

**DR.BABASAHEB AMBEDKAR MARATHWADA
UNIVERSITY,AURANGABAD.**



SAGAR BCA COLLEGE,JALNA

DEPARTMENT OF MANAGEMENT

(2021-22)

A

Project Report

On

“Pharmacy Management System”

Submitted By,

**Mr.Gawai Atul Bhimrao
BCA III Yr (VIth sem)**

Guided By:

Prof.Najardhane N.D.



Dr. Babasaheb Ambedkar Marathwada

University, Aurangabad

Sagar BCA College, Jalna.

(Department of Computer Management)

2021-22

C E R T I F I C A T E

This is to certify that project report entitled "Pharmacy Management System" submitted by Mr. Gawal Atul Bhimrao is as per the requirement of Dr. Babasaheb Ambedkar Marathwada University, Aurangabad in practical fulfillment of BCA (VI SEM) for the academic year (2021-22)


Guided BY
Head

Department of Management
Sagar BCA College, Jalna.


PRINCIPAL
Sagar College, Jalna
Principal


01-04-21
Examiner

DECLARATION

We hereby declare that, the project report submitted by us in partial fulfillment of B.CA. degree course is a genuine work, It has not been submitted either fully or partially to their or any other institute prior in any other connection. We have done the project as per the guidelines of our respected guide.

Guidance By



Prof. Najardhane N.D.

Head

Department of Management
Sagar BCA College, Jalna.

Submitted By



Mr. Gawai Atul Bhimrao

INTRODUCTION & BACKGROUND

Pharmacy management allows flexible scale within a single facility, and Permits each process to operate at maximum efficiency rather than having the speed of the entire process constrained by the slowest.

To keep the number of records for the flexible manual task it is necessary To make a software which provide flexible way for easier task .this project required management of any pharmacy college details in which keep all

The detail about student registration ,admission , courses required in the pharmacy college .the user which handle this software can easily store number of student record with current date and time . the user after any how can see the previous record and make or modify necessary changes .

Due to this software the user at any instance of time could not be need to more data entry. For developing the software we use references of visual basic Database , Ms-Access , phases of waterfall model etc.

The Project Becomes Consist Of Environmental (Hardware &Software Requirement)And Economical Requirement .

All forms is connected with the database tables but user cannot directly make changes in the structure of database and cannot do some like update . delete and insert etc. the user these operation using the fort end application only.

PROBLEM STATEMENT

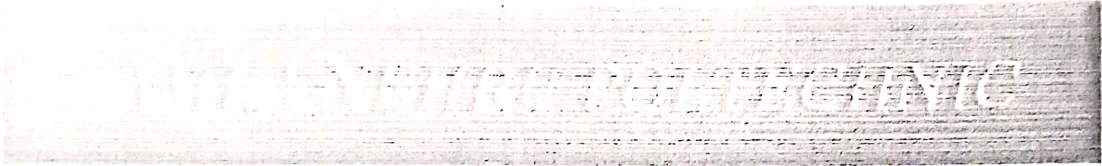
While developing the software there is a lot of things that we should have keep in mind, all the requirements and problems which can be arise in all that points we should keep in mind .this is very important to develop successful software .when we are developing this software we got many small and big problem on technical issue , the very first problem arise is the data form data base single cell data from list view that is primary key and transferring that particular row to sales section .

The topic we chosen 'stock management' is only because it is a common thing that anybody know little bit about the stock management as it concern to our daily life routine .the first and difficult step to take when developing any software is gathering the requirements we should know what to do first then only we can go ahead .

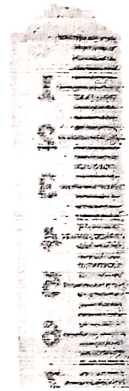
We take help from internet to read about the pharmacy management system and takes help of our guide which encourages us to go ahead. we are grateful our guide of project and to those who help us on every step and encourage us when are stop .

PROJECT DESCRIPTION





Pharmacy College



User Name

Passwrod

OK

Cancel



Database Report About us

Registration ID

R1000

Student Name

Mr Javeed

Date of Birth

06/05/1989

Gender

Male

Percentage

80%

Fee

200

College Name

Sir Sayeed College Aurangabad

Address

Plot No A-25, Kiz Kai Gate Aurangabad

R1000

R1001

R1008

R1002

New

Modify

Delete

Save

Cancel

Search

Close



Registration No.

Series

Student Name

Date of Exam

Class

City Percentage

MCQ Marks

College Name

Copyright

Q1 Mark

Q2 Mark

Q3 and Q4 Mark

Q5 Mark

Admission Date

Registration No.

Exam No.

Examiner

Signature

Stamp

Student ID

1000

Student Name

AB HANAN

Class

1 C-Pharmacy

Year

1 Year

Date

07/03/2011

Registration Fee

1000

Tuition Fee

7500

Exam Fee

500

Health Fee

500

Lab Fee

750

S1000

S1001

Page

Index

Print

Save

Screen

Search

Exit



Student ID S1000

Student Name Mr Mohsin

S1000

Class BCS

Year 8th Year

Subject JAVA

Subject VB 6.0

Subject Urdu

Subject C Programming

Subject VB.net 2010

Subject SQL Server 2010

New Modify Delete Save Cancel Search Close

Lecturer

Lecturer Name

Assignment

Subject

Course

MCA

EC-VBSQL

BCA, BCS

Home

Library

Course

Save

Cancel

Search

Close



by: [unclear] 10/01/2019

Lecturer ID: Lecturer Name:

