



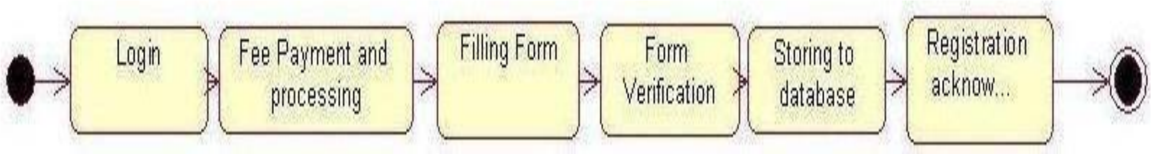
SEQUENCE DIAGRAM



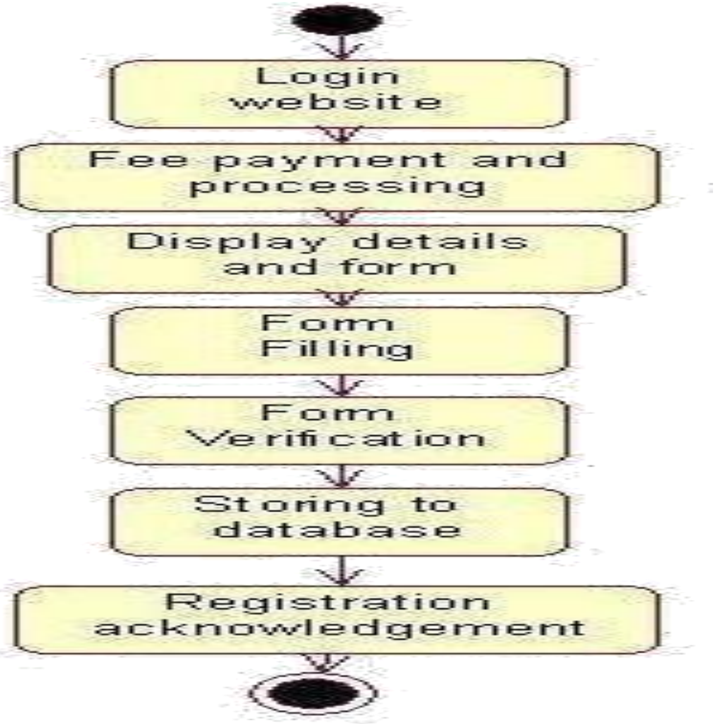
COLLABRATION DIAGRAM


2: Login Verification
7: Form verification
1: Login Website
4: Paying Fees
5: Fill Registration Form
Students
Reg Website
3: Display Registration details and form
6: Regitration Parameters
Registration Acknowledgement and details
8: Storing Form To Database
University Databse

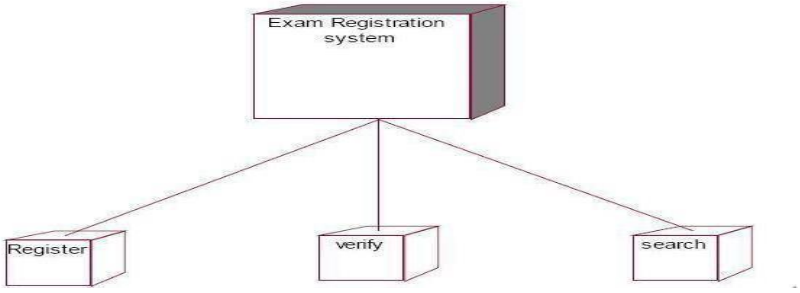
STATE CHART DIAGRAM



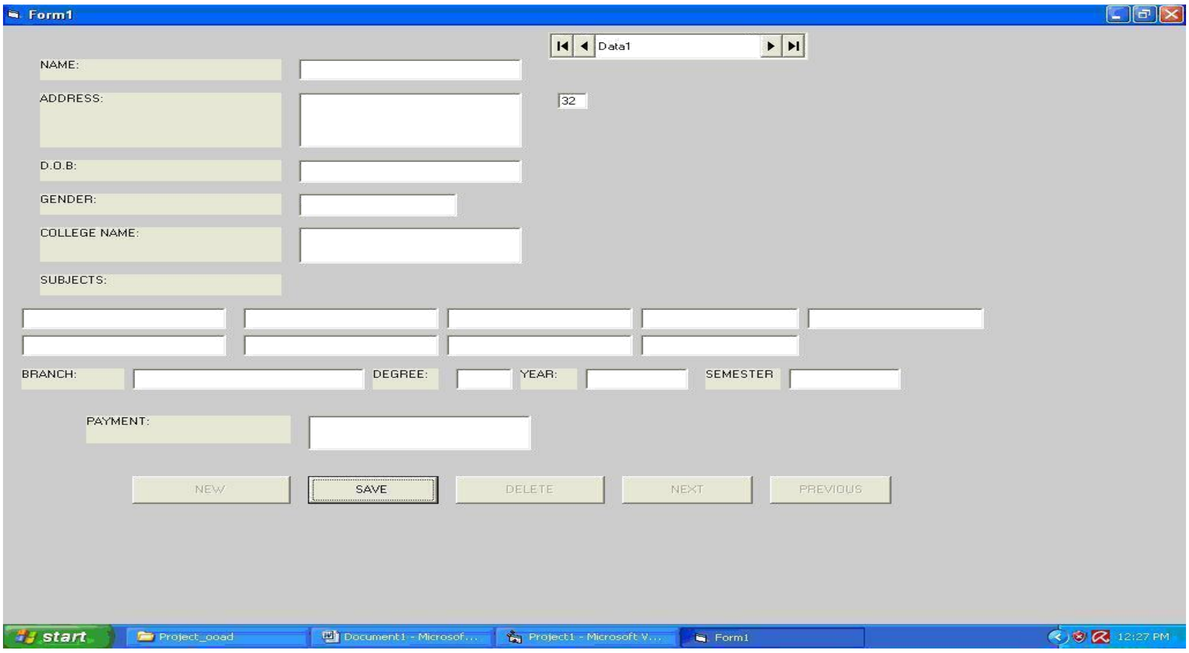
ACTIVITY DIAGRAM



DEPLOYMENT DIAGRAM



FORM



CODING

```
Dim ob1 as student
Private sub cmdregistrationform_click()
Set ob1=new students
ob1.Form_Filling
End sub
Public Sub Form_Filling()
datcollege.Recordset.AddNew
cmdSaveRecord.Enabled = True
cmdMovePrevious.Enabled = False
cmdMoveNext.Enabled = False
cmdNewRecord.Enabled = False
cmdDeleteRecord.Enabled = False
Form1.Show
End Sub
```

```
Dim ob2 as students
Private sub cmdregistrationform_click()
Set ob2=new students

ob2.Next
End sub
Private Sub cmdMoveNext_Click()
Form1.Show
End Sub
Private Sub next()
datcollege.Recordset.MoveNext
If datcollege.Recordset.EOF = True Then
datAuthors.Recordset.MoveLast
End If

Form1.Show
End Sub
Dim ob3 as students
Private sub cmdregistrationform_click()
Set ob3=new students
ob3.Previous
End sub
Private Sub previous()
Form1.show
End Sub
Dim ob4 as students
Private sub cmdregistrationform_click()
Set ob4=new students
ob4. Storing_registration
End Sub
Public Sub Storing_registration()
If MsgBox("Are you sure you want to save this record?", _ vbQuestion + vbYesNo +
vbDefaultButton2, _
"Confirm") = vbNo Then
datcollege.Recordset.Update
Exit Sub
End If
cmdSaveRecord.Enabled = False
cmdMovePrevious.Enabled = True
cmdMoveNext.Enabled = True
cmdDeleteRecord.Enabled = True
cmdNewRecord.Enabled = True
Form1.Show
End Sub
Private Sub cmdSaveRecord_Click()
Form1.Show
End Sub

Public Sub Processing()
datcollege.Recordset.MovePrevious
```

```
Form1.show
End Sub
Dim ob5 as students
Private sub cmdregistrationform_click()
Set ob5=new students
ob5. Delete
End Sub

Public Sub Delete()
On Error GoTo Delete_Error
If MsgBox("Are you sure you want to delete this record?", _
vbQuestion + vbYesNo + vbDefaultButton2, _

"Confirm") = vbNo Then
Exit Sub
End If
datcollege.Recordset.Delete
cmdMoveNext_Click
Exit Sub
Delete_Error:
MsgBox "This record cannot be deleted. Error code = " _
Err.Number & vbCrLf & Err.Description, _ vbCritical, "Cannot Delete"

End Sub
Private Sub cmdDeleteRecord_Click() form1.show
End Sub
```

CONCLUSION

Thus the project to develop Exam Registration system using Agro UML Software and to implements the software in Visual Basic is done successfully.

Ex. No.: 4
Date :

STOCK MAINTENANCE SYSTEM

AIM

To develop a project stock maintenance system using Agro UML and to implement the software in Visual Basic.

PROBLEM ANALYSIS AND PROJECT PLANNING

The Stock Maintenance System, initial requirement to develop the project about the mechanism of the Stock Maintenance System is caught from the customer. The requirement are analysed and refined which enables the end users to efficiently use Stock Maintenance System. The complete project is developed after the whole project analysis explaining about the scope and the project statement is prepared.

PROBLEM STATEMENT

The process of stock maintenance system is that the customer login to the particular site to place the order for the customer product. The stock maintenance system are described sequentially through steps

a. The customer login to the particular site.
b. They fill the customer details.
c. They place the orders for their product.
d. The vendor login and views the customer details and orders.

SOFTWARE REQUIREMENT SPECIFICATION

S.NO	CONTENTS
1.	INTRODUCTION
2.	OBJECTIVE
3.	OVERVIEW
4.	GLOSSARY
5.	PURPOSE
6.	SCOPE
7.	FUNCTIONALITY
8.	USABILITY
9.	PERFORMANCE
10.	RELIABILITY
11.	FUNCTIONAL REQUIREMENTS
12.	EXTERNAL INTERFACE REQUREMENTS

1. INTRODUCTION

This software specification documents full set of features and function for online stock maintenance system that is performed in company website. In this we give specification about the customer orders. It tells the usability, reliability defined in use case specification.

2. OBJECTIVE

The main objective of the stock maintenance system is to maintain the stock. It provides the vendor to maintain the stock in an precise manner.

3. OVERVIEW

The overview of the project is to design an online tool for the recruitment process which eases the work for the customer as well as the companies. Companies can create their company forms according to their wish in which the applicant can register.

4. GLOSSARY

TERMS	DESCRIPTION
CUSTOMER	The customer can have the username and password after login to the system. After login they directed to fill the customer details. And the customer places their order. After placing orders they lead to verify all the details in a single form. Then they places the order successfully.
VENDOR	Vendor has the login id. After login vendor verify the customer details and orders. And maintain the stocks
DATABASE	Database is used to verify thecustomer details and orders.
SOFTWARE REQUIREMENT SPECIFICATION	This software specification documents full set of features and function for stock maintenance system that is performed in company website.

5. PURPOSE

The purpose of stock maintenance system is to maintain the stock in an precise manner.

6. SCOPE

The scope of this stock maintenance system is to maintain the stock.

7. FUNCTIONALITY

The main functionality of the stock maintenance system is to maintain the stock.

8. USABILITY

User interface makes the stock maintenance system to be efficient. That is the system will help the customer to place the details and orders easily and helps the vendor to maintain the stock accurate. The system should be user friendly.

9. PERFORMANCE

It describes the capability of the system to maintain the stock without any loss of stock and performing it efficiently.

10. RELIABILITY

The stock maintenance system should be able to maintain the stock with correct updates from day to day placement of new orders from customer.

11. FUNCTIONAL REQUIREMENTS

Functional requirements are those refer to the functionality of the system. That is the services that are provided to the customer who places the orders.

SOFTWARE REQUIRMENTS

- Microsoft Visual Basic 6.0
- Agro UML
- Microsoft Access

HARDWARE REQUIRMENTS

- 1GB RAM
- Pentium IV Processor
- 100 GB HARDDISK

ANALYSIS MODELING

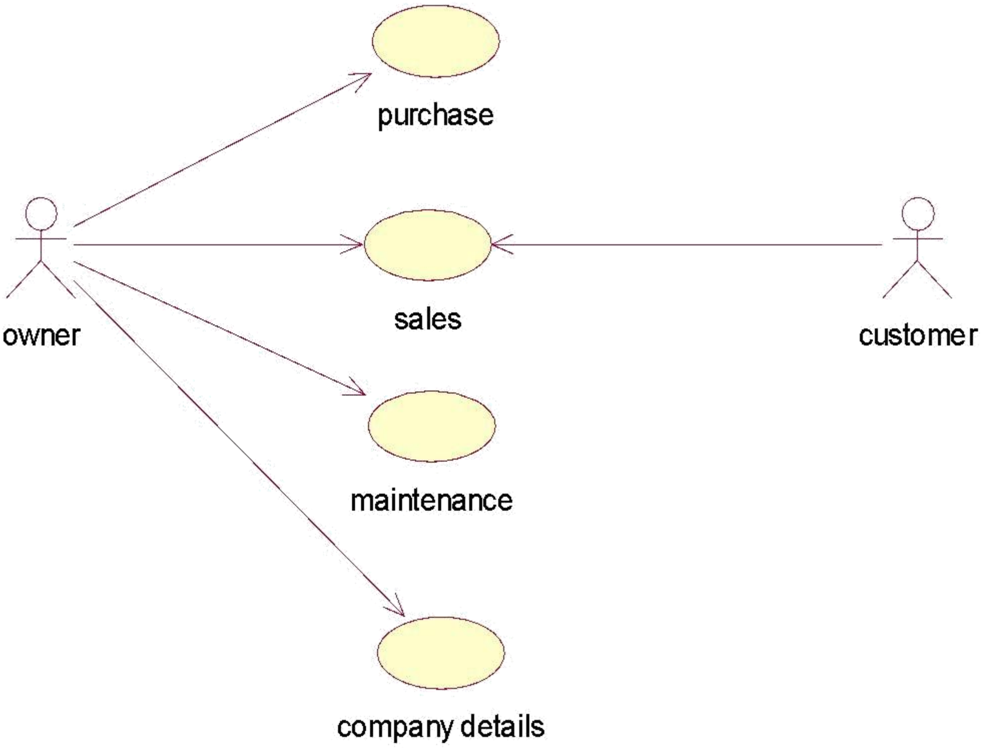
The project can be explained diagrammatically using the following diagrams:

UML DIAGRAMS:

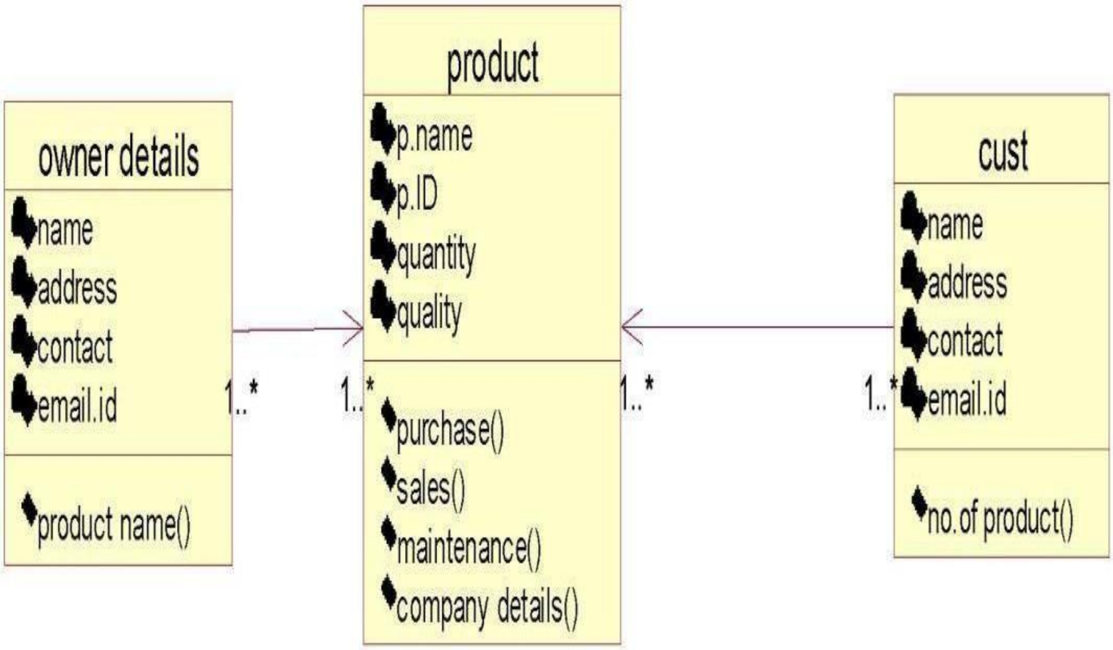
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

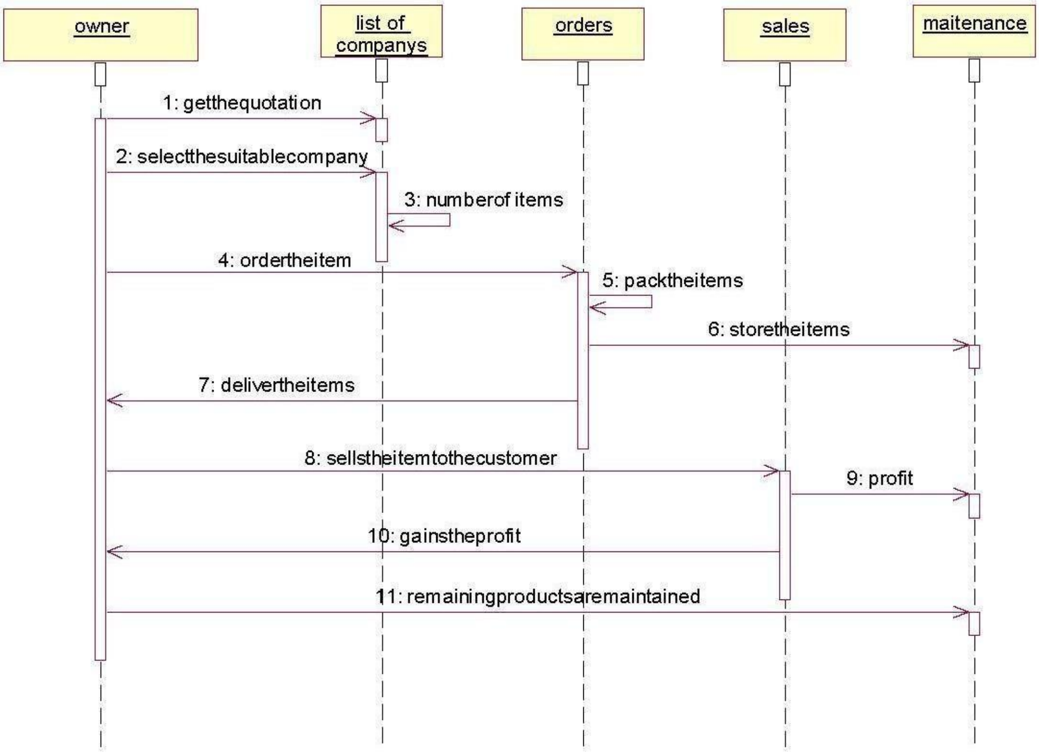
USE CASE DIAGRAM



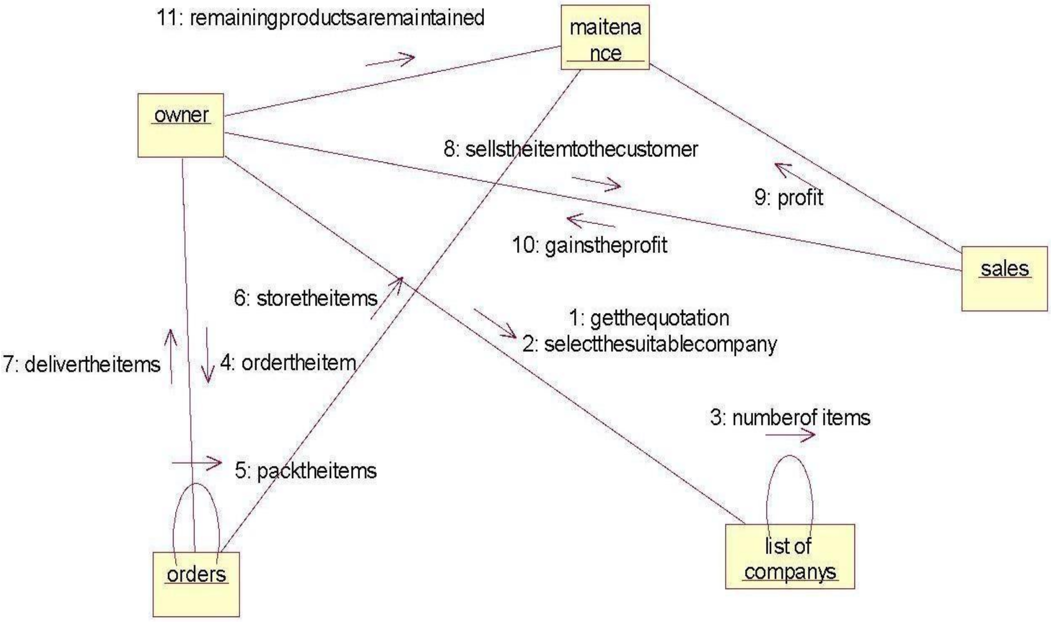
CLASS DIAGRAM



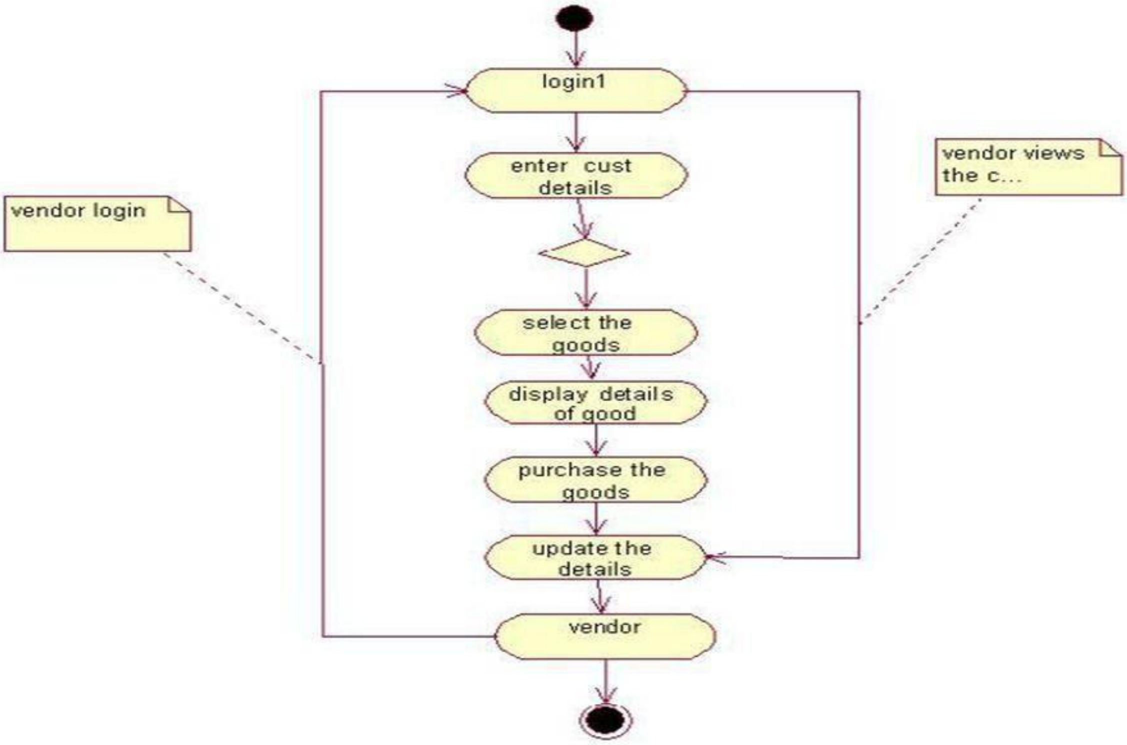
SEQUENCE DIAGRAM



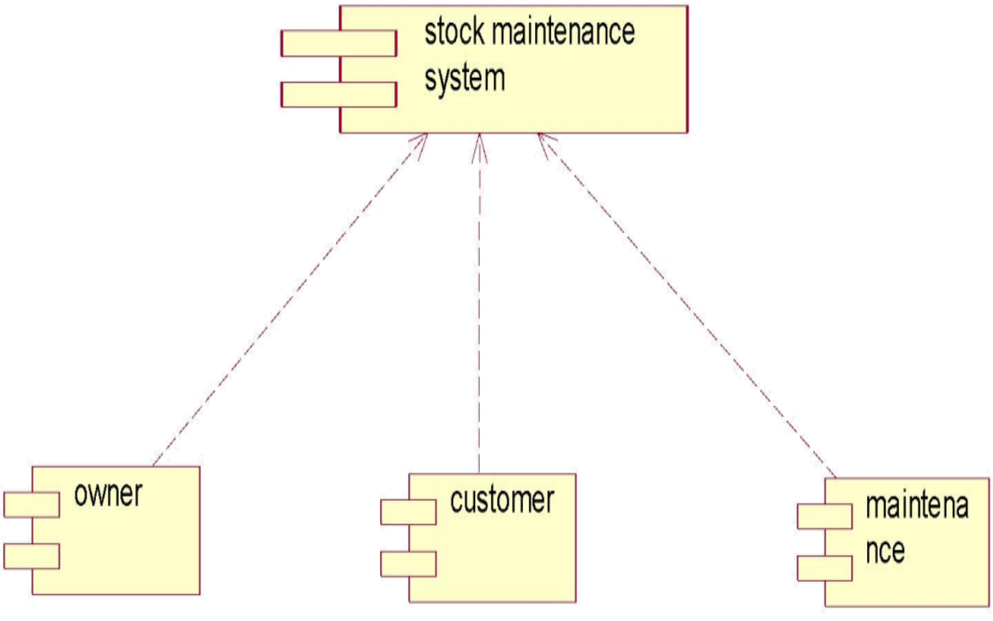
COLLABORATION DIAGRAM



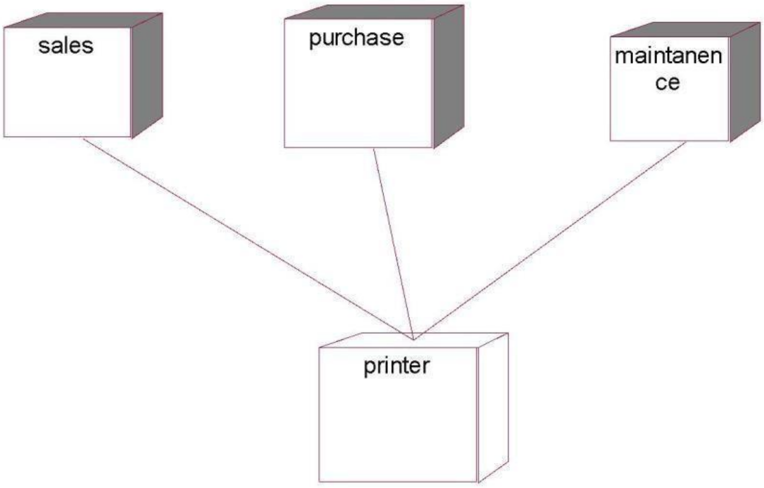
ACTIVITY DIAGRAM



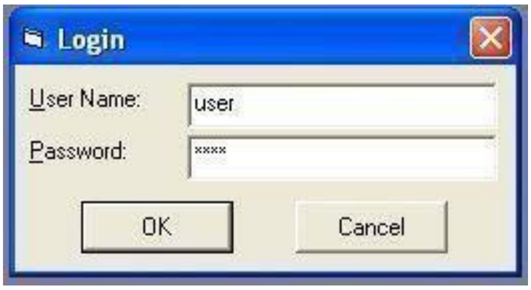
COMPONENT DIAGRAM



DEPLOYMENT DIAGRAM



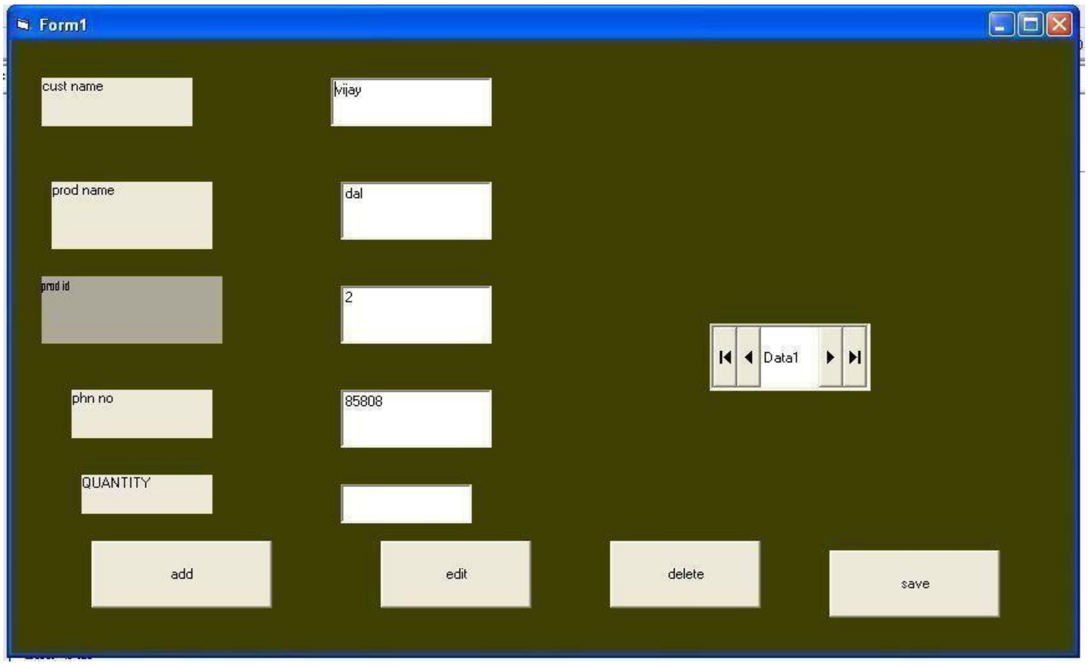
IMPLEMENTATION
LOGIN FORM



User Name: user

Password: ****

FORM1:



cust name vijay

prod name dal

prod id 2

phn no 85808

QUANTITY

CODING

```
Public LoginSucceededAs Boolean
Private Sub cmdCancel_Click()
'set the global var to false
'to denote a failed login
LoginSucceeded = False
Me.Hide
End Sub
Private Sub cmdOK_Click()
'check for correct password
If txtPassword = "cust" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form1.Show
Else
If txtPassword = "admin" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form2.Show
Else
MsgBox "Invalid Password, try again!", , "Login"
txtPassword.SetFocus
SendKeys "{Home}+{End}"
End If
End If
End Sub
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub
Private Sub Command2_Click()
Data1.Recordset.Edit
End Sub
Private Sub Command3_Click()
Data1.Recordset.Delete
End Sub
Private Sub Label1_Click()
Data1.Recordset.Update
End Sub
```

CONCLUSION:

Thus the project for Stock maintenance System was designed and codes are generated and then it was executed successfully.

Ex.No: 5
Date :

ONLINE COURSE RESERVATION

AIM

To design an object oriented model for online course reservation system.

PROBLEM ANALYSIS AND PROJECT PLANNING

The requirement form the customer is got and the requirements about the course registration are defined. The requirements are analysed and defined so that is enables the student to efficiency select a course through registration system. The project scope is identified and the problem statement is prepared.

PROBLEM STATEMENT

a. Whenever the student comes to join the course he/she should be provided with the list of course available in the college.

b. The system should maintain a list of professor who is teaching the course. At the end of the course the student must be provided with the certificate for the completion of the course.

SYSTEM REQUIEMENT SPECIFICATION

GLOSSARY

Generally a glossary is performed to define the entire domain used in the problem. It defines about the storage items that are familiar to the uses it provided these definitions and information about the attribute we are using in the particular project to the use,

DEFINITIONS

The glossary contain the working definition for the key concept in the course registration system

COURSE

The course which are offered by the institution

COURSE CATALOG

Thus a bridge for all the course offered by the institution.

GRADE

The ranking of a particular student for a particular course offered

CERTIFICATE

It is the proof for the completion the course

REGISTER

`One who register the course for the student

OBJECTIVES

a. The main purpose of creating the document about the software is to know about the list of the requirement in the software project part of the project to be developed.
b. It specifies the requirement to develop a processing software part that completes the set of requirement.

SCOPE

a. In this specification, we define about the system requirements that are about from the functionality of the system.
b. It tells the users about the reliability defined in use case specification

FUNCTIONALITY

- ✓ Many members of the process line to check
- ✓ It is the capability about which it can performed for its occurrences and transaction, we are have to carry over at sometimes

USABILITY

The user interface to make the transaction should be effectively

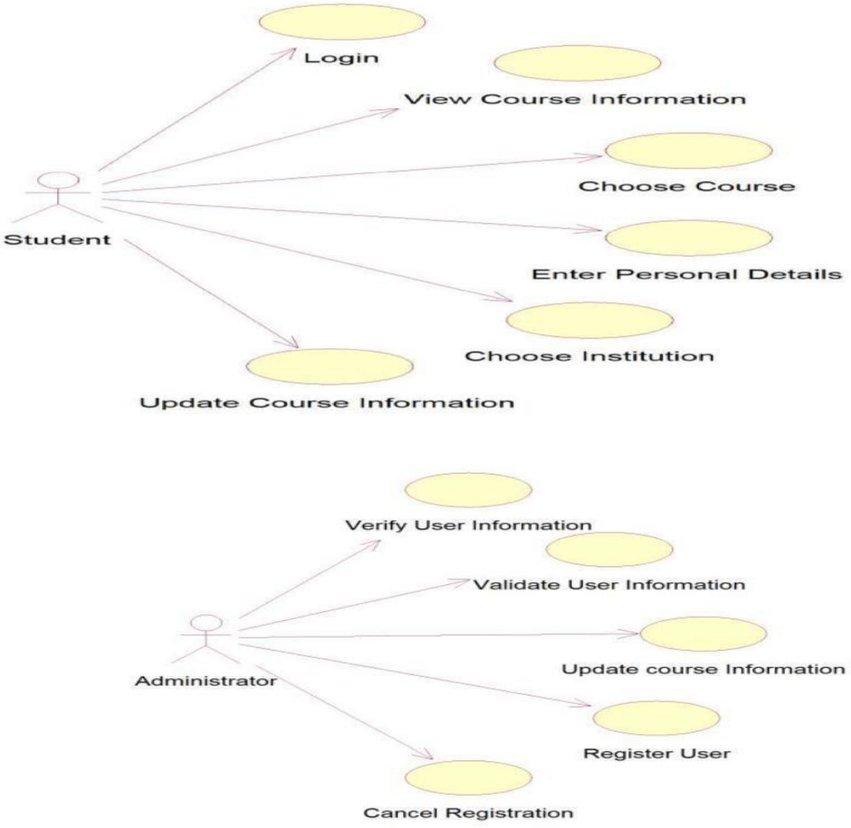
PERFORMANCE

Function for many user at sometimes efficiently (i.e.) without any ever occurrences