1 Mr Wz

(Period Name) Subject Name Class Name

8 to 9 C NCA

9 to 10 C++ BCA

11 to 12 C++ BCA

3 to 4 VB 5.0 ECS

2 to 3 VB 5.0 ECS

New Modify Delete Save Cancel Search Close

Developed By

ABHILASH

KRUSHNA

MAHASHABDE

OK

SAGAR COLLEGE JALNA



FUNCTION SPECIFICATION

Function perform:

- 1) User want capable of storing all types of information present any pharmacy college.
- 2) All the student record can be update easily.
- 3) All courses detail can be set without making any kinds of duplication.
- 4) Records can be set with current date and time.
- 5) Previous record can be easily seen.
- 6) The record of all student fees detail can maintain with due.
- 7) Records print should be easily taken.
- 8) Only the actual user can be seen the record and keep the record .unauthorized person could not be access data.
- 9) If any user can make invalid pass word.
- 10) To store the bake up.

Limitation and restriction :

- 1)The project required only for the pharmacy college
- 2)We cannot use this project for any management system except college of pharmacy.
- 3)The project has been secured hence unauthorized could not be access data.
- 4)Wherever the number are used , there make restriction for character .
- 5)Restriction of the character can be made in users ID.

USER INTERFACE DESIGN

DATABASE TABLE

LOGIN TABLE

Field name	Data type
USERNAME	TEXT(30)
PASSWORD	TEXT(20)

REGISTRATION TABLE

Field name	Data type
REG.. ID	TEXT(5)
STU. NAME	TEXT(30)
DATE OF BIRTH	DATE
GENDER	TEXT (5)
PERCENTAGE	TEXT(5)
CET MARKS	NUMBER
COL. NAME	TEXT(25)
ADDRESS	TEXT(30)

SUBJECT TABLE

FIELD NAME	DATA TYPE
REG., ID	TEXT(5)
STU. NAME	TEXT(30)
CLASS	TEXT
YEAR	TEXT
SUBJECT	TEXT

FEES DETAIL

FIELD NAME	DATA TYPE
REG., ID	TEXT(5)
STU. NAME	TEXT(30)
CLASS	TEXT
YEAR	TEXT
DATE	DATE
ADM.FEES	NUMBER
TUTION FEES	NUMBER
HOSTEL FEES	NUMBER

ADMISSION TABLE

FIELD NAME	DATA TYPE
REG., ID	TEXT(5)
DATE OF BIRTH	DATE
GENDER	TEXT(5)
PERCENTAGE	TEXT(5)
CET MARKS	NUMBER
COL. NAME	TEXT(25)
ADDRESS	TEXT(30)

HOSTEL TABLE

FIELD NAME	DATA TYPE
STU. ID	TEXT(5)
STU. NAME	TEXT(30)
CLASS	TEXT
YEAR	TEXT
BUILDING NO.	NUMBER
ROOM NO.	NUMBER

STAFF DETAIL TABLE

FIELD NAME	DATA TYPE
LEG.. ID	TEXT(5)
STU. NAME	TEXT(30)
QUALICATIONS	TEXT
SUBJECT	TEXT
CLASSES	TEXT

CLASSES TABLE DETAIL

FIELD NAME	DATA TYPE
LEC.. ID	TEXT(5)
STU.NAME	TEXT(30)
PEREIOD TIME	DATE/TIME
SUBJECT NAME	TEXT
CLASS NAME	TEXT

DESIGN SPECIFICATION

DESIGN SPECIFICATIONS

Before developing software is necessary to do total study about feasibility control. Feasibility it is method of developing software which refers to collect information in designing of software . There are different method for developing software that is called as models of software development.

There are total four models for software designing. try as follows,

1. waterfall model
2. prototype model
3. iterative model
4. spiral model

we studied all the models and found that waterfall model is convenient for our project and we choose the waterfall model for project designing.

Waterfall model

The waterfall model is one of the most used models. It was proposed by ROYCE. the model is called as sequential model or classic lifestyle.

in this model , project begins with feasibility analysis on successfully demonstrating the feasibility of project , the requirement analysis is completed . the code is integrated and testing , the system is installed . after this , the regular operation and maintenance of the system takes place .

The waterfall model is divided in to 6 steps,

1. system feasibility
 2. software requirement analysis (SRA)
 3. design
 4. testing
 5. maintenance
- 1) System Feasibility:

system feasibility is feasibility analysis conducted by them . in feasibility analysis , three basic element are going to determine that elements are called resources are used for designing and developing the software.

for example , economical resources (Finance) , operational resources which are also called as financial feasibility . when all the resources are available and software is feasible to develop, feasibility them prepares a report called and it is submit to the authorized person for entering into the next step of software development.

Software Requirement Analysis (SRA):

The requirement gathering process is intensified and focused specially on software, to understand the nature of the program (s) to be built, the software engineer (analyst) must understand the information domain for the software, requirement for the both system and software are documented and reviewed with the customer.

3) Design:

design is concerned with identifying software components, the intellectual design involves identifying the software components, decomposing them into processing module and conceptual data structures and specifying the interconnected among modules and data structures.

it also involves adaptation of existing code, modification of standard algorithm, invention of new algorithm, design, of data, representation and packing of software product.

4) Coding : The design must be translated into machine readable form. the code generation can be accomplished mechanistically.

5) Testing : once code has been generated, programs testing being this testing process focused on the logical internal of the software, ensuring that all the statements have been tested and on the function external that is coding test to uncover and ensure that defines and input will produce actual result that agree with required result.

6) Maintenance: Maintenance phase includes the enhancement of capabilities, adaption of the software to new processing environment and correction of software.

Out Put Of Waterfall Model

The output of project employing the waterfall model is not just the final program along and documentation is use it ,there are number of intermediate output that must be produced to produce a successful product .

The set of documents that should be produced in a project is dependent on how the process is implemented .the following is minimum set that should be produce in each project .

- 1.Requirement Documents
2. Project Plan
3. System Design Documents
- 4.Detailed Design Documents
- 5.Test Plans And Test Reports
- 6.Reviw Reports .

Advantages Of Waterfall Model

1. easy to understand even by non technical person .
2. each phase has well defined input and out put .
3. help the project manager in proper planning of the project .

Disadvantages Of Waterfall Model

1. the biggest drawback of water fall model is that it does not support iteration .
2. it is sequentially in nature ,one can't start with the system design until all the requirements are understood and represented .
3. the model is not suitable for new project because of uncertainty in the specification .

Equipment Configuration

Technical Requirements :

- a) hardware requirement :we require 128 MB RAM for VB and MS-ACCESS.20 Hard
- b)

2) Software Requirement Analysis (SRA)

The requirement gathering process is intensified and focused specially on software to understand the nature of the program(s) to be build, the software engineered (analyst) must understand the information domain for the software. Requirement for the both system and software are documented and reviewed with the customer.

3) Design :

Designed is concerned with identifying software components. The intellectual design involves identifying the software components, decoupling and decomposing them into processing module and conceptual data structures and specifying eh interconnected among components. The detail design is concerned with detail of how to implement among modules and data structures.

4) Coding : The design must be translated into machine readable form. The code generated steps perform these tasks. If design is performed in detail manner, code generated can be accomplished mechanistically.

5) Testing : Once code has been generated, program's testing begins. This testing process focused on the logical internal of the software, ensuing that all the statement have been tested and on the functional external that is conducting test to uncover and ensure that defines input will produce actual result that agree with required result.

6) Maintenance : Maintenance phase include the enhancement of capabilities, adaption of the software to new processing environment and correction of software bugs.

Output of waterfall model

The output of project employing the waterfall model is not just the final program along and documentation is to use it. There are number of intermediate output that must be produced to produce a successful product.

The set of documents that should be produced in a project is dependent on how the process is implemented. The following is minimum set that should be produce in each project.

1. Requirement documents
2. Project Plan
3. System Design Documents
4. Detailed Design Documents
5. Test plans and test Reports
6. Final Manual
7. Review Reports.