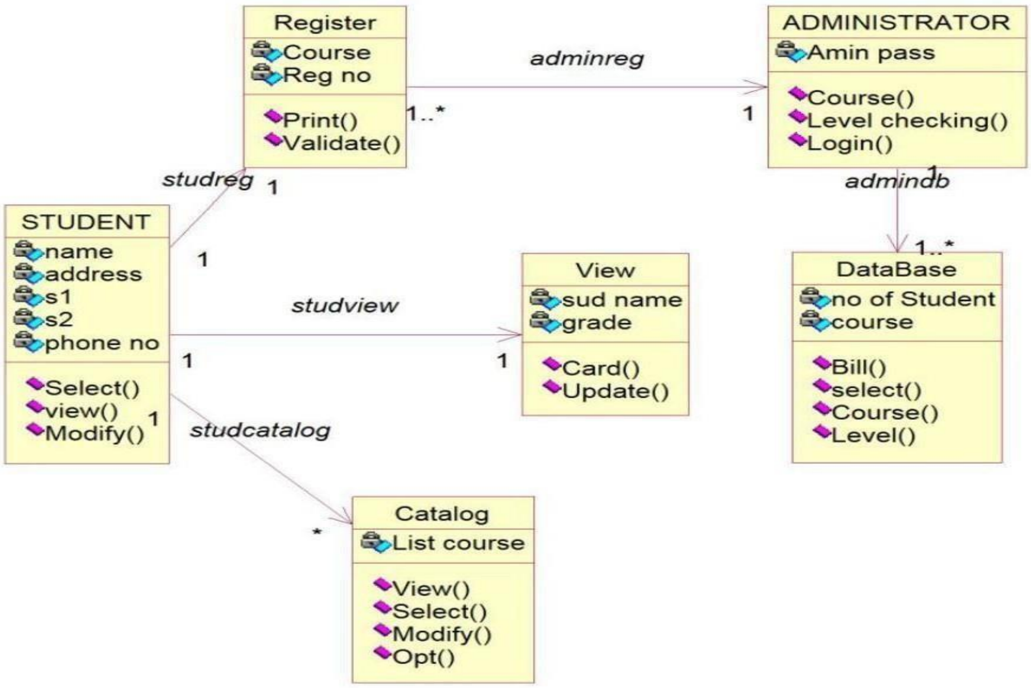
RELIABILITY

The system should be able to the user through the day to day transaction

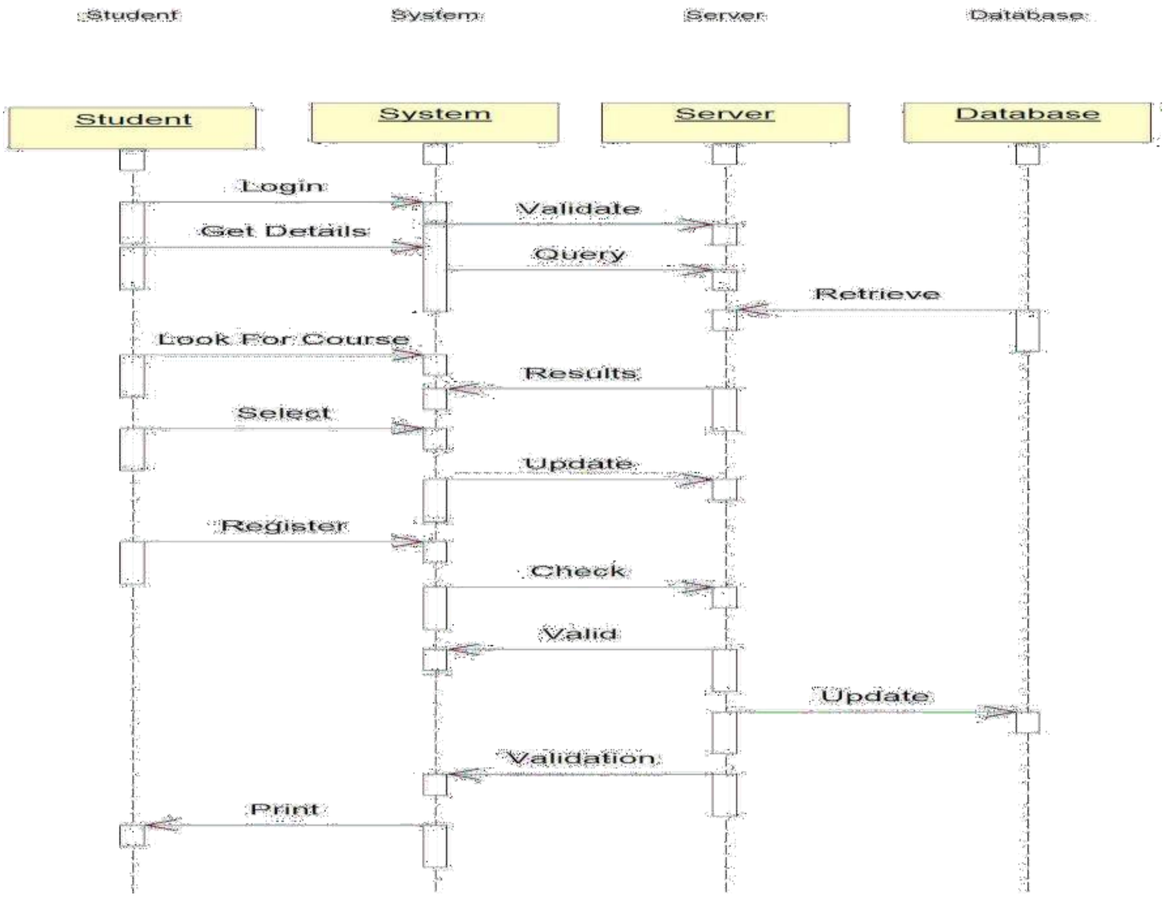
USE CASE DIAGRAM



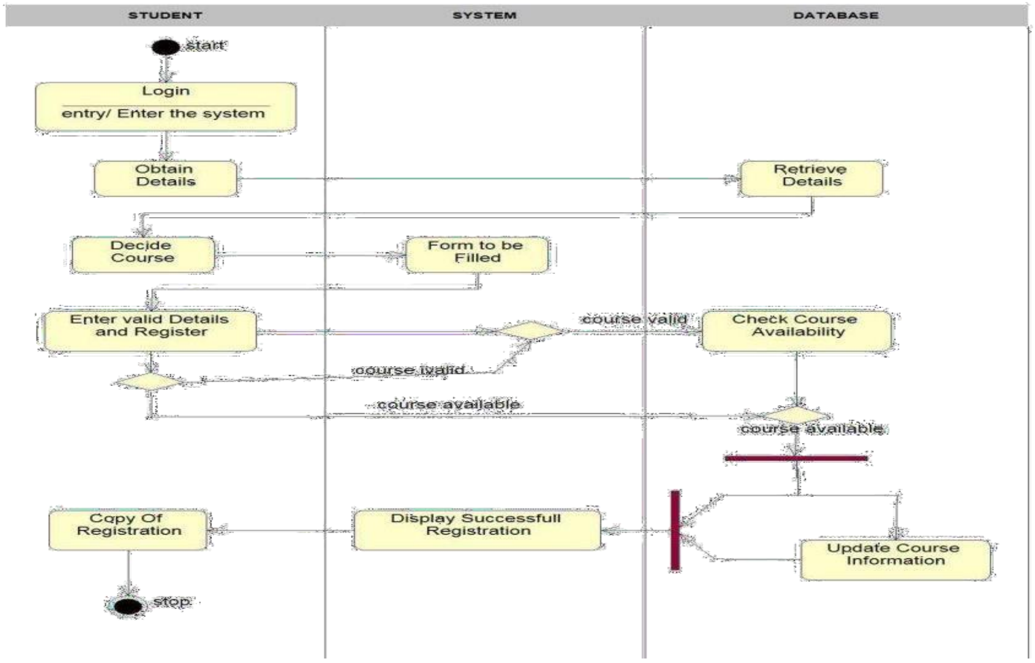
CLASS DIAGRAM



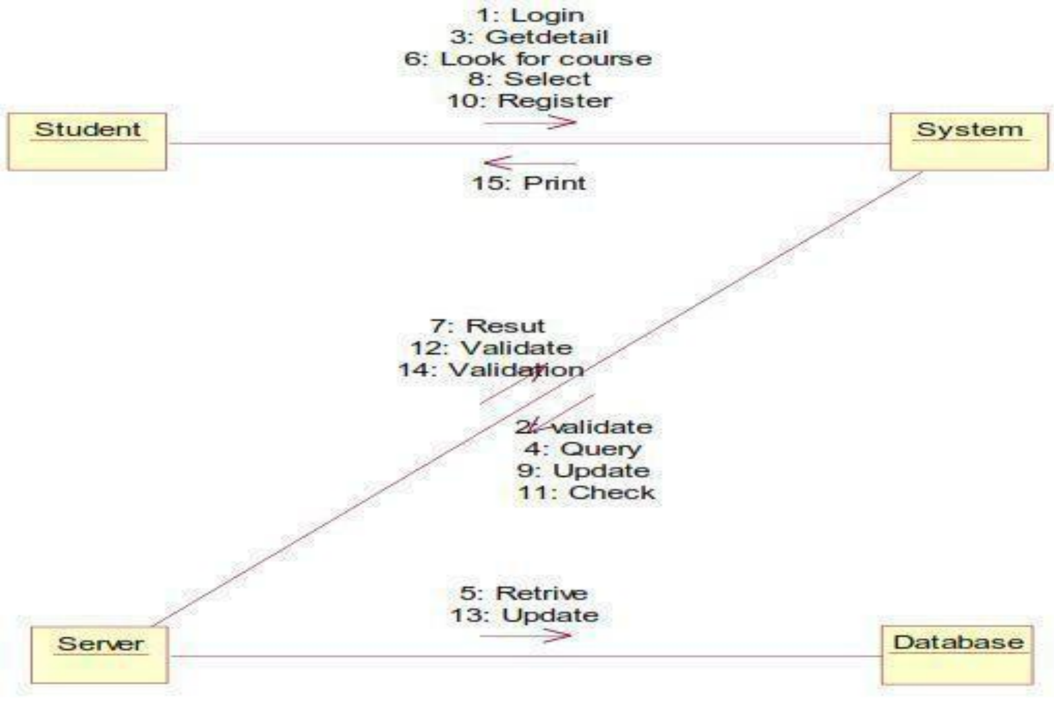
SEQUENCE DIAGRAM



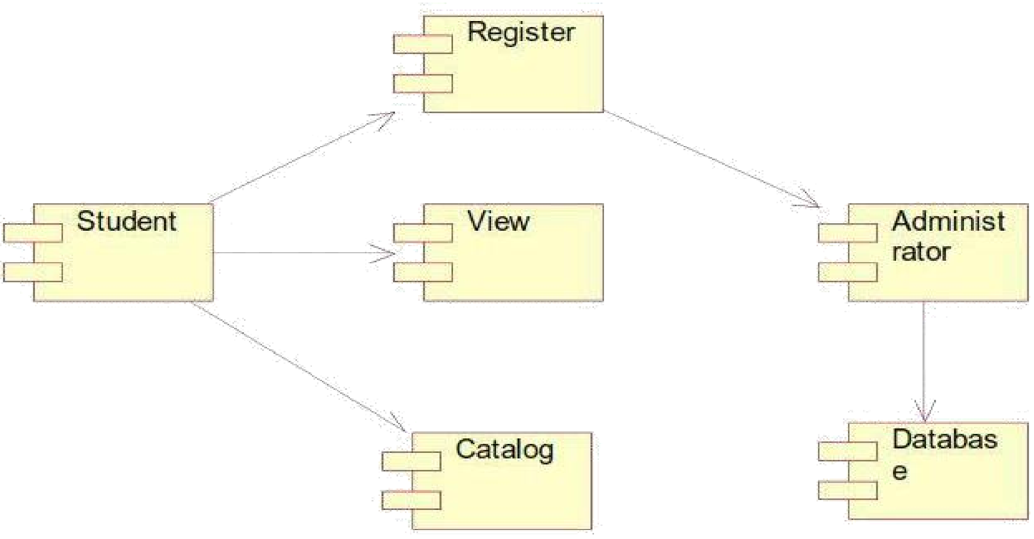
ACTIVITY DIAGRAM



COLLABRATION DIAGRAM

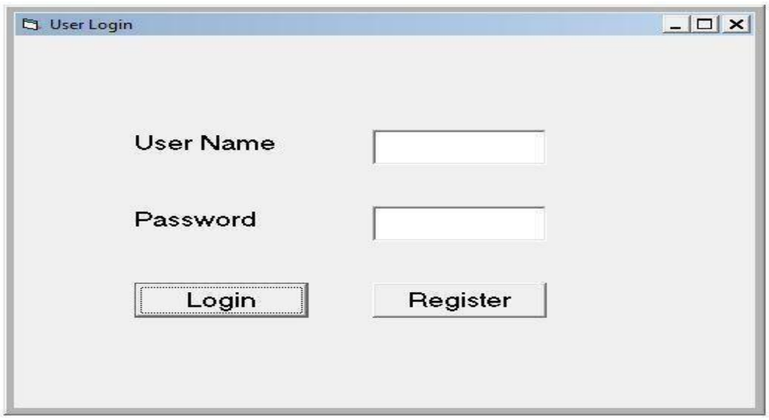


COMPONENT DIAGRAM

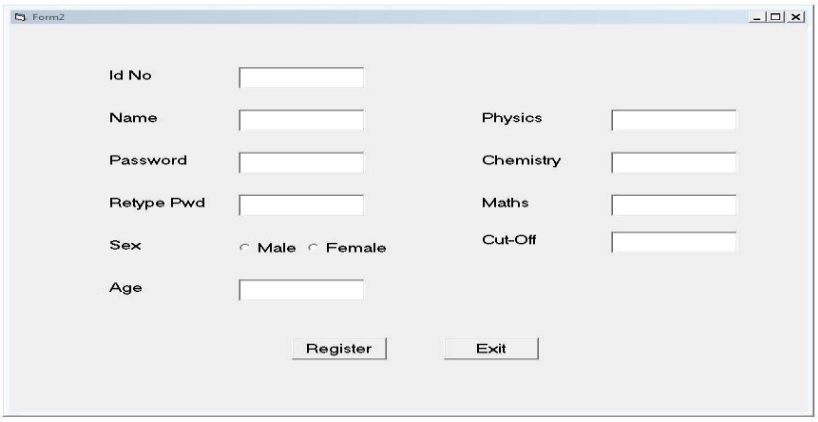


IMPLEMENTATION

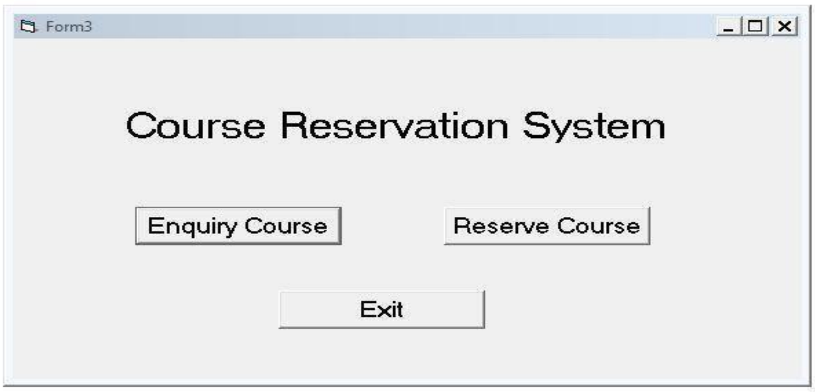
FORM1



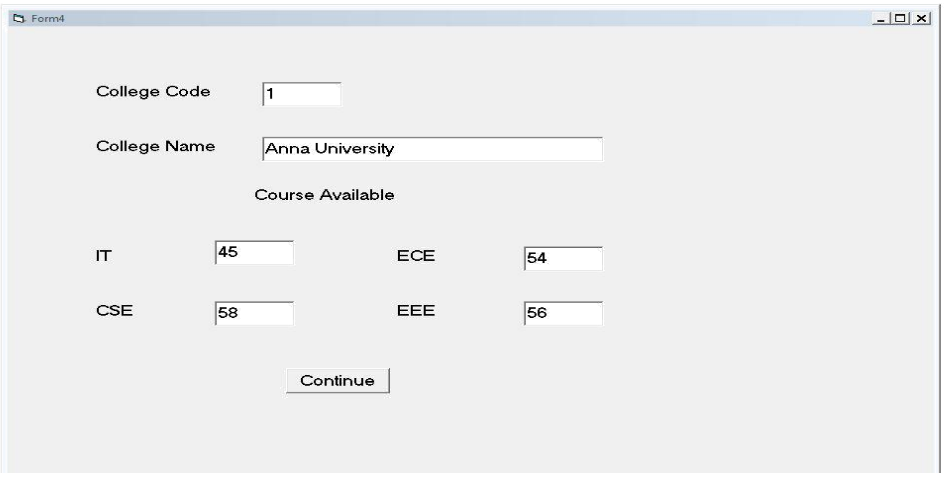
FORM2



FORM3



FORM4



College Code 1

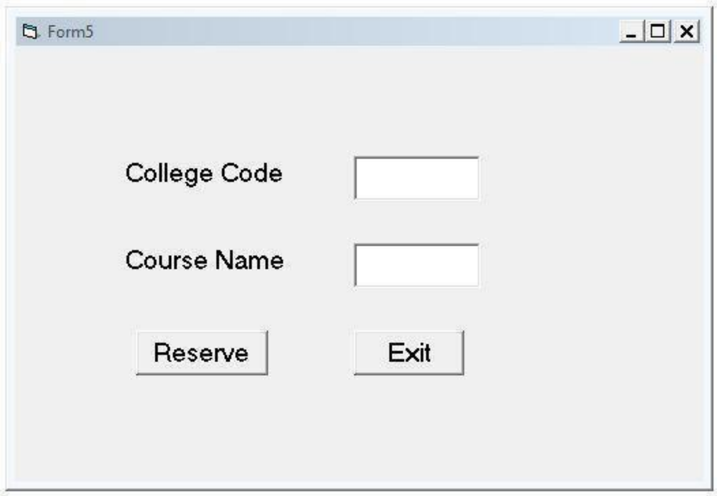
College Name Anna University

Course Available

IT 45 ECE 54

CSE 58 EEE 56

FORM5



CODING

```
Private Sub Command1_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim a As Boolean a =False
cn.Open "dsn=course"
rs.ActiveConnection = cnWithrs
CursorType = adOpenStatic
CursorLocation = adUseClient
LockType = adLockOptimistic
Open "select * from Students" End
With
rs.MoveFirst
While Notrs.EOF
If (Text1.Text = rs(1) And Text2.Text = rs(2)) Then a = True
Form3.Show
Form1.Hide  End  If
rs.MoveNext  Wend
If (a = False) Then
MsgBox ("Enter Correct UserName and Password") End If
End Sub

Private Sub Command2_Click()
Form2.Show
Unload Me
End Sub

Private Sub Command1_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
cn.Open "dsn=Course"
rs.ActiveConnection =
cn
If (Text3.Text = Text4.Text) Then
With rs
CursorType = adOpenStatic
CursorLocation = adUseClient
LockType = adLockOptimistic
Open "select * from
Students" End With
With rs
AddNew
Fields(0) = Val(Text1.Text)
Fields(1) = Text2.Text
Fields(2) = Text3.Text
If (Option1 = True) Then
Fields(3) = Option1.Caption
End If
If (Option2 = True) Then
Fields(3) = Option2.Caption
```

```
End If
Fields(4) = Val(Text5.Text)
Fields(5) = Text6.Text
Fields(6) = Text7.Text
Fields(7) = Text8.Text
Fields(8) = Text9.Text
Update
MsgBox ("Registration Success. Please Login")
Form1.Show
Unload Me
End With
Else
MsgBox ("Password doesn't match") End If
End SubPrivate Sub Command2_Click() Unload
Me
End Sub

Public Sub calCutoff()
Text9.Text = Val(Text6.Text) / 4 + Val(Text7.Text) / 4 + Val(Text8.Text) /2
End Sub

Private Sub Text6_Change()
calCutoff
End Sub

Private Sub Text7_Change()
calCutoff
End Sub

Private Sub Text8_Change()
calCutoff
End Sub

Private Sub Command1_Click()
Form4.Show
Unload Me
End Sub

Private Sub Command2_Click()
Form5.Show
Unload Me
End Sub
Private Sub Command1_Click()
Form3.Show
Unload Me
End Sub

Private Sub Text1_Change()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
```

```
cn.Open "dsn=course"
rs.ActiveConnection = cn
With rs
CursorType = adOpenStatic
CursorLocation = adUseClient
LockType = adLockOptimistic
Open "select * from Colleges" End
With
rs.MoveFirst
While Notrs.EOF
If (Val(Text1.Text) = rs(0)) Then
Text2.Text = rs(1)
Text3.Text = rs(2)
Text4.Text = rs(3)
Text5.Text = rs(4)
Text6.Text = rs(5) End If
rs.MoveNext Wend
End Sub

Private Sub Command1_Click()
Dim cn As New ADODB.Connection
Dcn.Open "dsn=Course"
rs.ActiveConnection = cn
With rs
CursorType = adOpenStat
CursorLocation = adUseClient
LockType = adLockOptimistic
Open "select * from Reservations" End
With
With rs
AddNew
Fields(0) = Form1.Text1.Text
Fields(1) = Text1.Text
Fields(2) = Text2.Text
Update
MsgBox ("Resrvation Success") End
With
End Sub
Private Sub Command2_Click() Unload
Me
End Sub
```

CONCLUSION

Thus the project Online Course Reservation System was designed and codes are generated and then it was executed successfully.

Ex.No:6
Date :

E-TICKETING

AIM

To develop the E-Ticketing System using Agro UML Software and to implement the software in visual basic.

PROBLEM ANALYSIS AND PROJECT PLANNING

In the E-Ticketing system the main process is a applicant have to login the database then the database verifies that particular username and password then the user must fill the details about their personal details then selecting the flight and the database books the ticket then send it to the applicant then searching the flight or else cancelling the process.

PROBLEM STATEMENT

The E-Ticketing system is the initial requirement to develop the project about the mechanism of the E-ticketing system what the process do at all.

a. The requirement are analysed and refined which enables the end users to efficiently use the E-ticketing system.
b. The complete project is developed after the whole project analysis explaining about scope and project statement is prepared.
c. The main scope for this project is the applicant should reserved for the flight ticket.
d. First the applicant wants to login to the database after that the person wants to fill their details.
e. Then the database will search for ticket or else the person will cancelled the ticket if he/she no need.

SOFTWARE REQUIREMENTS SPECIFICATION

S.No.	SOFTWARE REQUIREMENTS SPECIFICATION
	INTRODUCTION
1.1	PURPOSE
1.2	SCOPE
1.3	REFERENCE
1.4	TECHNOLOGY TO BE USED
1.5	TOOLS TO BE USED
1.6	OVERVIEW
	OVERALL DESCRIPTION
2.1	FUNCTIONALITY
2.2	USABILITY
2.3	PERFORMANCE
2.4	RELIABILITY

1.1 PURPOSE

The applicant should login to the database for reserving the ticket.

1.2 SCOPE

In the specification use define about the system requirements that are part from the functionality of the system. It tells the usability, reliability defined in the use case specification.

REFERENCES IEEE Software Requirement Specification format.

TECHNOLOGY TO BE USED Microsoft Visual Basic 6.0

TOOLS BE USED Agro UML (for developing UML Patterns)

OVERVIEW

SRS includes two sections overall description and specific requirements - Overall description will describe major role of the system components and inter-connections. Specific requirements will describe roles & functions of the actors.

2. OVERALL DESCRIPTION

FUNCTIONALITY

The database should be act as an main role of the e-ticketing system it can be booking the ticket in easy way.

USABILITY

The User interface makes the Credit Card Processing System to be efficient.

PERFORMANCE

It is of the capacities about which it can perform function for many users
at the same times efficiently that are without any error occurrence.

RELIABILITY

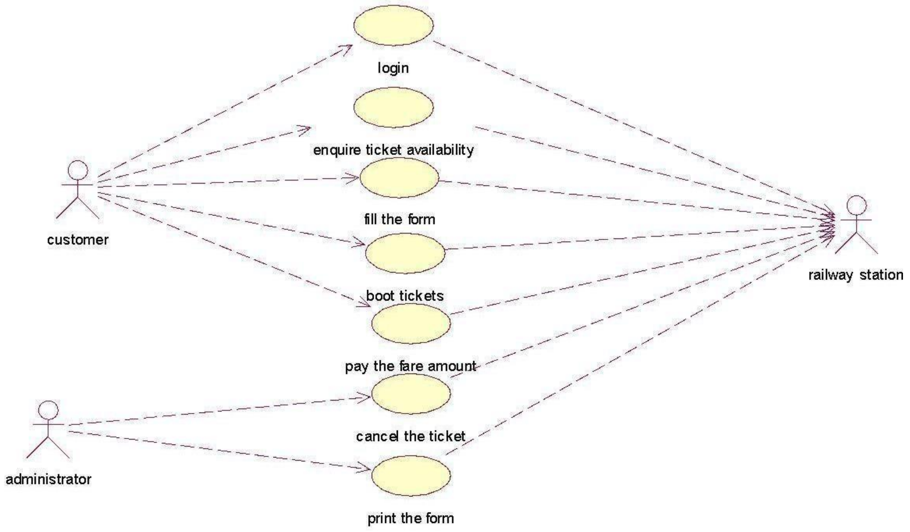
The system should be able to process the user for their corresponding request.

UML DIAGRAMS

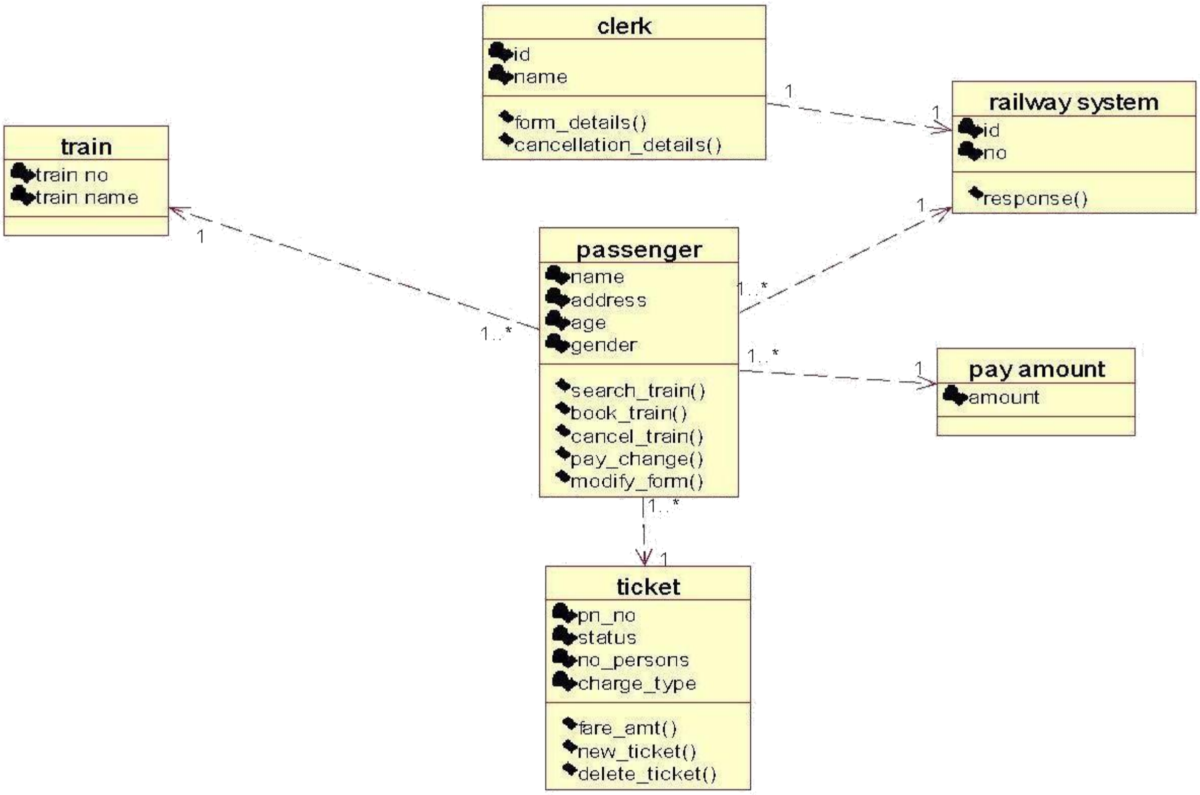
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

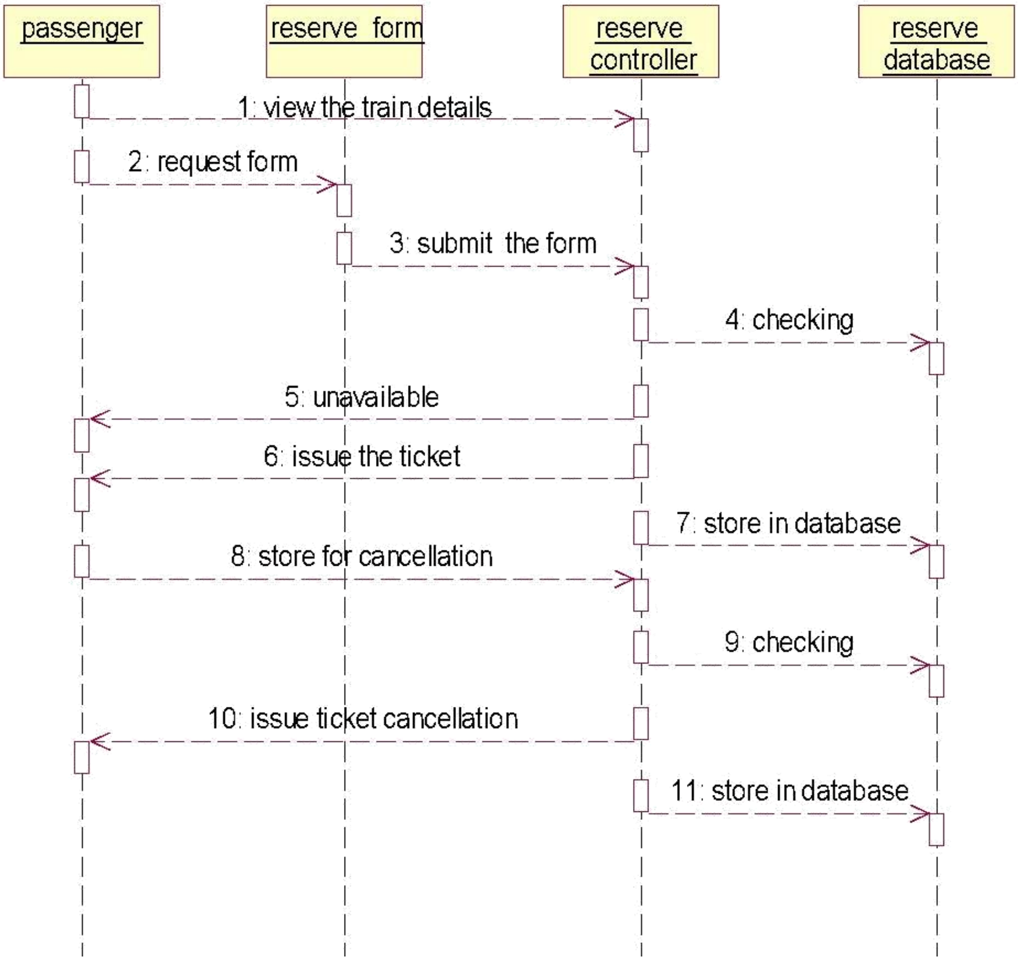
USECASE DIAGRAM:



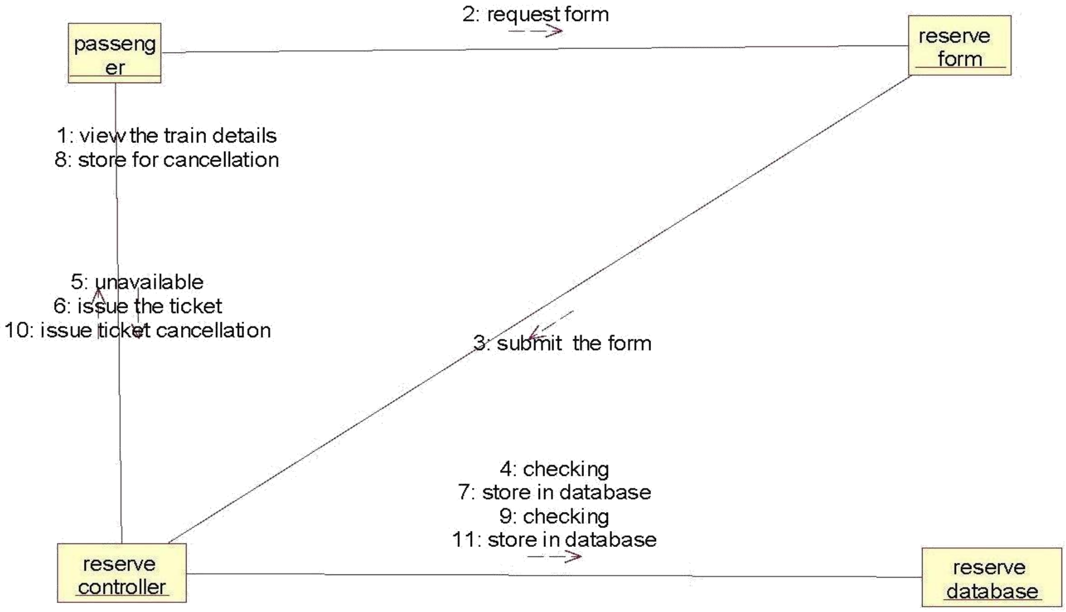
CLASS DAIGRAM



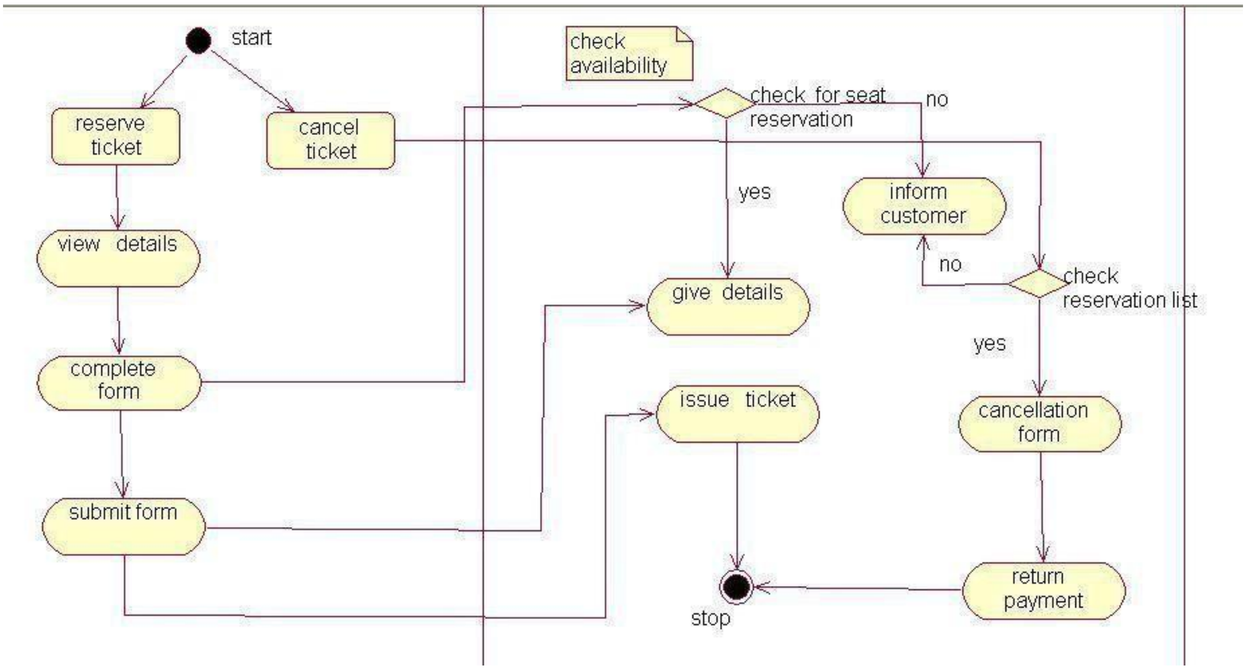
SEQUENCE DIAGRAM



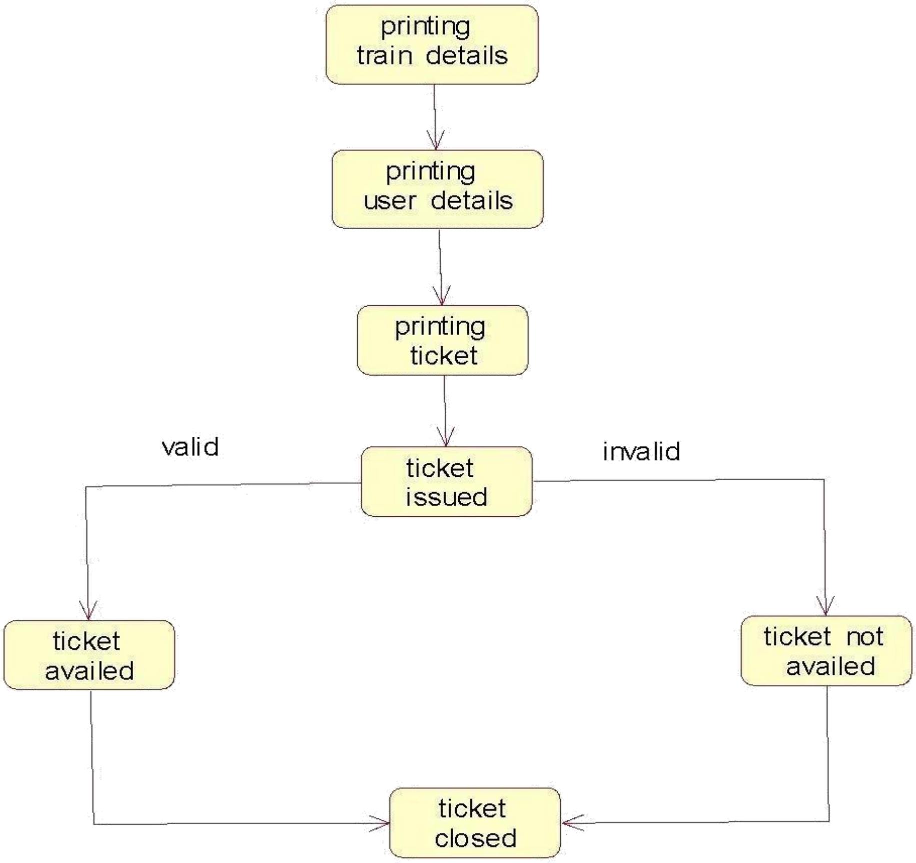
COLLABRATION DIAGRAM



ACTIVITY DIAGRAM



STATE CHART DIAGRAM



COMPONENT DIAGRAM

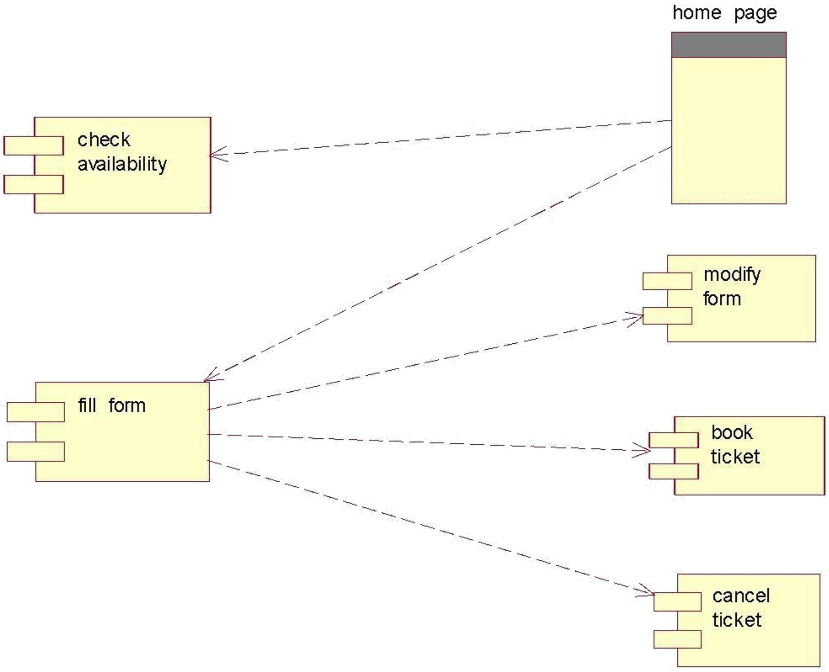
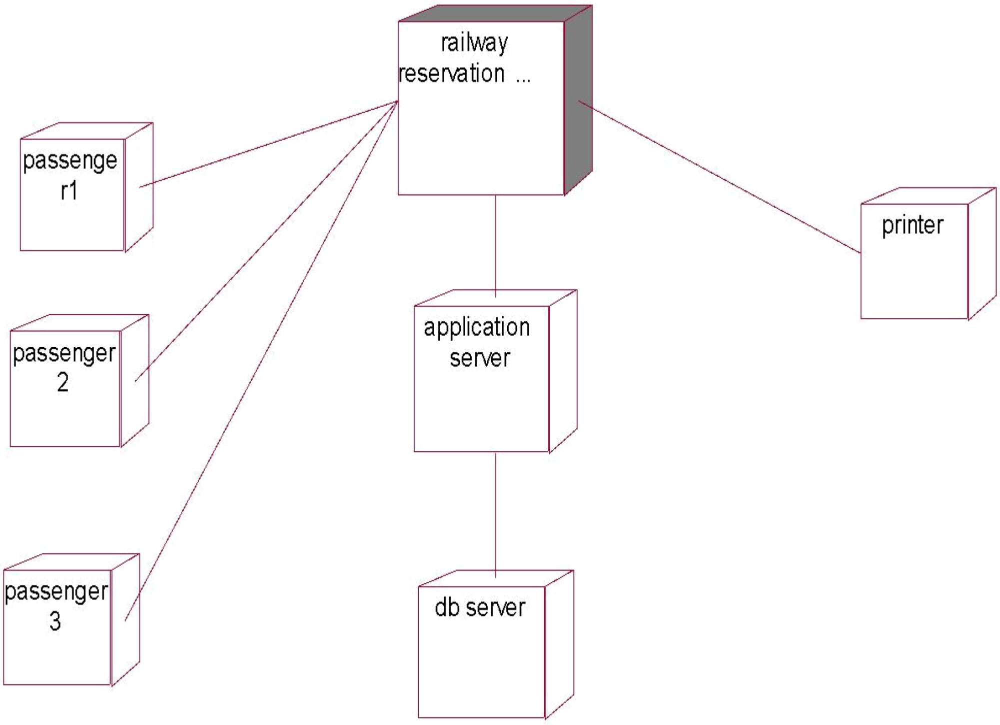
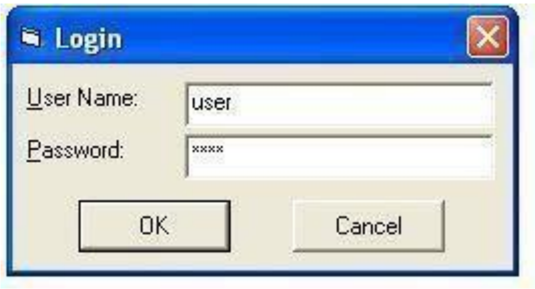


DIAGRAM DEPLOYMENT



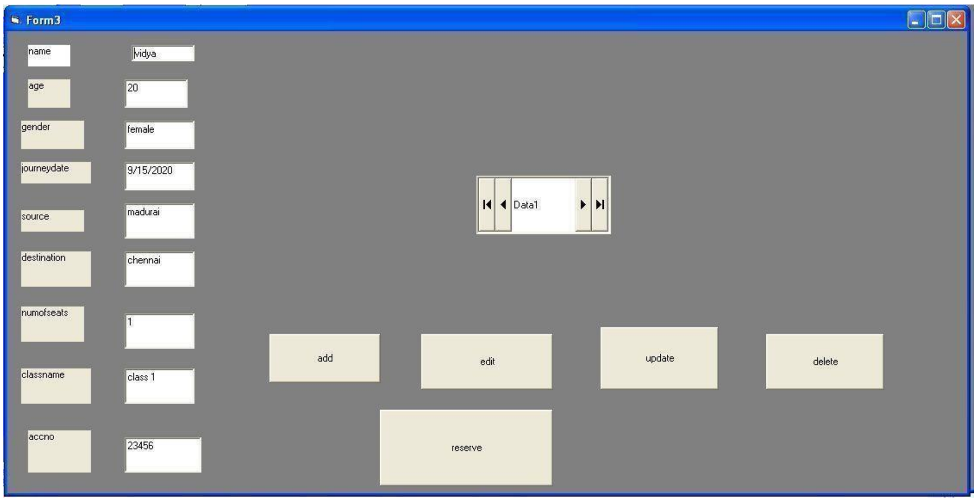
FORM1



User Name: user

Password: ****

FORM2



name vidya

age 20

gender female

journeydate 9/15/2020

source madurai

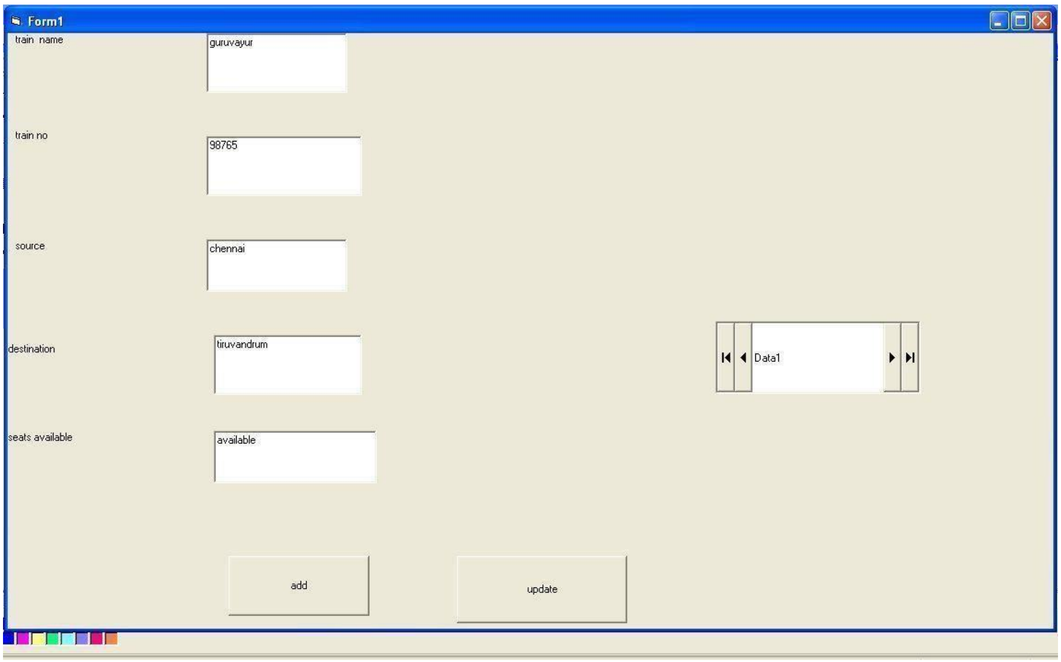
destination chennai

numofseats 1

classname class 1

accno 23456

FORM3



train name guruvayur

train no 98765

source chennai

destination tiruvandrum

seats available available

PROGRAM:
FORM1

```
Option Explicit
Public LoginSucceededAs Boolean
Private Sub cmdCancel_Click()
'set the global var to false
'to denote a failed login
LoginSucceeded = False
Me.Hide
End Sub

Private Sub cmdOK_Click()
'check for correct password
If txtPassword = "user" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form3.Show
Else
If txtPassword = "admin" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form1.Show
Else
MsgBox "Invalid Password, try again!", , "Login"
txtPassword.SetFocus
SendKeys "{Home}+{End}"
End If
End If
End Sub
```

FORM2

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
MsgBox "records are addded successfully"
End Sub

Private Sub Command2_Click()
Data1.Recordset.Edit
MsgBox "records are edited successfully"
End Sub

Private Sub Command3_Click()
Data1.Recordset.Update
MsgBox "records are updated successfully"
```

```
End Sub
Private Sub Command4_Click()
Data1.Recordset.Delete
MsgBox "records are deleted successfully"
End Sub

Private Sub Command5_Click()
Form2.Show
End Sub

Private Sub Command6_Click()
Data1.Recordset.exit
End Sub
```

FORM3

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub

Private Sub Command2_Click()
Data1.Recordset.Update
End Sub

Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub

Private Sub Command2_Click()
Data1.Recordset.Update
End Sub

Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub

Private Sub Command2_Click()
Data1.Recordset.Update
End Sub
```

CONCLUSION

Thus the project for E-Ticketing System was designed and codes are generated and then it was executed successfully.

Ex.No: 7
Date :

SOFTWARE PERSONNAL MANAGEMENT SYSTEM

AIM:

To develop a project employee management system using the Agro UML Software from the UML diagram and to implement the software in Visual Basic.

PROJECT ANALYSIS AND PROJECT PLANNING:

The employee management system is used to manage our personnel things such as maintaining databases in offices etc. this project is easy for the CEO to handle the details. This is personally used for CEO.

PROBLEM STATEMENT:

The CEO must enter the name and password to login the form and select the particular employee to view the details about that employee and maintaining the employee details personally. This process of employee management system are described sequentially through following steps,

- The CEO login to the employee management system.
- He/she search for the list of employees.
- Then select the particular employee.
- Then view the details of that employee.
- After displaying the employee details then logout.

SOFTWARE REQUIREMENTS SPECIFICATION

S.No.	SOFTWARE REQUIREMENTS SPECIFICATION
1.0	**INTRODUCTION**
1.1	PURPOSE
1.2	SCOPE
1.3	DEFINITION, ACRONYMS AND ABBREVIATIONS
1.4	REFERENCE
1.5	TECHNOLOGY TO BE USED
1.6	TOOLS TO BE USED
1.7	OVERVIEW
2.0	**OVERALL DESCRIPTION**
2.1	PRODUCTIVE DESCRIPTION
2.2	SOFTWARE INTERFACE
2.3	HARDWARE INTERFACE
2.4	SYSTEM FUNCTION
2.5	USER CHARACTERISTIC
2.6	CONSTRAINTS
2.7	ASSUMPTION AND DEPENDENCES

INTRODUCTION:

Purpose:

The main purpose of creating the document about the software is to know about the list of requirements that is to be developed.

Scope:

It specifies the requirements to develop a processing software part that complete the set of requirements. In this specification, we define about the system requirements that are apart from the functionality of system

References:

IEEE Software Requirements Specification format

Technology to Be Used:

Microsoft Visual Basic 6.0

Tools Be Used:

AGRO UML tool (for developing UML Patterns)

Overview:

SRS includes two sections overall description and specific requirements - Overall description will describe major role of the system components and inter-connections. Specific requirements will describe roles & functions of the actors.

OVERALL DESCRIPTION

Product Perspective:

The SPMP acts as an interface between the user and the database. This tries to handle the personnel databases easily.

Functionality:

Many members of the process live to check for the occurrence and transaction, we all have to carry over at sometime.

Usability:

The User interface makes the employee Management System to be efficient.

Performance:

It is the capability about which it can perform function for many users at the same time for the efficiency (i.e.) without any error occurrences.

Reliability:

The system should be able to the user through the day to day transactions.

Assumptions and dependencies:

The user must have the basic knowledge of computer and English language. The user must correctly login the database

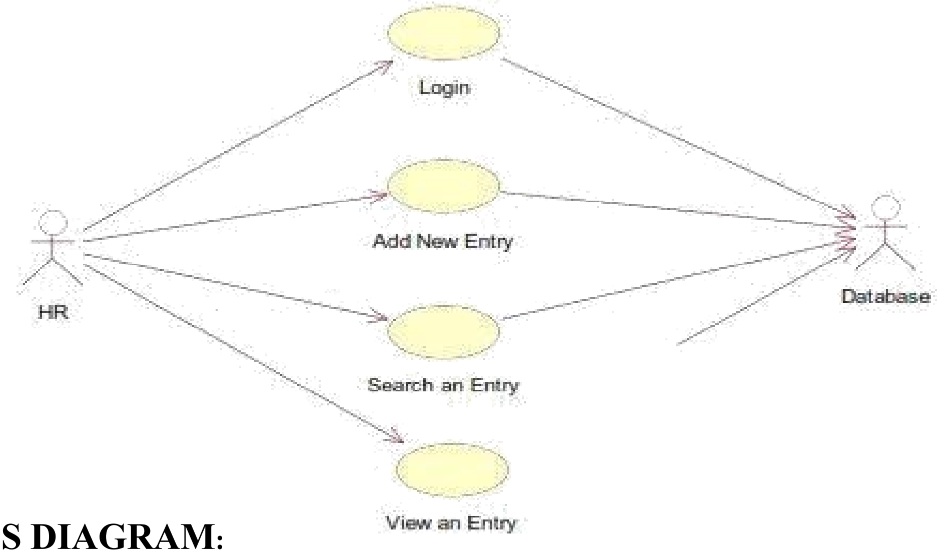
UML DIAGRAMS

The following UML diagrams describe the process involved in the online recruitment system

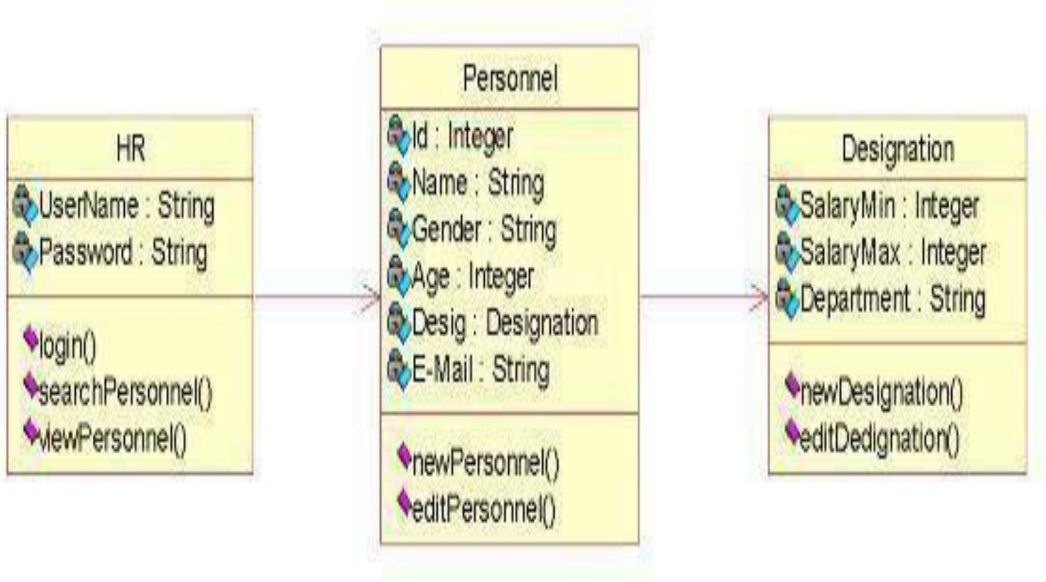
a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

The project can be explained diagrammatically using the following diagrams

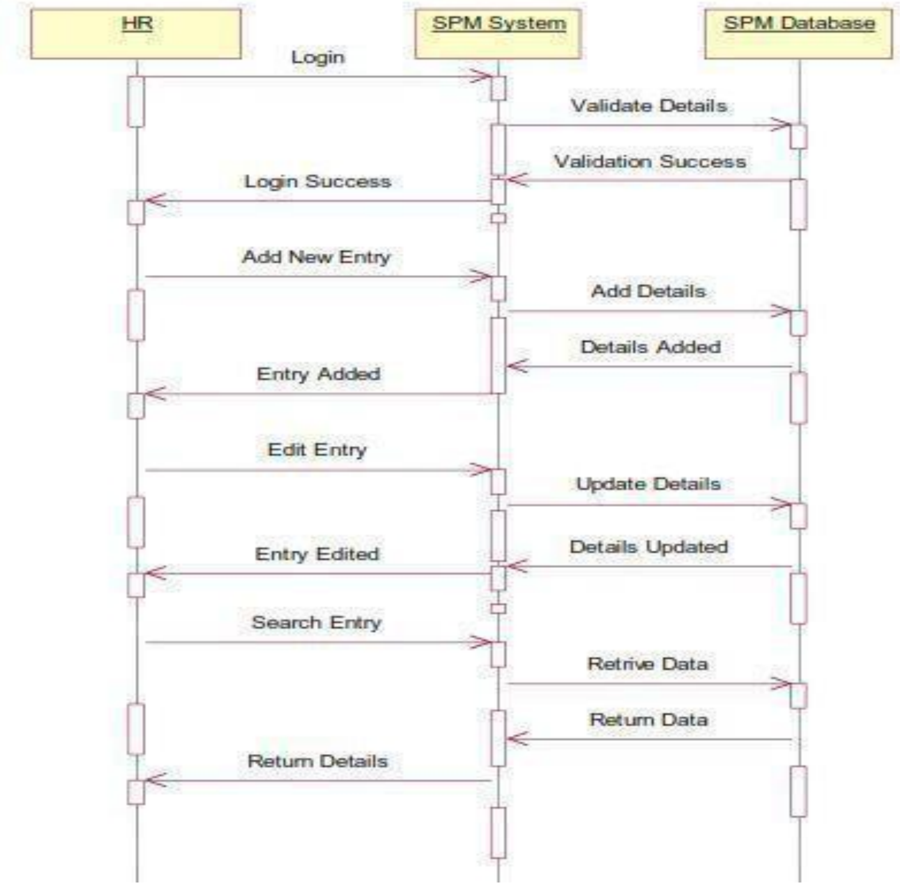
USE CASE DIAGRAM:



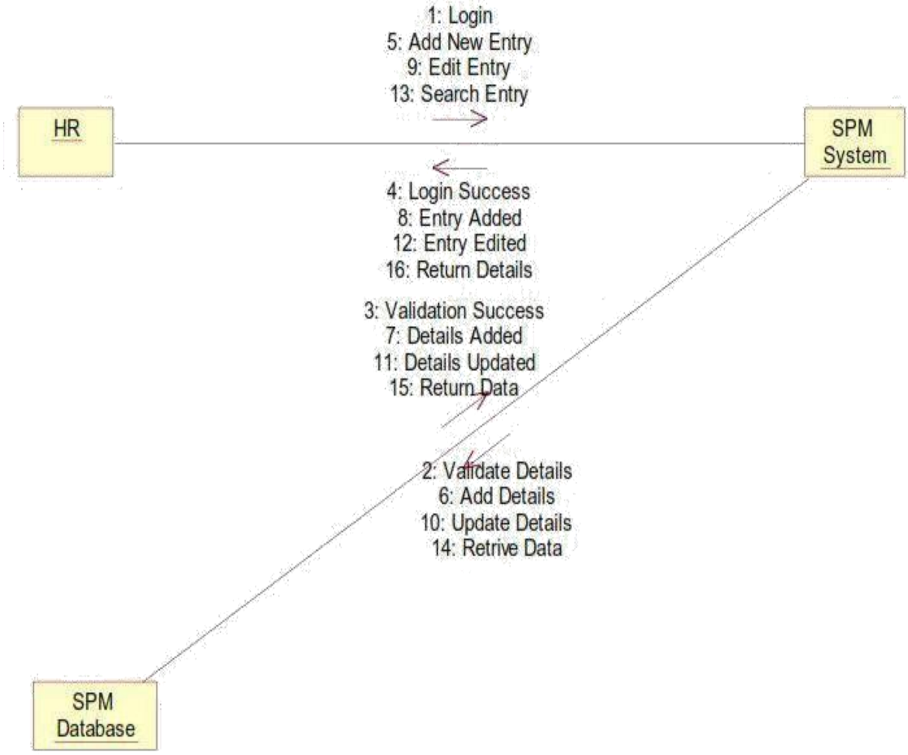
CLASS DIAGRAM:



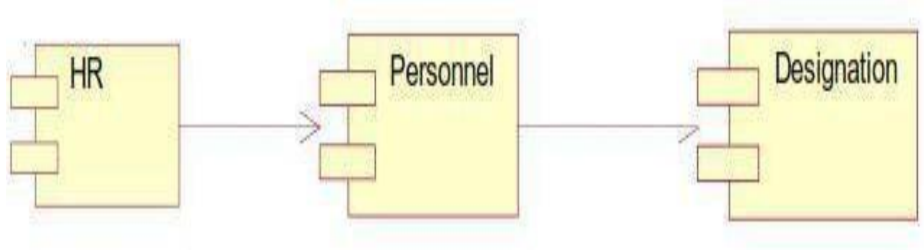
SEQUENCE DIAGRAM:



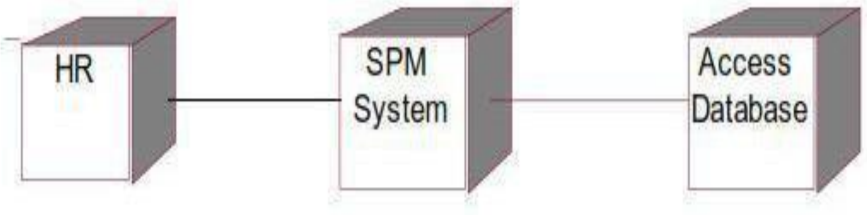
COLLABRATION DIAGRAM:



COMPONENT DIAGRAM:

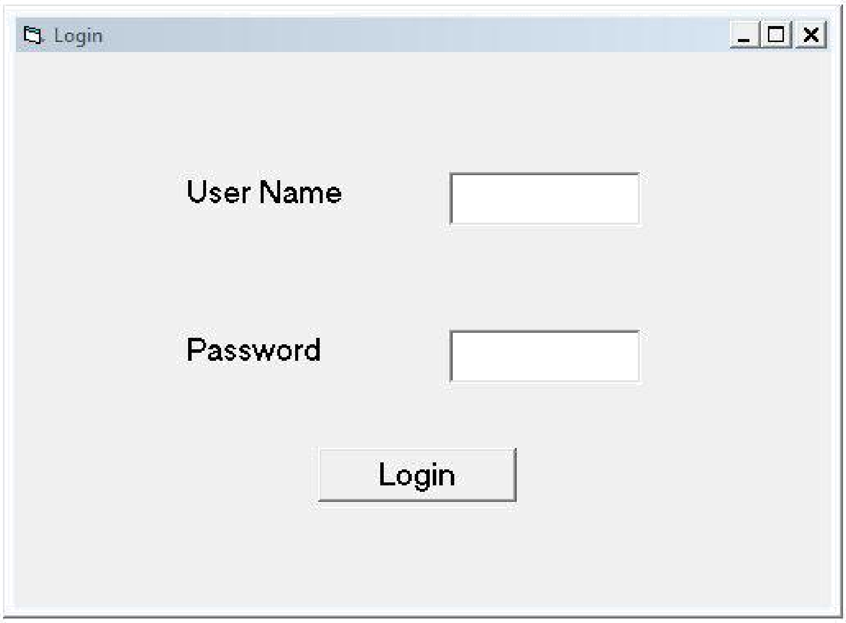


DEPLOYMENT DIAGRAM:

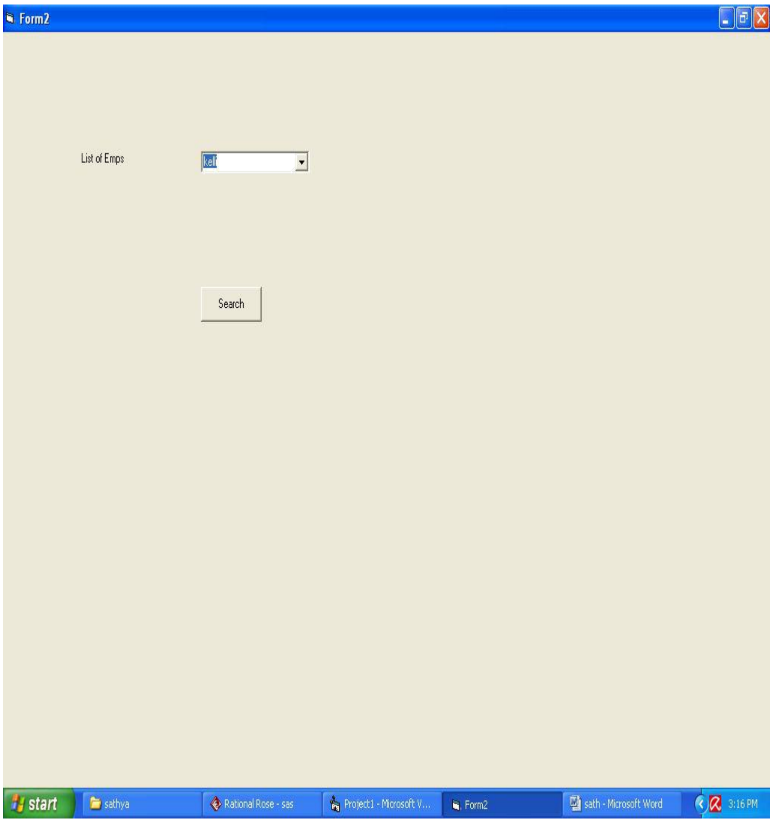


IMPLEMENTATION

FORM1:

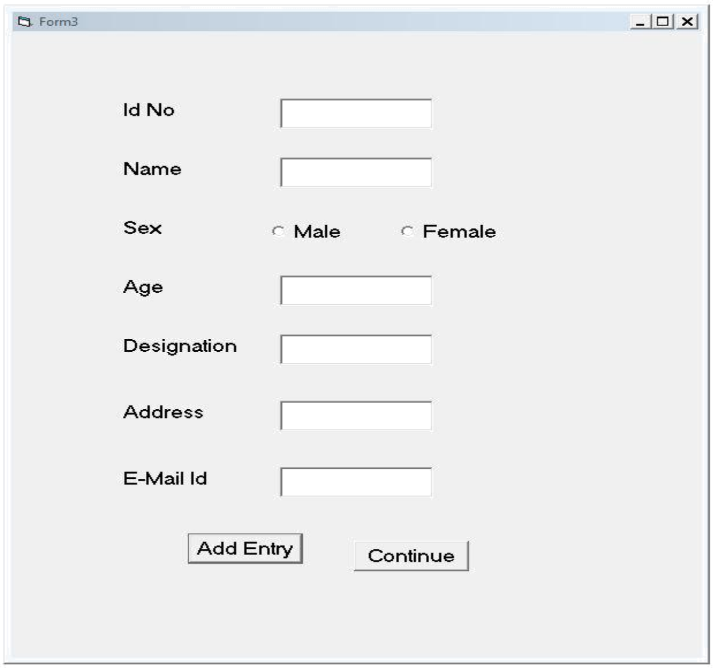


FORM2:



List of Emps kelli

FORM3:



Id No

Name

Sex ○ Male ○ Female

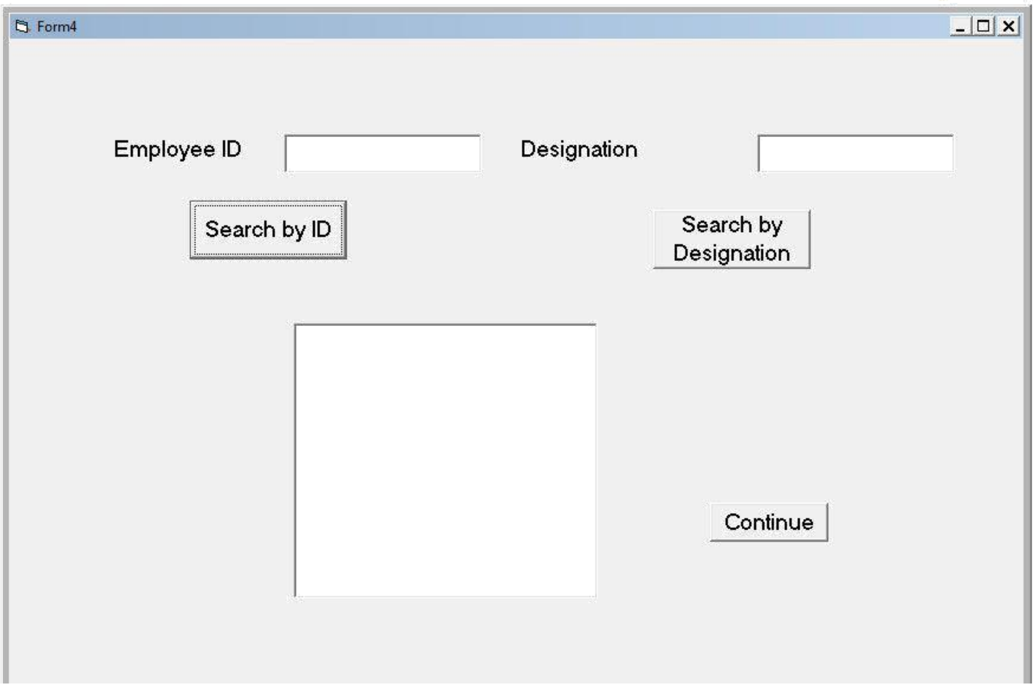
Age

Designation

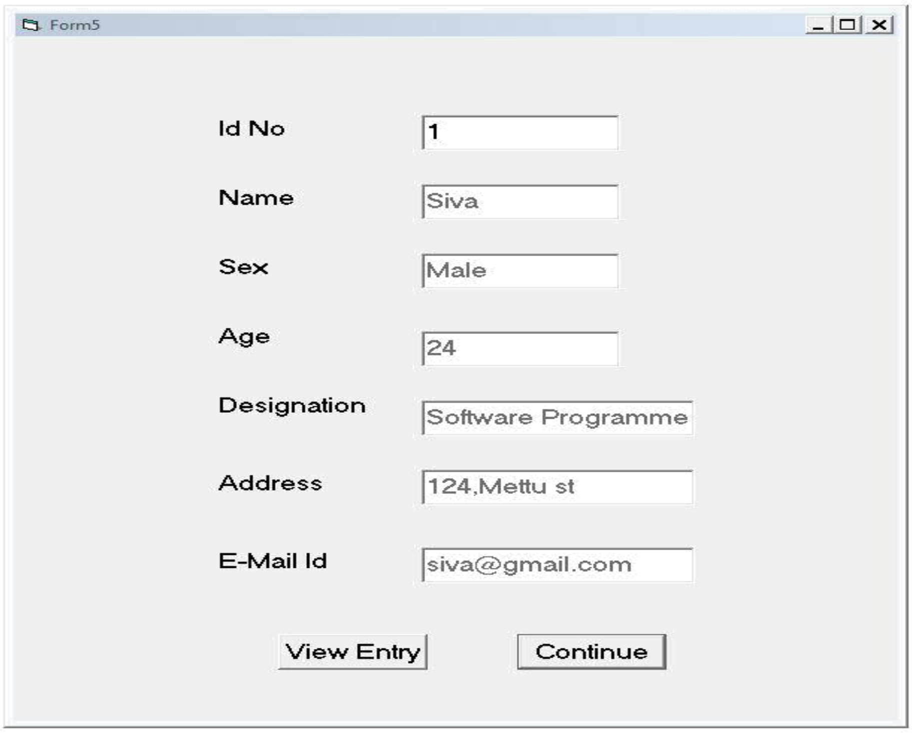
Address

E-Mail Id

FORM4:



FORM5:



Id No	1
Name	Siva
Sex	Male
Age	24
Designation	Software Programme
Address	124,Mettu st
E-Mail Id	siva@gmail.com

CODING

FORM1:

```
Private Sub Command1_Click() Dim a As Boolean
a = False
If (Text1.Text = "admin" And Text2.Text = "admin") Then a = True
Form2.Show
Unload Me
End If
If (a = False) Then
MsgBox ("Enter Correct Username and
Password") End If
End Sub
```

FORM2:

```
Private Sub Command1_Click()
Form3.Show
Unload Me
End Sub
Private Sub Command2_Click() Form4.Show
Unload Me
End Sub
Private Sub Command3_Click() Form5.Show
Unload Me
End Sub
Private Sub Command4_Click() Unload Me
End Sub
```

FORM3:

```
Private Sub Command1_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
cn.Open "dsn=Software" rs.ActiveConnection = cn
With rs
.CursorType = adOpenStatic
.CursorLocation = adUseClient
.LockType = adLockOptimistic
.Open "select * from Details" End With
With rs
.AddNew
.Fields(0) = Val(Text1.Text)
.Fields(1) = Text2.Text
If (Option1 = True) Then
.Fields(2) = Option1.Caption
End If
If (Option2 = True) Then
.Fields(2) = Option2.Caption
End If
.Fields(3) = Val(Text3.Text)
.Fields(4) = Text4.Text
```

```
.Fields(5) = Text5.Text
.Fields(6) = Text6.Text.
Update End With Text1.Text = "" Text2.Text = "" Text3.Text = "" Text4.Text = ""
Text5.Text = "" Text6.Text = "" Option1 = False Option2 = False
End Sub
```

FORM4:

```
Private Sub Command1_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim a As Boolean a = False
cn.Open "dsn=software" rs.ActiveConnection = cn With rs
CursorType = adOpenStatic
CursorLocation = adUseClient
LockType = adLockOptimistic
Open "select * from Details" End With
rs.MoveFirst While Notrs.EOF
If (Val(Text1.Text) = rs(0)) Then Text3.Text = Text3.Text + rs(1) + ", " Text3.Text =
Text3.Text
+ rs(2) + ", " Text3.Text = Text3.Text + Str$(rs(3)) + ", " Text3.Text =
Text3.Text + rs(4) + ", " Text3.Text = Text3.Text + rs(5) + ", " Text3.Text =
Text3.Text + rs(6) + ", "a = True End If rs.MoveNext Wend
If (a = False) Then
MsgBox ("Enter correct Employee ID") End If
Text1.Text = ""
End Sub
Private Sub Command2_Click()
Dim cn As New ADODB.Connection Dim rs As New ADODB.Recordset
Dim a As                          Boolean a = False
cn.Open "dsn=software" rs.ActiveConnection = cn
With rs
CursorType = adOpenStatic
CursorLocation = adUseClient
LockType = adLockOptimistic
Open "select * from Details" End With
rs.MoveFirst
While Notrs.EOF
If (Text2.Text = rs(4)) Then
Text3.Text = Text3.Text + Str$(rs(0)) + ", " Text3.Text = Text3.Text + rs(1) + ", "
Text3.Text = Text3.Text + rs(2) + ", "
Text3.Text = Text3.Text + Str$(rs(3)) + ", " Text3.Text = Text3.Text + rs(5) + ", "
Text3.Text = Text3.Text + rs(6) + ". "
a =  True
End If
rs.MoveNext
Wend
If (a = False) Then
MsgBox ("Enter correct Designation") End If
Text2.Text = ""
End Sub
```

```
Private Sub Command3_Click() Form2.Show
Unload Me
End Sub
```

FORM5:

```
Private Sub Command1_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim a As Boolean a = False
cn.Open "dsn=software" rs.ActiveConnection = cnWithrs
CursorType = adOpenStatic
CursorLocation = adUseClient
LockType = adLockOptimistic
Open "select * from Details" End With
rs.MoveFirst
While Notrs.EOF
If (Val(Text1.Text) = rs(0)) Then
Text2.Text = rs(1) Text3.Text = rs(2) Text4.Text = rs(3) Text5.Text = rs(4)
Text6.Text = rs(5)
Text7.Text = rs(6) a = True End If rs.MoveNext Wend If (a = False) Then
MsgBox ("Enter correct ID") End If
End Sub
```

CONCLUSION:

Thus the project for Software Personnel management System was designed and codes are generated and then it was executed successfully.

Ex.No : 8

Date :

CREDIT CARD PROCESSING

AIM

To develop a project credit card system using the Agro UML from the UML diagram and to implement the software in Visual Basic

PROBLEM ANALYSIS AND PROJECT PLANNING

The Credit Card Processing System which is use to purchasing an item from any shop mall, and it is used to maintain the limitation of credit card balance and current transaction process could be update via credit card machine. This project mainly used for large amount of item can be easy to buy from anywhere and required transaction process should bemaintained them.

PROBLEM STATEMENT

The customer should select the item to be purchase from the shop by using credit card payment then the vendor should give a bill for the selected item .The customer should givehis card to swap and request for the kind of amount transaction. After processing the transaction, the CREDIT CARD MACHINE should give the balance print statement or receipt.

- Customer should select the item from the shop.
- Vendor makes the bill for the selected item.
- Customer gives the credit card to the vendor to swap the card.
- They required amount transaction is done by the card reader.
- Vendor will issue the balance statement to the customer.
- Customers put the signature in the receipt and return to the vendor.

SOFTWARE REQUIREMENTS SPECIFICATION

S.No.	SOFTWARE REQUIREMENTS SPECIFICATION
1.0	**INTRODUCTION**
1.1	PURPOSE
1.2	SCOPE
1.3	REFERENCE
1.4	TECHNOLOGY TO BE USED
1.5	TOOLS TO BE USED
1.6	OVERVIEW
2.0	**OVERALL DESCRIPTION**
2.1	PRODUCT PERSPECTIVE
2.2	FUNCTIONALITY
2.3	USABILITY
2.4	PERFORMANCE
2.5	RELIABILITY
2.6	ASSUMPTION AND DEPENDENCES

PURPOSE

If the entire process of 'Issue of Credit Card ' is done in a manual manner then it would take several months for the Credit Card to reach the applicant. Considering the fact that the number of applicants for Credit Card is increasing every year, an Automated System becomes essential to meet the demand. So this system uses several programming and database techniques to elucidate the work involved in this process. As this is a matter of National Security, the system has been carefully verified and validated in order to satisfy it.

SCOPE

In the specification use define about the system requirements that are part from the functionality of the system. It tells the usability, reliability defined in the use case specification.

REFERENCES

IEEE Software Requirement Specification format.

TECHNOLOGY TO BE USED

Microsoft Visual Basic 6.0

TOOLS TO BE USED

Agro UML (for developing UML Patterns)

OVERVIEW

SRS includes two sections overall description and specificrequirements - Overall description will describe major role of the system components and inter-connections. Specific requirements will describe roles & functions of the actors.

OVERALL DESCRIPTION

PRODUCT PERSPECTIVE

The CCP acts as an interface between the'Customer' and the 'Card Reader'. This system tries to make the transaction as simple as possible and at the same time not risking the security of data transaction process. This minimizes the time duration in which the user receives the item.

FUNCTIONALITY

Many members of the process lives to checking for the occurrence and transaction we all have to carry over sometimes user interface to make the transaction to be efficient.

USABILITY

The User interface makes the Credit Card Processing System to be efficient.

PERFORMANCE

It is of the capacities about which it can perform function for many users at the same times efficiently that are without any error occurrence.

RELIABILITY

The system should be able to process the user for their corresponding request.

ASSUMPTION AND DEPENDENCIES

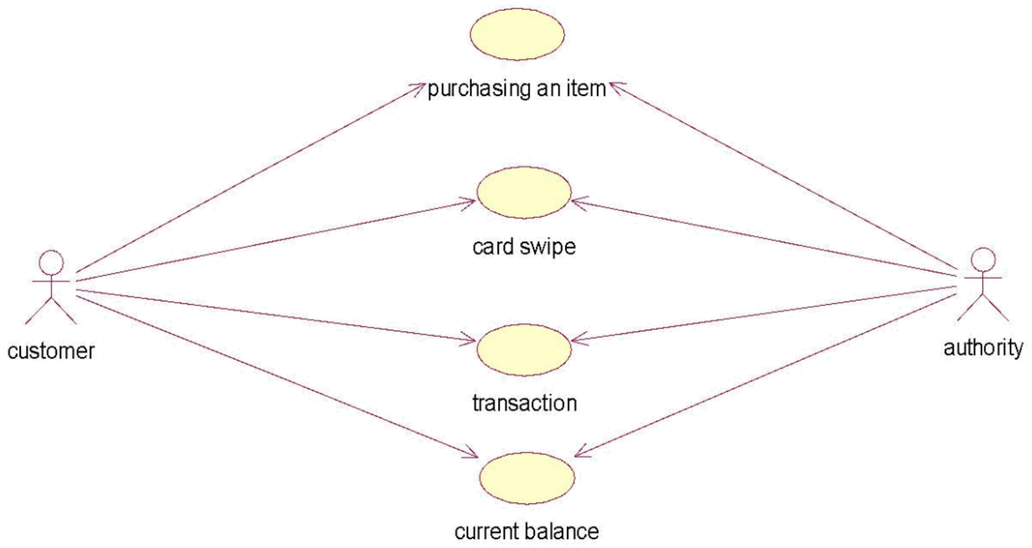
The Vendor and Customer must have basic knowledge of computers and English Language. The vendor may be required to delivered the item purchased by the customer.

UML DIAGRAMS

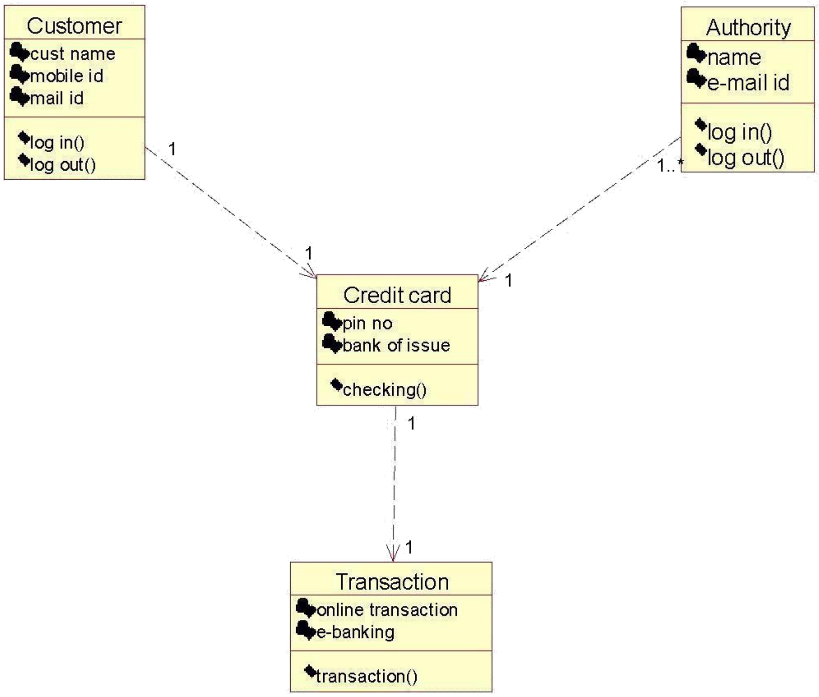
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

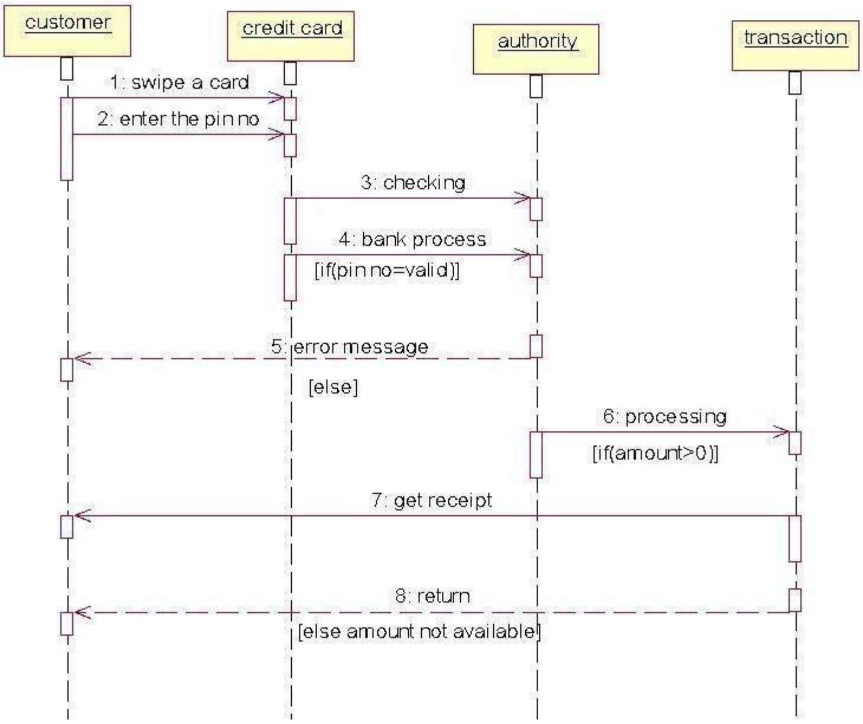
USE CASE DIAGRAM



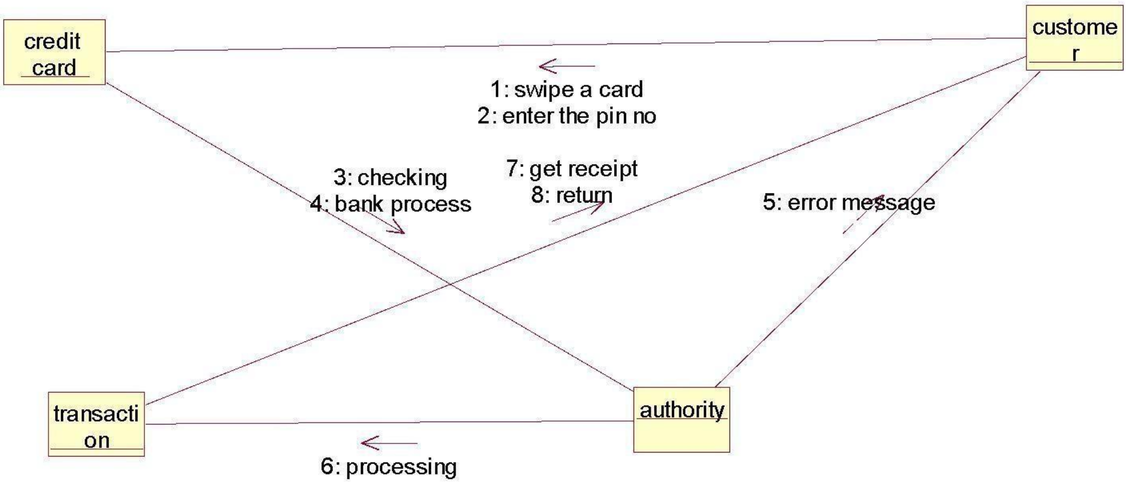
CLASS DIAGRAM



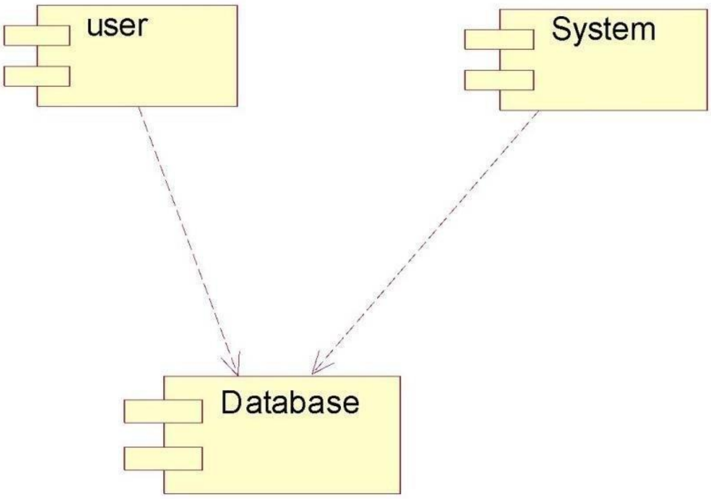
SEQUENCE DIAGRAM



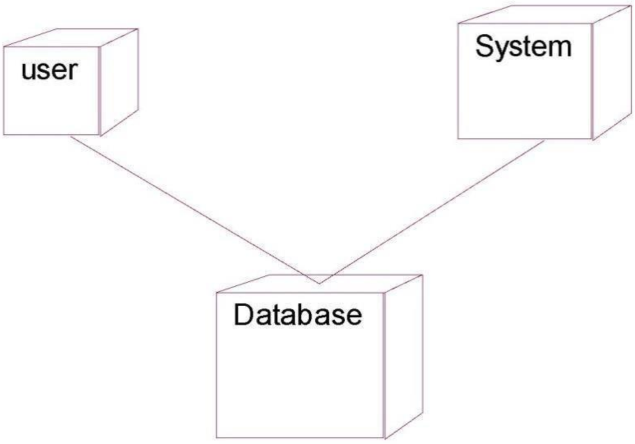
COLLABORATION DIAGRAM



COMPONENT DIAGRAM

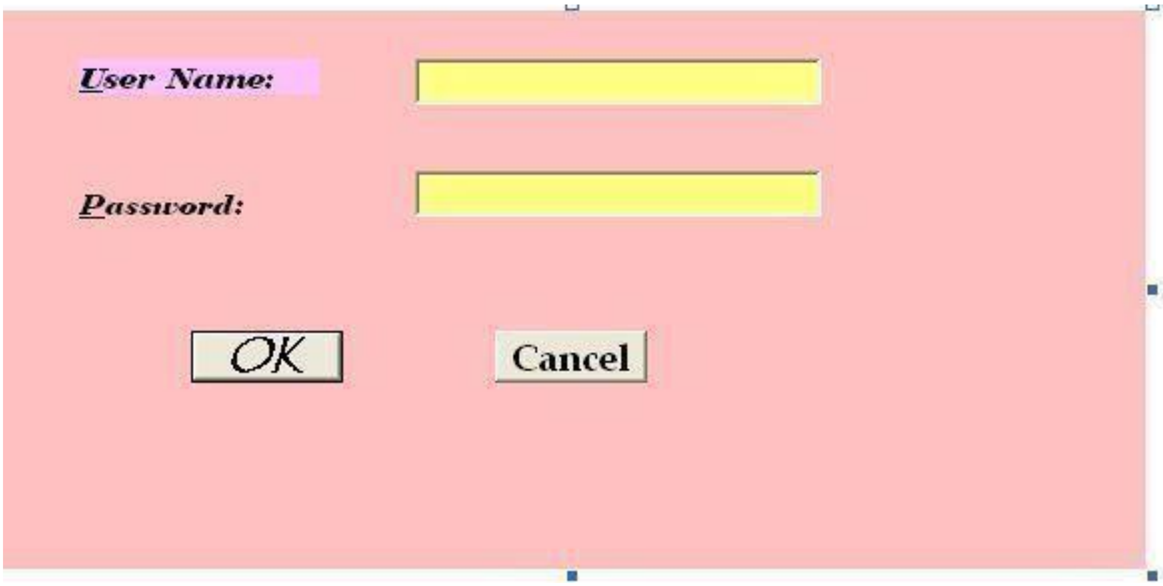


DEPLOYMENT DIAGRAM

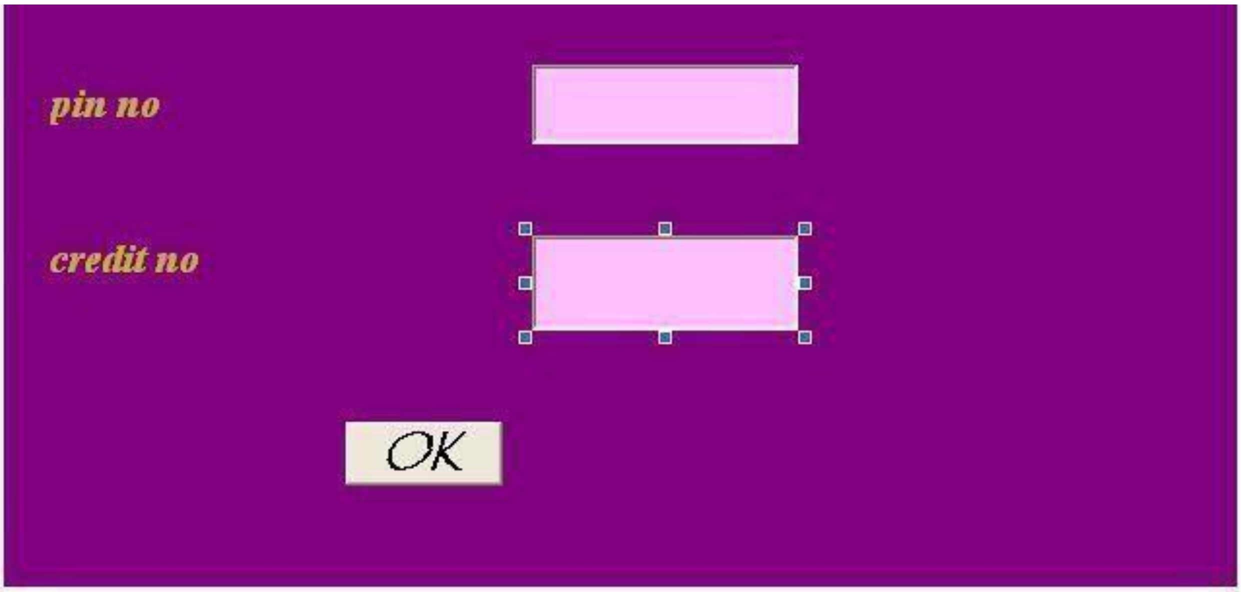


IMPLEMENTATION

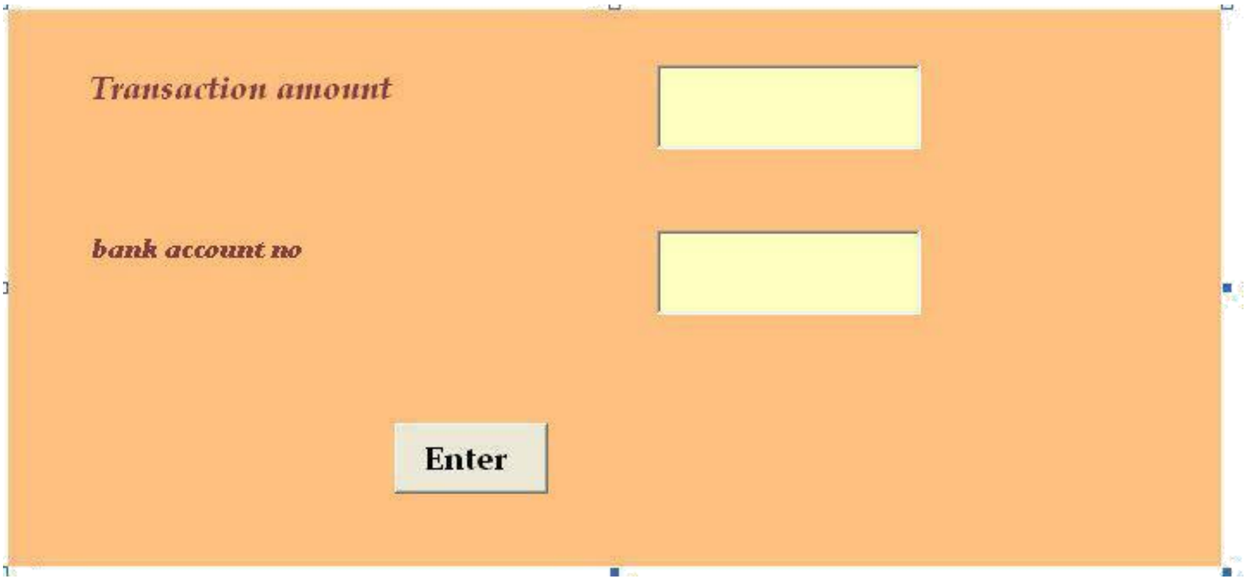
LOGIN FORM:



FORM 1:



FORM 2:



PROGRAM:

LOGIN FORM:

```
Private Sub cmdOK_Click()
'check for correct password
If txtPassword = "user" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form2.Show

Else
If txtPassword = "admin" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form2.Show
End If
End Sub
```

FORM 1:

```
Private Sub ok_Click()
'check for correct password
If pin = "2089" And cno = "2089" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
LoginSucceeded = True
Me.Hide
Form3.Show
Else
If pin = "1234" And cno = "1234" Then
'place code to here to pass the
'success to the calling sub
'setting a global var is the easiest
\LoginSucceeded = True
\Me.Hide
Form3.Show
Else
MsgBox "Invalid pinno, try again!", , "Login"
pin.SetFocus
SendKeys "{Home}+{End}"
End If
End If
End Sub
```

FORM 2:

```
Private Sub enter_Click()
If tamt<= 1580 Then
MsgBox "transaction succeded"
Else
MsgBox "amount not available"
End If
End Sub
```

CONCLUSION:

Thus the project for Credit Card Processing System was designed and codes are generated and then it was executed successfully.

Ex.No: 9
Date:

E-BOOK MANAGEMENT SYSTEM

AIM

To develop a project E-Book Management system using Agro UML Software and to implement the software in Visual Basic.

PROBLEM ANALYSIS AND PROJECT PLANNING

E-Book Management System gives an idea about how books are maintained in the particular websites. The books that are to be purchased, the books that are to be sold are maintained here. . Further some additional details of the current books that is available in the store are also given. E-Book Management System in this project is done in an authorized way. The password and user id has been set here.

PROBLEM STATEMENT

The website has to be maintained properly since the whole E-Book purchase process can be improved. E-Book management in this project gives the idea about how E-Books are maintained in a particular concern. The book details which includes the number of books available, no of pages and price. E-Book management system the E-Book management in this project is understood by going through the modules that is being involved.

SOFTWARE REQUIREMENT SPECIFICATION

S.NO	CONTENTS
1.	INTRODUCTION
2.	OBJECTIVE
3.	OVERVIEW
4.	GLOSSARY
5.	PURPOSE
6.	SCOPE
7.	FUNCTIONALITY
8.	USABILITY
9.	PERFORMANCE
10.	RELIABILITY
11.	FUNCTIONAL REQUIREMENTS
12.	EXTERNAL INTERFACE REQUREMENTS

1. INTRODUCTION

E-Book management gives an idea about how E-Books are maintained in the particular concern. The E-Books that are to be purchased, the E-Books that are to be sold aremaintained here. Further some additional details of the current E-Book list that is available in the website is also given. E-Book management in this project is done in an authorized way.

2. OBJECTIVE

The main objective of this project is to overcome the work load and time consumption which makes the maintenance of the E-Book in an organization as a tedious process. This project provides complete information about the details of the E-Book to the customers. This project identifies the amount of book available, . Separate modules have been created for purchasing, viewing book details, and delivery details.

3. OVERVIEW

The overview of the project is to Storing of information about the E-Books and updating the E-Book list for each organization which is using this system, keeps track of all the information about the E-Books purchased that are made by the customers, having registration feature of adding up new customers to the organization are provided in this system.

4. GLOSSARY

TERMS	DESCRIPTION
CUSTOMER	Customer will purchase the books from the Website .
DATABASE	Database is used to store the books and details of books.
ADMIN	Handles all the support features and the technical works in the application.
SOFTWARE REQUIREMENT SPECIFICATION	This software specification documents full set of features and function for E-Book management system that is performed in application.

5. PURPOSE

The purpose of E-Book management system is to store and sell the books in a website effectively.

6. SCOPE

The scope of this E-Book management is to maintain the book details after the purchase and list of reaming books available in the same book type.

7. FUNCTIONALITY

The main functionality of E-Book maintenance system is to store and sell E-Books for a website.

8. USABILITY

User interface makes the E-Book management system to be efficient. That is the system will help the admin to maintain stock details easily and helps the store to handle the stocks effectively. The system should be user friendly.

9. PERFORMANCE

It describes the capability of the system to perform the E-Book management system of the store without any error and performing it efficiently.

10. RELIABILITY

The E-Book management system should be able to serve the customer with correct information and day-to-day update of E-Book list details.

11. FUNCTIONAL REQUIREMENTS

Functional requirements are those refer to the functionality of the system. That is the services that are provided to the website which maintains E-Books in online database.

12. EXTERNAL INTERFACE REQUIREMENTS

SOFTWARE REQUIREMENTS

1. **Front end:** Agro UML.
2. **Back end:** Visual Basic 6.0.

HARDWARE REQUIREMENTS

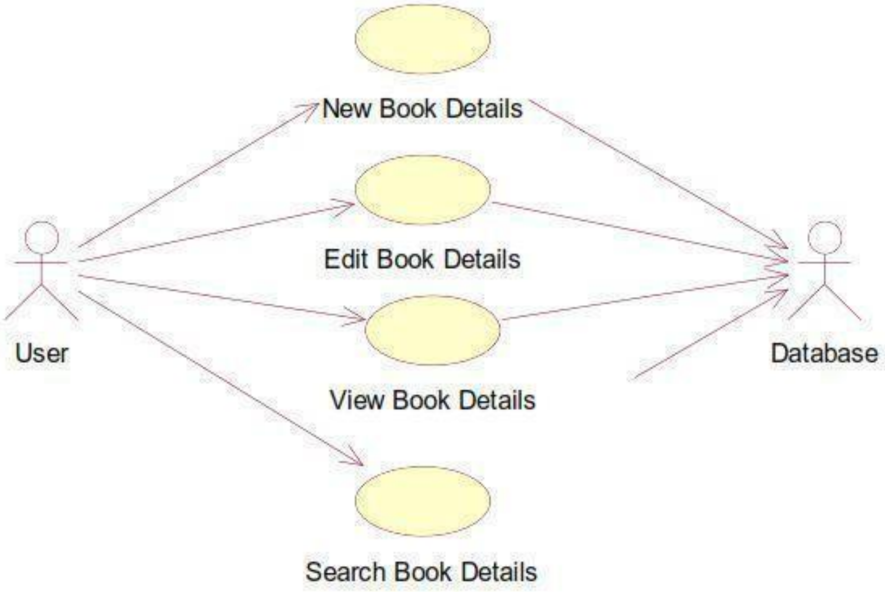
1. **Processor:** Pentium 4.
2. **RAM : 256** MB
3. **Operating system:** Microsoft windows XP.
4. **Free disk space : 1**GB

UML DIAGRAMS

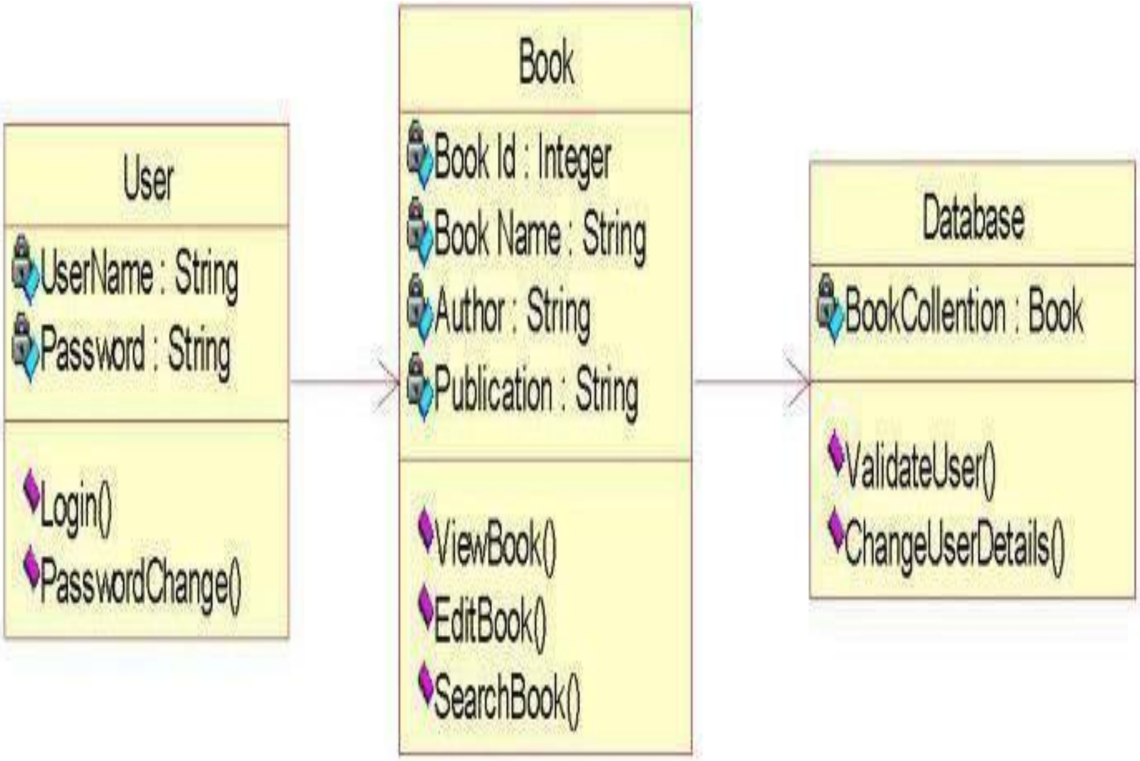
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

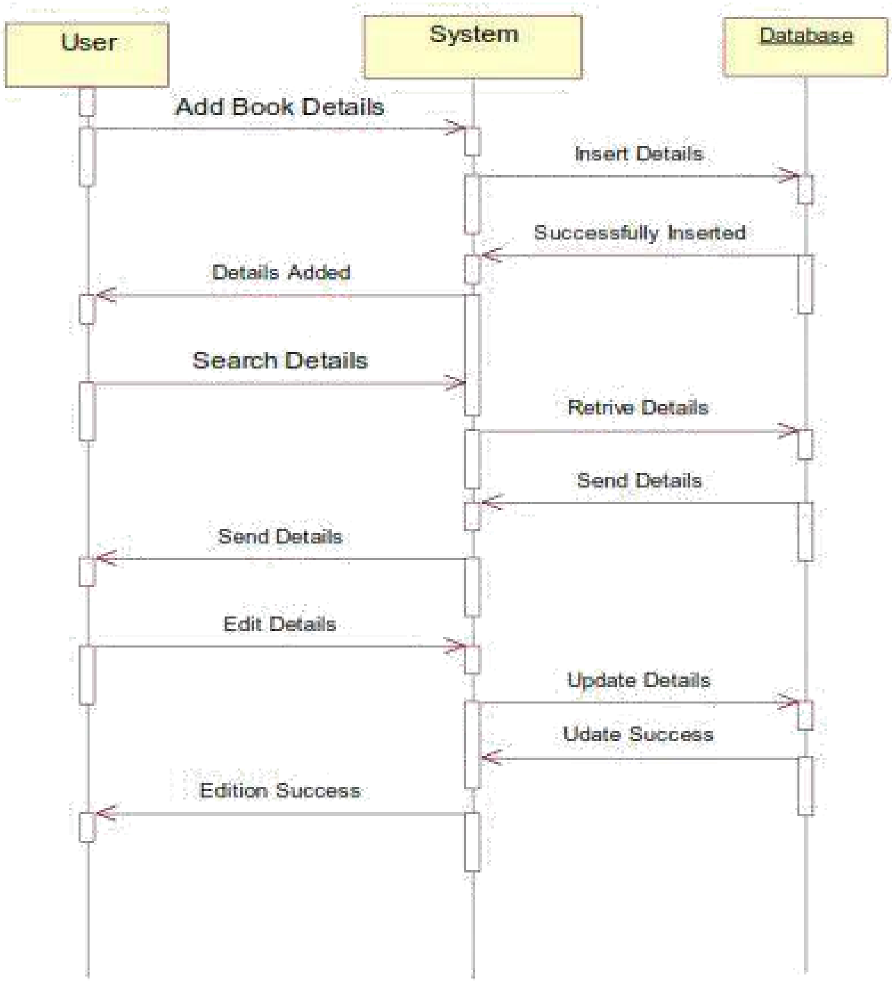
USE CASE DIAGRAM



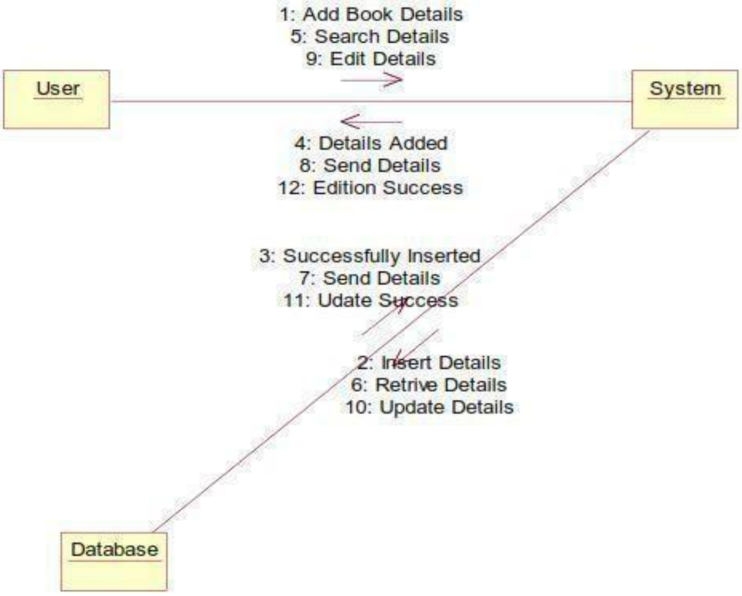
CLASS DIAGRAM



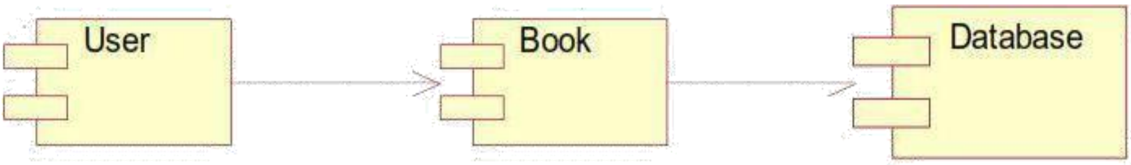
SEQUENCE DIAGRAM



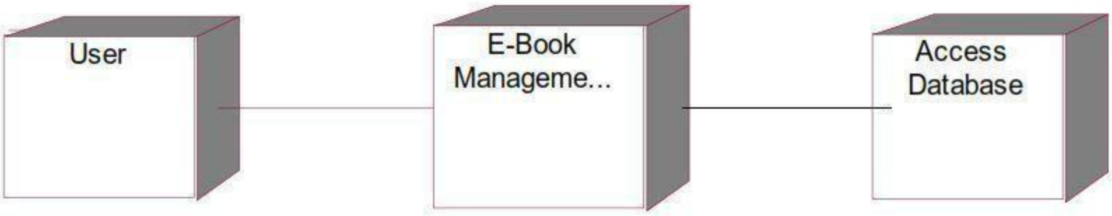
COLLABORATION DIAGRAM



COMPONENT DIAGRAM

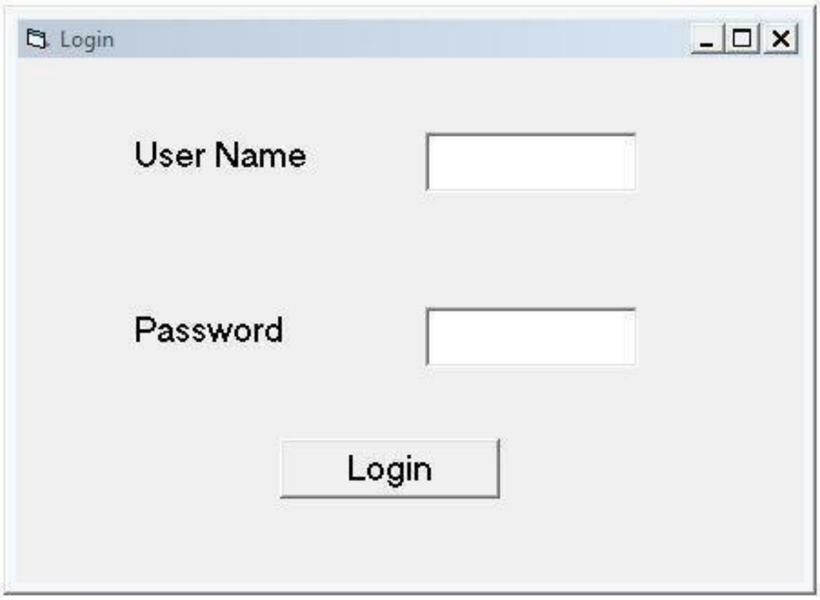


DEPLOYMENT DIAGRAM

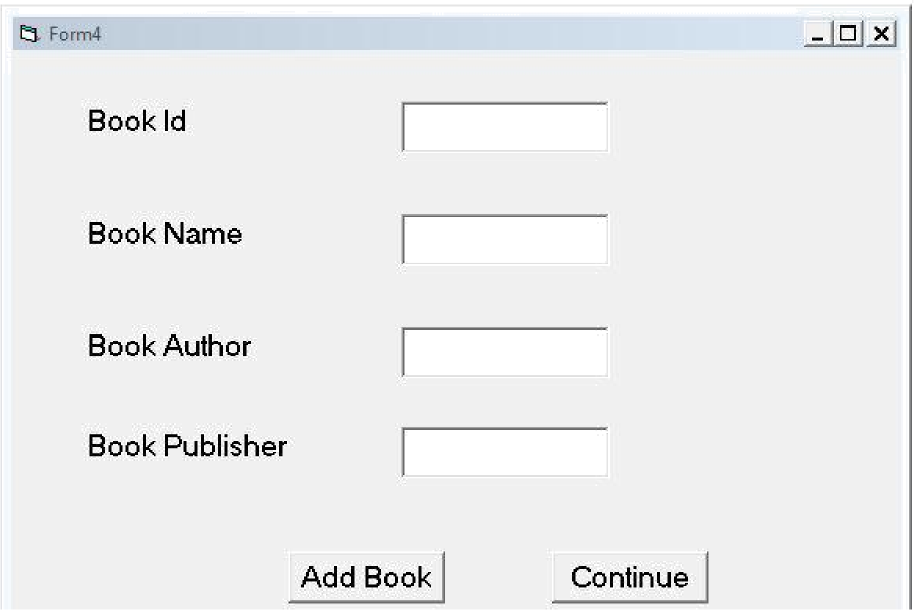


IMPLEMENTATION

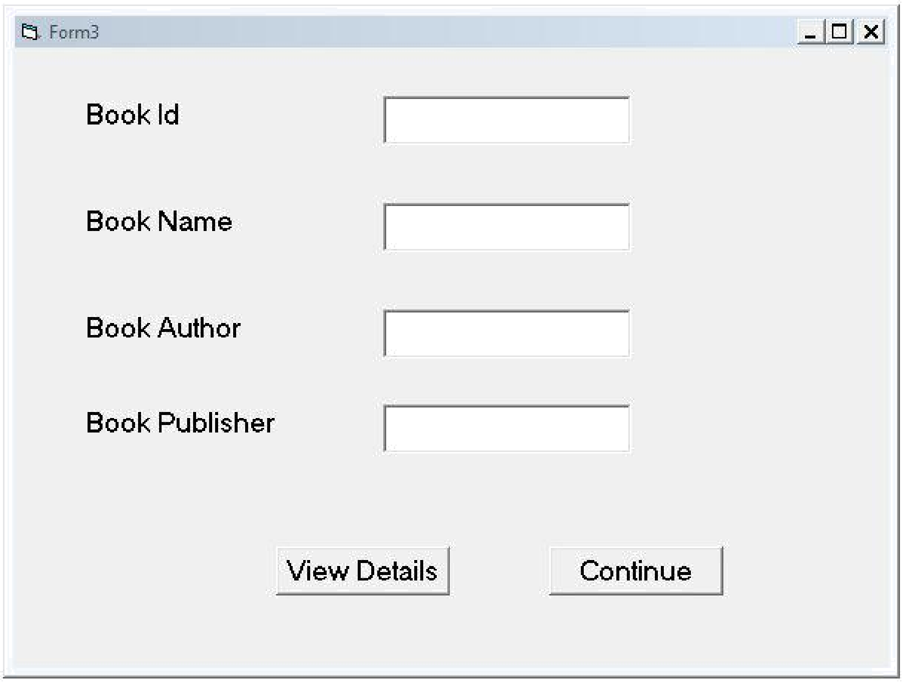
FORM 1:



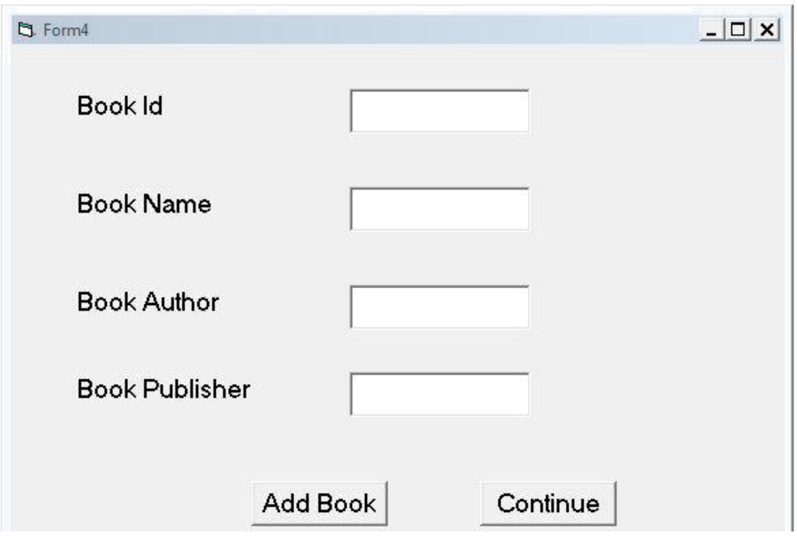
FORM 2:


Form4
Book Id
Book Name
Book Author
Book Publisher
Add Book
Continue

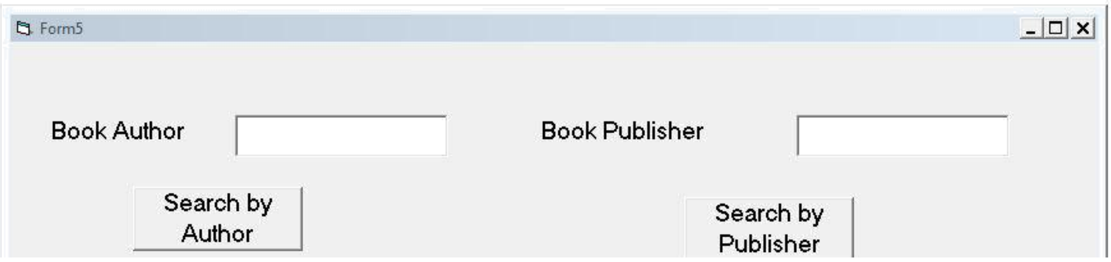
FORM 3:


Form3
Book Id
Book Name
Book Author
Book Publisher
View Details
Continue

FORM 4:


Form4
Book Id
Book Name
Book Author
Book Publisher
Add Book
Continue

FORM 5:


Form5
Book Author
Book Publisher
Search by Author
Search by Publisher

PROGRAM:

FORM 1:

```
Private Sub Command1_Click()
Dim a As Boolean
a = False
If (Text1.Text = "admin" And Text2.Text = "admin") Then a = True
Form2.Show
Unload Me End
If
If (a = False) Then
MsgBox ("Enter Correct Username and Password")
End If
End Sub
```

FORM 2:

```
Private Sub Command1_Click()
Form3.Show
Unload Me
End Sub

Private Sub Command2_Click()
Form4.Show
Unload Me
End Sub

Private Sub Command3_Click()
Form5.Show
Unload Me
End Sub

Private Sub Command4_Click() Unload
Me
End Sub
```

FORM 3:

```
Private Sub Command1_Click()
Dim cn As New ADODB.Connection Dim rs
As New ADODB.Recordset Dim a As Boolean
a = False
cn.Open "dsn=E-Book"
rs.ActiveConnection = cn
With rs
.CursorType = adOpenStatic
.CursorLocation = adUseClient
.LockType = adLockOptimistic
.Open "select * from Details" End
```

```
With
rs.MoveFirst
While Not rs.EOF
If (Val(Text1.Text) = rs(0)) Then
Text2.Text = rs(1)
Text3.Text = rs(2)
Text4.Text = rs(3)
a = True End If
rs.MoveNext Wend
If (a = False) Then
MsgBox ("Enter correct ID") End If
End Sub

Private Sub Command2_Click()
Form2.Show
Unload Me
End Sub
```

FORM 4:

```
Private Sub Command1_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
cn.Open "dsn=E-Book"
rs.ActiveConnection = cn
With rs
.CursorType = adOpenStatic
.CursorLocation = adUseClient
.LockType = adLockOptimistic
.Open "select * from Details" End
With
With rs
.AddNew
.Fields(0) = Val(Text1.Text)
.Fields(1) = Text2.Text
.Fields(2) = Text3.Text
.Fields(3) = Text4.Text
.Update End With
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
End Sub

Private Sub Command2_Click()
Form2.Show
Unload Me
End Sub
```

FORM 5:

```
Private Sub Command1_Click()
Dim cn As New ADODB.Connection Dim rs
As New ADODB.Recordset Dim a As
Boolean
a = False
cn.Open "dsn=E-Book"
rs.ActiveConnection = cn
With rs
.CursorType = adOpenStatic
.CursorLocation = adUseClient
.LockType = adLockOptimistic
.Open "select * from Details" End
With
rs.MoveFirst
While Not rs.EOF
If (Text1.Text = rs(2)) Then
Text3.Text = Text3.Text + Str$(rs(0)) + ", "
Text3.Text = Text3.Text + rs(1) + ", " Text3.Text =
Text3.Text + rs(3) + ".          "
a = True End If
rs.MoveNext Wend
If (a = False) Then
MsgBox ("Enter correct Author Name") End If
End Sub

Private Sub Command2_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim a As Boolean a =
False
cn.Open "dsn=E-Book"
rs.ActiveConnection = cn
With rs
.CursorType = adOpenStatic
.CursorLocation = adUseClient
.LockType = adLockOptimistic
.Open "select * from Details" End
With
rs.MoveFirst
While Not rs.EOF
If (Text2.Text = rs(3)) Then
Text3.Text = Text3.Text + Str$(rs(0)) + ", "
Text3.Text = Text3.Text + rs(1) + ", "
Text3.Text = Text3.Text + rs(2) + ". "
a = True End If
rs.MoveNext Wend
If (a = False) Then
MsgBox ("Enter correct Publisher Name") End If
End Sub
```

```
Private Sub Command3_Click()
Form2.Show
Unload Me
End Sub
```

CONCLUSION

Thus the project for E-book management System was designed and codes are generated and then it was executed successfully.

Ex.No :10
Date :

RECRUITMENT SYSTEM

AIM

To develop a project on online recruitment system using Agro UML and to implement the project in Visual Basic.

PROBLEM ANALYSIS AND PROJECT PLANNING

The Online Recruitment System is an online website in which applicant can register themselves and then attend the exam. Examination will be conducted at some venue. The details of the examination, venue & Date of the examination will be made available to them through the website. Based on the outcome of the exam the applicant will be short listed and the best applicant is selected for the job.

PROBLEM STATEMENT

The process of applicants is login to the recruitment system and register for the job through online. The resume is processed by the company and the required applicant is called for the test. On the basis of the test marks, they are called for next level of interview. Finally the best applicant is selected for the job. This process of online recruitment system are described sequentially through following steps,

- The applicant login to the online recruitment system.
- They register to the company for the job.
- They appear for examination.
- Based on the outcome of the exam, the best applicant is selected.
- The recruiter informs the applicant about their selection.

SOFTWARE REQUIREMENT SPECIFICATION

S.NO	CONTENTS
1.	INTRODUCTION
2.	OBJECTIVE
3.	OVERVIEW
4.	GLOSSARY
5.	PURPOSE
6.	SCOPE
7.	FUNCTIONALITY
8.	USABILITY
9.	PERFORMANCE
10.	RELIABILITY
11.	FUNCTIONAL REQUIREMENTS
12.	EXTERNAL INTERFACE REQUREMENTS

1. INTRODUCTION

This software specification documents full set of features and function for online recruitment system that is performed in company website. In this we give specification about the system requirements that are apart from the functionality of the system to perform the recruitment of the jobseekers. It tells the usability, reliability defined in use case specification.

2. OBJECTIVE

The main objective of Online Recruitment System is to make applicants register themselves online and apply for job and attend the exam. Online Recruitment System provides online help to the users all over the world.

3. OVERVIEW

The overview of the project is to design an online tool for the recruitment process which ease the work for the applicant as well as the companies. Companies can create their company forms according to their wish in which the applicant can register.

4. GLOSSARY

TERMS	DESCRIPTION
APPLICANT	Applicant can register himself. After registration, he will be directed to his homepage. Here he can update his profile, change password and see the examination details and all.
RECRUITER	Recruiter verify applicant details and conduct examination, approve ordisapprove applicant attending examination and provides results about the selected applicant.
DATABASE	Database is used to verify login and store the details of selected applicants.
READER	Anyone visiting the site to read about online recruitment system.
USER	Applicant and the reader
SOFTWARE REQUIREMENT SPECIFICATION	This software specification documents full set of features and function for online recruitment system that is performed incompany website.

The main functionality of recruitment system is to recruit the applicant for the job in their company.

8. USABILITY

User interface makes the Recruitment system to be efficient. That is the system will help the applicant to register easily and helps the companies to recruit the applicant effectively. The system should be user friendly.

9. PERFORMANCE

It describes the capability of the system to perform the recruitment process of the applicant without any error and performing it efficiently.

10. RELIABILITY

The online recruitment system should be able to serve the applicant with correct information and day-to-day update of information.

11. FUNCTIONAL REQUIREMENTS

Functional requirements are those refer to the functionality of the system. That is the services that are provided to the applicant who apply for the job.

12. EXTERNAL INTERFACE REQUIREMENTS

SOFTWARE REQUIREMENTS

3. **Front end:** Agro UML.
4. **Back end:** Visual Basic 6.0.

HARDWARE REQUIREMENTS

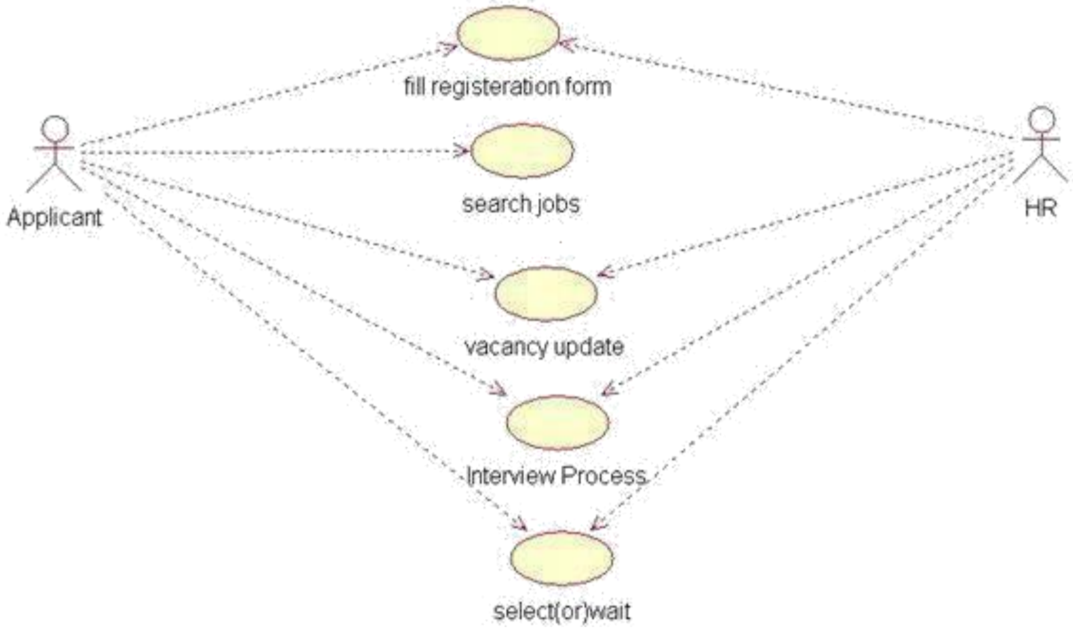
2. **Processor:** Pentium 4.
5. **RAM : 256** MB
6. **Operating system :** Microsoft windows XP.
7. **Free disk space : 1**GB

UML DIAGRAMS:

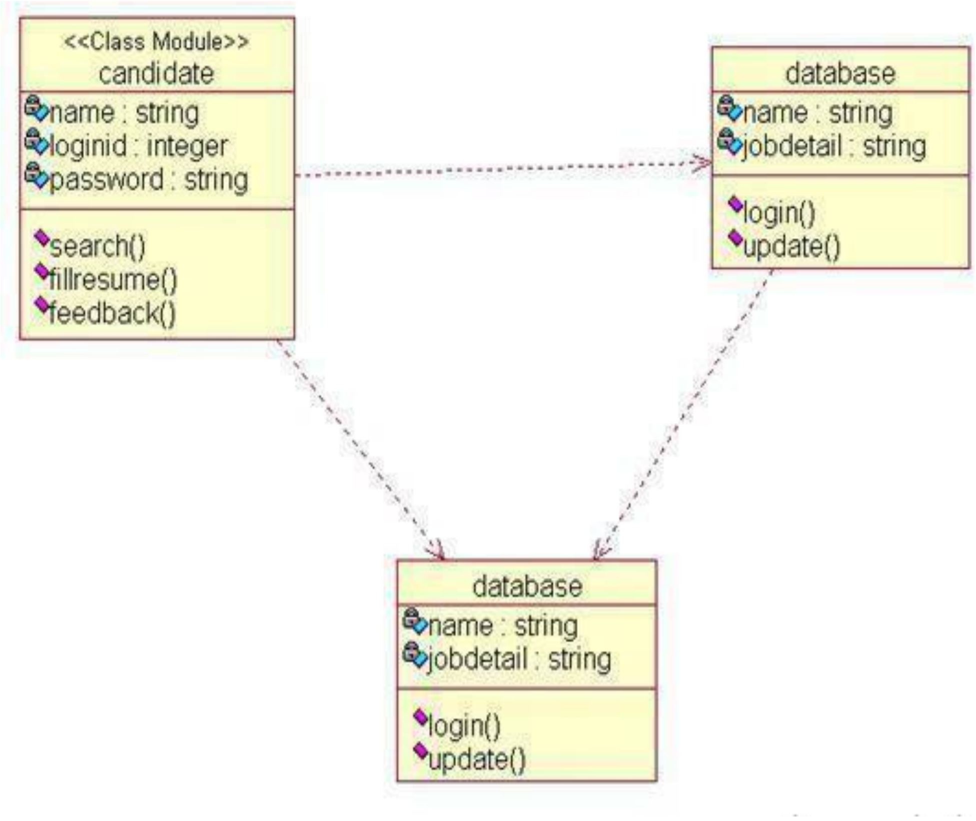
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

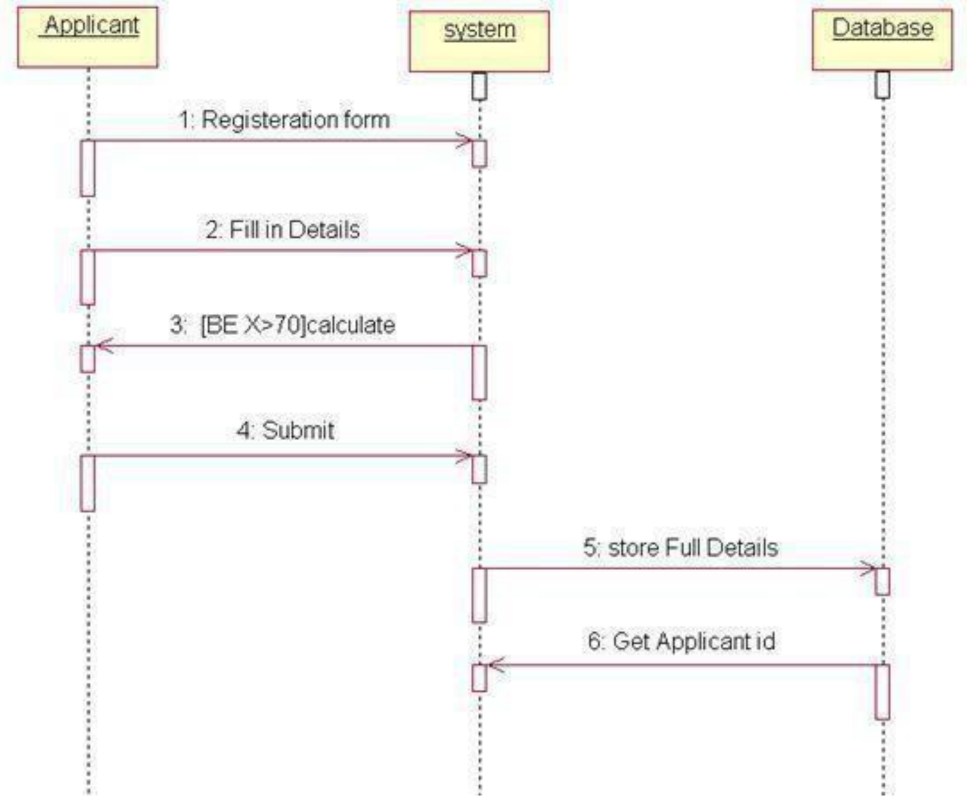
USE CASE DIAGRAM



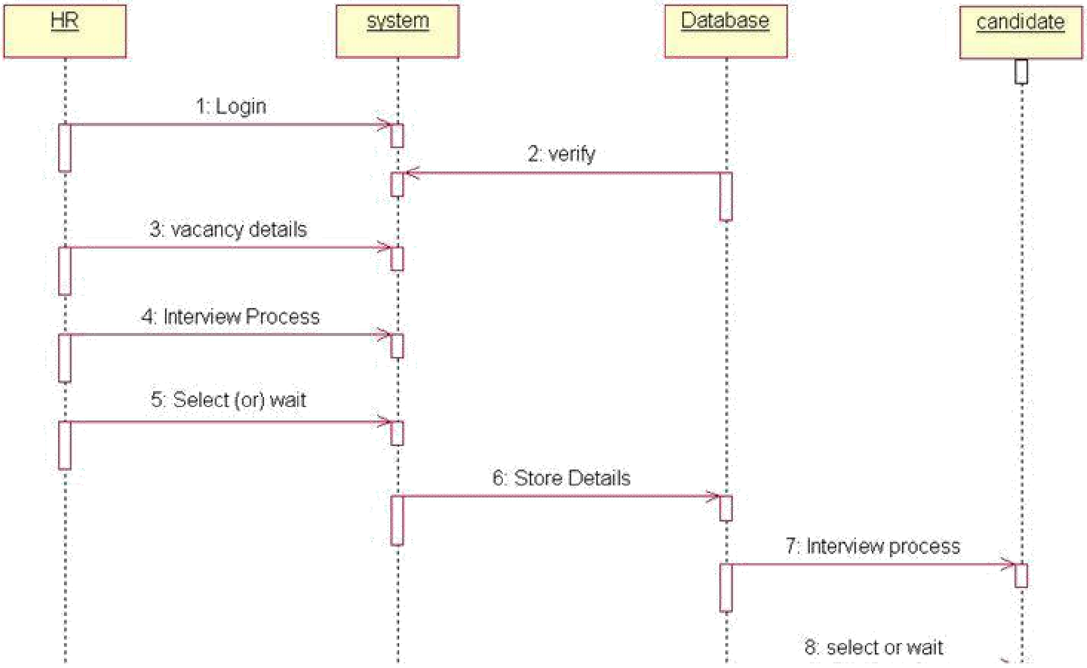
CLASS DIAGRAM



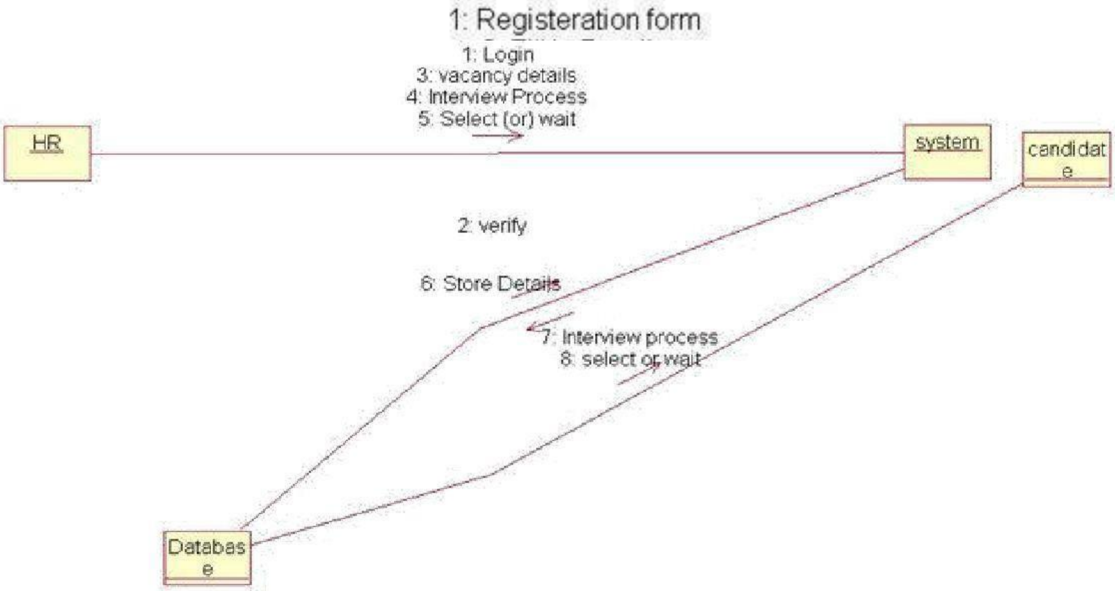
SEQUENCE DIAGRAM (APPLICANT)



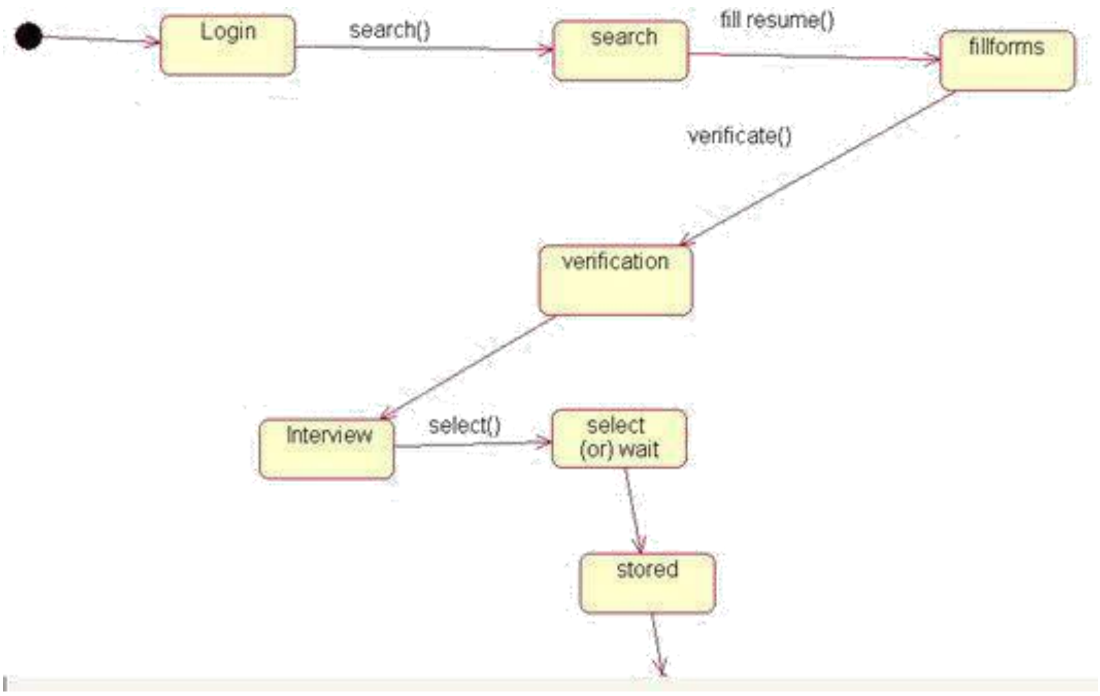
SEQUENCE DIAGRAM



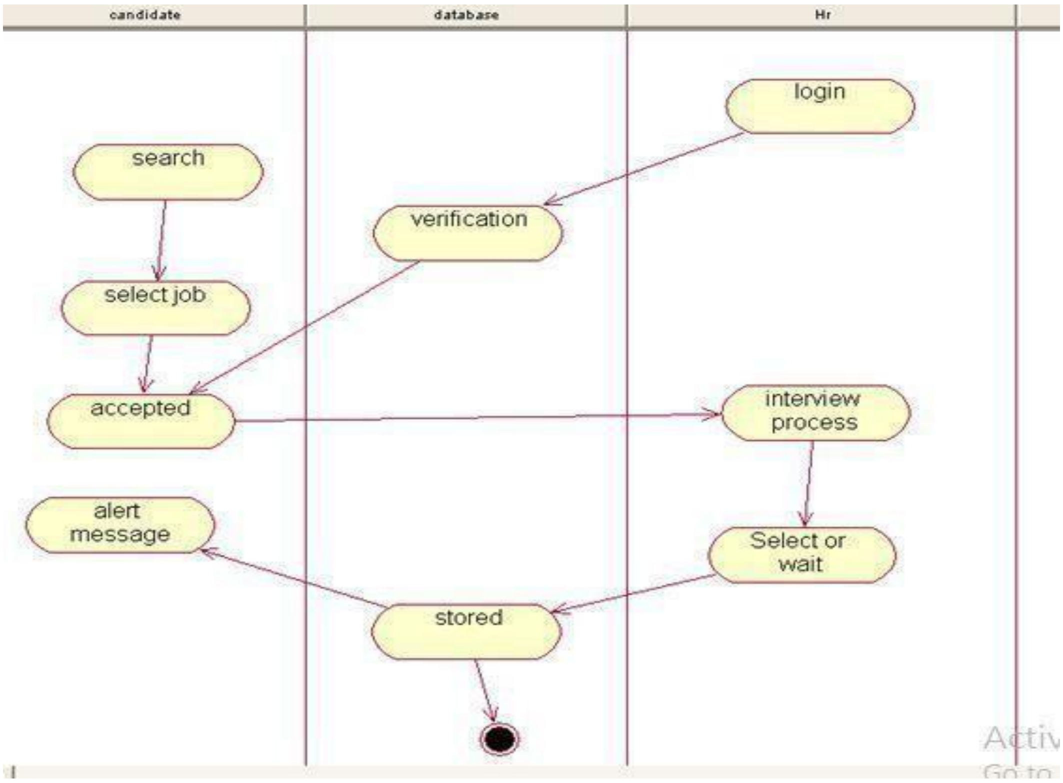
COLLABORATION DIAGRAM (APPLICANT)



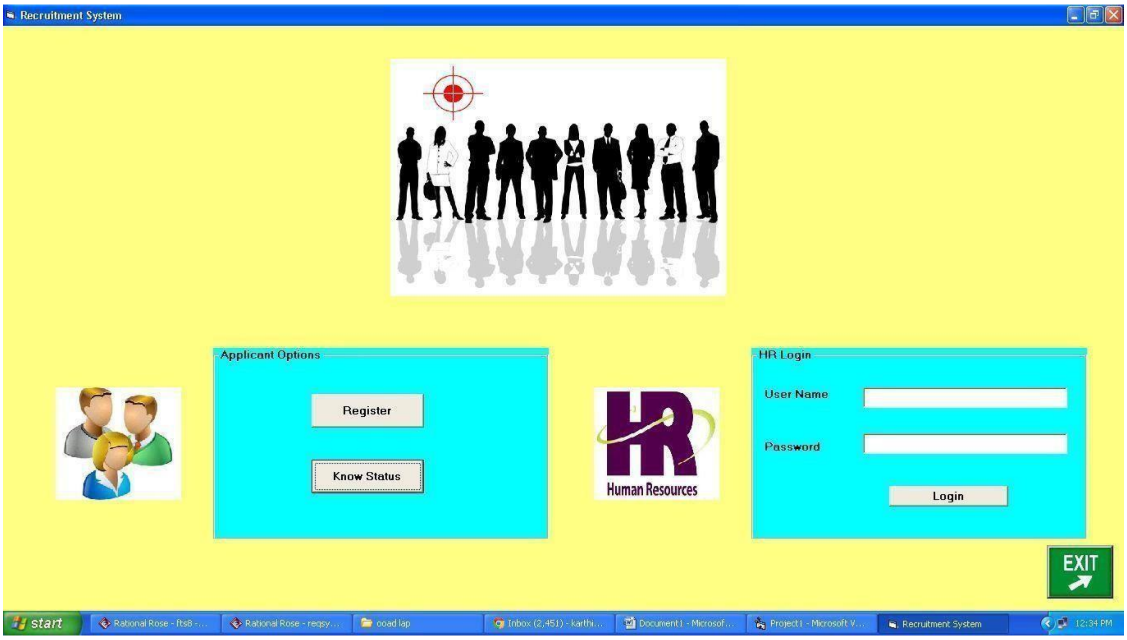
COLLABORATION DIAGRAM (HR)



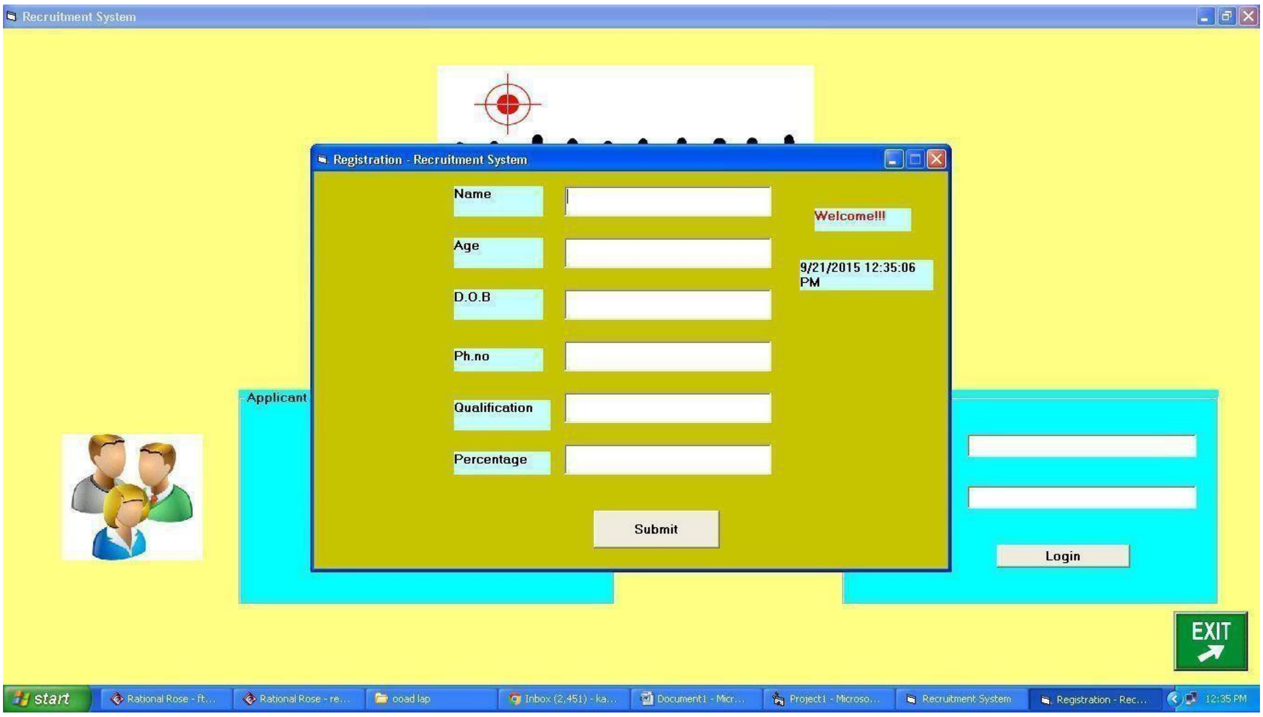
STATE ACTIVITY DIAGRAM



HOME FORM:

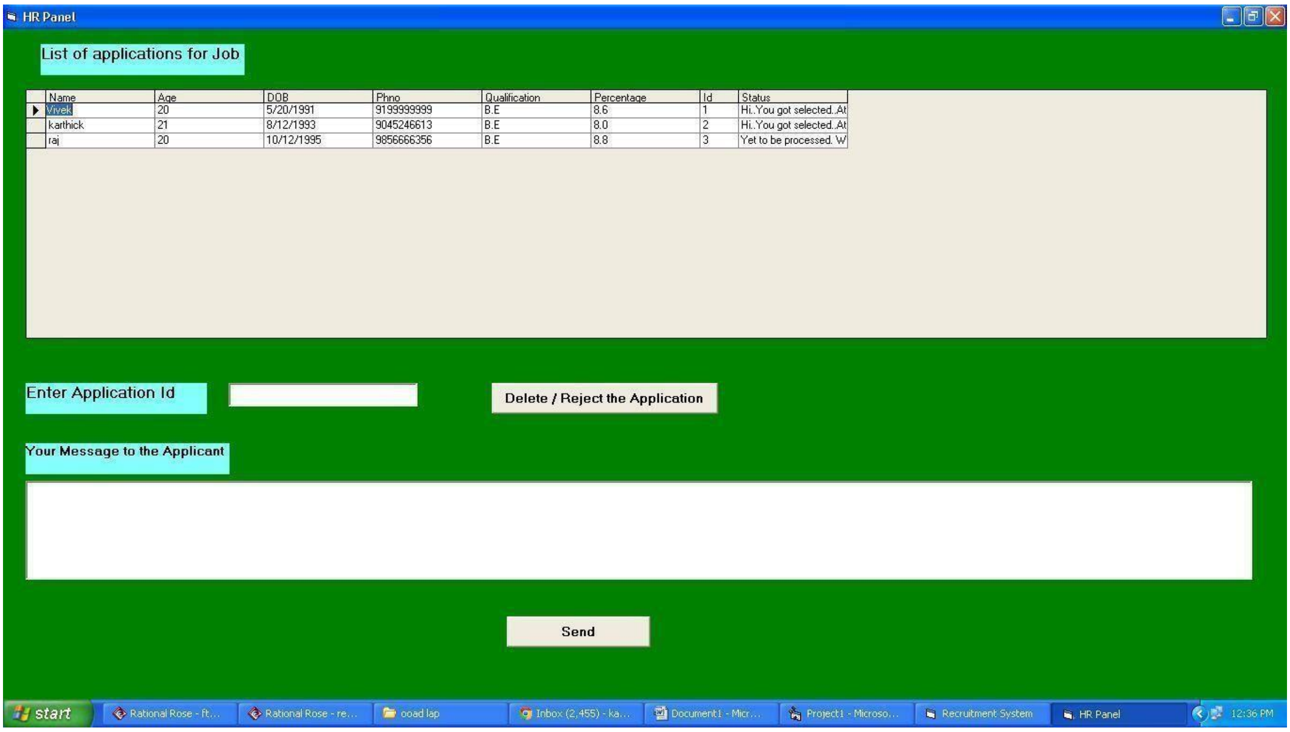


REGISTER FORM:



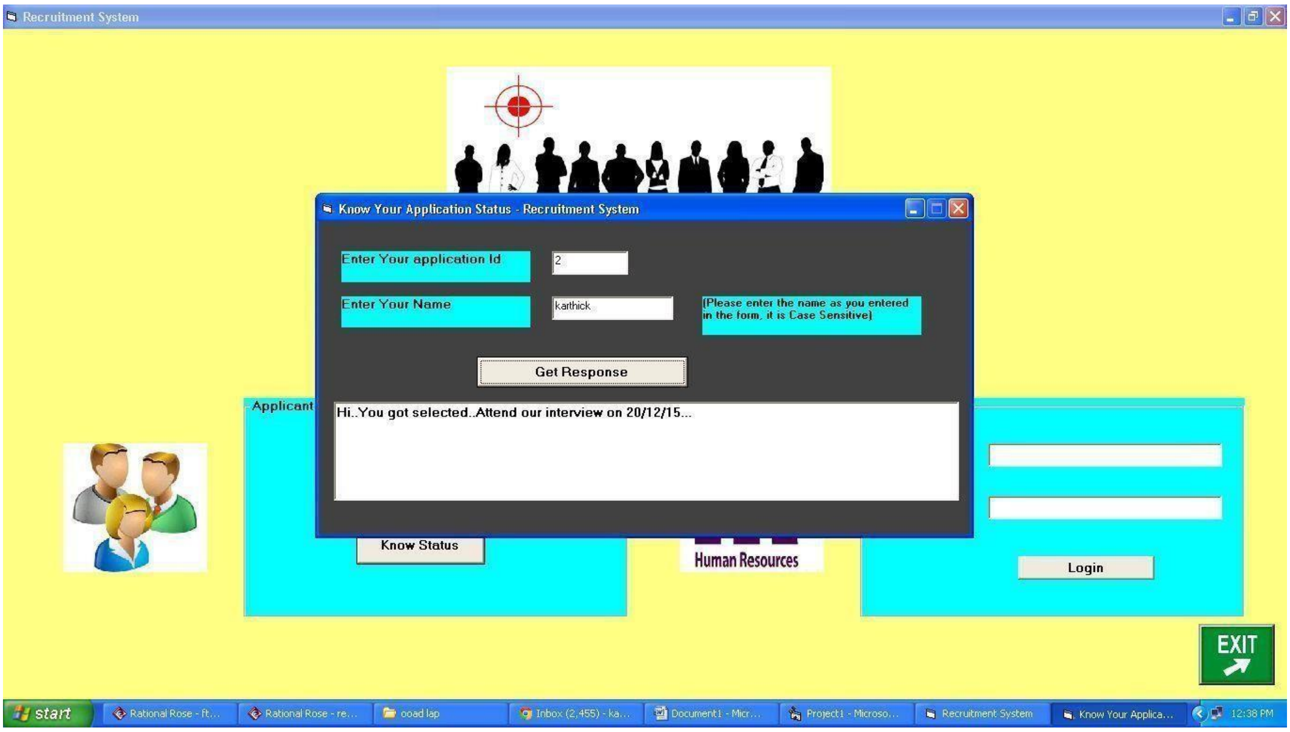
9/21/2015 12:35:06 PM

HR FORM:



Name	Age	DOB	Phno	Qualification	Percentage	Id	Status
Vivek	20	5/20/1991	9199999999	B.E	8.6	1	Hi..You got selected..At
karthick	21	8/12/1993	9045246613	B.E	8.0	2	Hi..You got selected..At
raj	20	10/12/1995	9856666356	B.E	8.8	3	Yet to be processed. W

STATUS FORM:



Enter Your application Id: 2

Enter Your Name: karthick

[Please enter the name as you entered in the form, it is Case Sensitive]

Hi..You got selected..Attend our interview on 20/12/15...

PROGRAM:

HOME FORM:

```
Private Sub Command1_Click()
Status.Show
End Sub
Private Sub Command2_Click()
Register.Show
End Sub

Private Sub Command4_Click()
If hr_username = "admin" And hr_password = "admin" Then
hr_username = ""
hr_password = ""
Hr.Show
Else
MsgBox "Invalid Username / Password", vbCritical, "Recruitment System”
End
End Sub
Private Sub Image2_Click()
Unload Me
End Sub
```

REGISTER FORM:

```
Dim   c  as Integer
Private Sub Command2_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Cn.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + App.Path + "\record.
Mdb;Persist Security Info=False"
R s.Open "record", cn, AdOpenKeyset, adLockPessimistic, AdCmdTable
c = c + 1
r s.AddNew
rs("Name") = Text1
r s("Age") = Text2
r s("DOB") = Text3
r s (" P hno") = Text4
rs("Qualification") = Text5
rs("Percentage") = Text6
rs("Id") = c
rs("Status") = "Yet to be processed. Waiting for the response from HR. Stay Tuned
for updates"
Msg Box "Registration Successful...Your Application id is " & c & "", vbInformation,
"Recruitment System"
rs.Update
rs.Close
cn.Close
Unload Me
End Sub

Private Sub Form_Load()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Cn.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + App.Path +
"\record.mdb;Persist Security Info=False"
R s.Open "select * from record", cn, adOpenKeyset, adLockOptimistic
c = rs.RecordCount
End Sub

Private Sub Timer1_Timer()
Label8.Caption = Now
End Sub
```

HR FORM:

```
Private Sub Command1_Click()
On Error Resume Next
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
C n.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + App.Path +
"\record.mdb;Persist Security Info=False"
R s.Open "update record set status='" + Text2.Text + "' where Id=" + Text1.Text +
"", cn, adOpenKeyset, adLock Optimisti
Msg Box "Response sent successfully", vbInformation, "Recruitment System"
```

```
Unload Me
Me.Show
End Sub

Private Sub Command2_Click()
On Error Resume Next
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
cn.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + App.Path +
"\record.mdb;Persist Security Info=False"
rs.Open "delete from record where Id=" & Text1.Text & "", cn,
adOpenKeyset, adLockOptimistic MsgBox "Delete successfully..",
vbInformation, "Recruitment System" Unload Me
Me.Show
End Sub

Private Sub Form_Load()
On Error Resume Next
Dim oconn As New ADODB.Connection
Dim rs As New ADODB.Recordset
Dim strSQL As String
strSQL = "select * from record"
Set oconn = New ADODB.Connection
oconn.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + App.Path +
"\record.mdb;Persist Security Info=False"
rs.CursorType = adOpenStatic
rs.CursorLocation = adUseClient
rs.LockType = adLockOptimistic
rs.Open strSQL, oconn, adOpenKeyset,
adLockOptimistic Set
DataGrid1.DataSource = rs
End Sub
```

STATUS FORM:

```
Private Sub Command1_Click()
Dim cn As New ADODB.Connection
Dim rs As New ADODB.Recordset
cn.Open "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" + App.Path +
"\record.mdb;Persist Security Info=False"
rs.Open "select * from record where Id=" & Text1.Text & "", cn, adOpenKeyset,
adLockOptimist
If (rs(0).Value = Text2.Text) Then
Text3.Text = rs(7).Value
Else
MsgBox "Please verify the details you have given", vbCritical, "Recruitment System"
End If
End Sub
```

CONCLUSION:

Thus the project for Recruitment System was designed and codes are generated and then it was executed successfully.

Ex.No: 11
Date :

FOREIGN TRADING SYSTEM

AIM

To design a project Foreign Trading System using Agro UML Software and to implement the software in Visual Basic.

PROJECT ANALYSIS AND PROJECT PLANNING

The initial requirements to develop the project about the mechanism of the Foreign Trading System is bought from the trader. The requirements are analysed and refined which enables the analyst (administrator) to efficiently use the Foreign Trading System. The complete project analysis is developed after the whole project analysis explaining about the scope and the project statement is prepared.

PROBLEM STATEMENT

The steps involved in Foreign Trading System are:

- The forex system begins its process by getting the username and password from the trader.
- After the authorization permitted by the administrator, the trader is allowed to perform the sourcing to know about the commodity details.
- After the required commodities are chosen, the trader places the order.
- The administrator checks for the availability for the required commodities and updates it in the database.
- After the commodities are ready for the trade, the trader pays the amount to the administrator.
- The administrator in turn provides the bill by receiving the amount and updates it in the database.
- The trader logouts after the confirmation message has been received.