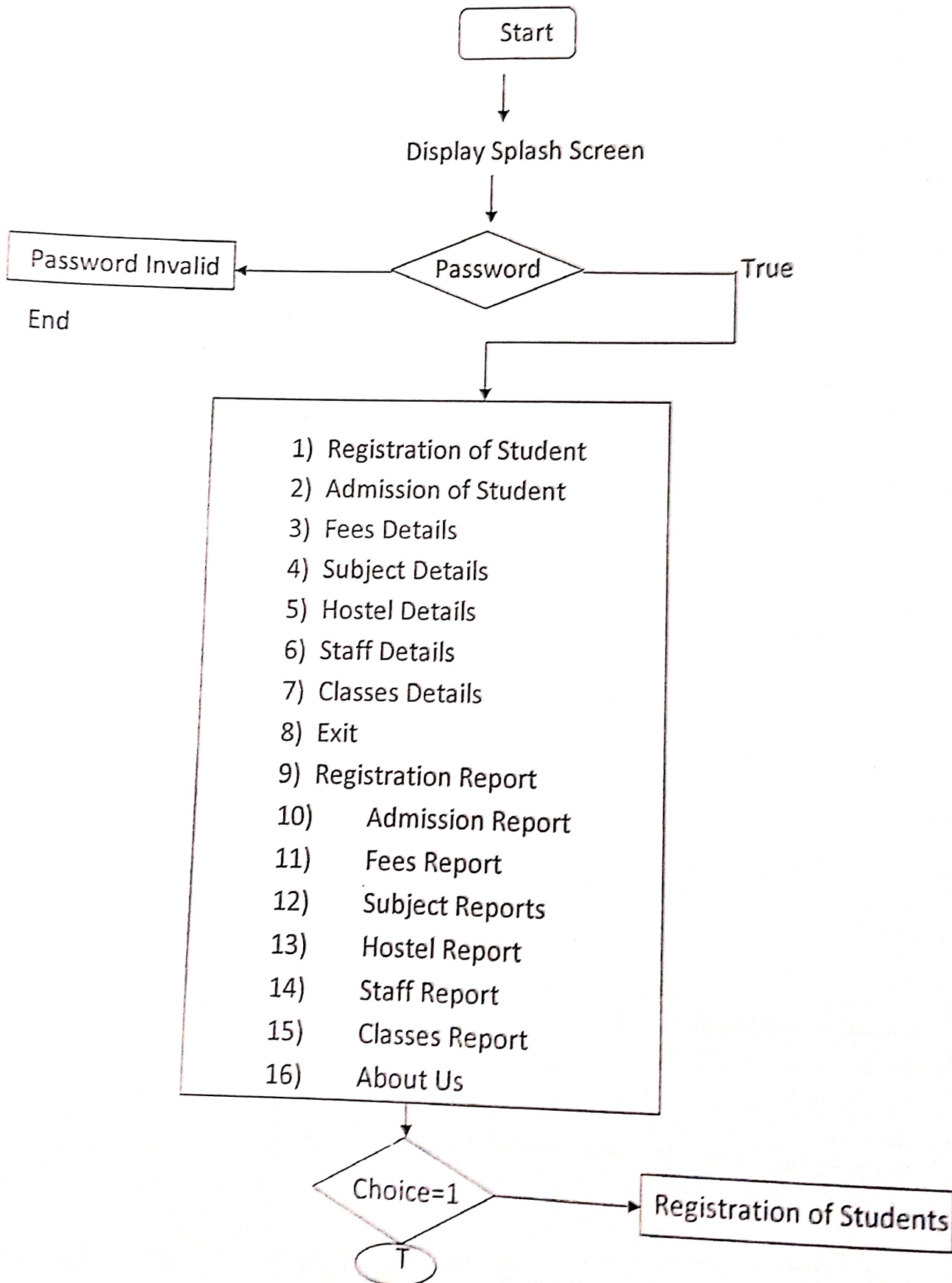
Advantages of Waterfall Model

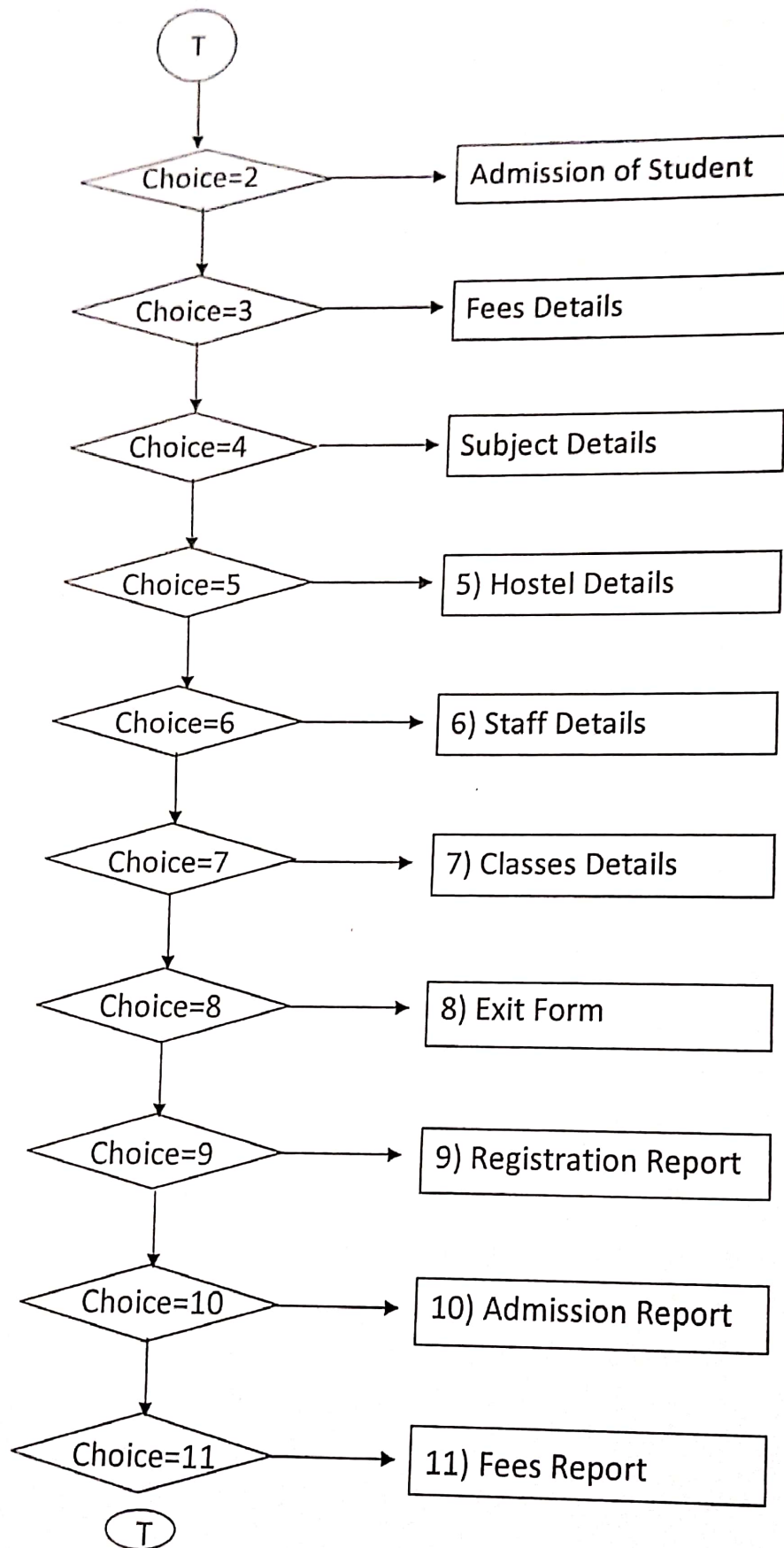
1. Easy to understand even by non technical person.
2. Each phase has well defined input and output.
3. Help the project manager in proper planning of the project.