SOFTWARE REQUIREMENT SPECIFICATION

S.NO	CONTENTS
1.	INTRODUCTION
2.	OBJECTIVE
3.	OVERVIEW
4.	GLOSSARY
5.	PURPOSE
6.	SCOPE
7.	FUNCTIONALITY
8.	USABILITY
9.	PERFORMANCE
10.	RELIABILITY
11.	FUNCTIONAL REQUIREMENTS

1. INTRODUCTION

International trade is exchange of capital, goods, and services across international borders or territories. In most countries, it represents a significant share of gross domestic product (GDP). While international trade has been present throughout much of history (see Silk Road, Amber Road), its economic, social, and political importance has been on the rise in recent centuries. Industrialization, advanced transportation, globalization, multinational corporations, and outsourcing are all having a major impact on the international trade system. Increasing international trade is crucial to the continuance of globalization. Without international trade, nations would be limited to the goods and services produced within their own borders.

2. OBJECTIVE

The main objective of Foreign Trading System is to make the traders to do trading process easily through online as the forex is open 24 hours a day.

3. OVERVIEW

The overview of the project is to design an online tool for the foreign trading process and it oversees the implementation, administration and operations covered in foreign trade.

4. GLOSSARY

TERMS	**DESCRIPTION**
TRADER	Person who trades for the commodities.
ADMINISTRATOR	One who coordinates the entire trading process.
DATABASE	All the transaction details are stored here.
READER	Person who is viewing the website.
USER	The traders and the viewers are the users.
SOFTWARE REQUIREMENTSPECIFICATION	This software specification documentsfull set of features and function forforeign trading system.

5. PURPOSE

The primary purpose of the forex is to assist international trade and investment, by allowing businesses to convert one currency to another currency. That is, In a typical foreign exchange transaction, a trader purchases a quantity of one currency by paying the quantity of another currency.

6. SCOPE

The are a lot of advantages in Forex Trading as compared to many other financial trading ,like futures or stock trading. The Forex market is open 24 hour a day. Being the market available 24 hours a day, this gives the trader to choose which time they would like to trade. It requires only minimum beginning capital to start the Forex trade. Forex Trading has outstanding liquidity as it never closes.

7. FUNCTIONALITY

Transfer purchasing power between countries. Obtain credit for international trade transactions. Minimize exposure to the risks of exchange rate changes.

8. USABILITY

The interface to make the trader access the system will be efficient.

9. PERFORMANCE

The capability that the system performs on the whole will be efficient and reliable without any error occurrence.

10. RELIABILITY

The system should be able to maintain its function throughout the transactions in the future.

11. FUNCTIONALITY REQUIREMENTS

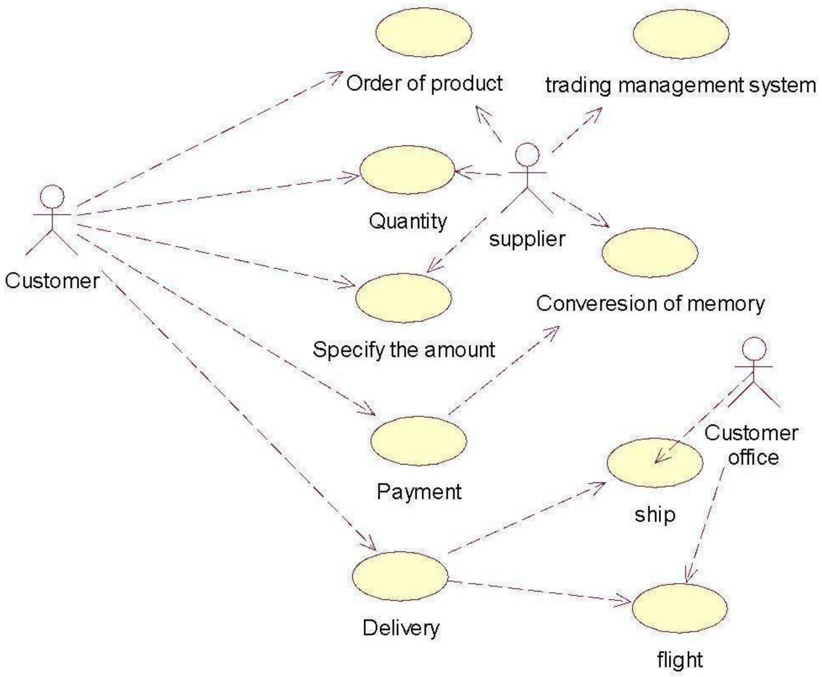
Functional requirements refers to the functionality of the system. The services that are provided to the trader who trades.

UML DIAGRAMS:

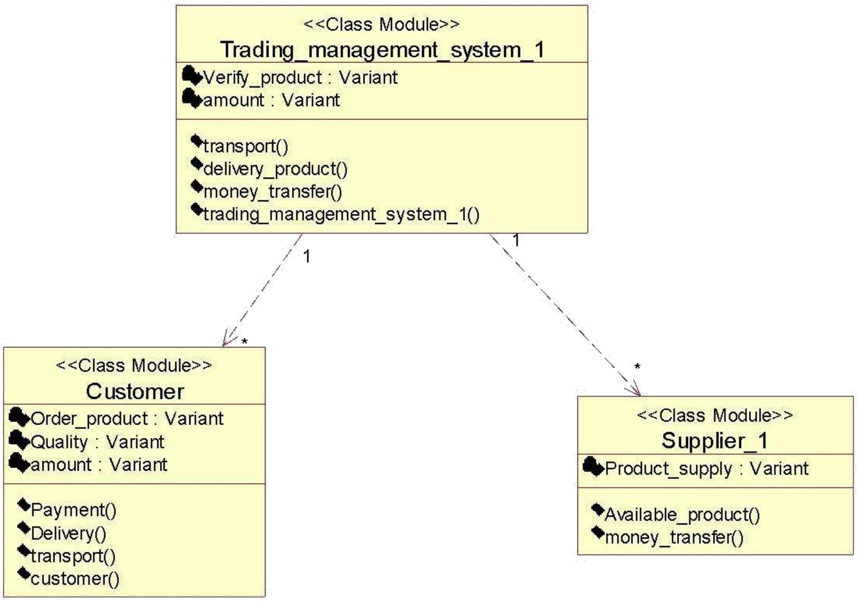
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

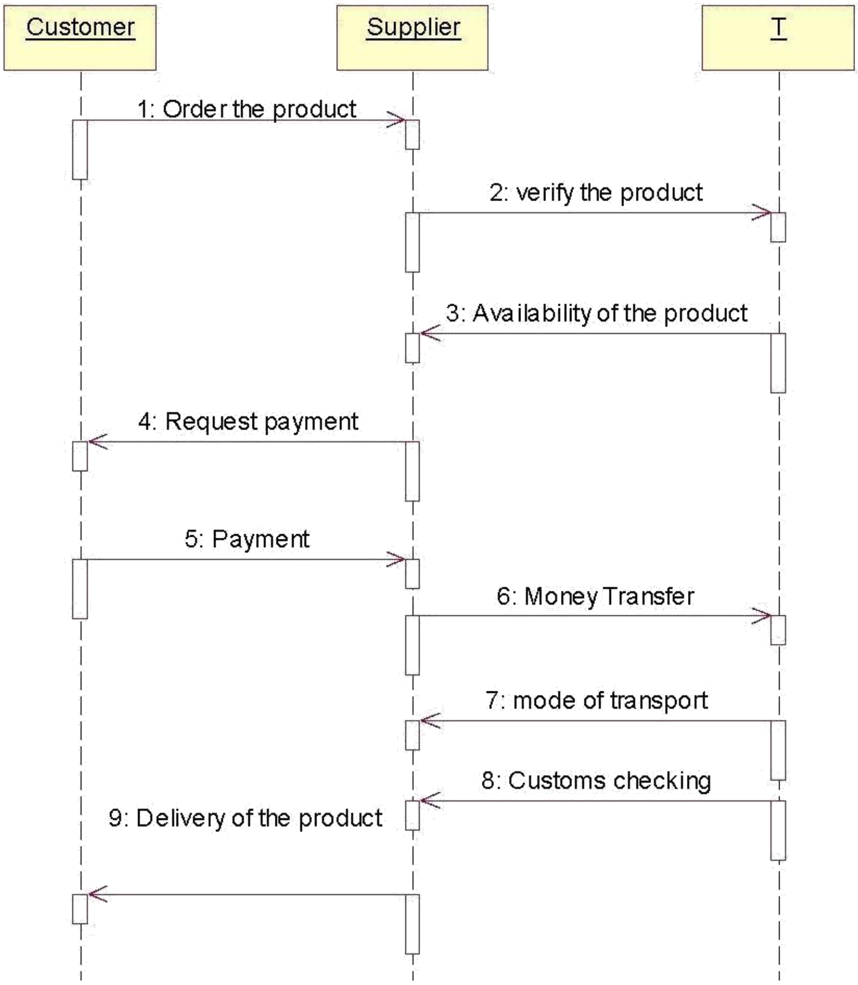
USE CASE DIAGRAM



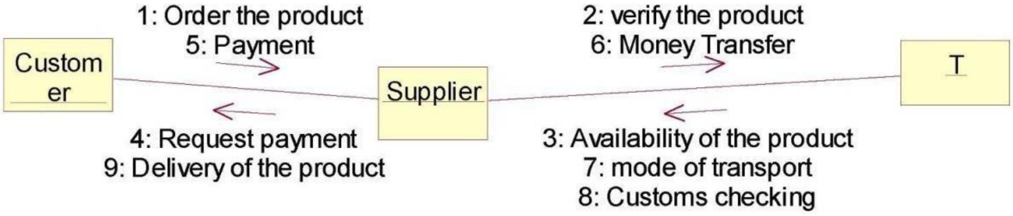
CLASS DIAGRAM



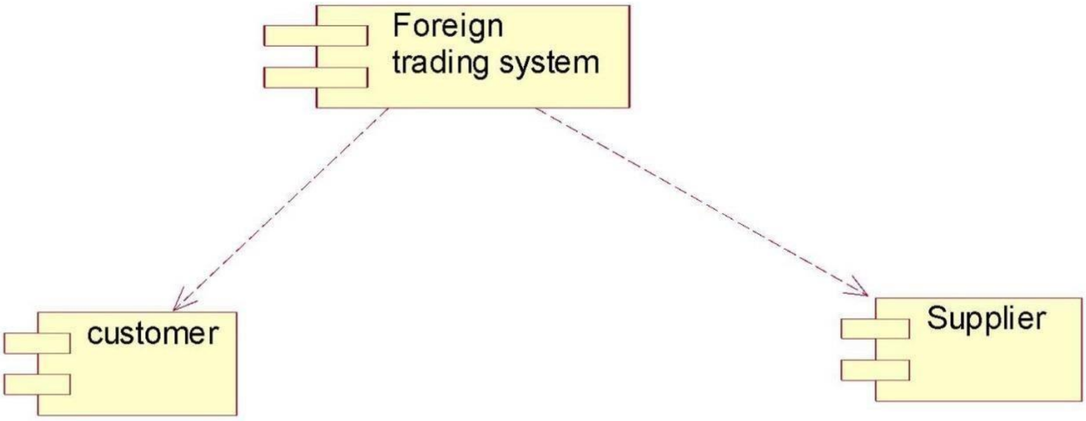
SEQUENCE DIAGRAM



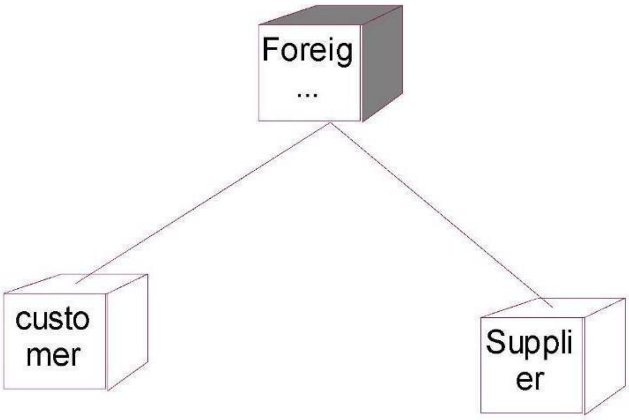
COLLABORATION DIAGRAM



COMPONENT DIAGRAM

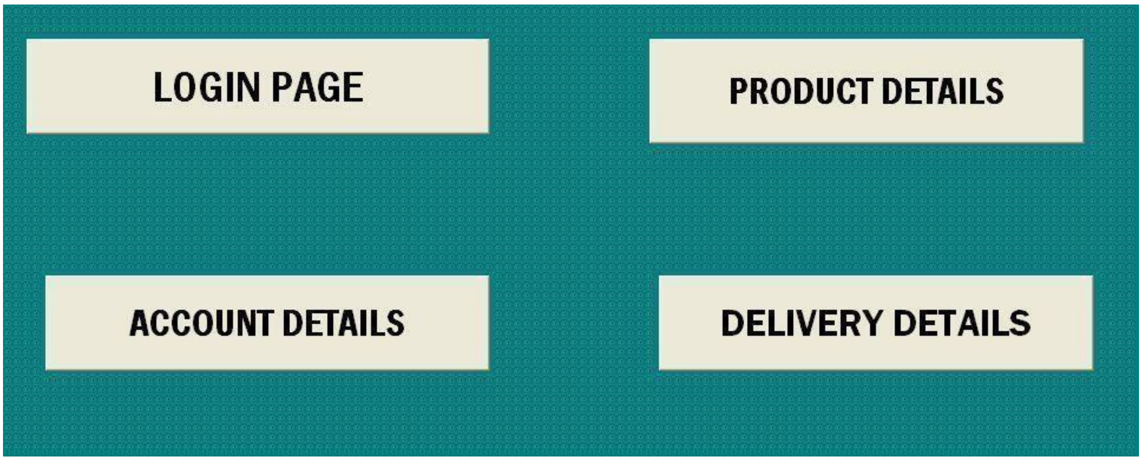


DEPLOYMENT DIAGRAM

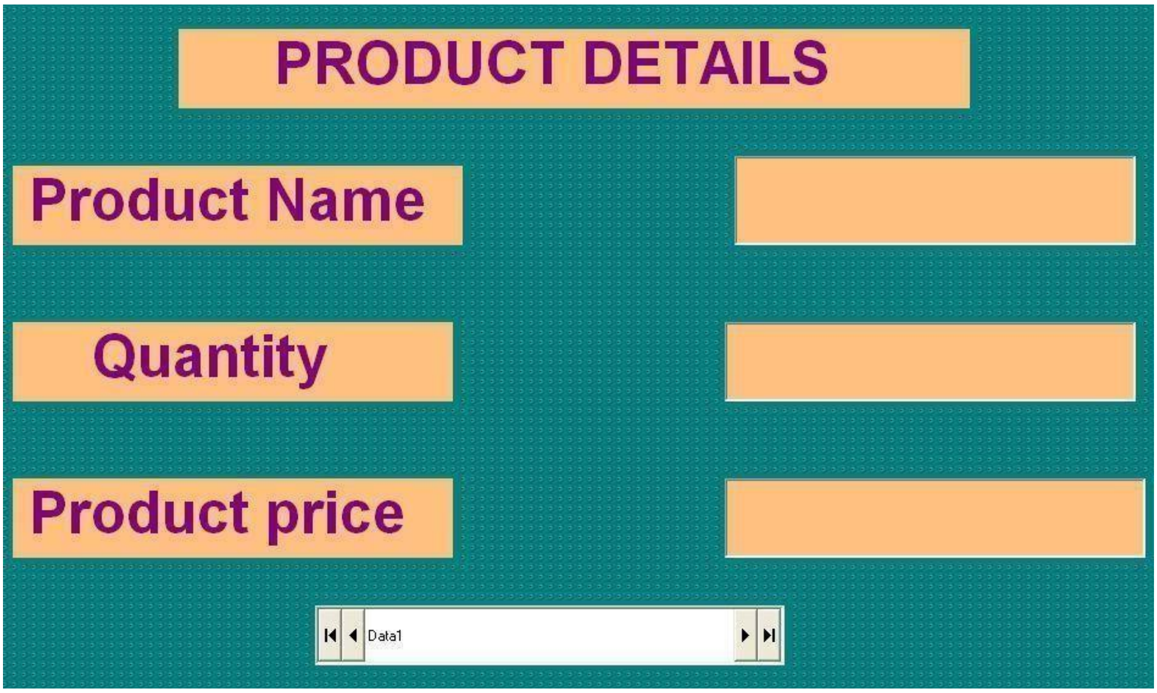


IMPLEMENTATION:

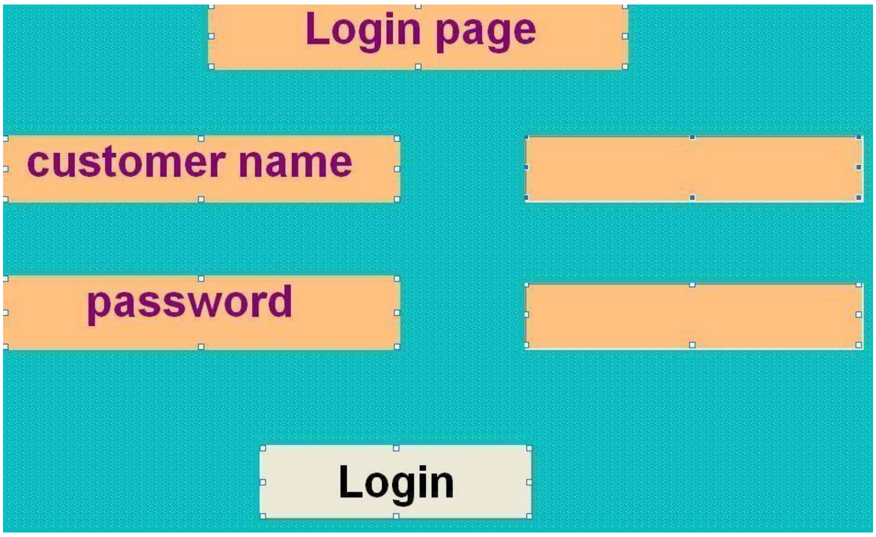
FORM1:



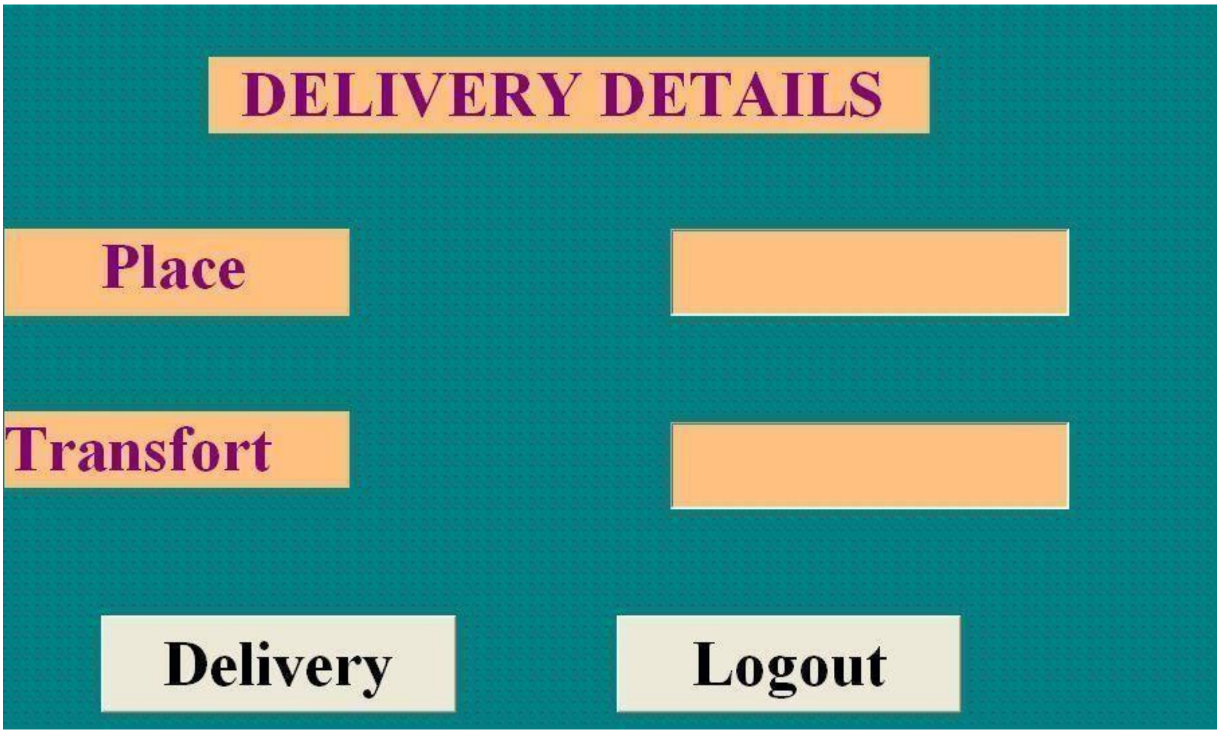
FORM2:


PRODUCT DETAILS
Product Name
Quantity
Product price
Data1

FORM3:


Login page
customer name
password
Login

FORM4:



PROGRAM:

FORM1:

```
Private Sub Command1_Click()
Form3.Show
End Sub

Private Sub Command2_Click(Index As Integer)
Form2.Show
End Sub

Private Sub Command3_Click()
form4.Show
End Sub

Private Sub Command5_Click()
form5.Show
End Sub
```

FORM2:

```
Private Sub Command1_Click()
MsgBox "products purchased"
Form1.Show
End Sub

Private Sub Command2_Click()
Data1.Recordset.AddNew
End Sub
Private Sub Command3_Click()
Data1.Recordset.Edit
End Sub

Private Sub Command4_Click()
Data1.Recordset.Update
End Sub

Private Sub Command5_Click()
Data1.Recordset.Delete
End Sub

Private Sub Data1_Validate(Action As Integer, Save As Integer)
End Sub
```

FORM3:

```
Private Sub Command1_Click()
Text1.Text = Text1.Text
If Text2 = "user" Then
LoginSucceeded = True
Form1.Show
Else
If Text2 = "supplier" Then
LoginSucceeded = True
Form1.Show
Else
MsgBox "Invalid password, Try again!"
End If
End If
End Sub
```

FORM4:

```
Private Sub Command1_Click()
text1.Text = text1.Text
If text2 = "ship" Then
MsgBox "products are delivered"
Else
If text2 = "flight" Then
MsgBox "products are delivered"
Else
MsgBox "Invalid values, Try again!"
End If
End If
End Sub
```

CONCLUSION:

Thus the project for foreign trading System was designed and codes are generated and then it was executed successfully.

Ex.No : **12**
Date :

CONFERENCE MANAGEMENT SYSTEM

AIM

To develop a project on Conference management system using Agro UML Software and to implement the project in Visual Basic.

PROBLEM ANALYSIS AND PROJECT PLANNING

The Conference Management System is an online website in which candidate can submit the paper and register themselves and then attend the conference. The paper will be reviewed. The details of the conference, date and time will be made available to them through the website. After getting the confirmation details the candidate should submit the revised and camera ready paper. Then the registration process will be done.

PROBLEM STATEMENT

The process of the candidates is to login the conference system and submit the paper through online. Then the reviewer reviews the paper and sends the acknowledgement to the candidate either paper selected or rejected. This process of on conference management system are described sequentially through following steps,

- The candidate login to the conference management system.
- The paper title is submitted.
- The paper is been reviewed by the reviewer.
- The reviewer sends acknowledgement to the candidate.
- Based on the selection, the best candidate is selected.
- Finally the candidate registers all details.

SOFTWARE REQUIREMENT SPECIFICATION

S.NO	CONTENTS
1.	INTRODUCTION
2.	OBJECTIVE
3.	OVERVIEW
4.	GLOSSARY
5.	PURPOSE
6.	SCOPE
7.	FUNCTIONALITY
8.	USABILITY
9.	PERFORMANCE
10.	RELIABILITY
11.	FUNCTIONAL REQUIREMENTS

1. INTRODUCTION

This software specification document consist full set of features and function for online conference management system. In this we give specification about the system requirements that are apart from the functionality of the system to perform the candidate paper valuation. It tells the usability, reliability defined in use case specification.

2. OBJECTIVE

The main objective of Conference Management System is to accomplish paper submission online, update the presentation details and confirm registration. Conference management system provides online help to the users all over the world.

3. OVERVIEW

The overview of the project is to design a process which ease the work for the candidate as well as the reviewer. Candidate can easily submit the paper and go for registration.

4. GLOSSARY

TERMS	DESCRIPTION
CANDIDATE	The candidate can login and submit the paper to the reviewer. After getting acknowledgement the candidate will submit the revised and camera ready paper then registration process will be carried out
REVIEWER	Reviewer will reviews the paper and sending acknowledgement to the candidate
DATABASE	Database is used to verify login and store the details of selected candidates.
SOFTWARE REQUIREMENT SPECIFICATION	This software specification documents full set of features and function for conference management system.

5. PURPOSE

The purpose of the conference management system is that the system can easily review the process. The main process in this document is the submission of paper by the candidate, reviewing process by the reviewer and sending of acknowledgement to the candidates whose paper is selected.

6. SCOPE

The scope of this conference management process is to select the best candidate from the list of candidates based on their performance in the process.

7. FUNCTIONALITY

The main functionality of conference system is to select the candidate for the presentation in conference.

8. USABILITY

The user interface to make the process should be effective that is the system will help the candidates to register easily. The system should be user friendly.

9. PERFORMANCE

It describes the capability of the system to perform the conference process of the candidate without any error and performing it efficiently.

10. RELIABILITY

The conference system should be able to serve the applicant with correct information and day-to-day update of information.

11. FUNCTIONAL REQUIREMENTS

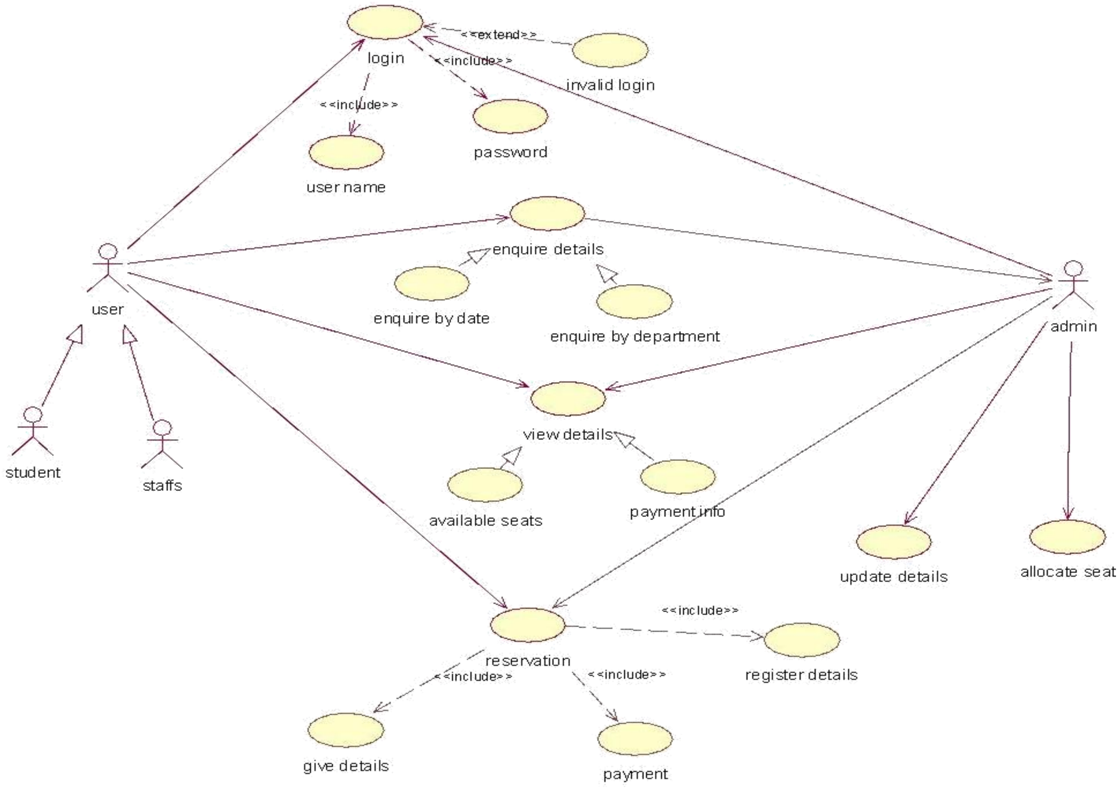
Functional requirements are those that refer to the functionality of the system that is the services that are provided to the candidate who register for the conference.

UML DIAGRAMS:

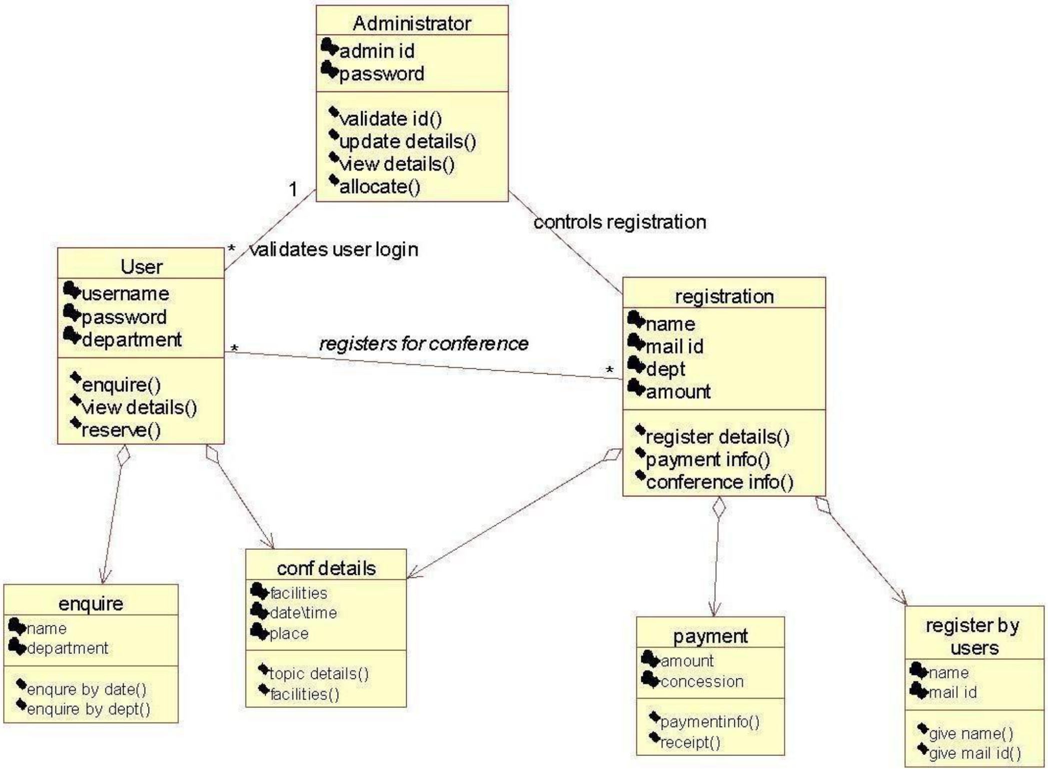
The following UML diagrams describe the process involved in the online recruitment system

a) Use case diagram
b) Class diagram
c) Sequence diagram
d) Collaboration diagram
e) State chart diagram
f) Activity diagram
g) Component diagram
h) Deployment diagram
i) Package diagram

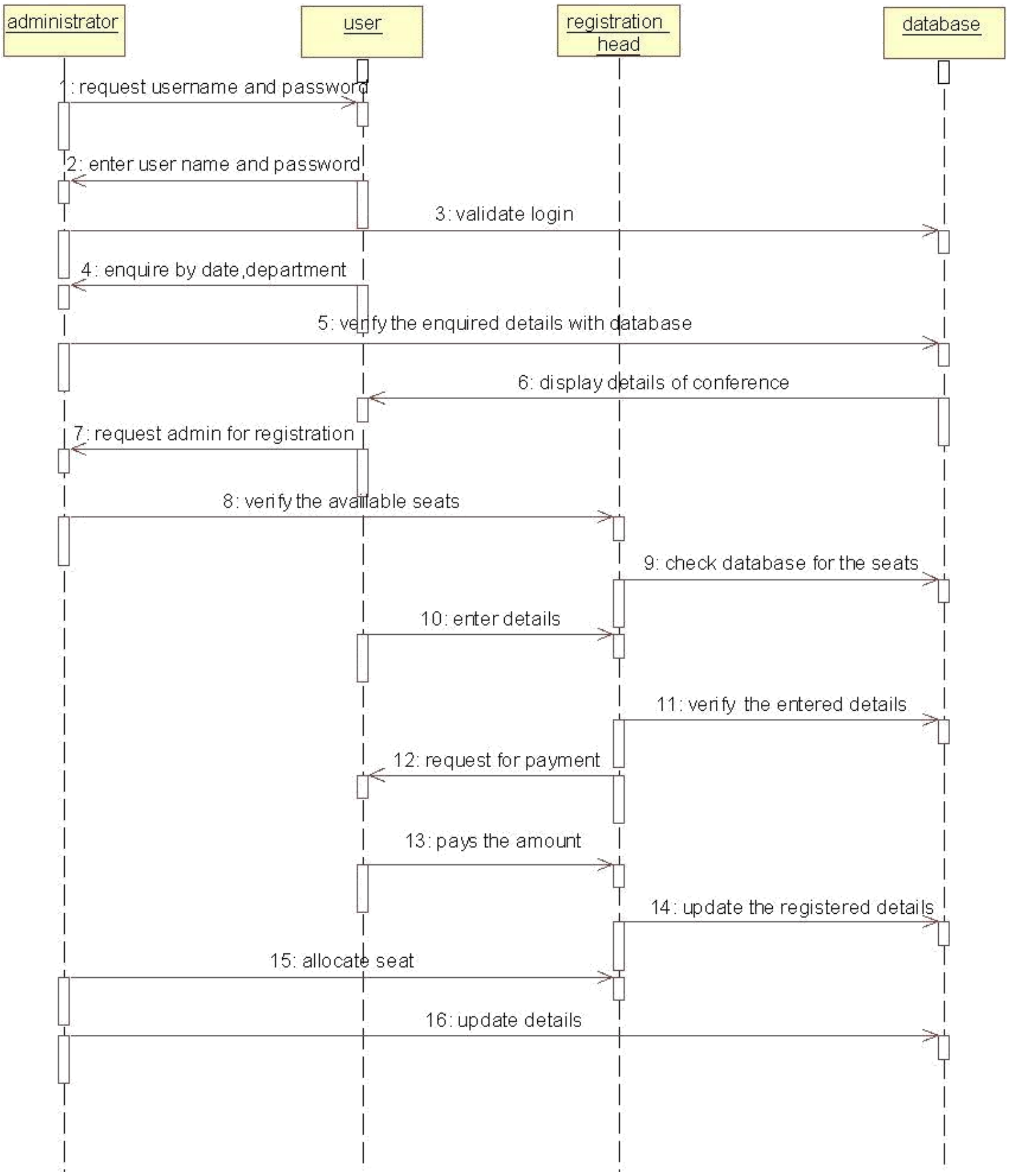
USE CASE DIAGRAM:



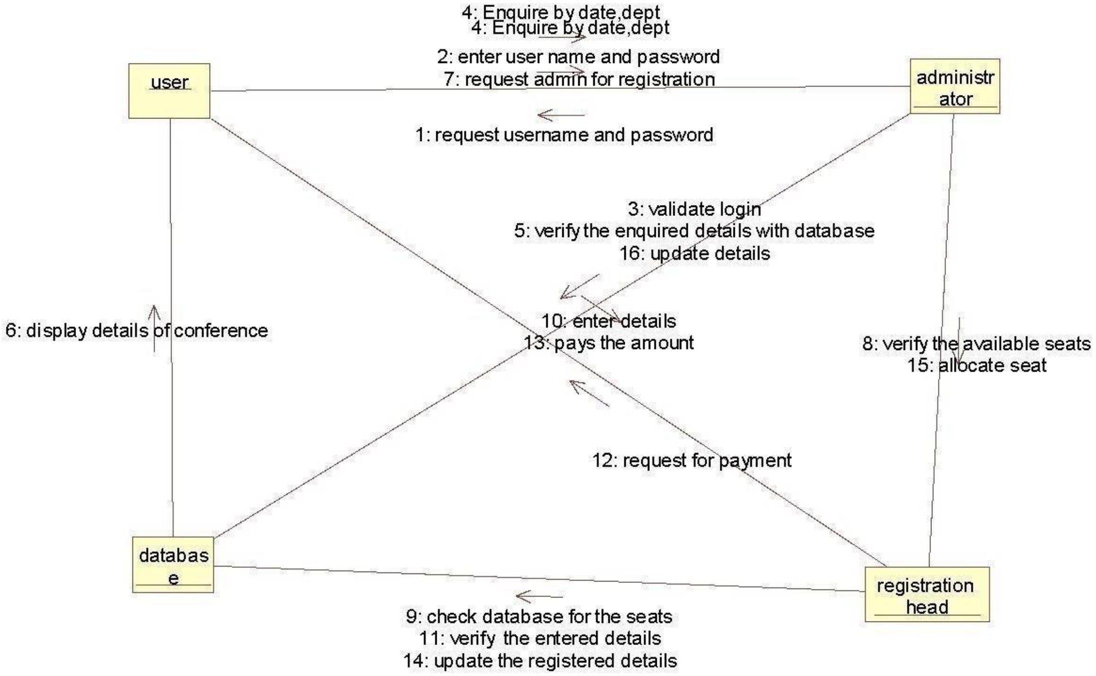
CLASS DIAGRAM:



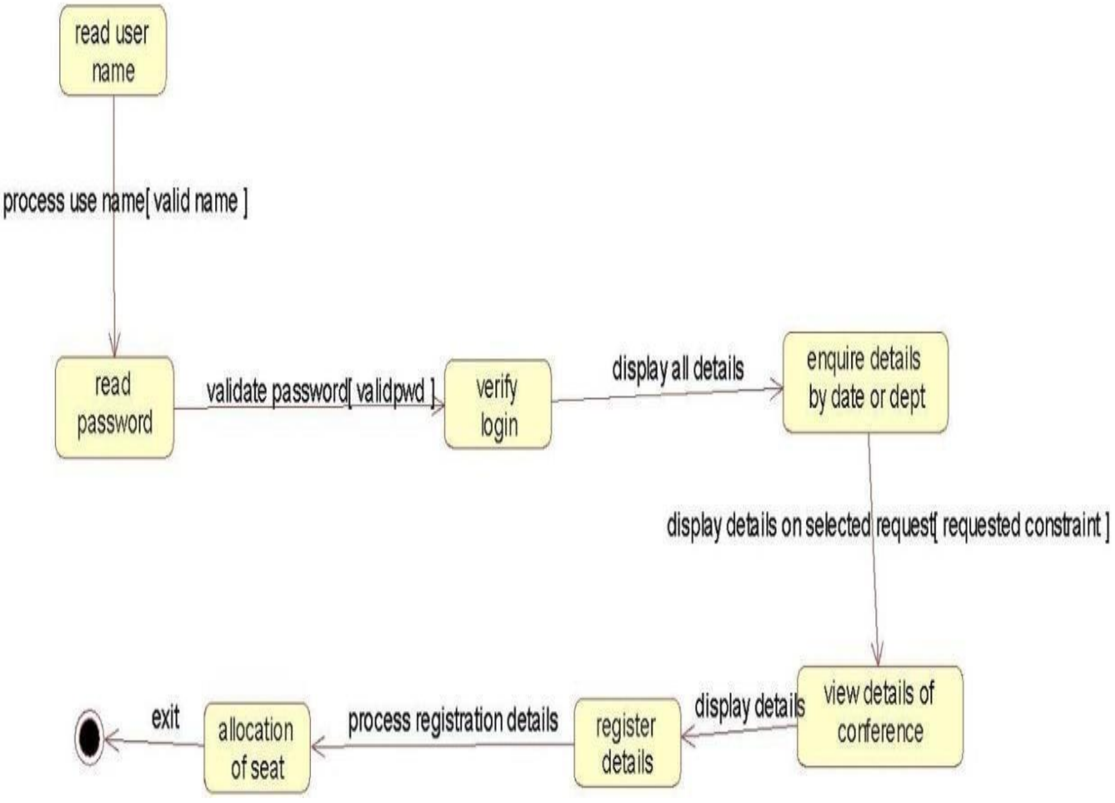
SEQUENCE DIAGRAM:


administrator
user
registration head
database
1: request username and password
2: enter user name and password
3: validate login
4: enquire by date,department
5: verify the enquired details with database
6: display details of conference
7: request admin for registration
8: verify the available seats
9: check database for the seats
10: enter details
11: verify the entered details
12: request for payment
13: pays the amount
14: update the registered details
15: allocate seat
16: update details

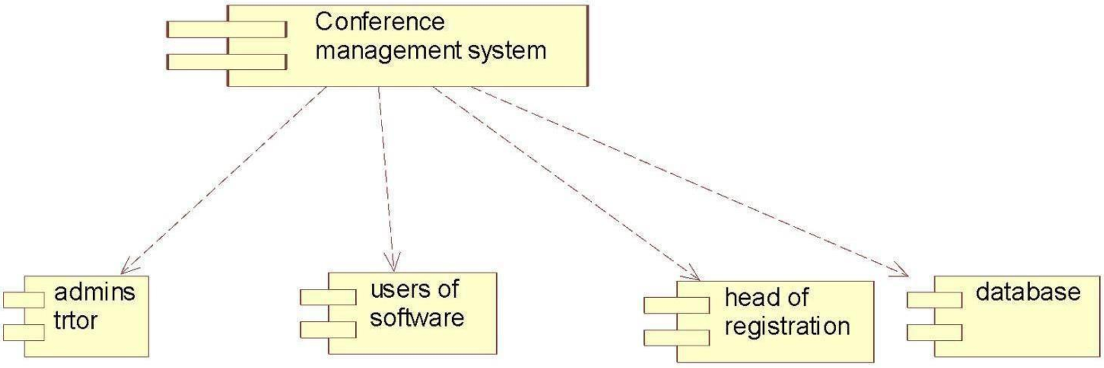
COLLABORATION DIAGRAM


4: Enquire by date,dept
4: Enquire by date,dept
2: enter user name and password
7: request admin for registration
1: request username and password
user
administr
ator
3: validate login
5: verify the enquired details with database
16: update details
6: display details of conference
10: enter details
13: pays the amount
8: verify the available seats
15: allocate seat
12: request for payment
databas
e
registration
head
9: check database for the seats
11: verify the entered details
14: update the registered details

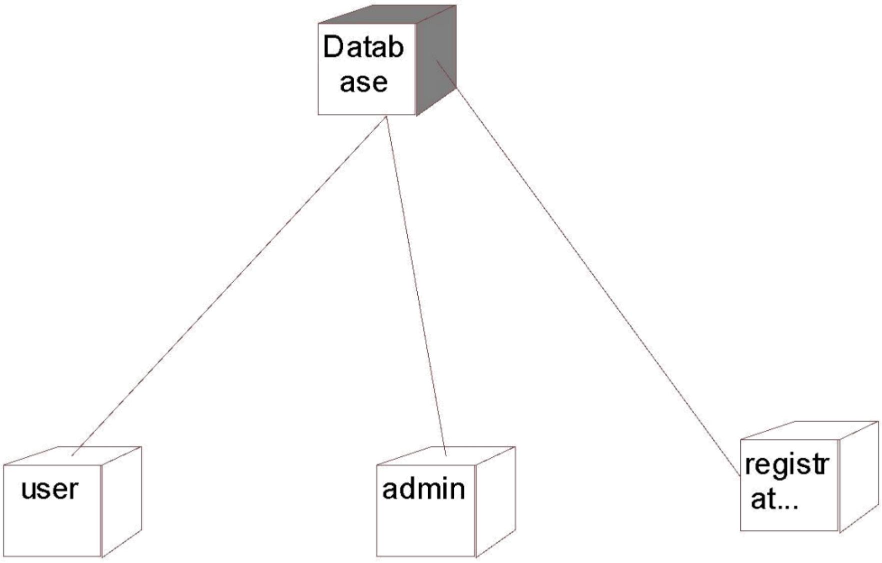
STATE CHART DIAGRAM:



COMPONENT DIAGRAM:

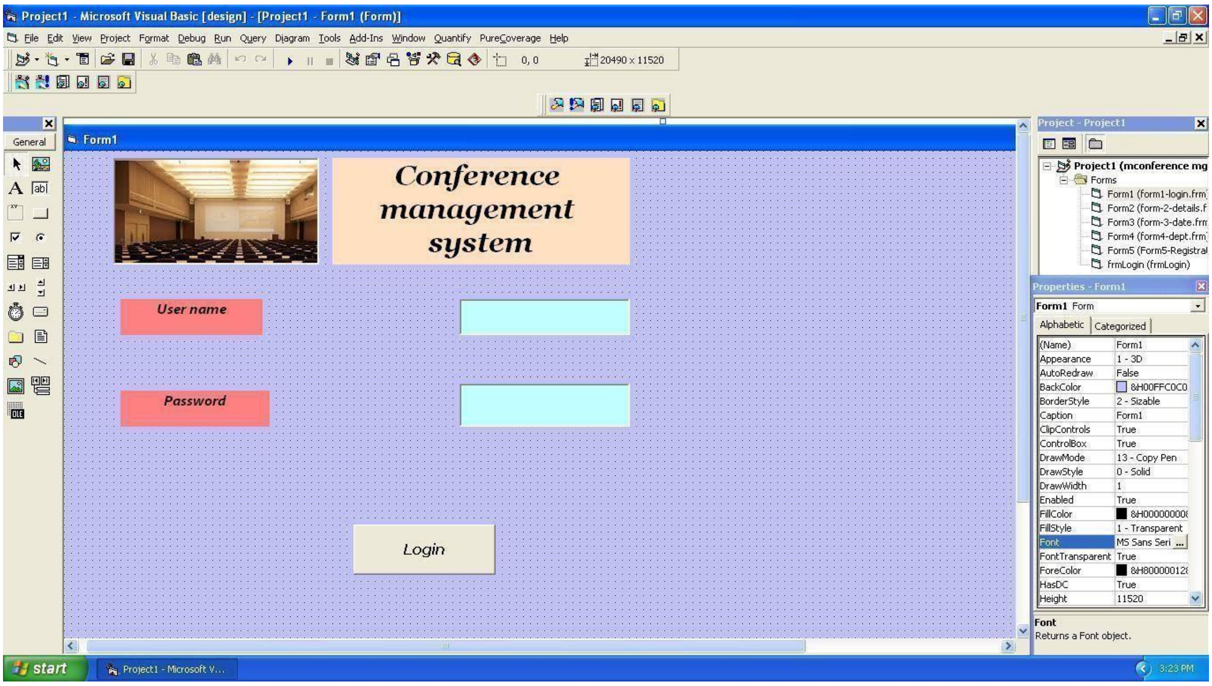


DEPLOYMENT DIAGRAM:



IMPLEMENTATION:

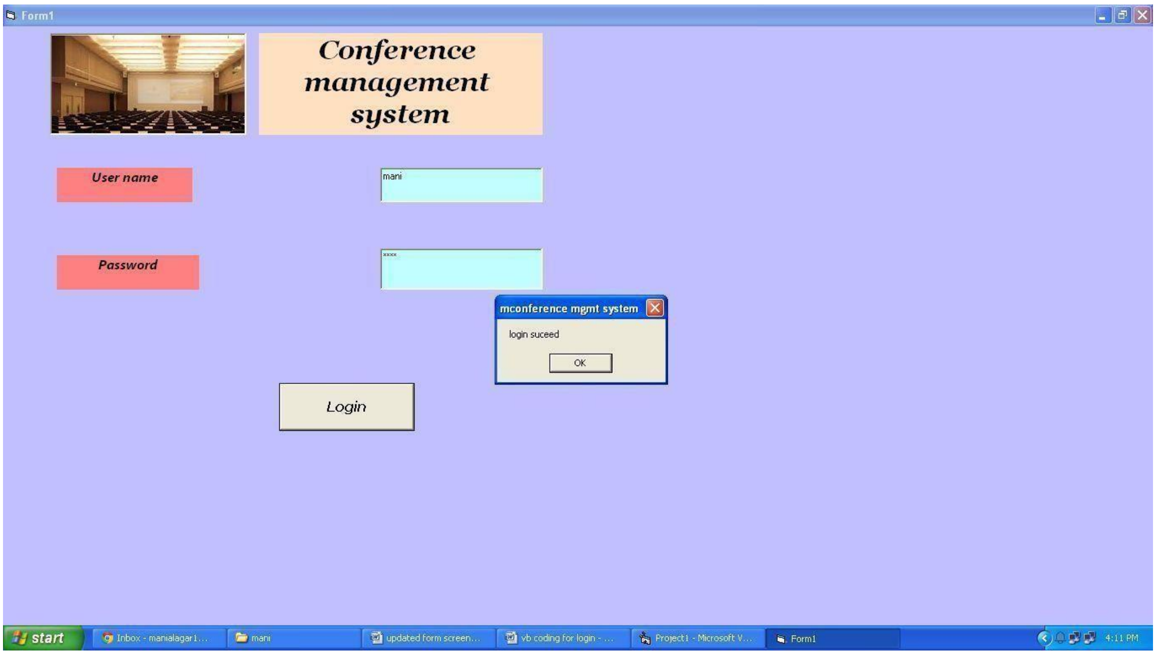
LOGIN:



(Name)	Form1
Appearance	1 - 3D
AutoRedraw	False
BackColor	&H00FFC0C0
BorderStyle	2 - Sizable
Caption	Form1
ClipControls	True
ControlBox	True
DrawMode	13 - Copy Pen
DrawStyle	0 - Solid
DrawWidth	1
Enabled	True
FillColor	&H0000000C
FillStyle	1 - Transparent
Font	MS Sans Seri
FontTransparent	True
ForeColor	&H8000012
HasDC	True
Height	11520

Font
Returns a Font object.

LOGIN VALIDATION:



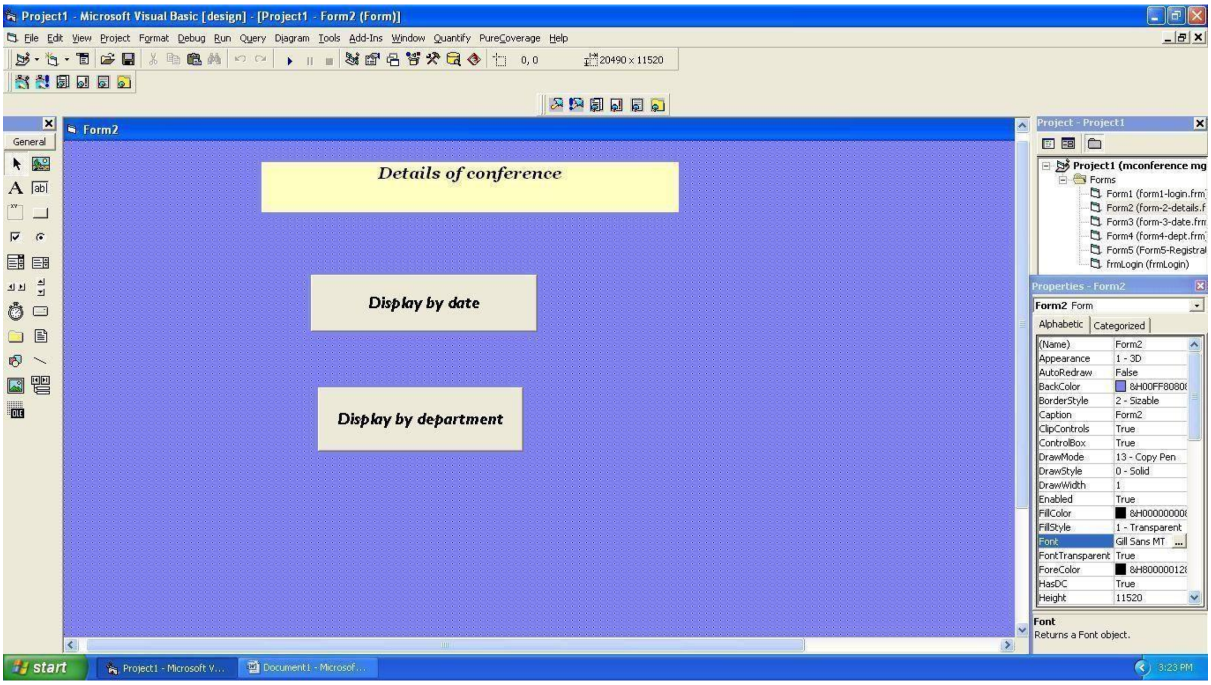
User name: mani

Password: ****

mconference mgmt system

login suceed

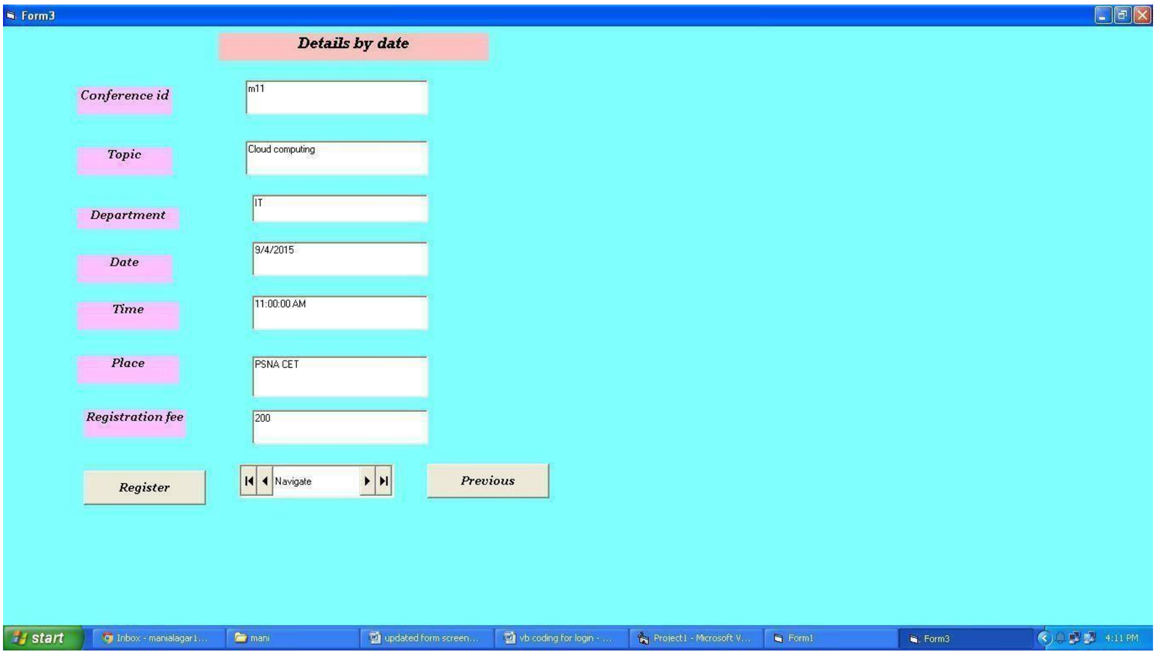
ENQUIRY OF CONFERENCE:



(Name)	Form2
Appearance	1 - 3D
AutoRedraw	False
BackColor	&H00FF8080(
BorderStyle	2 - Sizable
Caption	Form2
ClipControls	True
ControlBox	True
DrawMode	13 - Copy Pen
DrawStyle	0 - Solid
DrawWidth	1
Enabled	True
FillColor	&H00000000(
FillStyle	1 - Transparent
Font	Gill Sans MT
FontTransparent	True
ForeColor	&H80000012(
HasDC	True
Height	11520

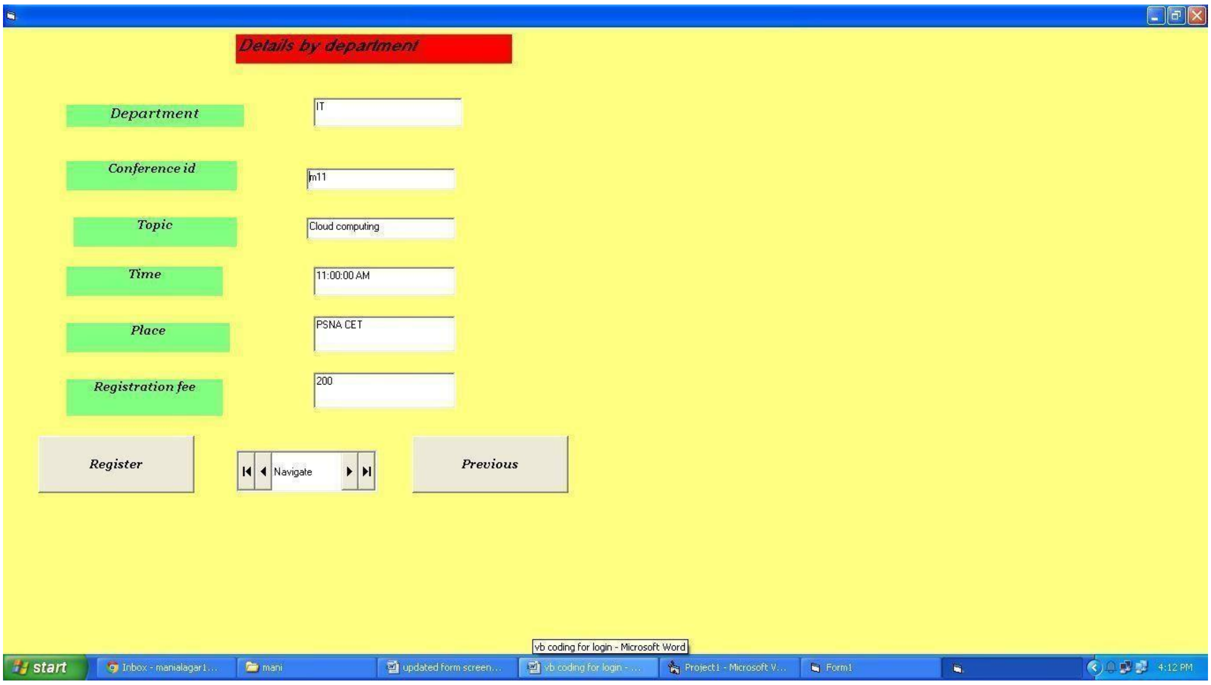
Font
Returns a Font object.

DETAILS BY DATE:



Conference id	m11
Topic	Cloud computing
Department	IT
Date	9/4/2015
Time	11:00:00 AM
Place	PSNA CET
Registration fee	200

DETAILS BY DEPARTMENT:



Details by department

Department: IT

Conference id: m11

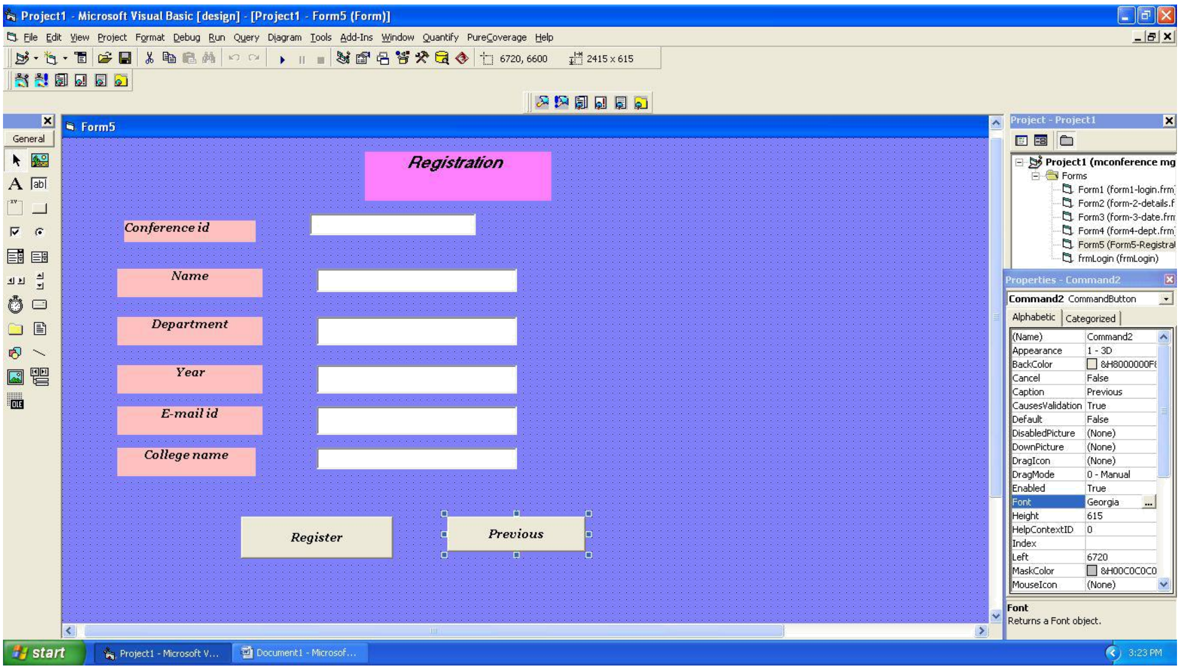
Topic: Cloud computing

Time: 11:00:00 AM

Place: PSNA CET

Registration fee: 200

REGISTRATION FORM:



Project1 - Microsoft Visual Basic [design] - [Project1 - Form5 (Form)]

6720, 6600

2415 x 615

Form5

Registration

Conference id

Name

Department

Year

E-mail id

College name

Project - Project1

Project1 (mconference mg

Forms

Form1 (form1-login.frm

Form2 (form-2-details.f

Form3 (form-3-date.frm

Form4 (form4-dept.frm

Form5 (Form5-Registra

frmLogin (frmLogin)

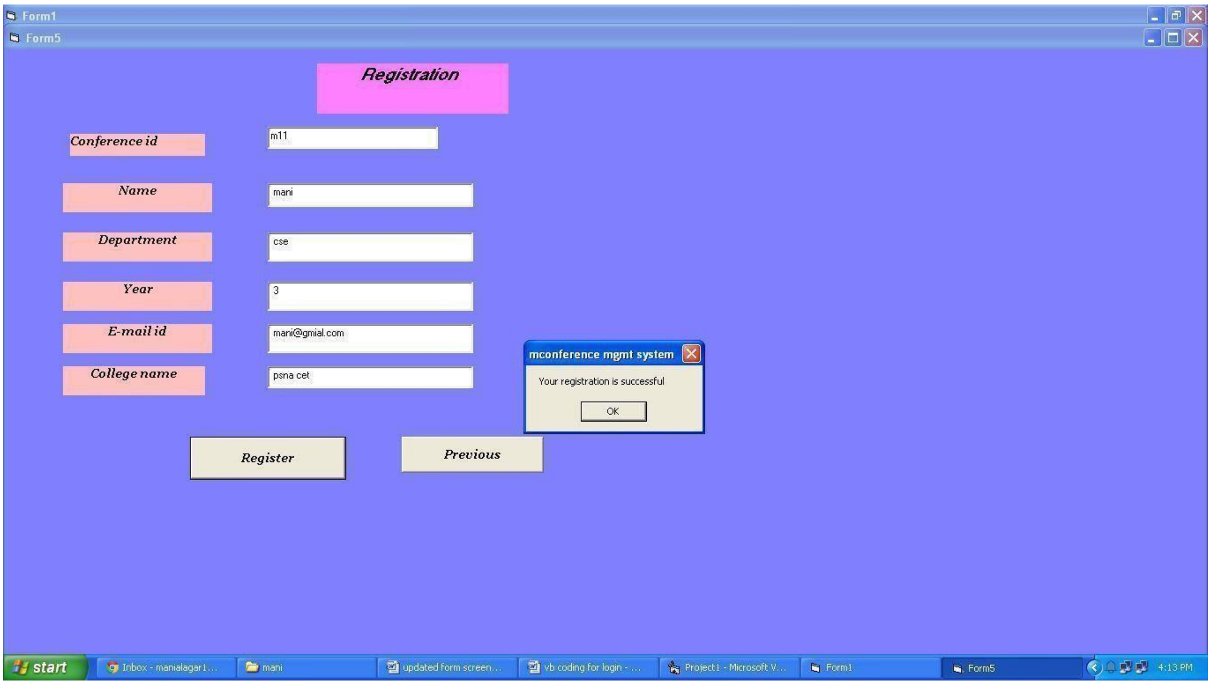
Properties - Command2

Command2 CommandButton

(Name) Command2
Appearance 1 - 3D
BackColor &H8000000F
Cancel False
Caption Previous
CausesValidation True
Default False
DisabledPicture (None)
DownPicture (None)
DragIcon (None)
DragMode 0 - Manual
Enabled True
Font Georgia
Height 615
HelpContextID 0
Index
Left 6720
MaskColor &H00C0C0C0
MouseIcon (None)

Font
Returns a Font object.

REGISTRATION FORMAFTER RESERVATION:



Conference id	m11
Name	mani
Department	cse
Year	3
E-mail id	mani@gmial.com
College name	psna cet

mconference mgmt system

Your registration is successful

FORM 1:

```
Dim b as New Class
Private Sub Command1_Click ()
Set b = New Class
login
End Sub

Private Sub
Form Load()
Data1.Visibl
e = False
End Sub
```

FORM2
ENQUIRY OF CONFERENCE:

```
Private Sub command1_click ()
Form3.Show
Unload Me
End Sub

Private Sub Command2_click ()
Form4.Show
Unload Me
End Sub
```

FORM3
DETAILS BY DATE:

```
Private Sub command2_click ()
Form5.Show
Unload Me
End Sub

Private Sub command1_click ()
Form2.Show
Unload Me
End Sub

Form 4Details by department:
Private Sub command1_click ()
Form5.Show
Unload Me
End Sub

Private Sub command2_click ()
Form2.Show
Unload Me
End Sub
```

FORM5
REGISTRATION FORM:

```
Private Sub command1_click ()
MsgBox (“your registration is successful”);
Form2.Show
Unload Me
End Sub
```

Administrator:

```
Option Explicit
'##ModelId=55A6283A0000
Private admin_id As Variant
'##ModelId=55A62842008C
Private password As Variant
'##ModelId=55A62B570138
Public NewPropertyAs User
'##ModelId=55A62B6003B9
Public NewProperty2 As registration
'##ModelId=55B89B210188
Public NewProperty3 As User
'##ModelId=55B8A0E703D8
Public NewProperty4 As registration
'##ModelId=55B8A1AB00EA
Public NewProperty5 As registration
'##ModelId=55B8A28A0242
Public NewProperty6 As registration
'##ModelId=55A6280051F4
```

```
Public Sub validate_id()
Do
If Form1.Data1.Recordset.EOF Then Form1.Data1.Recordset.MoveFirst
If (Form1.Text1.Text = Form1.Data1.Recordset.Fields (0)) and
(Form1.Text2.Text = Form1.Data1.Recordset.Fields (1)) Then
MsgBox ("login succeeds")
Form2.Show
Exit Do
Else
Form1.Data1.Recordset.MoveNext
End If
Loop Until Form1.Data1.Recordset.EOF
If Form1.Data1.Recordset.EOF Then
MsgBox ("invalid login")
End If
End Sub

'##ModelId=55A62859005D
Public Sub update_details()
End Sub

'##ModelId=55A6286002DE
Public Sub view_details()
End Sub

'##ModelId=55A6296900FA
Public Sub allocate ()
End Sub
```

Conference details:

```
Option Explicit
'##ModelId=55B8A6E8031C
Private facilities As Variant
'##ModelId=55B8A6EE0157
Private date_time As Variant
'##ModelId=55B8A6F60213
Private place As Variant
'##ModelId=560279BC0261
Private mdepartmentObject As New department
'##ModelId=55F92E340148
Public NewProperty As department
'##ModelId=55B8A6FC003E
Public Sub topic_details()
End Sub

'##ModelId=55B8A71D0271
Public Sub facilities2 ()
End Sub

'##ModelId=560279BC0271
```

```
Private Sub department_view_details_by_dept() Call
mdepartmentObject.view_details_by_dept
End Sub
```

Date:

```
Option Explicit
'##ModelId=55F92DF301E4
Private topic As Variant
'##ModelId=55F92DF6000F
Private place As Variant
'##ModelId=55F92DF803B9
Private registration_fee As Variant
'##ModelId=560279BC037A
Private menquireObject As New enquire
'##ModelId=55F92E1D036B
Public Sub view_details_by_date()
End Sub

'##ModelId=560279BC038A
Private Sub enquire_enqure_by_date()
Call menquireObject.enqure_by_date
End Sub

'##ModelId=560279BC038B
Private Sub enquire_enquire_by_dept()
Call menquireObject.enquire_by_dept
End Sub
```

Department:

```
'##ModelId=55F92DCE0232
Private date_time As Variant
'##ModelId=55F92DD603B9
Private place As Variant
'##ModelId=55F92DDA029F
Private registration_fee As Variant
'##ModelId=55F92E020148
Private topic As Variant
'##ModelId=560279BC01B5
Private menquireObject As New enquire
'##ModelId=55F92E0803A9
Public Sub view_details_by_dept()
End Sub

'##ModelId=560279BC01C5
Private Sub enquire_enqure_by_date()
Call menquireObject.enqure_by_date
End Sub

'##ModelId=560279BC01C6
Private Sub enquire_enquire_by_dept()
```

```
Call menquireObject.enquire_by_dept
End Sub
```

Enquire:

```
Option Explicit
'##ModelId=55B8A66B035B
Private name As Variant
'##ModelId=55B8A66F034B
Private department As Variant
'##ModelId=55B8A677009C
Public Sub enqure_by_date()
End Sub

'##ModelId=55B8A683037A
Public Sub enquire_by_dept()
End Sub
```

Payment:

```
Option Explicit
'##ModelId=55B8A5E000CB
Private amount As Variant
'##ModelId=55B8A5E500AB
Private concession As Variant
'##ModelId=55B8A5F5035B
Public Sub paymentinfo()
End Sub

'##ModelId=55B8A624030D
Public Sub receipt()
End Sub
```

Registerby users:

```
Option Explicit
'##ModelId=55B8A51E03B9
Private name As Variant
'##ModelId=55B8A5220138
Private mail_id As Variant
'##ModelId=55B8A5A60177
Public NewProperty As registration'##ModelId=55B8A52803C8
Public Sub give_name()
End Sub

'##ModelId=55B8A5480167
Public Sub give_mail_id()
End Sub
```

Registration:

```
Option Explicit
'##ModelId=55A6288200FA
Private name As Variant
'##ModelId=55A6288B038A
Private mail_id As Variant
'##ModelId=55B89077002E
Private dept As Variant
'##ModelId=55B8909B0177
Private amount As Variant
'##ModelId=560279BA032C

Private mAdministratorObject As New
Administrator
'##ModelId=55B8A1CA0167 Public
NewPropertyAs User
'##ModelId=55B8A1D803D8
Public NewProperty2 As User
'##ModelId=55B8A1E20167
Public NewProperty3 As User
'##ModelId=55B8A1F2003E
Public NewProperty4 As User
'##ModelId=55B8A1FF02EE
Public NewProperty5 As User
'##ModelId=55B8A28A0244
Public NewProperty6 As Administrator
'##ModelId=55B8A5C3007D
Public NewProperty7 As register_by_users
'##ModelId=55B8A6300157
Public NewProperty8 As payment
'##ModelId=55B8A7500261
Public NewProperty9 As conf_details
'##ModelId=55B8A7680290
Public NewProperty10 As User
'##ModelId=55B8A77901E6
Public NewProperty11 As User
'##ModelId=55A628A100BB
Public Sub register_details()
End Sub

'##ModelId=55A628A60399
Public Sub payment_info()
End Sub

'##ModelId=55B890DB002E
Public Sub conference_info()
End Sub

'##ModelId=560279BA034B
Private Sub Administrator_validate_id()
```

```
Call mAdministratorObject.validate_id
End Sub

'##ModelId=560279BA035B
Private Sub Administrator_update_details()
Call mAdministratorObject.update_details
End Sub

'##ModelId=560279BA035C
Private Sub Administrator_view_details()
Call mAdministratorObject.view_details
End Sub

'##ModelId=560279BA036B
Private Sub Administrator_allocate()
Call mAdministratorObject.allocate
End Sub

'##ModelId=560279BA036C
Private Property Set Administrator_ (ByVal RHS As registration)
 Set mAdministratorObject. = RHS
End Property

'##ModelId=560279BA0399
Private Property Get Administrator_ () As registration Set Administrator_ =
mAdministratorObject.
End Property
```

Users:

```
Option Explicit

'##ModelId=55A627D3029F
Private username As Variant
'##ModelId=55A627DD02AF
Private password As Variant
'##ModelId=55A627E503A9
Private department As Variant
'##ModelId=560279BB02FD
Private mAdministratorObject As New
Administrator
'##ModelId=55A62AFB002E Public
NewProperty As registration
'##ModelId=55A62B3000AB
Public NewProperty2 As registration
'##ModelId=55A62B37033C
Public NewProperty3 As Administrator
'##ModelId=55B89B210186
Public NewProperty4 As Administrator
'##ModelId=55B8A6910261
Public NewProperty5 As enquire
```

```
'##ModelId=55B8A72C03D8
Public NewProperty6 As conf_details
'##ModelId=55B8A7720186
Public NewProperty7 As registration
'##ModelId=55B8A77901E4
Public NewProperty8 As registration
'##ModelId=55A62804001F
Public Sub enquires ()
End Sub

'##ModelId=55A6282B01E4
Public Sub view_details()
End Sub

'##ModelId=55A6295102DE
Public Sub reserve ()
End Sub

'##ModelId=55F937D5031C
Public Sub login ()
Do
If Form1.Data1.Recordset.EOF Then Form1.Data1.Recordset.MoveFirst
If (Form1.Text1.Text = Form1.Data1.Recordset.Fields (0)) And (Form1.Text2.Text =
Form1.Data1.Recordset.Fields (1)) Then
MsgBox ("login suceed")
Form2.Show
Exit Do
Else
Form1.Data1.Recordset.MoveNext
End If
Loop Until Form1.Data1.Recordset.EOF
If Form1.Data1.Recordset.EOF Then
MsgBox ("invalid login")
End If
End Sub

„##ModelId=560279BB030D
Private Sub Administrator_validate_id()
Call mAdministratorObject.validate_id
End Sub

'##ModelId=560279BB031C
Private Sub Administrator_update_details()
Call mAdministratorObject.update_details
End Sub

'##ModelId=560279BB031D
Private Sub Administrator_view_details()
Call mAdministratorObject.view_details
End Sub
```

```
'##ModelId=560279BB032C
Private Sub Administrator_allocate()
Call mAdministratorObject.allocate
End Sub

'##ModelId=560279BB032D
Private Property Set Administrator_NewProperty(ByVal RHS As User)
Set mAdministratorObject.NewProperty = RHS
End Property

'##ModelId=560279BB034B
Private Property Get Administrator_NewProperty() As User
Set Administrator_NewProperty = mAdministratorObject.NewProperty End Property
'##ModelId=560279BB035C
Private Property Set Administrator_NewProperty2 (ByVal RHS As registration) Set
mAdministratorObject.NewProperty2 = RHS End Property
'##ModelId=560279BB037A
Privae Property Get Administrator_NewProperty2 () As registration
Set Administrator_NewProperty2 = mAdministratorObject.NewProperty2
End Property

'##ModelId=560279BB038B
Private Property Set Administrator_NewProperty3 (ByVal RHS As User)
Set mAdministratorObject.NewProperty3 = RHS
End Property

'##ModelId=560279BB03B9
Private Property Get Administrator_NewProperty3 () As User
Set Administrator_NewProperty3 = mAdministratorObject.NewProperty3
End Property

'##ModelId=560279BB03BB
Private Property Set Administrator_NewProperty4 (ByVal RHS As registration) Set
mAdministratorObject.NewProperty4 = RHS End Property
'##ModelId=560279BC0000
Private Property Get Administrator_NewProperty4 () As registration
Set Administrator_NewProperty4 = mAdministratorObject.NewProperty4
End Property

'##ModelId=560279BC000F
Private Property Set Administrator_NewProperty5 (ByVal RHS As registration) Set
mAdministratorObject.NewProperty5 = RHS
End Property

'##ModelId=560279BC002E
Private Property Get Administrator_NewProperty5 () As registration
Set Administrator_NewProperty5 = mAdministratorObject.NewProperty5
End Property
```

```
'##ModelId=560279BC003F
Private Property Set Administrator_NewProperty6 (ByVal RHS As registration) Set
mAdministratorObject.NewProperty6 = RHS End Property
'##ModelId=560279BC006D
Private Property Get Administrator_NewProperty6 () As registration
Set Administrator_NewProperty6 = mAdministratorObject.NewProperty6 End
Property
```

CONCLUSION:

Thus the project for Conference Management System was designed and codes are generated and then it was executed successfully.

Ex. No: 13
Date :

BPO MANAGEMENT SYSTEM

AIM

To develop a project Business process out sourcing (BPO) management system Using Agro UML software and to implement the software in Visual Basic.

PROBLEM ANALYSIS AND PROJECT PLANNING

Generally outsourcing can be defined as an organization entering into a contract with another organization to operate and managed one or more of its business processes.
There are many problems faced by the BPO one among them is meeting their targets and leaving the concern very often and switch to another company. In this project we deal with the inbound system of the BPO. In inbound system the agent calls the customer from his database to sell his product.

PROBLEM STATEMENT

In this BPO inbound system, the process undergoing is that the agent tries to sell his product so that the agent gets the details of the customer from the database and pitches about his product and makes the sales successful. The communication is done through the telephone. Telephone is the major component used for this customer satisfaction service. The steps are as follows:

- The agent login to the website and enters the username and password .It checks for authorization.
- If the username and password is correct, it allows the agent to get the details of the customer from the database.
- Now the agent makes the call to the customer and pitches about the product.
- If the customer is satisfied, agent sells the product else disconnects the call.
- Agent proceeds with the another call.

SOFTWARE REQUIREMENT SPECIFICATION

S.NO	CONTENTS
1.	INTRODUCTION
2.	OBJECTIVE
3.	OVERVIEW
4.	GLOSSARY
5.	PURPOSE
6.	SCOPE
7.	FUNCTIONALITY
8.	USABILITY
9.	PERFORMANCE
10.	RELIABILITY
11.	FUNCTIONAL REQUIREMENTS
12.	EXTERNAL INTERFACE REQUREMENTS

1. INTRODUCTION

BPO is typically categorized into back office outsourcing-which includes internal business functions such as human resources or finance and accounting, and front office out souring -which includes customer related services such as contact centre services. BPO that is contracted outside a company‟s country is called offshore outsourcing. BPO that is contracted to a company‟s neighbouring country is called near shore out sourcing .Given the proximity of BPO to the information technology industry, it is categorized as an information technology enabled service or ITES. Knowledge process out sourcing (KPO) and legal process out sourcing (LPO) are some of the sub-segments of business process outsourcing industry. In the following SRS the front office outsourcing is explained in detail.

2. PURPOSE

The purpose of this system is to provide information about the customer need from inside and outside world. With the reduction in communication costs and improved bandwidths and associated infrastructure, BPO as a segment is witnessing massive growth. One of the key challenges that BPO companies is that to provide data entry/data validation services is an efficient and effective way of getting the source documents from different customers and accurately route the same of different operators for processing.

3. SCOPE

Developing a good BPO management system. BPO is a way in which it helps to increase company flexibility. As part of BPO, documents need to be managed between the outsourcing company and the offshore company. Multiple clients need to be managed by the BPO companies.

4. GLOSSARY

TERMS	DESCRIPTION
CUSTOMER	Person who is seeking information.
AGENT	People who receives the query.
DATABASE	Collection of all information monitored by the BPO system.
READER	Anyone visiting the site to read about Member BPO management system.
SOFTWARE REQUIREMENT SPECIFICATION	This software specification documents full set of features and function for online recruitment system that is performed in company website.

7. REFERENCES

Business process out sourcing the competitive advantage by Rick L. Click, Thomas N.Duening-2005 .Sir document is referred from the standard IEEE format from fundamentals of software engineering by Raj Mall (2004) page no: 356

8. FUNCTIONALITY

Many customers of the process to check for its occurrences and other works .we all have to carry over at same time.

9. USABILITY

The user interface to make the BPO management to be efficient.

10. PERFORMANCE

It is the capability about which it can perform function for many user efficiently at the same time without any error occurrence.

11. SYSTEM ENVIRONMENT

The BPO system is embedded in a larger system involving several management systems .we describe this environment as communication system between customer and agent through voice chat .the administrator of the system uses FTP for moving files from one place to another.

12. FUNCTIONAL REQUIREMENTS

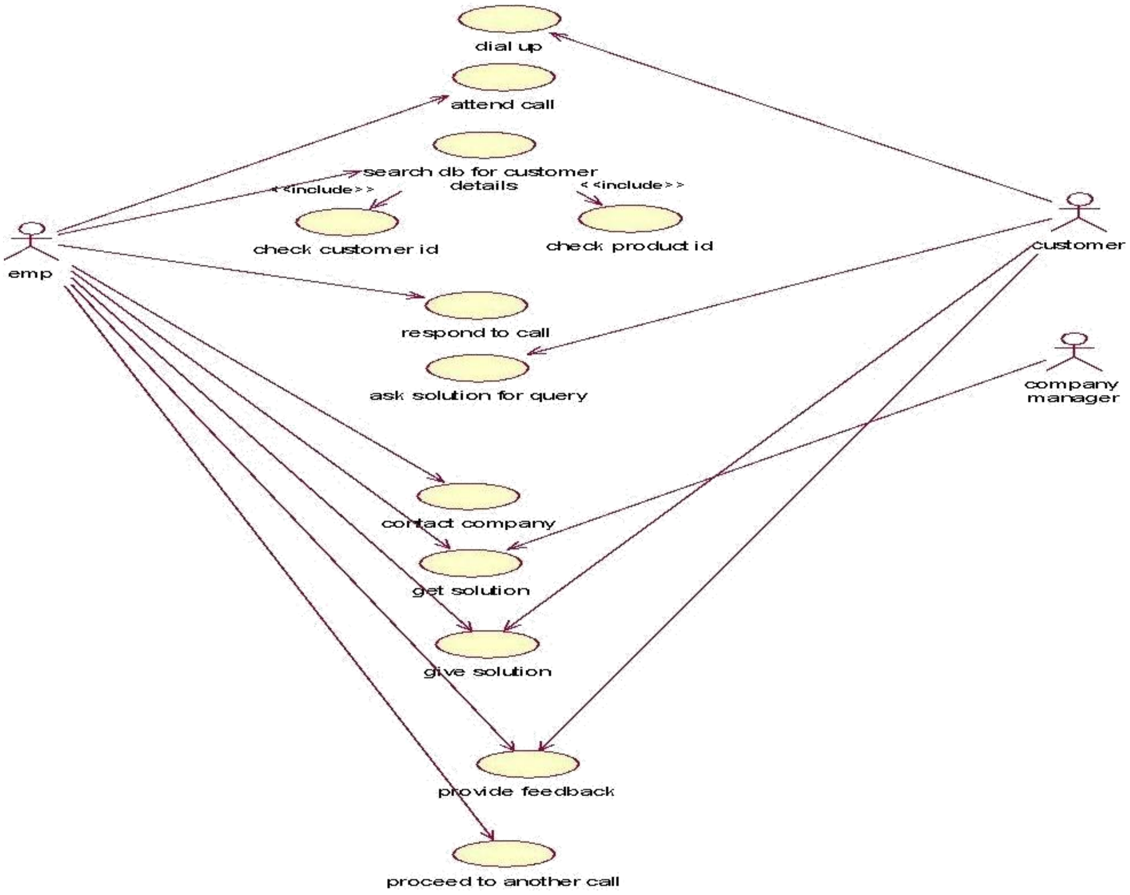
Functional requirements are those refer to the functionality of the system. i.e. what services it will provide to the user. Non functional (supplementary) requirements pertain to other information needed to produce the system correctly and detailed separately.

UML DIAGRAMS:

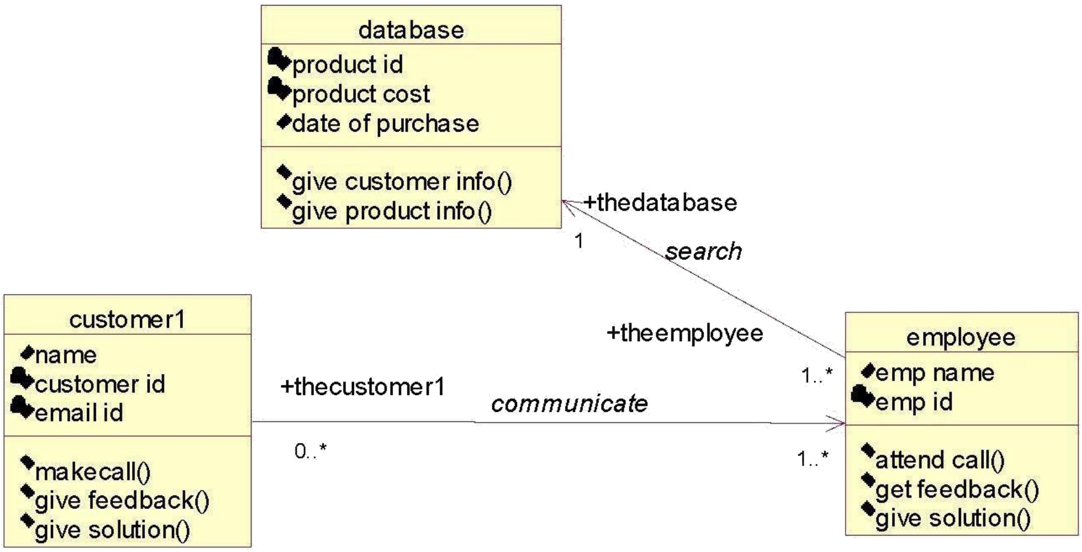
The following UML diagrams describe the process involved in the online recruitment system

a. Use case diagram
b. Class diagram
c. Sequence diagram
d. Collaboration diagram
e. State chart diagram
f. Activity diagram
g. Component diagram
h. Deployment diagram
i. Package diagram

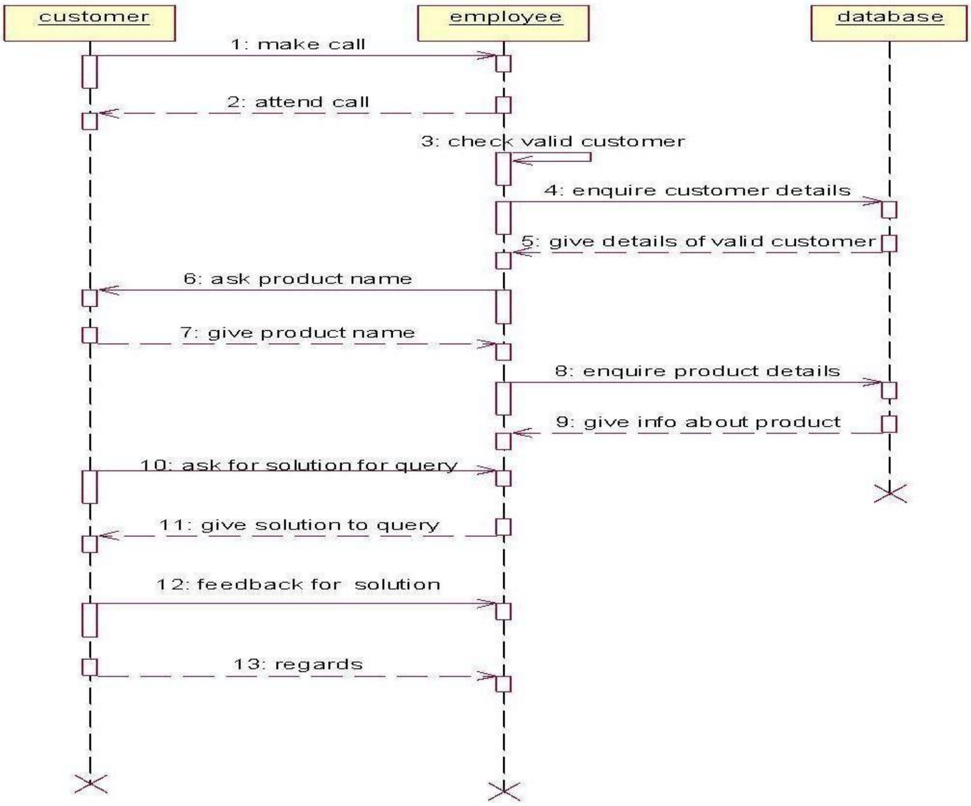
USE CASE DIAGRAM



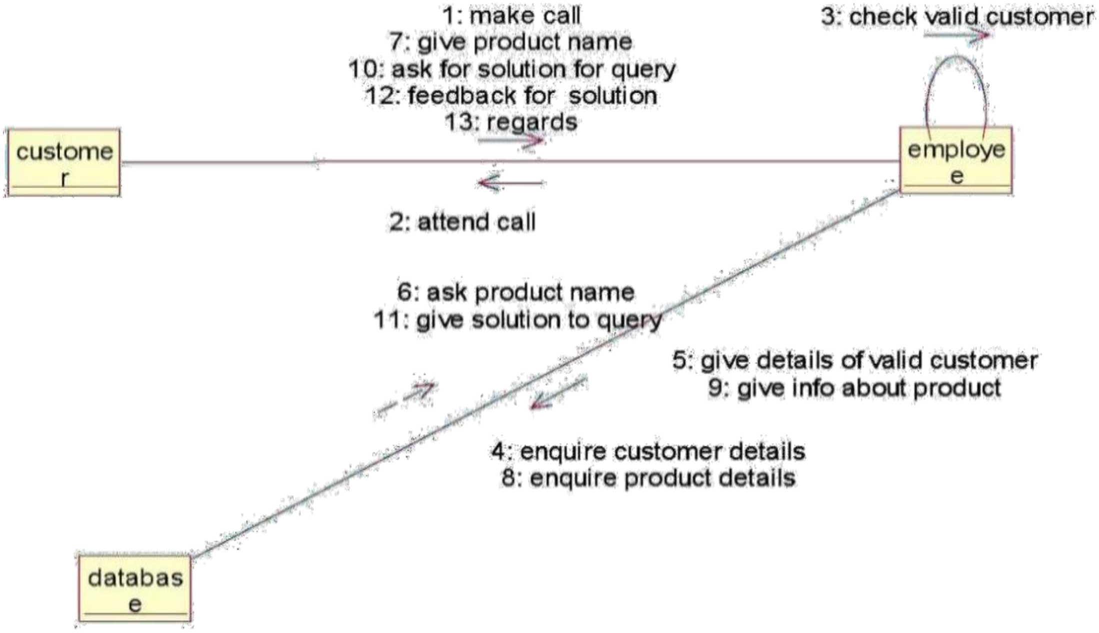
CLASS DIAGRAM



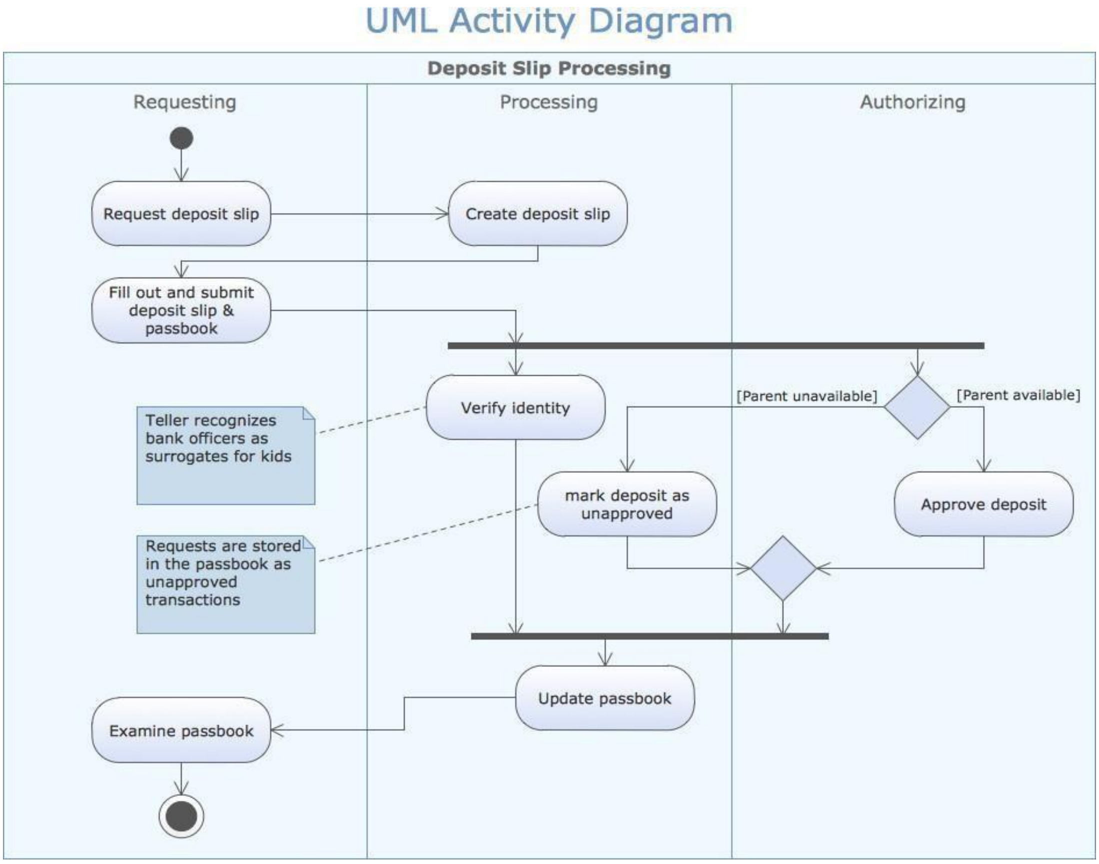
SEQUENCE DIAGRAM


customer
employee
database
1: make call
2: attend call
3: check valid customer
4: enquire customer details
5: give details of valid customer
6: ask product name
7: give product name
8: enquire product details
9: give info about product
10: ask for solution for query
11: give solution to query
12: feedback for solution
13: regards

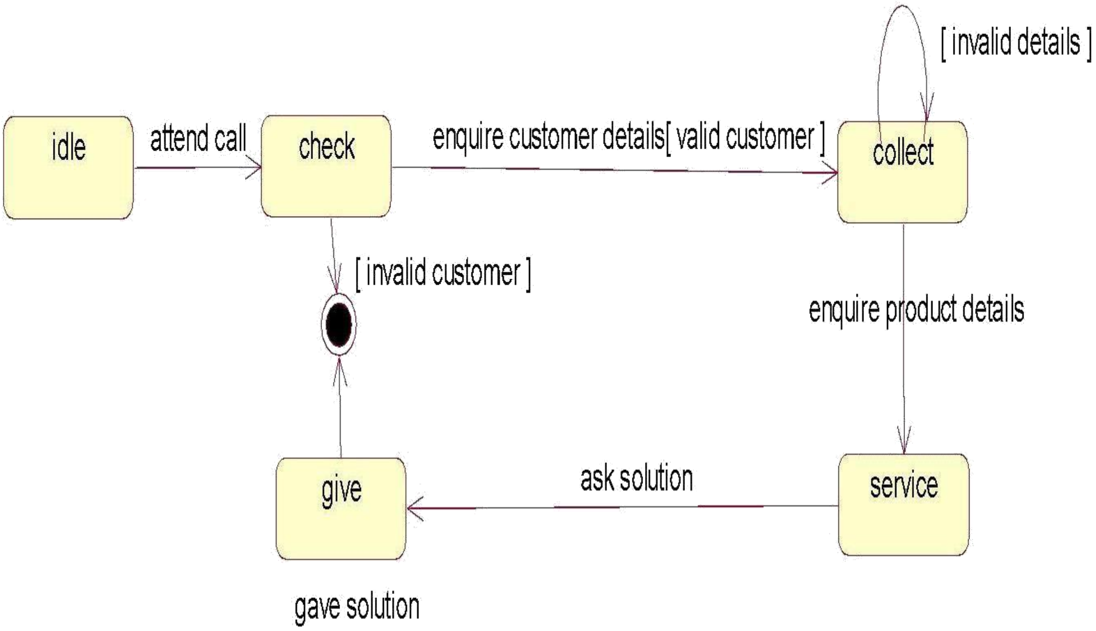
COLLABORATION DIAGRAM


1: make call
7: give product name
10: ask for solution for query
12: feedback for solution
13: regards
3: check valid customer
custome r
employe e
2: attend call
6: ask product name
11: give solution to query
5: give details of valid customer
9: give info about product
4: enquire customer details
8: enquire product details
databas e