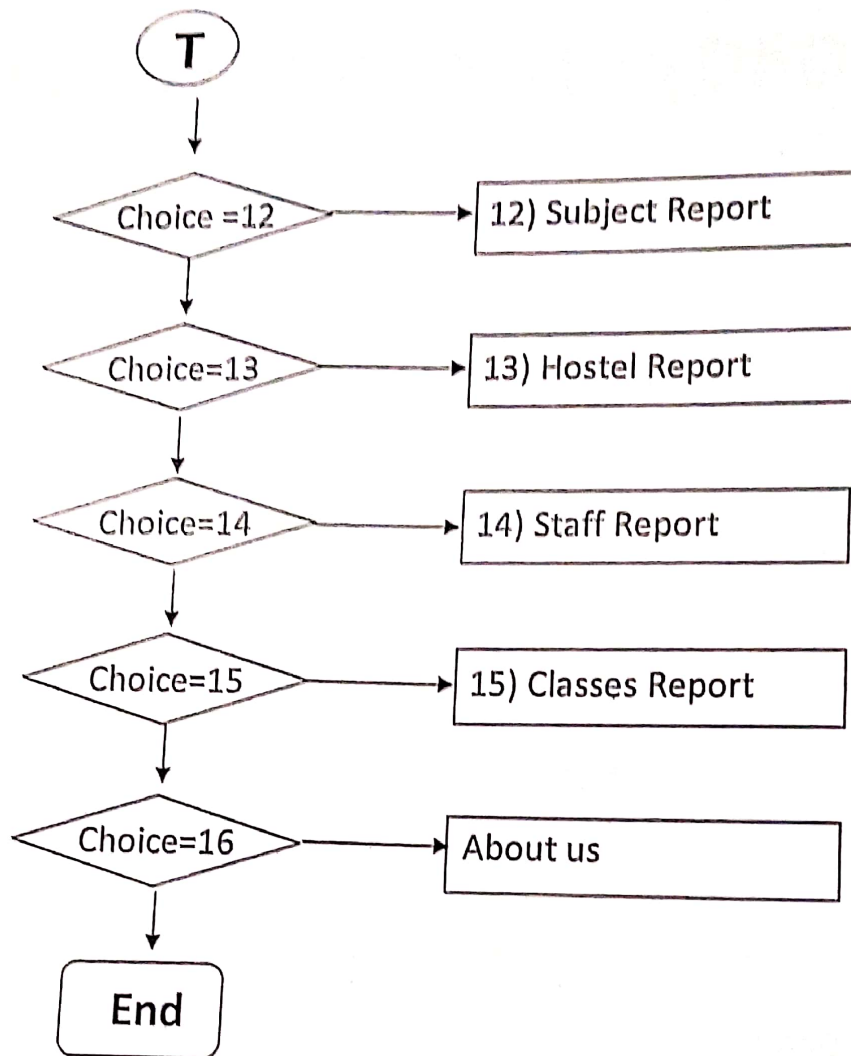
Disadvantages of Waterfall Model

1. The biggest drawback of waterfall model is that it does not support iteration.
2. It is sequentially in nature, one can't start with the system design until all the requirement are understood and represented.
3. The Model is not suitable for new projects because of uncertainty in the specification.