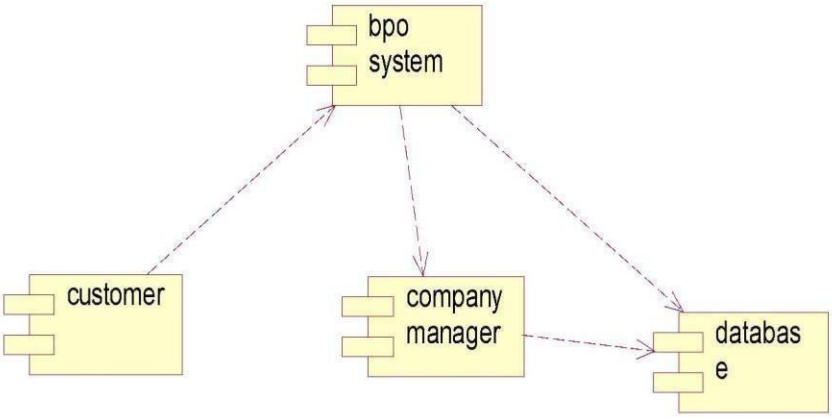
ACTIVITY DIAGRAM



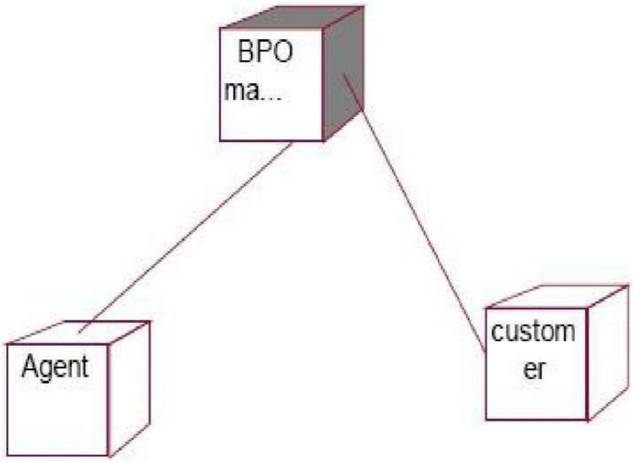
STATE CHART DIAGRAM



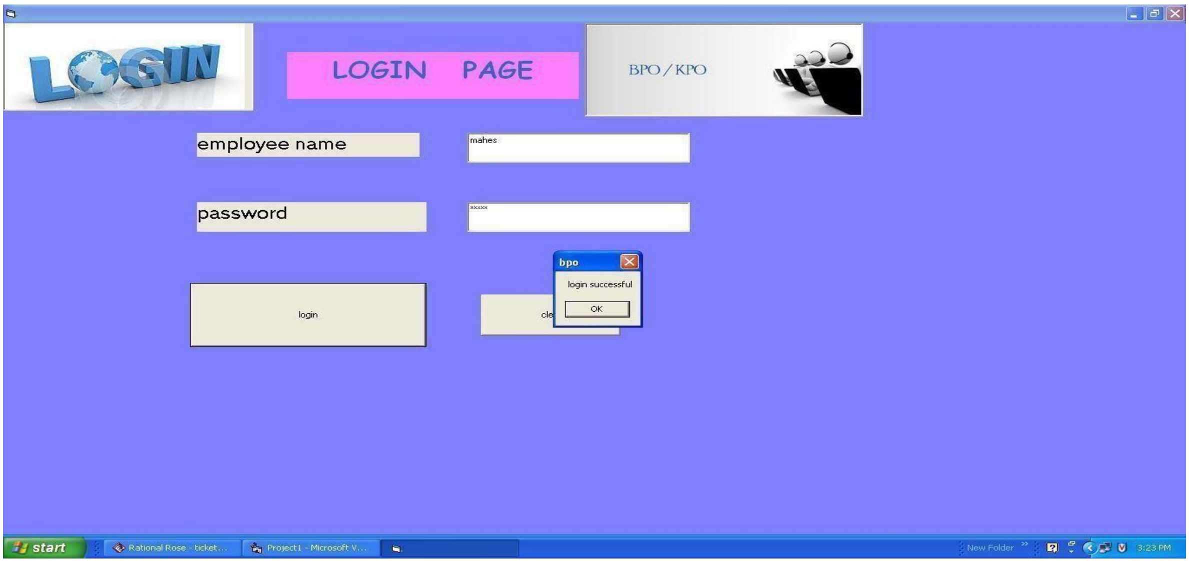
COMPONENT DIAGRAM



DEPLOYMENT DIAGRAM



FORM 1:



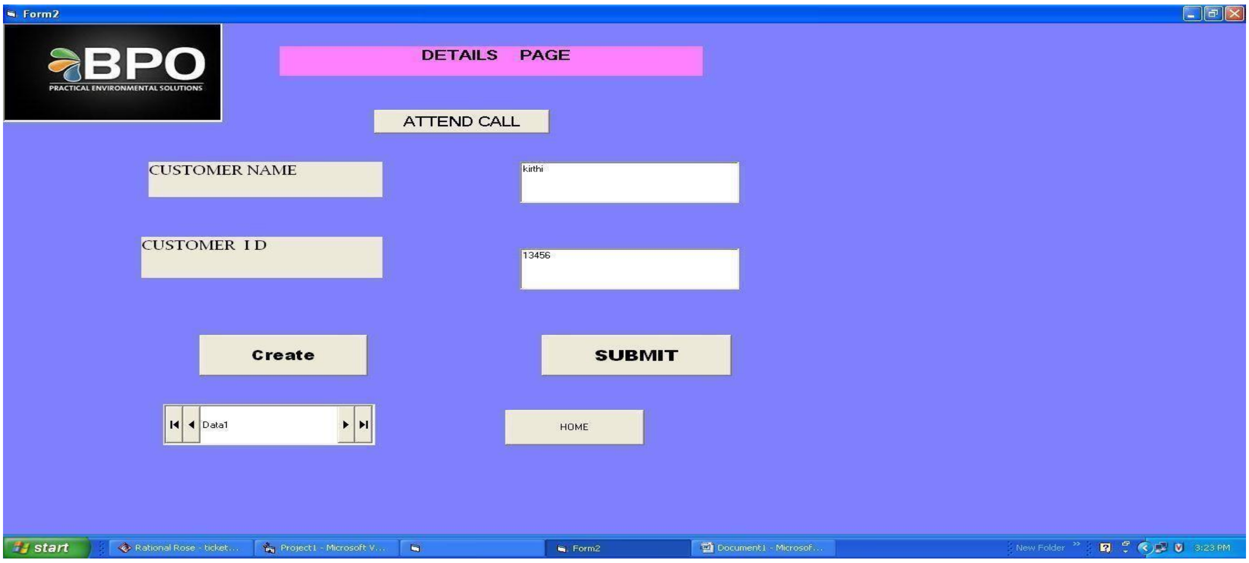
employee name: mahes

password: *****

bpo

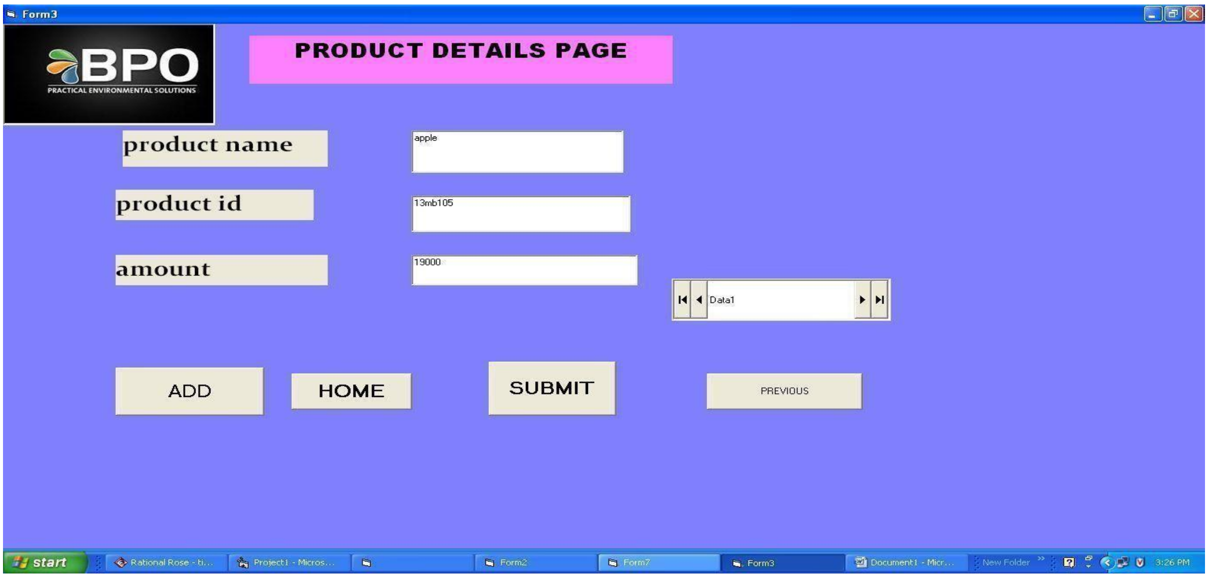
login successful

FORM 2:



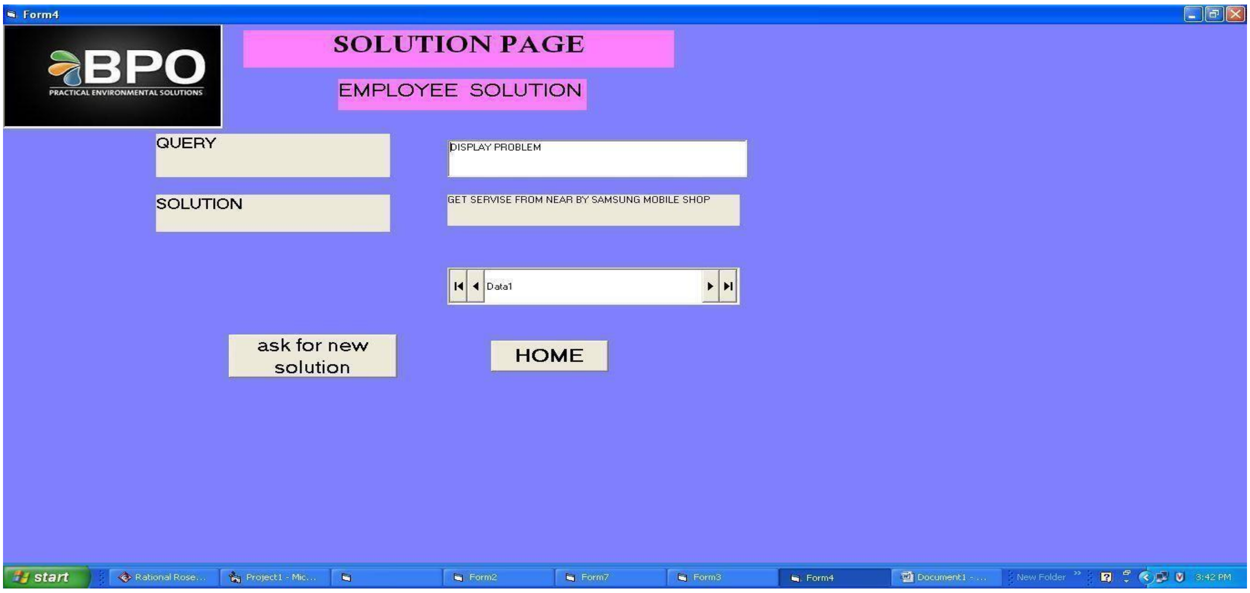
CUSTOMER NAME	kirthi
CUSTOMER I D	13456

FORM 3:



product name	apple
product id	13mb105
amount	19000

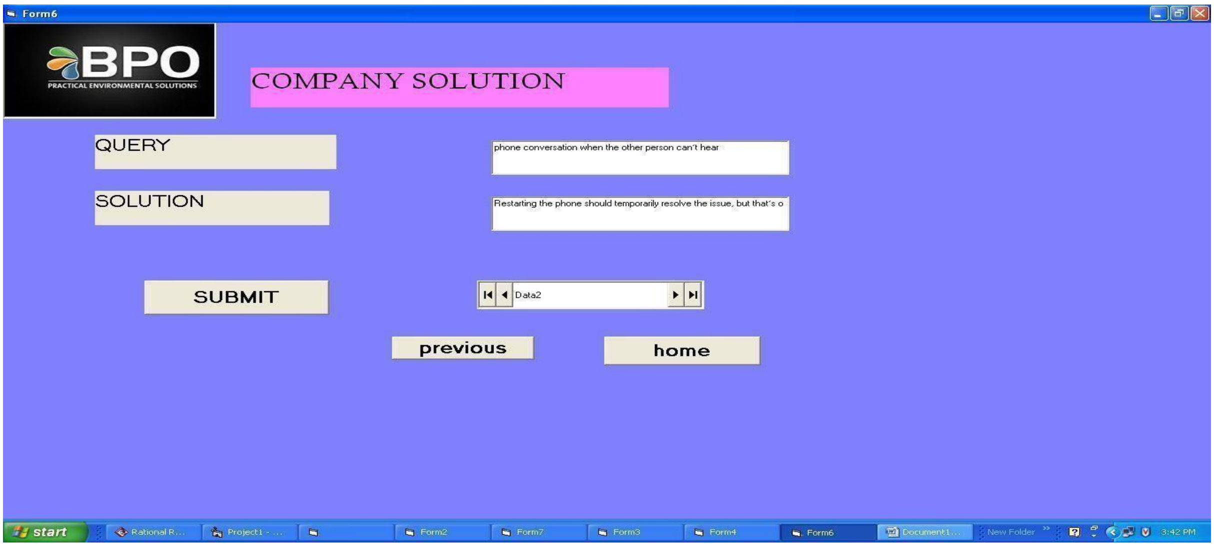
FORM 4:



QUERY: DISPLAY PROBLEM

SOLUTION: GET SERVISE FROM NEAR BY SAMSUNG MOBILE SHOP

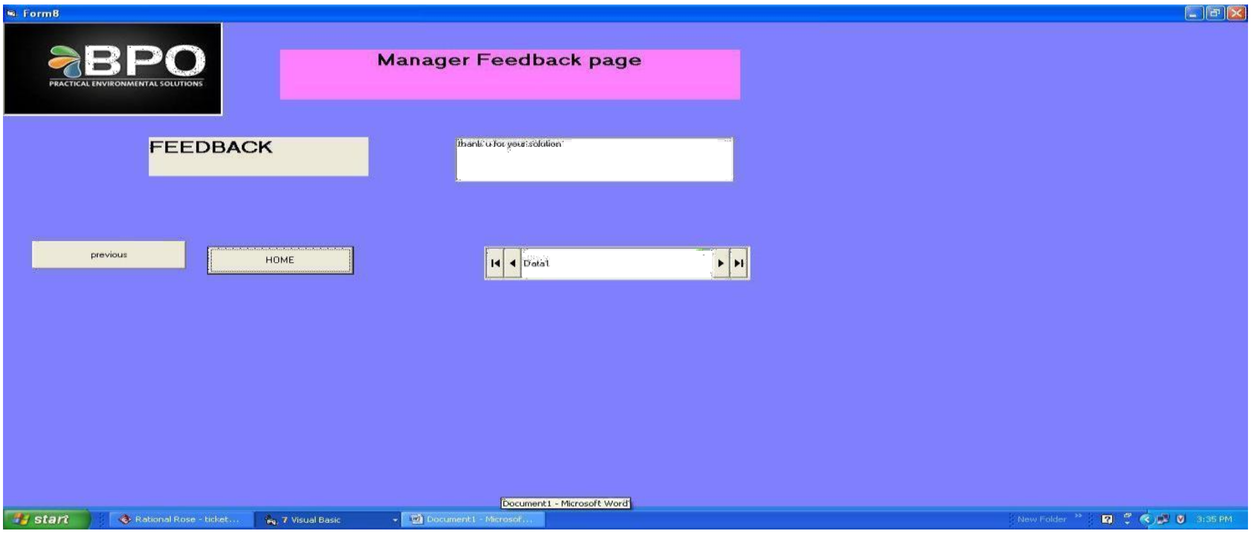
FORM 5:



QUERY: phone conversation when the other person can't hear

SOLUTION: Restarting the phone should temporarily resolve the issue, but that's o

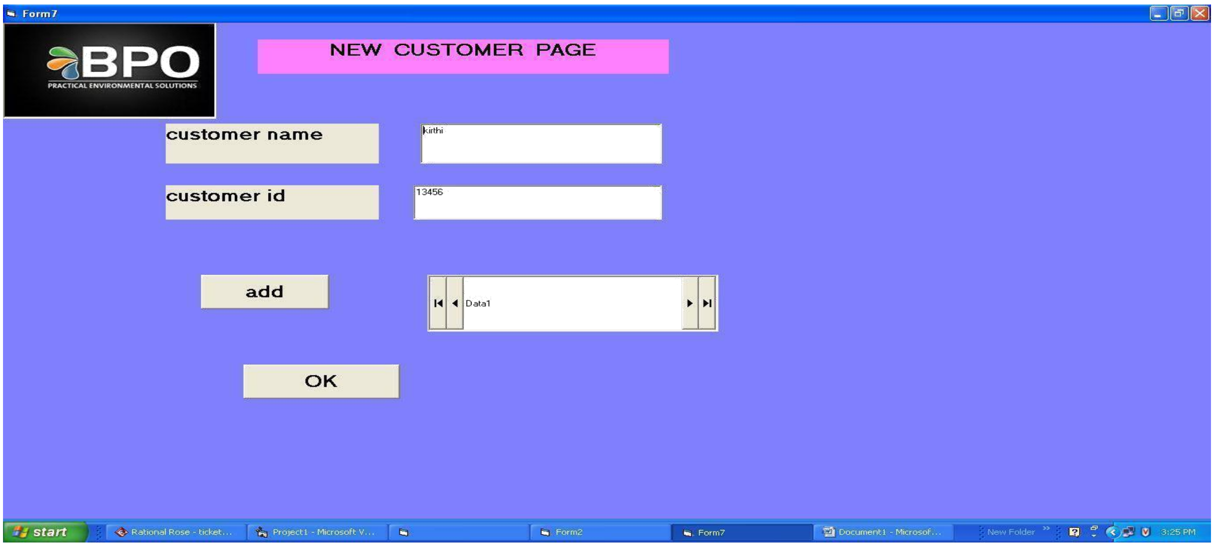
FORM 6:


Form8
BPO
PRACTICAL ENVIRONMENTAL SOLUTIONS
previous
HOME
Data1
Document1 - Microsoft Word
start
Rational Rose - ticket...
7 Visual Basic
Document1 - Microsof...
New Folder
3:35 PM

Manager Feedback page

FEEDBACK: thank u for your solution

FORM 7:

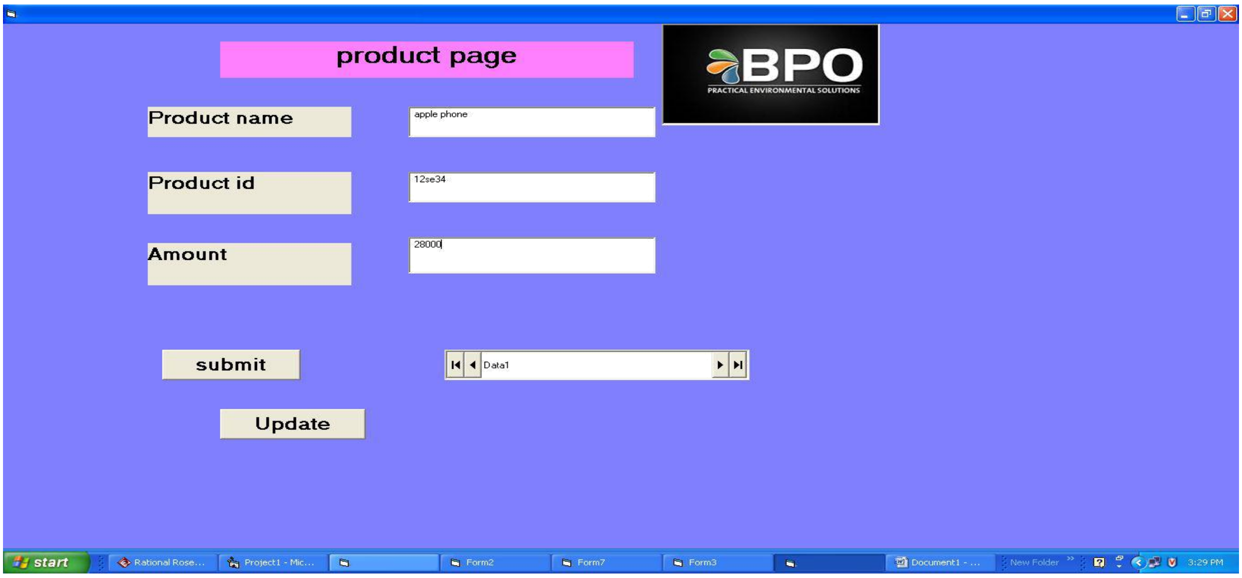

Form7
BPO
PRACTICAL ENVIRONMENTAL SOLUTIONS
add
Data1
OK
start
Rational Rose - ticket...
Project1 - Microsoft V...
Form2
Form7
Document1 - Microsof...
New Folder
3:25 PM

NEW CUSTOMER PAGE

customer name: kirthi

customer id: 13456

FORM 8:



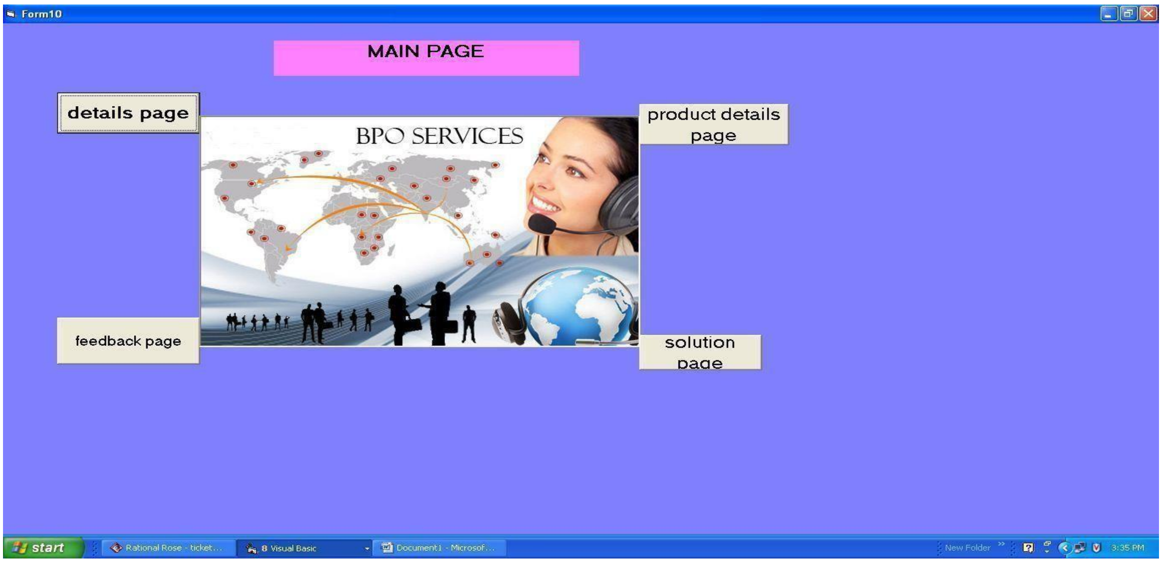
product page

Product name: apple phone

Product id: 12se34

Amount: 28000

FORM 9:



Form10

MAIN PAGE

PROGRAM:
FORM 1:

```
Private Sub Command1_Click ()
Dim count As Integer
For count = 0 To 2
If Data1.Recordset.EOF = False Then
If Data1.Recordset.Fields (0) = Text1.Text and Data1.Recordset.Fields (1) =
Text2.Text Then
MsgBox ("login successful")
Form2.Show
Exit For
'#unload Me
Else
Msg Box ("invalid login")
Exit For
End If
End If
Next count
End Sub

Private Sub Command2_Click()
Text1.Text = ""
Text2.Text = ""
End Sub

Private Sub Form _Load ()
Data1.Visible = Falls
End Sub

Private Sub Command1_Click ()
Msg Box "call attended"
End Sub
```

FORM 2:

```
Private Sub Command2_Click ()
Form10.Show
End Sub

Private Sub Command3_Click ()
Form3.Show
End Sub

Private Sub Command4_Click ()
Form8.Show
End Sub
```

FORM 3:

```
Sub Command1_Click ()
        Form9.Show
        End Sub

        Private Sub Command3_Click ()
        Form10.Show
        End Sub

        Private
        Private Sub Command4_Click ()
        Form4.Show
        End Sub

        Private Sub Command5_Click ()
        Form2.Show
        End Sub
```

FORM 4:

```
        Private Sub Command1_Click()
        Form10.Show
        End Sub

        Private Sub Command2_Click()
        Msg Box "check in company solution"
        Form6.Show
        End Sub

        Dim a As customer1
        Private Sub Command1_Click ()
        Msg Box "call ended"
        End Sub

        Private Sub Command2_Click ()
        Msg Box "feedback is saved in your database"
        Form7.Show
        End Sub

        Private Sub Command3_Click ()
        Set a = New customer1
        a. give feedback
        End Sub

        Private Sub Command4_Click ()
        Form10.Show
        End Sub

        Private Sub Command6_Click()
        Form4.Show
        End Sub
```

FORM 5:

```
Private
Sub Command1_Click()
Form5.Show
End Sub

Private Sub Command2_Click()
Form10.Show
End Sub

Private Sub Command3_Click()
Form4.Show
End Sub
```

FORM 6:

```
Private Sub Command1_Click()
Form5.Show
End Sub

Private Sub Command2_Click()
Form10.Show
End Sub
```

FROM 7:

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub

Private Sub Command2_Click()
Form3.Show
End Sub
```

FORM 8:

```
Private Sub Command1_Click()
Data1.Recordset.AddNew
End Sub

Private Sub Command2_Click()
Form5.Show
End Sub
```

FORM 9:

```
Private Sub Command1_Click()
Form2.Show
End Sub

Private Sub Command2_Click()
Form3.Show
End Sub

Private Sub Command3_Click()
Form4.Show
End Sub

Private Sub Command4_Click()
Form5.Show
End Sub
```

CONCLUSION:

Thus the project for BPO managementSystem was designed and codes are generated and then it was executed successfully.

Ex.No.: 14
Date :

LIBRARY MANAGEMENT SYSTEM

AIM

To analyse, design and develop code for Library Management System using Agro UML software.

PROBLEM STATEMENT

The case study titled Library Management System is library management software for the purpose of monitoring and controlling the transactions in a library. This case study on the library management system gives us the complete information about the library and the daily transactions done in a Library. We need to maintain the record of new s and retrieve the details of books available in the library which mainly focuses on basic operations in a library like adding new member, new books, and up new information, searching books and members and facility to borrow and return books. It features a familiar and well thought out, an attractive user interface, combined with strong searching, insertion and reporting capabilities. The report generation facility of library system helps to get a good idea of which are this borrowed by the members, makes users possible to generate hard copy.

OVERALL DESCRIPTION

The Online Course Reservation System is an integrated system that has four modules as part of it. The four modules are,

1) **Login for Student:** Using this module student login to the system using his/her unique user name and password.
2) **Search a book:** The details are stored in book table in database by giving the details about the selecting books
3) **Formfor Registration:** In this module the new user can apply for his/her library membership
4) **ReturnBook:**In this module the user can return the book.
5) **Renewal Book:** In this module the user can renewal the book before the due date is exceed otherwise paying the fine.

SOFTWARE REQUIRMENTS

- Microsoft Visual Basic 6.0
- Agro UML
- Microsoft Access

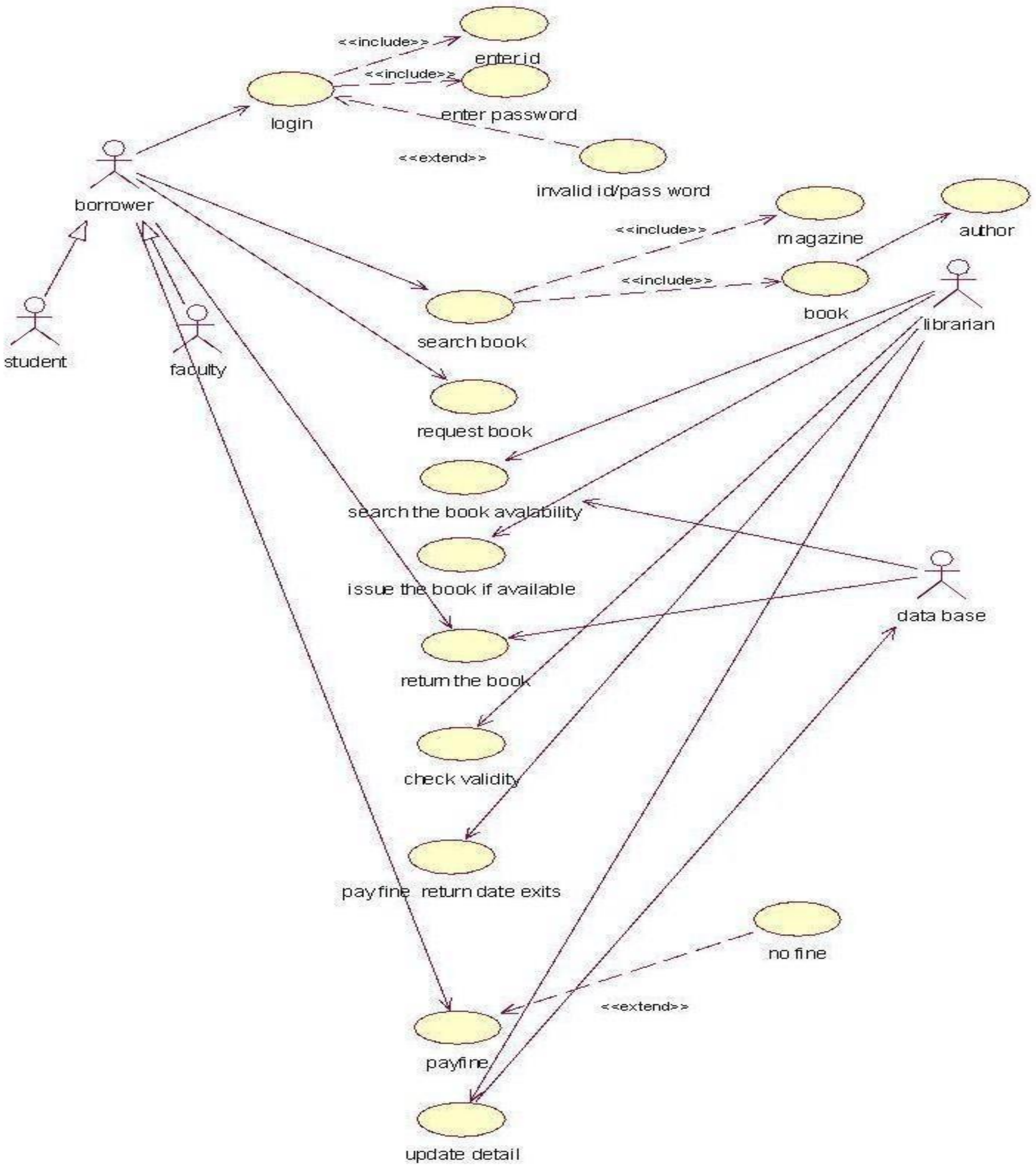
HARDWARE REQUIRMENTS

- 128MB RAM
- Pentium III Processor

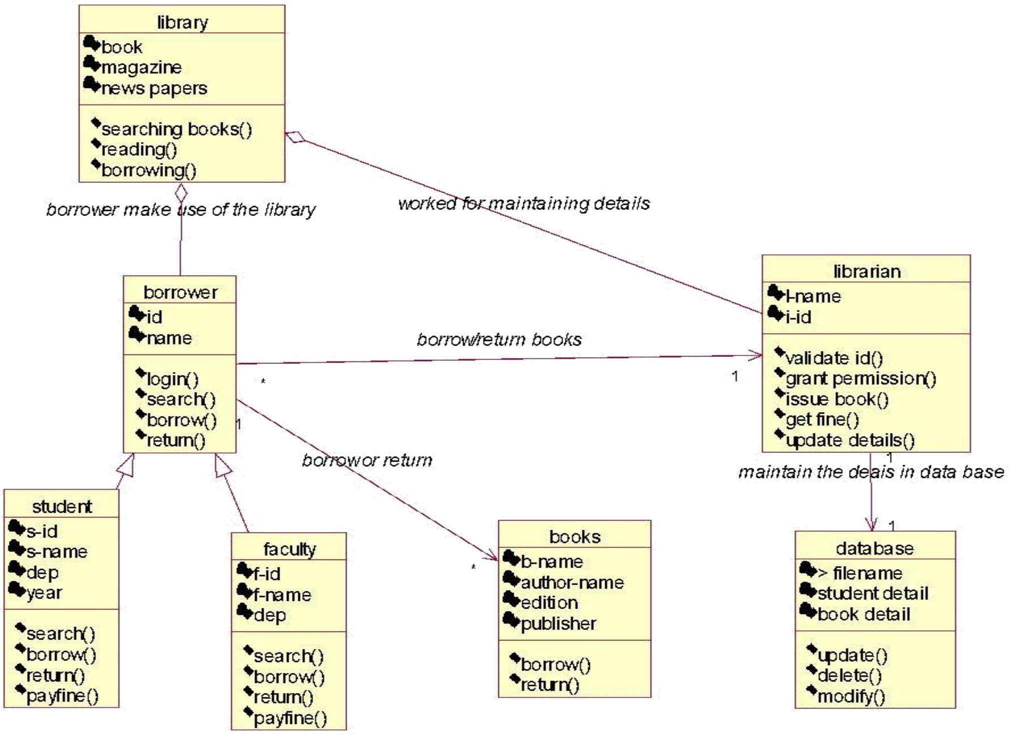
ANALYSIS MODELING

The project can be explained diagrammatically using the following diagrams:

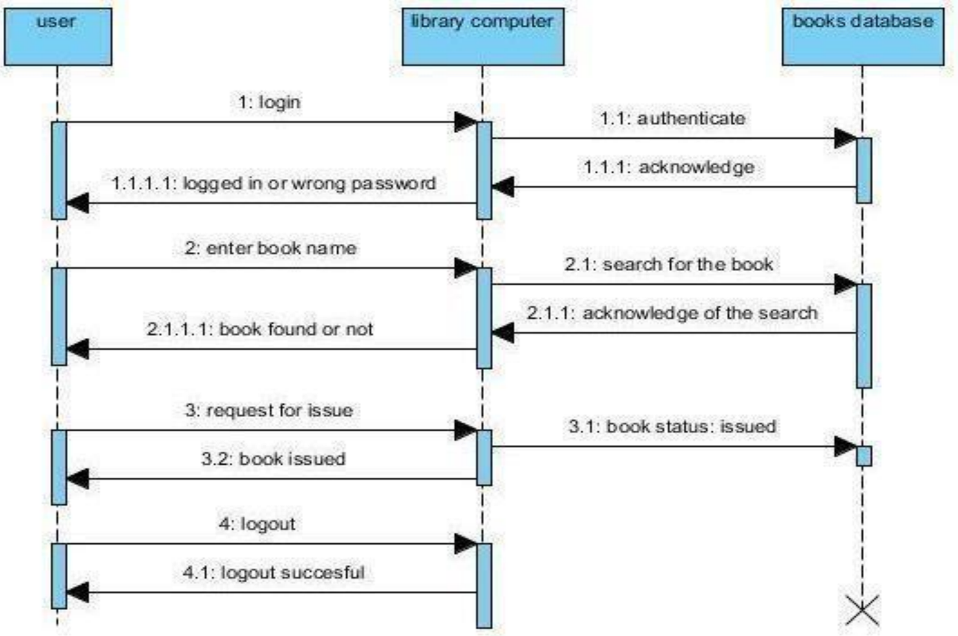
USE CASE DIAGRAM:



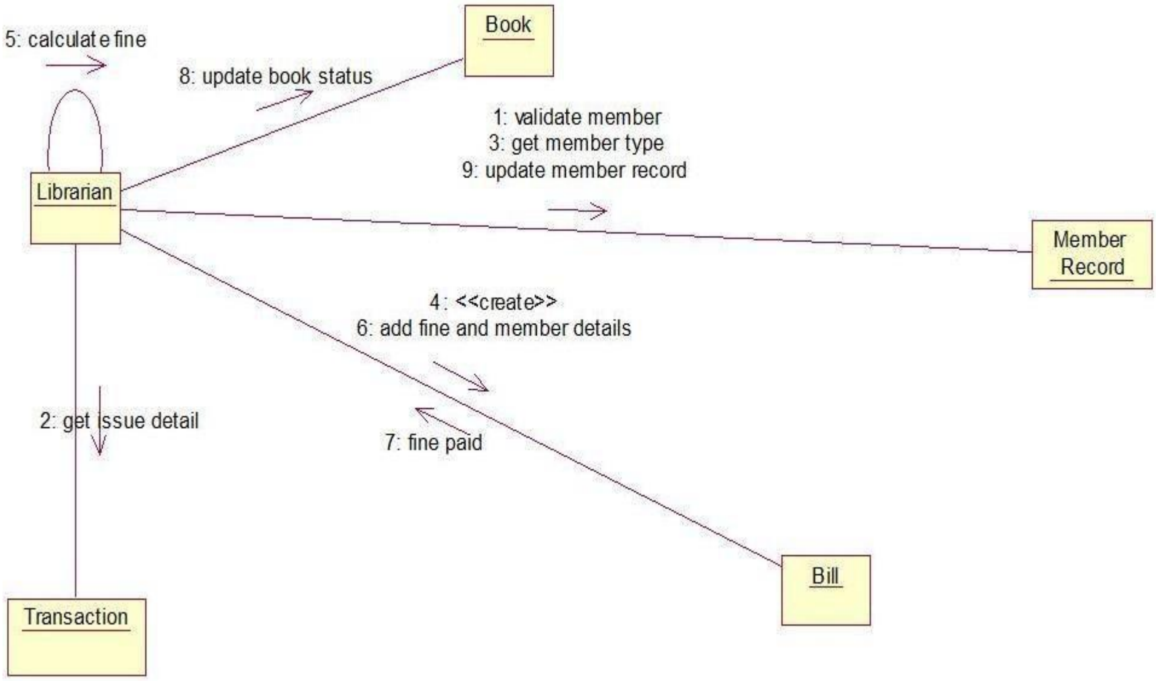
CLASS DIAGRAM:



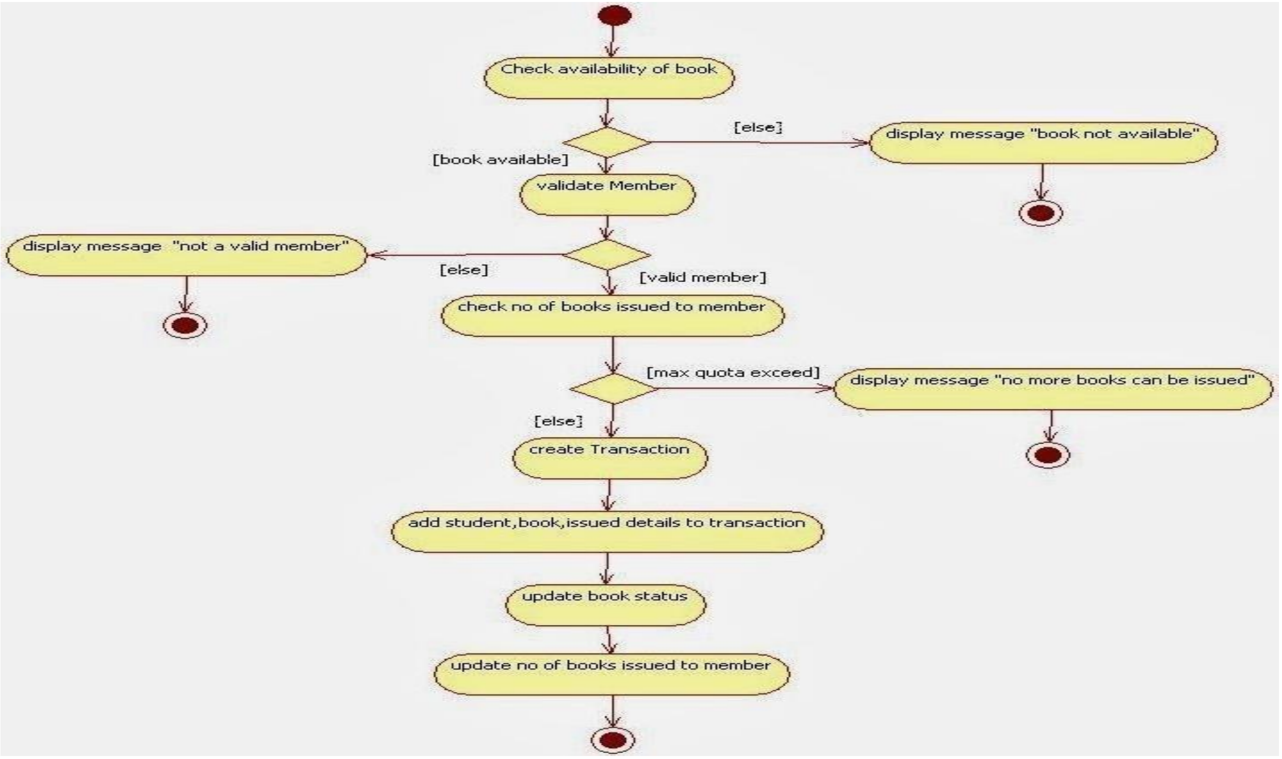
SEQUENCE DIAGRAM:



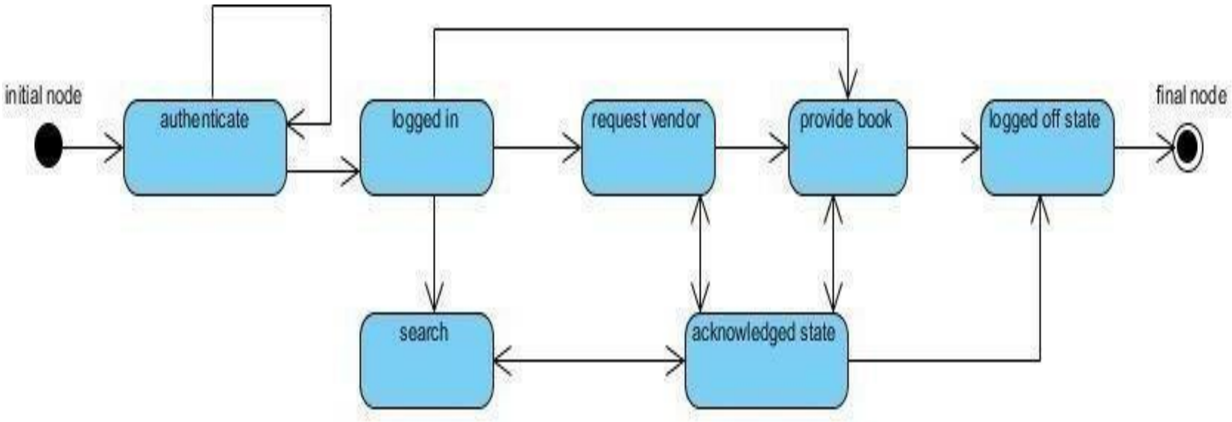
COLLABORATION DIAGRAM



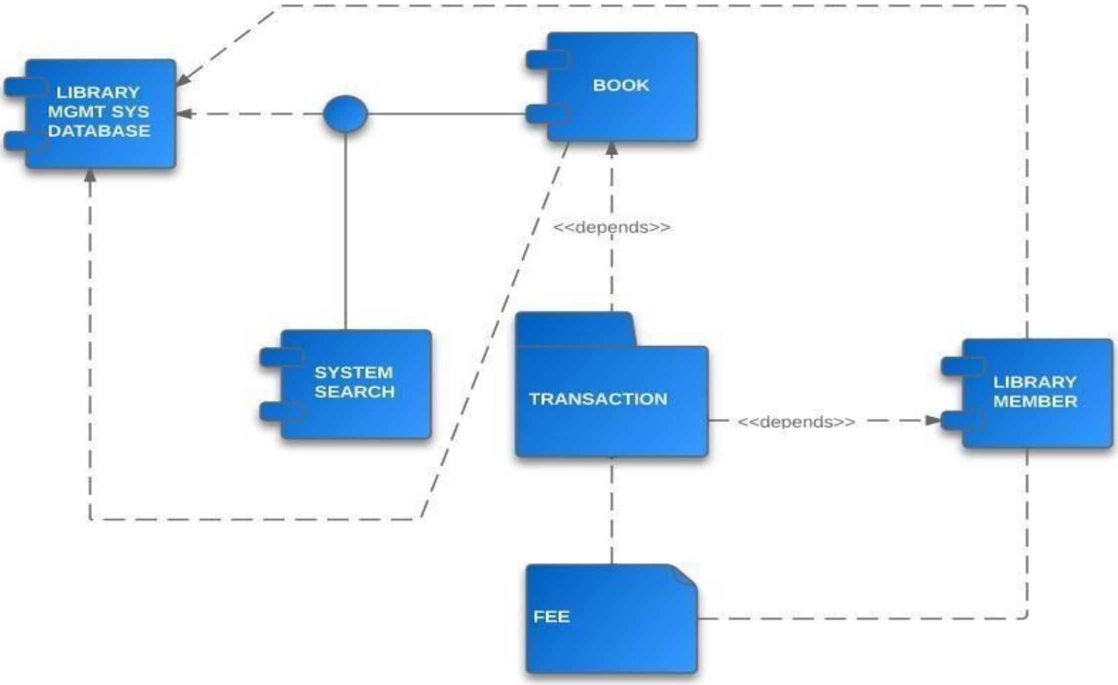
ACTIVITY DIAGRAM:


Check availability of book
[else]
display message "book not available"
[book available]
validate Member
display message "not a valid member"
[else]
[valid member]
check no of books issued to member
[max quota exceed]
display message "no more books can be issued"
[else]
create Transaction
add student,book,issued details to transaction
update book status
update no of books issued to member

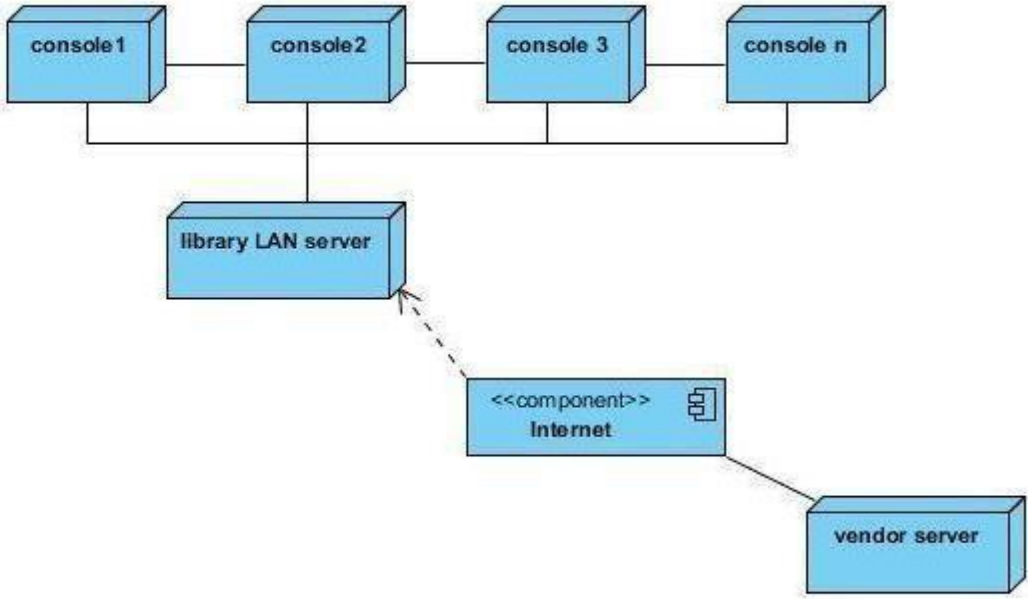
STATE DIAGRAM:



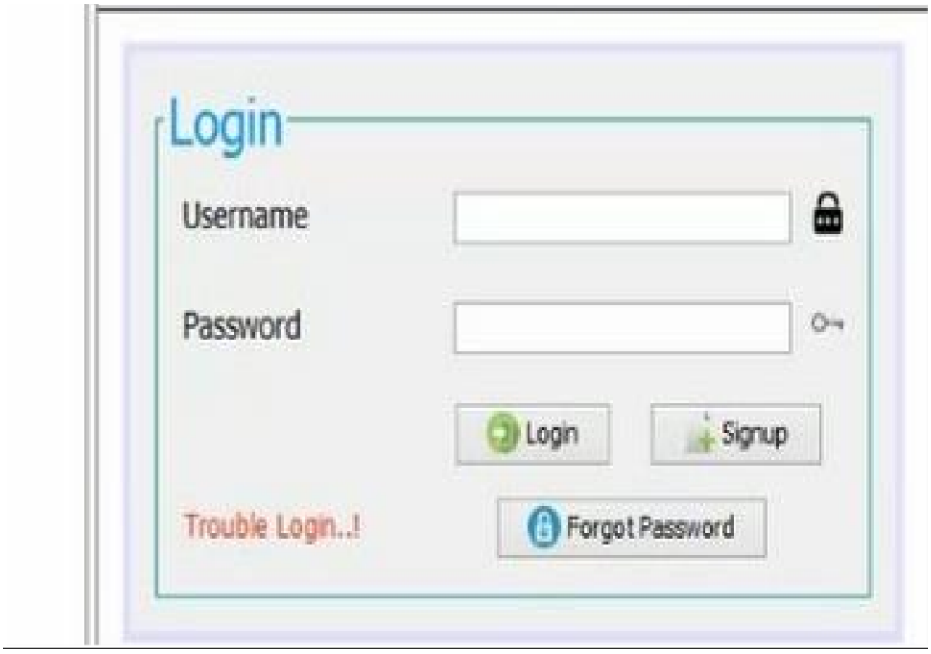
COMPONENT DIAGRAM:



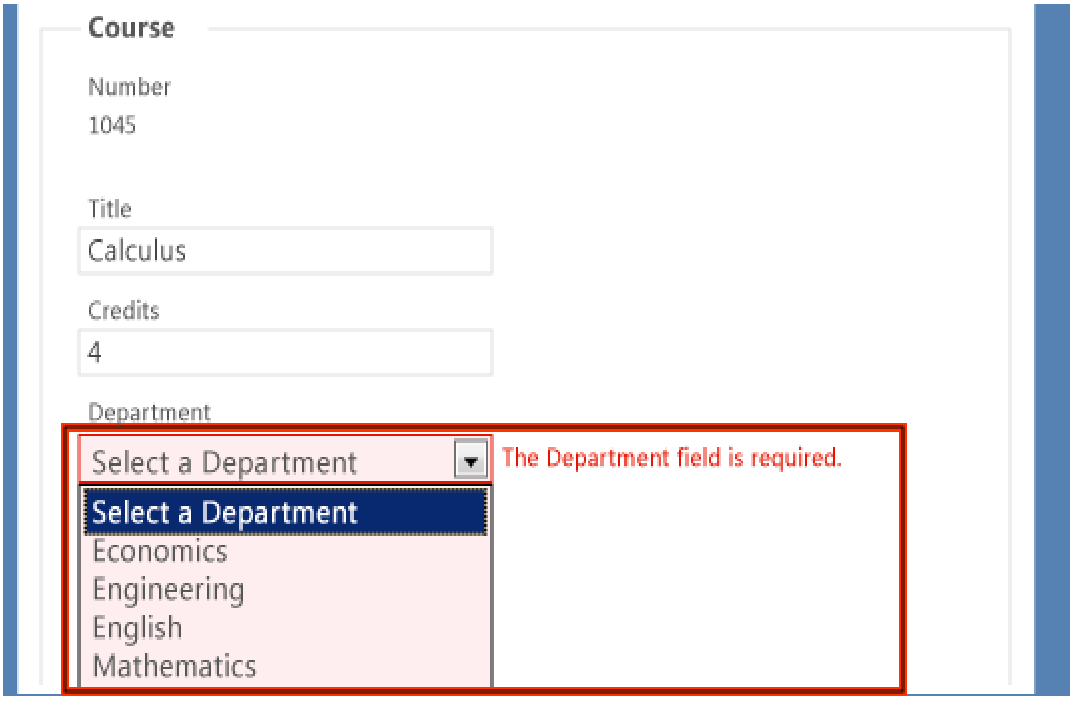
DEPLOYMENT DIAGRAM:



Login form:



Choice form:



Course

Number
1045

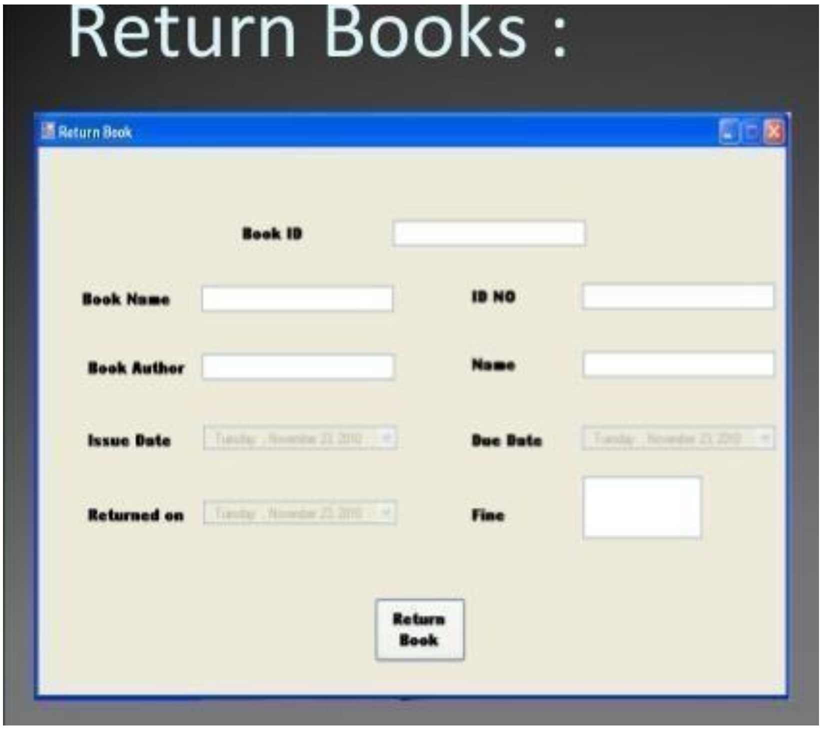
Title
Calculus

Credits
4

Department
Select a Department

The Department field is required.

Return form:



Return Books :

Return Book

Book ID

Book Name

ID NO

Book Author

Name

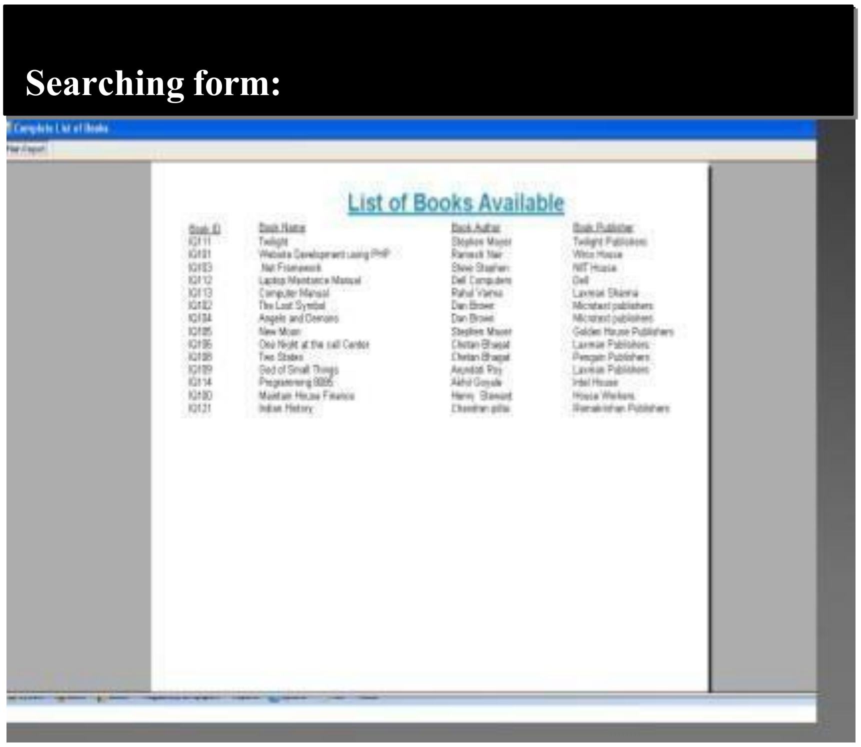
Issue Date: Tuesday , November 23, 2010

Due Date: Tuesday , November 23, 2010

Returned on: Tuesday , November 23, 2010

Fine

Searching form:

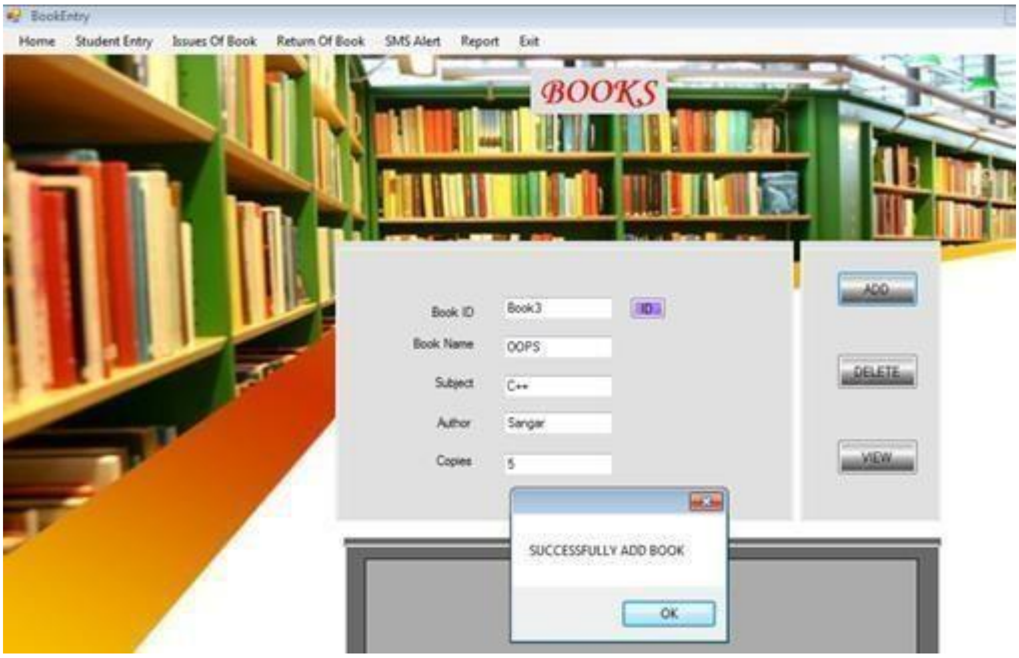


Searching form:

List of Books Available

Book ID	Book Name	Book Author	Book Publisher
IG111	Twilight	Stephen Meyer	Twilight Publishers
IG101	Website Development using PHP	Ramesh Nair	White House
IG103	.Net Framework	Steve Stephen	NIIT House
IG112	Laptop Maintance Manual	Dell Computers	Dell
IG113	Computer Manual	Rahul Varma	Laxman Sharma
IG102	The Lost Symbol	Dan Brown	Microtard publishers
IG104	Angels and Demons	Dan Brown	Microtard publishers
IG105	New Moon	Stephen Meyer	Golden House Publishers
IG106	One Night at the call Center	Chetan Bhagat	Laxman Publishers
IG108	Two States	Chetan Bhagat	Penguin Publishers
IG109	God of Small Things	Arundati Roy	Laxman Publishers
IG114	Programming 8085	Akhil Goyale	Intel House
IG100	Maintain House Finance	Henry Steward	House Workers
IG121	Indian History	Chandran pillai	Ramakrishan Publishers

Renewal form:



Book ID: Book3

Book Name: OOPS

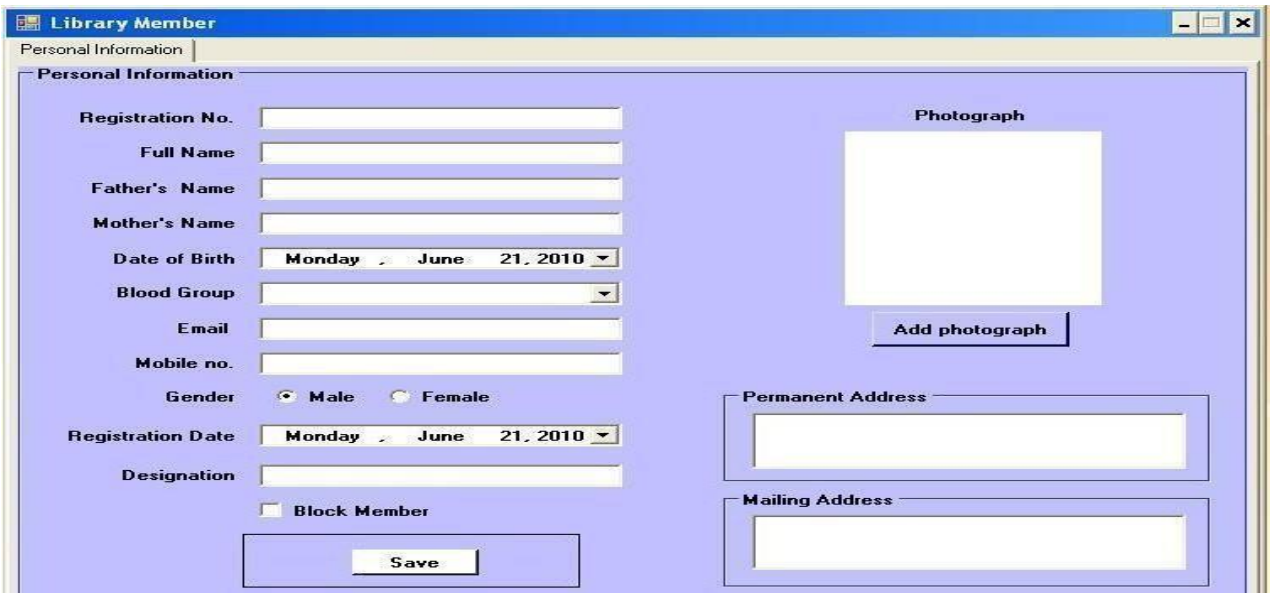
Subject: C++

Author: Sangar

Copies: 5

SUCCESSFULLY ADD BOOK

Registration form:



Library Member

Personal Information

Registration No.

Full Name

Father's Name

Mother's Name

Date of Birth: Monday , June 21, 2010

Blood Group

Email

Mobile no.

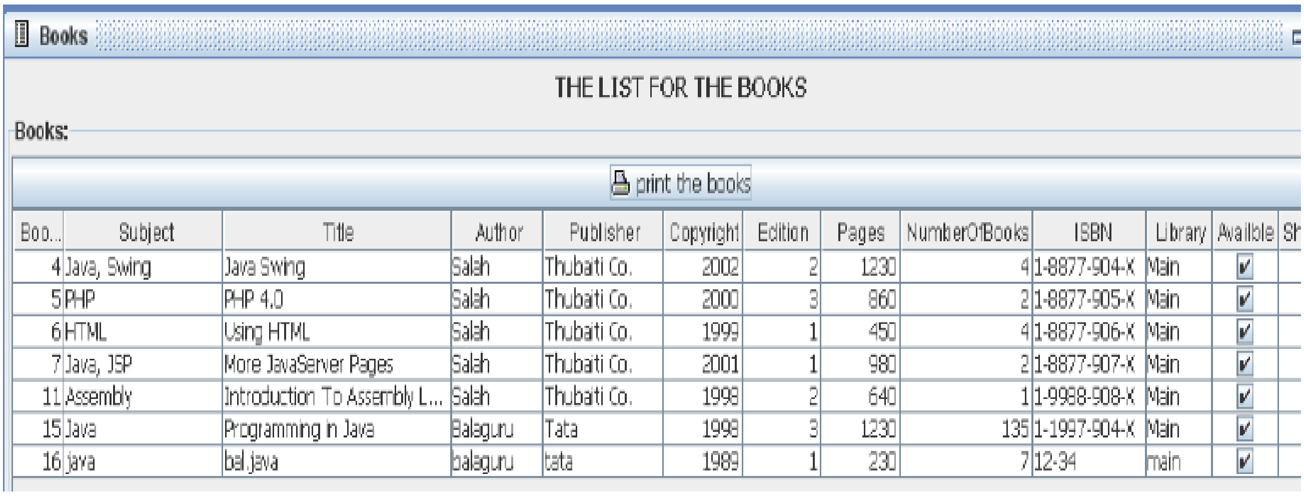
Gender: ● Male ○ Female

Registration Date: Monday , June 21, 2010

Designation

□ Block Member

View books:



THE LIST FOR THE BOOKS

Books:

Boo...	Subject	Title	Author	Publisher	Copyright	Edition	Pages	NumberOfBooks	ISBN	Library	Availble	Sh
4	Java, Swing	Java Swing	Salah	Thubaiti Co.	2002	2	1230	4	1-8877-904-X	Main	☑	
5	PHP	PHP 4.0	Salah	Thubaiti Co.	2000	3	860	2	1-8877-905-X	Main	☑	
6	HTML	Using HTML	Salah	Thubaiti Co.	1999	1	450	4	1-8877-906-X	Main	☑	
7	Java, JSP	More JavaServer Pages	Salah	Thubaiti Co.	2001	1	980	2	1-8877-907-X	Main	☑	
11	Assembly	Introduction To Assembly L...	Salah	Thubaiti Co.	1998	2	640	1	1-9988-908-X	Main	☑	
15	Java	Programming in Java	Balaguru	Tata	1998	3	1230	135	1-1997-904-X	Main	☑	
16	java	bal.java	balaguru	tata	1989	1	230	7	12-34	main	☑	

CODING:

```
Private Sub Command1_Click()
Do
If Form1.Data1.Recordset.EOF Then Form1.Data1.Recordset.MoveFirst
If (Form1.Text1.Text = Form1.Data1.Recordset.Fields(0)) And
(Form1.Text2.Text = Form1.Data1.Recordset.Fields(1)) Then
Msg Box "login succeed"
Form2.Show
Exit Do
Else
Form1.Data1.Recordset.MoveNext
End If
Loop Until Form1.Data1.Recordset.EOF
If Form1.Data1.Recordset.EOF Then
MsgBox ("invalid login")
End If
End Sub
Private Sub Label1_Click()
Form7.Show
End Sub
Private Sub Command2_Click()
Form7.Show
End Sub
Private Sub Command1_Click()
Form3.Show
End Sub
Private Sub Command2_Click()
Form4.Show
End Sub
Private Sub Command3_Click()
Form5.Show
End Sub
Private Sub Command4_Click()
Form6.Show
End Sub
Private Sub Command1_Click()
If (Text1.Text = "Internet Programming" Or Text1.Text = "ooad" Or Text1.Text
= "computer graphics") Then
MsgBox "Book is successfully returned........ "
Form2.Show
Else
MsgBox "Renewal time exceeding ...... "
End If
End Sub
Private Sub Combo1_Change()
End Sub
Private Sub Command1_Click()
Dim a As String
a = Text1.Text
```

```
Dim b As String
b = Text2.Text
Dim i As Integer
For i = 0 To 5
If ((a = Data1.Recordset.Fields(0)) And (b =
Data1.Recordset.Fields(1)))
Then MsgBox "successfully issued......thank u!!!!!"
Exit For
Else
MsgBox "Sorry....ur book not found. .try again!!!!!"
End If
Next i
End Sub
Private Sub Command2_Click()
Form2.Show
End Sub
Private Sub Command1_Click()
If (Text1.Text = "Internet Programming" Or Text1.Text = "ooad" Or Text1.Text =
"computer graphics") Then
MsgBox "Book is successfully renewaled. ...... "
Form2.Show
Else
MsgBox "Renewal time exceeding......."
End If
End Sub
Private Sub Command1_Click()
Data1.Recordset.AddNew
MsgBox "successfully registered. ...... "
Form2.Show
End Sub
```

CONCLUSION

Thus the project for Library Management System was designed and codes are generated and then it was executed successfully.

Ex.No.: 15
Date :

STUDENT INFORMATION SYSTEM

AIM

To analyse, design and develop code for Student Information System using Agro URL software.

PROBLEM STATEMENT

The student information system is a software application for schools, colleges and universities to manage student data. These system are capable of holding students' test scores, assessment scores, etc. through an electronic grade book. They are also used to hold records of the students' attendance, track student schedules, handling inquiries from students, enrolling new students, managing any other records relevant to students.

Students can enter their profile details and update it, if any. They can view their exam schedule and their results. Staffs can view students‟ profile and enter marks into system. These details are stored in the database which can be retrieved whenever necessary.

OVERALL DESCRIPTION

The Student Information System is an integrated system that has four modules as part of it.

The three modules are:

1) **Login for student and staff:** Using this module both student and staff can login to thesystem using his/her unique username and password respectively.
2) **Student update:** In this module, the students register his/her details in the system, which is stored in the database. They can view their exam details and results.
3) **Staff update:**In this module, the staffs can register students‟ details in the system. They are stored in the database.

SOFTWARE REQUIRMENTS

- Microsoft Visual Basic 6.0
- Agro UML
- Microsoft Access

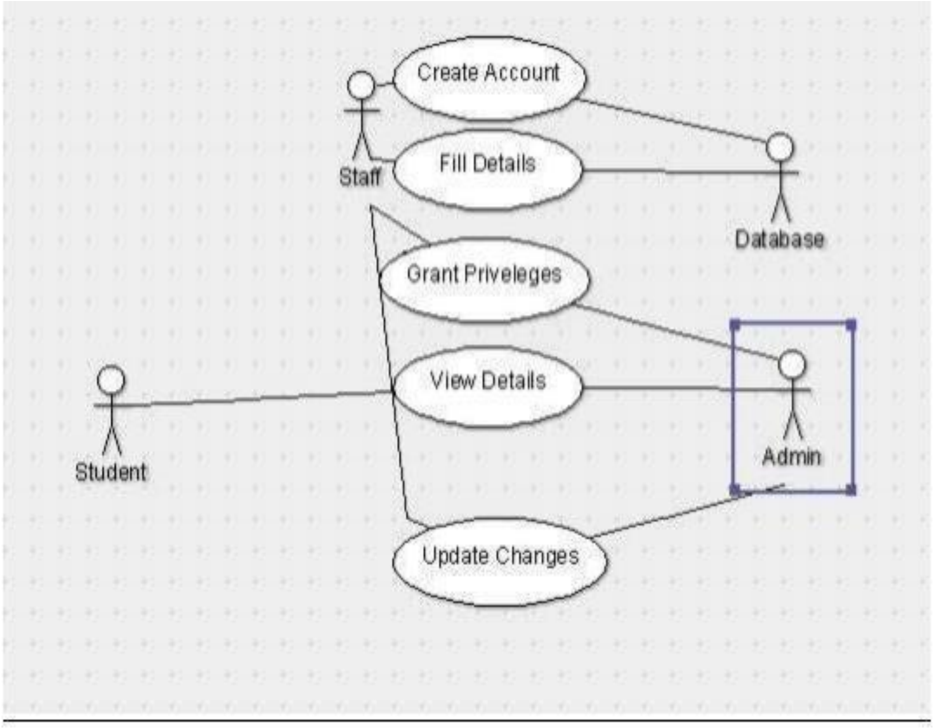
HARDWARE REQUIRMENTS

- 128MB RAM
- Pentium III Processor

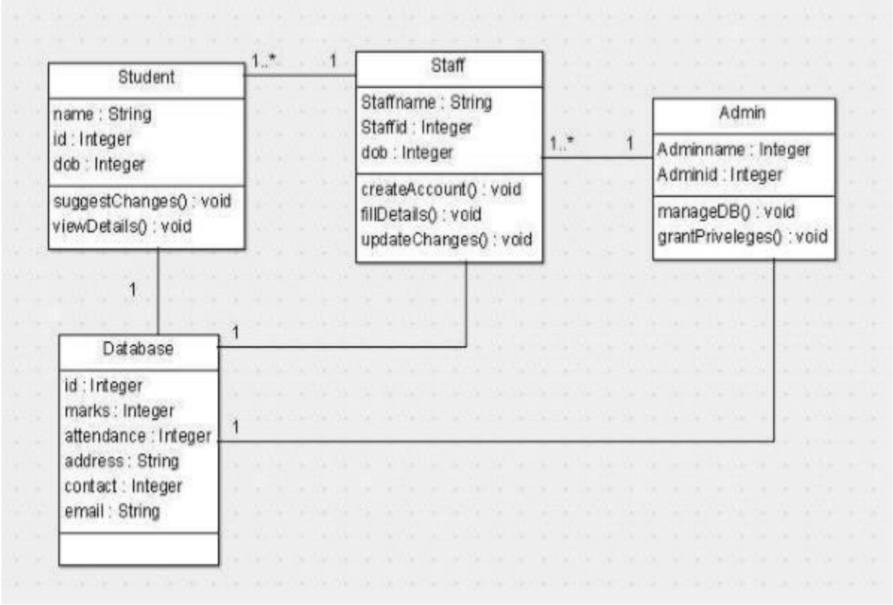
ANALYSIS MODELING

The project can be explained diagrammatically using the following diagrams:

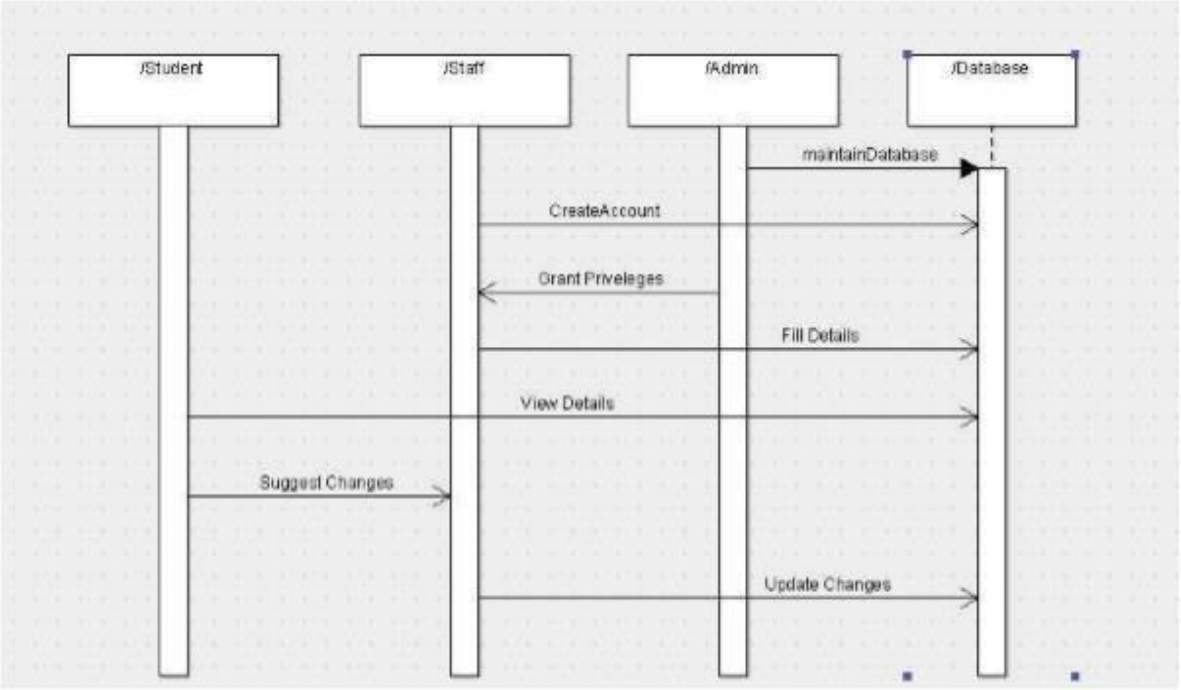
USE CASE DIAGRAM



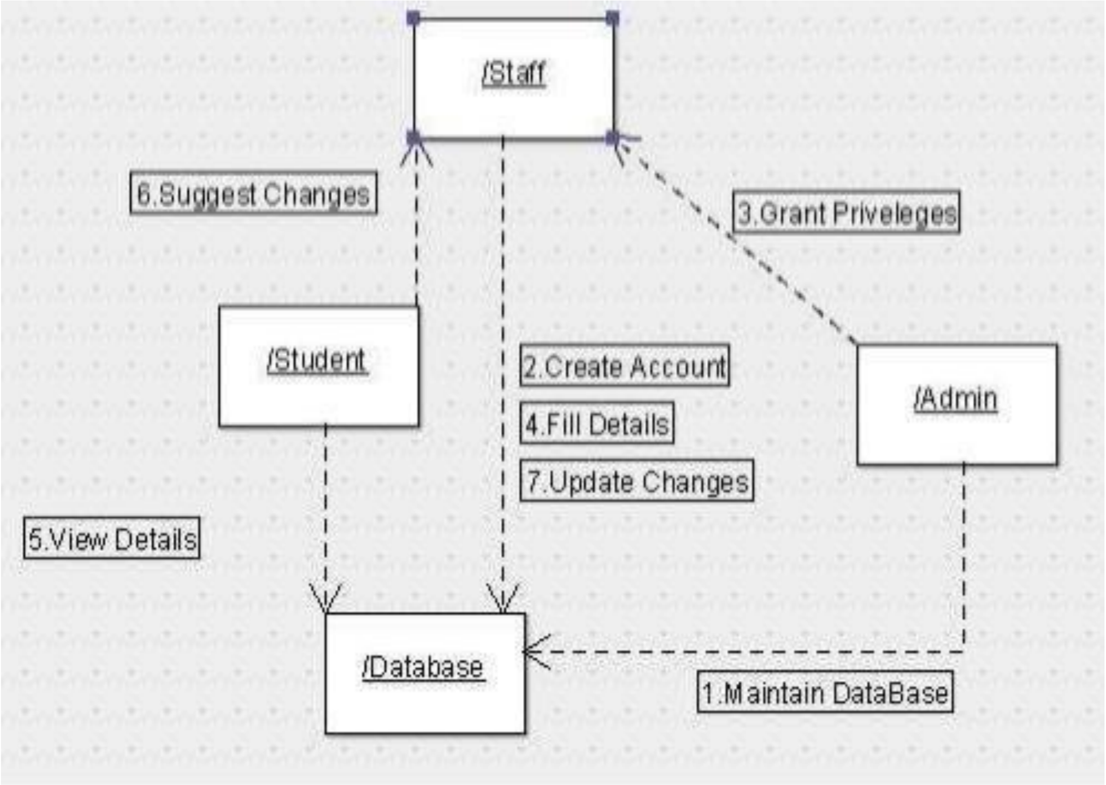
CLASS DIAGRAM



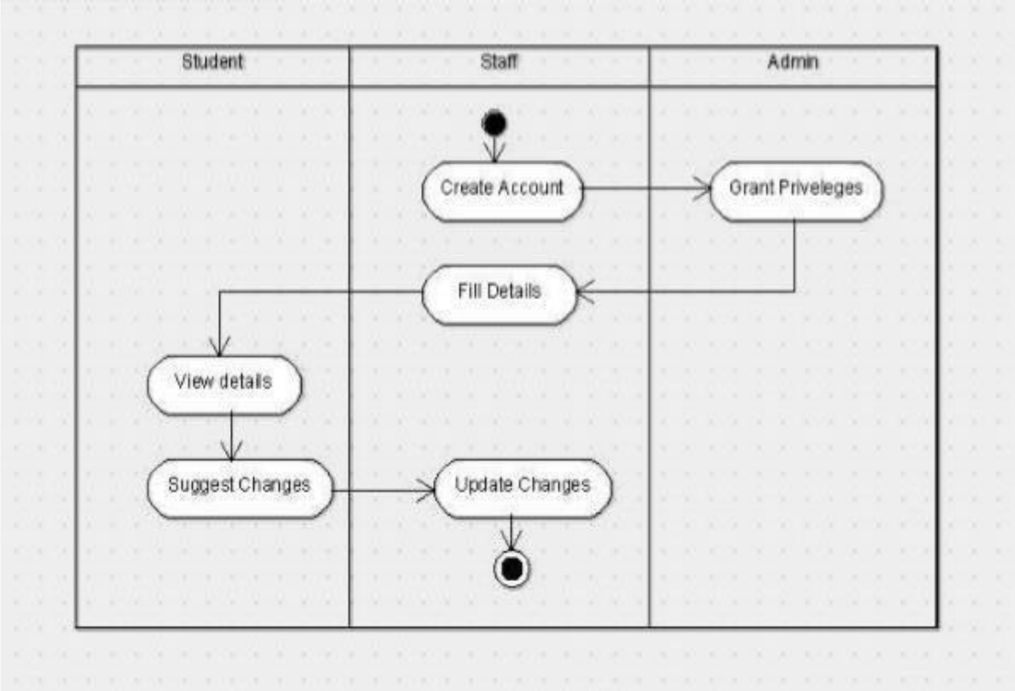
SEQUENCE DIAGRAM



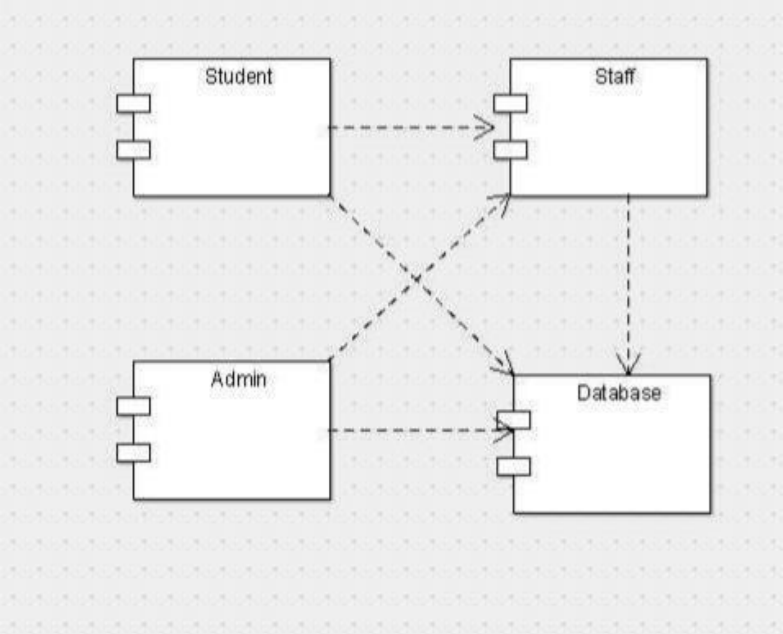
COLLABORATION DIAGRAM



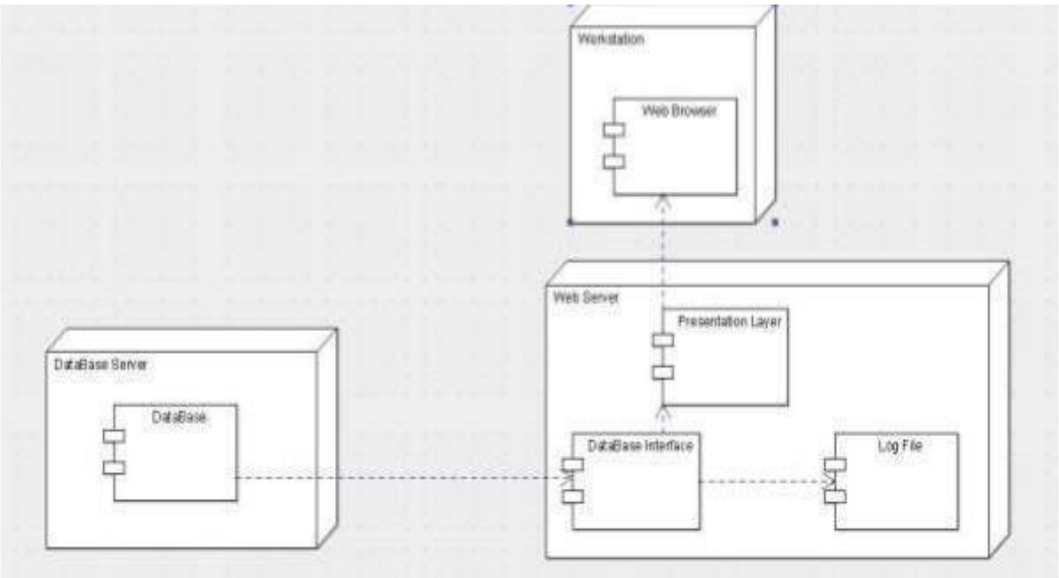
ACTIVITY DIAGRAM



COMPONENT DIAGRAM



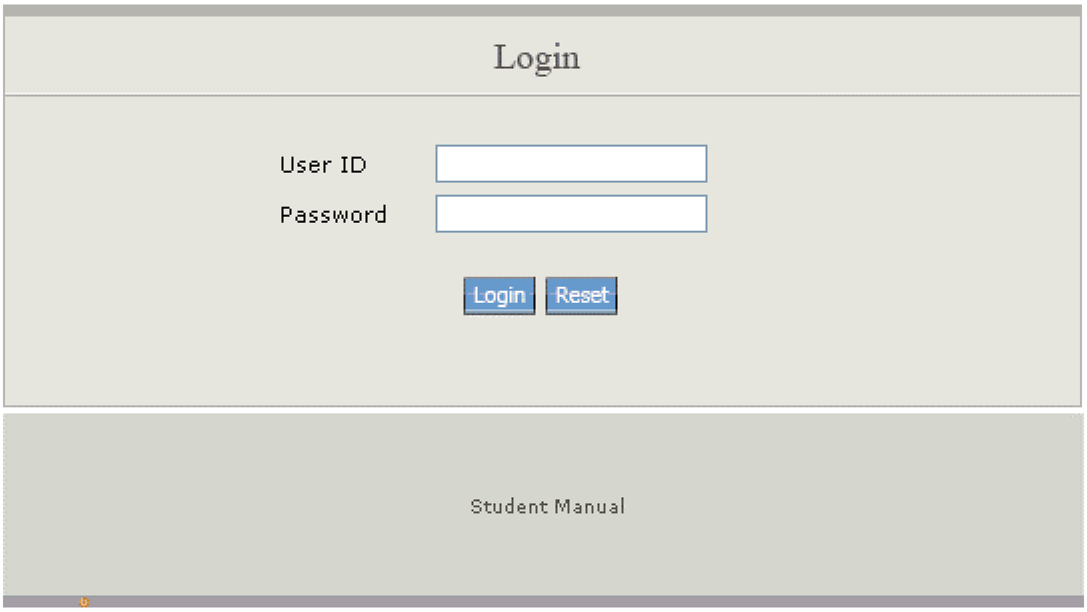
DEPLOYMENT DIAGRAM



IMPLEMENTATION

FORMS

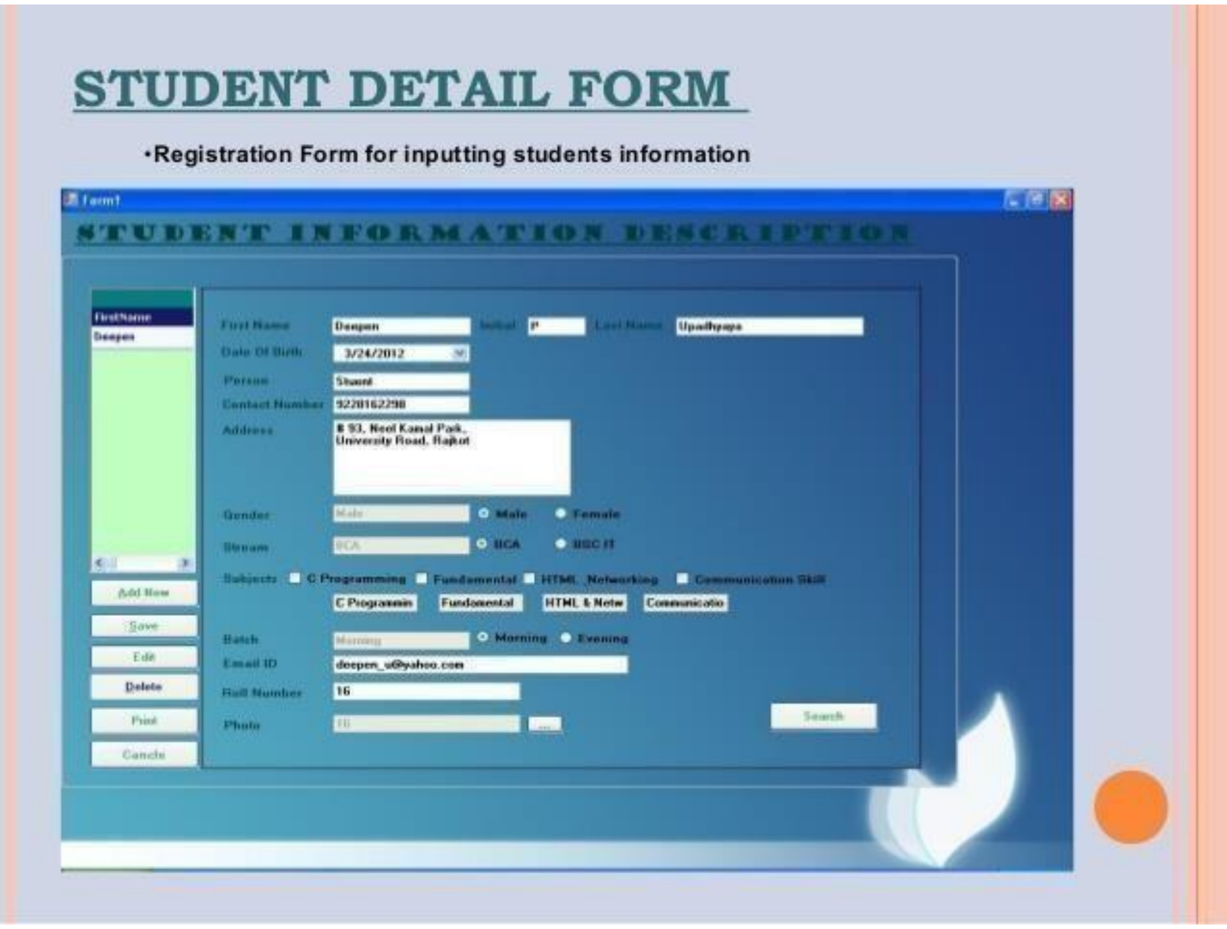
LOGIN PAGE



EXAM SCHEDULE

CA Final Time Table Nov 2015				
Group - I				
Paper No.	Subject	Date	Day	Timings (IST)
1	Financial Reporting	1-11-2015	Sunday	2.00 P.M. to 5.00 P.M.
2	Strategic Financial Management	3-11-2015	Tuesday	2.00 P.M. to 5.00 P.M.
3	Advanced Auditing and Professional Ethics	5-11-2015	Thursday	2.00 P.M. to 5.00 P.M.
4	Corporate and Allied Laws	7-11-2015	Saturday	2.00 P.M. to 5.00 P.M.
5	Advanced Management Accounting	09-11-2015	Monday	2.00 P.M. to 5.00 P.M.
6	Information Systems Control and Audit	12-11-2015	Thursday	2.00 P.M. to 5.00 P.M.
7	Direct Tax Laws	14-11-2015	Saturday	2.00 P.M. to 5.00 P.M.
8	Indirect Tax Laws	16-11-2015	Monday	2.00 P.M. to 5.00 P.M.

STUDENT PROFILE



STUDENT DETAIL FORM

- Registration Form for inputting students information

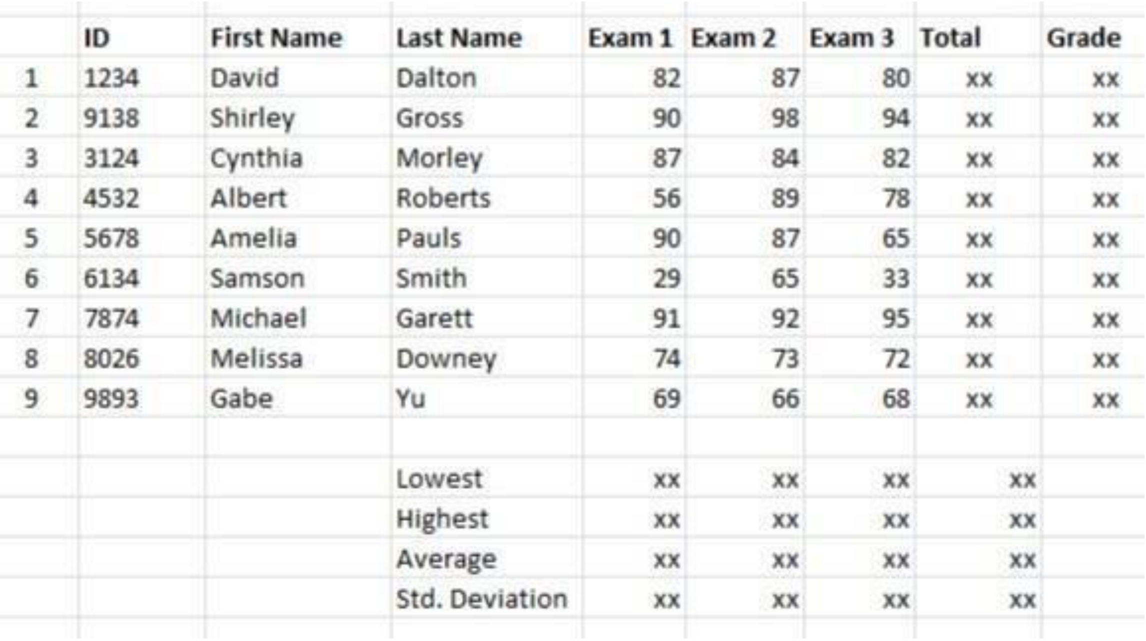
Form1

STUDENT INFORMATION DESCRIPTION

FirstName	
Deepen	

First Name	Deepen	Initial	P	Last Name	Upadhyaya
Date Of Birth	3/24/2012				
Person	Stuent				
Contact Number	9228162298				
Address	# 93, Neel Kamal Park, University Road, Rajkot				
Gender	Male	○ Male	○ Female		
Stream	BCA	○ BCA	○ BSC IT		
Subjects	□ C Programming	□ Fundamental	□ HTML Networking	□ Communication Skill	
	C Programmin	Fundamental	HTML & Netw	Communicatio	
Batch	Morning	○ Morning	○ Evening		
Email ID	deepen_u@yahoo.com				
Roll Number	16				
Photo	16				

SERIAL TEST MARK



	ID	First Name	Last Name	Exam 1	Exam 2	Exam 3	Total	Grade
1	1234	David	Dalton	82	87	80	xx	xx
2	9138	Shirley	Gross	90	98	94	xx	xx
3	3124	Cynthia	Morley	87	84	82	xx	xx
4	4532	Albert	Roberts	56	89	78	xx	xx
5	5678	Amelia	Pauls	90	87	65	xx	xx
6	6134	Samson	Smith	29	65	33	xx	xx
7	7874	Michael	Garett	91	92	95	xx	xx
8	8026	Melissa	Downey	74	73	72	xx	xx
9	9893	Gabe	Yu	69	66	68	xx	xx
			Lowest	xx	xx	xx	xx	
			Highest	xx	xx	xx	xx	
			Average	xx	xx	xx	xx	
			Std. Deviation	xx	xx	xx	xx	

CODINGS:

```
Private Sub Command1_Click()
Dim count As Integer
For count = 0 To 5
If Data1.Recordset.EOF = False Then
If Data1.Recordset.Fields(0) = Text1.Text And Data1.Recordset.Fields(1) =
Text2.Text Then
MsgBox ("login successful")
Form6.Show
Exit For
'#unloadMe
Else
MsgBox ("invalid login")
Exit For
End If
End If
Next count
End Sub

Private Sub Command2_Click()
Dim count As Integer
For count = 0 To 5
If Data2.Recordset.EOF = False Then
If Data2.Recordset.Fields(0) = Text3.Text And Data2.Recordset.Fields(1) =
Text4.Text Then
MsgBox ("login successful")
Form2.Show
Exit For
```

```
„#unloadMe
Else
MsgBox ("invalid login")
Exit For
End If
End If
Next count
End Sub
Private Sub Command3_Click()
Text1.Text = ""
Text2.Text = ""
End Sub
Private Sub Command4_Click()
Text3.Text = ""
Text4.Text = ""
End Sub
Private Sub Command1_Click()
Form3.Show
End Sub
Private Sub Command2_Click()
Form5.Show
End Sub
Private Sub Command1_Click()
Form1.Show
End Sub
Private Sub Command2_Click()Dim count As Integer
For count = 0 To 5
If Data1.Recordset.EOF = False Then
If Text6.Text = Data1.Recordset.Fields(0) Then
'MsgBox ("login successful")
Text1.Text = Data1.Recordset.Fields(1)
Text2.Text = Data1.Recordset.Fields(2)
Text3.Text = Data1.Recordset.Fields(3)
Text4.Text = Data1.Recordset.Fields(4)
Text5.Text = Data1.Recordset.Fields(5)
Exit For
'#unloadMe
Else
MsgBox ("invalid reg no")
Exit For
End If
End If
Next count
End Sub
Private Sub Command3_Click()
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
```

```
Text6.Text = ""
End Sub
Private Sub Command1_Click()
Form4.Show
End Sub
Private Sub Command2_Click()
Dim count As Integer
For count = 0 To 4
If Data1.Recordset.EOF = False Then
If Data1.Recordset.Fields(1) = Text2.Text Then
Text1.Text = Data1.Recordset.Fields(0)
Text3.Text = Data1.Recordset.Fields(2)
Text4.Text = Data1.Recordset.Fields(3)
Text5.Text = Data1.Recordset.Fields(4)
Text6.Text = Data1.Recordset.Fields(5)
Exit For
'#Unload Me
Else
MsgBox ("invalid register number")
End If
End If
Next count
End Sub

Private Sub Command3_Click()
Text1.Text = ""
Text2.Text = ""
Text3.Text = ""
Text4.Text = ""
Text5.Text = ""
Text6.Text = ""
End Sub

Dim a As DB
Private Sub Command1_Click()
Set a = New DB
a.store
End Sub
Private Sub Command2_Click()
Form1.Show
End Sub
```

CONCLUSION

Thus the project for Student Information System was designed and codes are generated and then it was executed successfully.