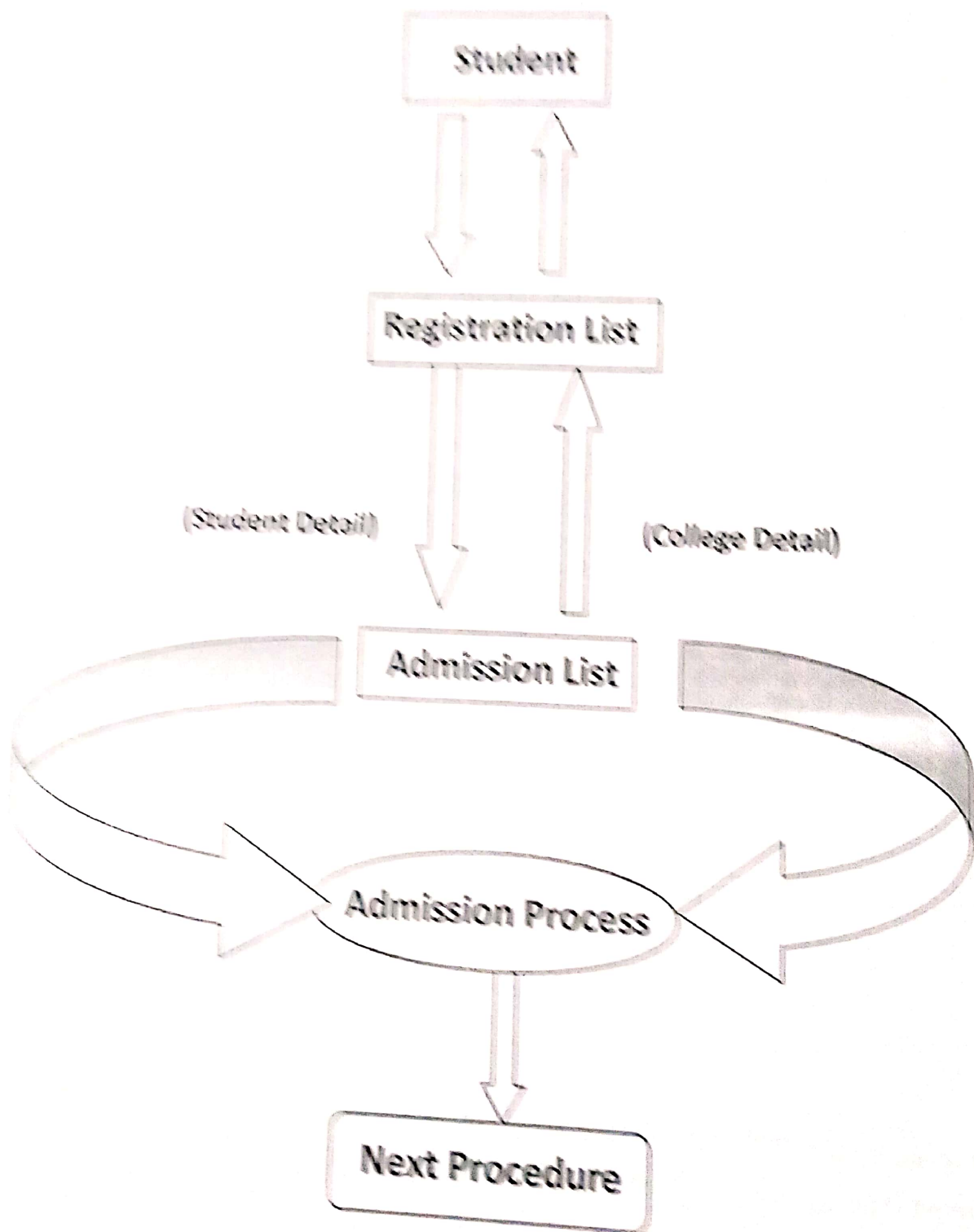
Flow Chart







DATA FLOW DIAGRAM (DFD)



Equipment Configuration

Technical Requirement :

- a) Hardware Requirement : We Require 128 MB RAM for VB and MS-Access. 20 GB Hard Dist. Pentium IV or Equivalent Processor to Run Project Efficiently
- b) Software Requirement : MS-Visual Basic 6.0, MS-ACCESS 2007, OFFICE-2007 or OFFICE XP

Economical Requirement :

- a) We need complete Computer setup with printer. Cost of this project is Rs.50,000/-
- b) We need Stationary like printing papers of reports, floppy or CD for back of data, ink for printer which is included in cost of project.

Environmental Requirement :

- a) Staff must have knowledge of how to operate computer. Printing of data-reports, to save data to get back up copies on floppy or CD. This training must be given to staff.

Implementation Languages (VB & MS Access)

Visual Basic

Visual Basic is available in three versions, each geared to meet a specific set of development requirements.

The visual basic learning edition allows programmers to easily create powerful applications for Microsoft windows NT. It includes all intrinsic controls, plus grid, tab and data bound controls, Documentations provided with this edition includes the learned VB and Cd plus the Microsoft developers network (MSDNTM) library CD's containing full online documentation.

The enterprise edition allows professionals to create robust distributed application in a team setting. It includes all the features professional edition, plus Back office tools such as SQL servers, Microsoft transaction server, internet information server visual Source Safe, SNA server and more. Printed documentation provided with enterprise edition includes the visual studio. Enterprise feature book plus Microsoft developers network CD's containing full online documentation.

The structure of Visual Basic Project

Form module :

Form module can contain textual descriptions of the form and its controls, including there property settings. They can also contain form level declaration of constant, variables and external procedure , event procedures general procedures.

Class Module :

Class modules are similar to form modules, accept that they have no visible user interface. You can use class module.

To create your own object, including code for method and properties.

MS ACCESS :

A database is collection of related and ordered information, organized in such a way that information can be accessed quickly and easily. A database can consist of one or more tables of information that are related in some way. A Database Management System (DBMS) is a computer based system to manage a database, or a collection of database or files.

Microsoft Access 200 is a DBMS package from Microsoft. It is one of the products in the Microsoft office 2000 suite. MS-ACCESS supports RDBMS features like setting primary key, relations between more than one table. Designing queries, reports.

- A primary key is to be set to a table, which uniquely identifies each record.
- A query that displays specified columns and data from a tables(s).

MS-ACCESS supports SQL-Structured query language which is used to manipulate data, retrieve data from more than one tables or for query designing.

The three basic functions provided by SQL are :

1. Data Definition Language (DDL) : Consist of commands to crate the objects such as tables, views, indexes etc. E.g. Create, alter these are the various commands which are used in defining of the database (database many be anyone but we are using MS-ACCESS)
2. Data Manipulation Language (DML) : Uses for query, insertion, deletion and updation of data stored in the database.
E.g. delete, update etc are the data manipulation commands which we use for the manipulation of data into our database
3. Data Control Language (DCL) : This is used for controlling data and their access to the database.

E.g. Select and grant etc is the examples of data control language.

Splash Screen

```
Private Sub CmdCancel_Click ()
```

```
    Unload Me
```

```
End Sub
```

```
Private Sub cmdOK_Click()
```

```
    If txtPassword.Text = "786" Then
```

```
        Unload Me
```

```
    Else
```

```
        MsgBox"Password Invalid !!!!!!!!!!!!!!!!!!!!!!!", vbInformation, "Error"
```

```
    End if
```

```
End Sub
```


MDI Form

```
Private Sub help_Click()
```

```
End Sub
```

```
Private Sub MDIForm_Load()
```

```
End Sub
```

```
Private Sub MnAboutus_Click()
```

```
    frmAbout.Show
```

```
End Sub
```

```
Private Sub MnAdmission_Click()
```

```
    frmAdmission.Show
```

```
End Sub
```

```
Private Sub MnClasses_Click()
```

```
    frmClasses.Show
```

```
End Sub
```

```
Private Sub MnExit_Click()
```

```
    Unload Me
```

```
End Sub
```

```
Private Sub MnFeesDetails_Click()
```

```
    frmFeesDetails.Show
```

```
End Sub
```

```
Private Sub MnHostelDetails_Click()
```

```
    frmHostelDetails.Show
```

```
End Sub
```

```
Private Sub MnRegistration_Click()
```

```
    frmRegistration.Show
```

```
End Sub
```

```
Private Sub MnRptAdmission_Click()
```

```
    RptAdmission.Show
```

```
End Sub
```

```
Private Sub MnRptClassess_Click()
```

```
    MnRptClassess.Show
```

```
End Sub
```

```
Private Sub MnRptClasses_Click()
```

```
    RptClasses.Show
```

```
End Sub
```

```
Private Sub MnRptFees_Click()
```

```
    RptFeesDetails.Show
```

```
End Sub
```

```
Private Sub MnRptHostel_Click()
```

```
    RptHostelDetails.Show
```

End Sub

Private Sub MnRptRegistration_Click()

RptRegistration.Show

End Sub

Private Sub MnRptStaff_Click()

RptStaff.Show

End Sub

Private Sub MnRptSubject_Click()

RptSubjectDetails.Show

End Sub

Private Sub MnStaffDetails_Click()

frmStaff.Show

End Sub

Private Sub MnSubjectDetails_Click()

frmSubjectDetails.Show

End Sub

Student Registration Form

Private Sub BtnUp_Click(Index As Integer)

If index = 0 Then

AddNew

ElseIf index = 1 Then

Modify

FillList

ElseIf index = 2 Then

Delete

FillList

ElseIf index = 3 Then

Save

FillList

ElseIf Index = 4 Then

Cancel

FillList

ElseIf Index = 5 Then

Unload Me

End Sub

```
Private Sub BtnUp_MouseDown(Integer, Button As Integer, Shift As Integer, X As  
Single, Y As Single)
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = False
```

```
        lblBtn(index).FontBold = True
```

```
        lblBn(Index).ForeColor = vbWhite
```

```
    End If
```

```
End Sub
```

```
Private Sub BtnUp_MouseUp(index As Integer, Button As Integer, Shift As Integer,X  
As Single, Y as Single)
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = True
```

```
        lblBtn(Index)FontBold = False
```

```
        lblBn(Index).ForeColor = vbBlack
```

```
    End If
```

```
    If index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
Private Sub Timer1_Time()
```

```
    lblFld(11).Caption = "Time:" + Str(Time)
```

```
End Sub
```

```
Function AddNew()
```

```
    On Error GoTo hand
```

```
        ADO.Recordset.AddNew
```

```
hand:
```

```
    If Err.Number <> 0 Then
```

```
        MsgBox Err.Description, vbInformation, "Error"
```

```
    End If
```

```
End Function
```

```
Function Modify()
```

```
    On Error GoTo hand:
```

```
        ADO.Recordset.Update
```

```
        ADO.Refresh
```

```
Hand:
```

```
    If Err.Number <> 0 Then
```

```
        MsgBox Err.Description, vbInformation, "Error"
```

```
End Function
```

```
Function Save ()
```


On Error GoTo hand :

ADO.Recordset.Save

ADO.Refresh

Hand :

If Err.Numbr<> 0 Then

MsgBox Err.Description,vbInformation, "Error"

End If

End Function

Function Delete()

On Error GoTo hand :

ADO.Recordset.Delete

ADO.Refresh

Hand :

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Fuction

Foundation Cancel()

On Error GoTo hand :

ADO.Recordset.Cancel

ADO.Refresh

Hand :

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Fuction

Funtion FillList()

ADO.Refresh

ADO.Recordset.MoveFirst

ADO.Recordset.MoveLast

ADO.Refresh

LstFld.Clear

While Not ADO.Recordset.EOF

LstFld.AddItem ADO.Recordset.Fields(0). Value

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Fuction

Student Admission Form

Private Sub Btnup_Click(Index As Integer)

 If Index = 0 Then

 AddNew

 Elseif Index = 1 Then

 Modify

 FillList

 Elseif Index = 2 Then

 Delete

 FillList

 Elseif Index = 3 Then

 Save

 FillList

 Elseif Index = 4 Then

 Cancel

 FillList

 Elseif Index = 5 Then

 Unload Me

End If

End Sub




```
Private Sub BtnUp_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X  
As Single, Y As Single )
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = False
```

```
        lblBtn(Index).FontBold = True
```

```
        lblBtn(Index).ForeColor = vbWhite
```

```
    End If
```

```
End Sub
```

```
Private Sub BtnUp_MouseUp(Index As Integer, Button As Integer, Shift As Integer, X As  
Single, Y As Single.
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = True
```

```
        lblbtn(Index).FontBold = False
```

```
        lblBtn(Index).ForeColor = vbBlackk
```

```
    End If
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

Modify

FillList

ElseIf Index = 2 Then

Delete

FillList

ElseIf Index = 3 Then

Save

FillList

ElseIf Index = 4 Then

Cancel

FillList

ElseIf Index = 5 Then

Unload Me

End If

End Sub

Private Sub Form_Load()

lblFid(12).Caption = "Date: " + Str(Date)

FillList

End Sub

```
Private Sub lblBtn_MouseDown(Index AS Integer, Button As Integer, Shift As Integer, X  
As Single, Y As Single)
```

```
    BtnUp(Index).Visible = False
```

```
    lblBtn(Index).FontBold = True
```

```
    lblBtn(Index).ForeColor = vbWhite
```

```
End Sub
```

```
Private Sub lblBtn_MouseUp(Index AS Integer, Button As Integer, Shift As Integer, X As  
Single, Y As Single.
```

```
    BtnUp(Index).Visible = True
```

```
    lblBtn(Index).FontBold = False
```

```
    lblBtn(Index).ForeColor = vbBlack
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```


Elseif Index = 3 Then

Save

FillList

Elseif Index = 4 Then

Cancel

FillList

Elseif Index = 5 Then

Unload Me

End If

End Sub

Private Sub LstFld_Click()

ADO.Refresh

While Not ADO.Recordset.EOF

If LstFld.Text = ADO.Recordset.Fields(0). Value Then

Exit Sub

End If

ADO.Recordset.MoveNext

Wend

ADO.Refresh



End Sub

Private Sub Timer1_Timer()

 lblFld(11).Caption = "Time : " + Str(Time)

End Sub

Function AddNew()

 On Error GoTo hand :

 ADO.Recordset.AddNew

Hand :

 If Err.Number<> 0 Then

 MsgBox Err.Description,vbInformation, "Error"

 End If

End Fucntion

Function Modify()

 On Error GoTo hand:

 ADO.Recordset.Update

 ADO.Refresh

hand:

 If Err.Number<> 0 Then

 MsgBox Err.Description,vbInformation, "Error"

End If

End Function

Function Save()

On Error GoTo hand:

ADO.Recordset.Save

ADO.Refresh

hand:

If Err.Number<>0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function Delete ()

On Error GoTo hand:

ADO.Recordset.Delete

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function Cancel()

On Error Goto hand:

ADO.Recordset.Cancel

ADO.Refresh

hand:

If Err.Number <> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function FillList()

ADO.Refresh

ADO.Recordset.MoveFirst

ADO.Recordset.MoveLast

ADO.Refresh

LstFld.Clear

While Not ADO.Recordset.EOF

LstFld.AddItem ADO.Recordset.Fields(0).Value

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Function

Fees Details

Private Sub BtnUp_Click(Index As Integer)

 If Index = 0 Then

 AddNew

 ElseIf Index = 1 Then

 Modify

 FillList

 ElseIf Index = 2 Then

 Delete

 FillList

 ElseIf Index = 3 Then

 Save

 FillList

 ElseIf Index = 4 Then

 Cancel

 FillList

 ElseIf Index = 5 Then

 Unload Me

End If

End Sub

Private Sub BtnUp_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X As Single, Y As Single,

 If Button = 1 Then

 BtnUp(Index).Visible = False

 lblBtn(Index).FontBold = True

 lblBtn(Index).ForeColor = vbWhite

 End If

End Sub

Private Sub BtnUp_MouseUp(Index As Integer, Button As Integer, Shift As Integer, X As Single, Y as Single)

 If Button = 1 Then

 BtnUp(Index).Visible = True

 lblBtn(Index).FontBold = False

 lblBtn(Index).ForeColor = vbBlack

 End If

If Index = 0 Then

 AddNew

Elseif Index = 1 Then

Modify

FillList

Elseif Index = 2 Then

Delete

FillList

Elseif Index = 3 Then

Save

FillList

Elseif Index = 4 Then

Cancel

FillList

Elseif Index = 5 Then

Unload Me

End If

End Sub

Private Sub Form_Load()

 IbIFld(12).Caption = "Date: " + Str(Date)

 FillList

End Sub



```
Private Sub lbIBtn_MouseDown(Index As Integer, Button As Integer, Shift AS Integer, X  
As Single, Y As Single)
```

```
    BtnUp(Index).Visible = False
```

```
    lbIBtn(Index).FontBold = True
```

```
    lbIBtn(Index).ForeColor=vbWhite
```

```
End Sub
```

```
Private Sub lbIBtn_MouseUp(Index As Integer, Button As integer, Shift As Integer, X as  
Single, Y As Single.
```

```
    BtnUp(Index).Visible = True
```

```
    lbIBtn(Index).FontBold = False
```

```
    lbIBtn(Index).ForeColor = vbBlack
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```

Elseif Index = 3 Then

Save

FillList

Elseif Index = 4 Then

Cancel

FillList

Elseif Index = 5 Then

Unload Me

End If

End Sub

Private Sub LstFld_Click()

ADO.Refresh

While Not ADO.Recordset.EOF

If LstFld.Text = ADO.Recordset.Fields(0).Value Then

Exit Sub

End If

ADO.Recordset.MoveNext

Wend

ADO.Refresh



End Sub

Private Sub Timer1_Timer()

lblFld(11).Caption = "Time:" + Str(Time)

End Sub

Function AddNew()

On Error GoTo hand:

ADO.Recordset.AddNew

hand:

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function Modify()

On Error GoTo hand:

ADO.Recordset.Update

ADO.Refresh

hand:

If Err.Number<>0 Then

MsgBox Err.Description,vbInformation, "Error"

End If

End Function

Function Save()

On Error GoTo hand:

ADO.Recordset.Save

~~ADO.Refresh~~

hand:

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function Delete()

On Error GoTo hand:

ADO.Recordset.Delete

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Description ,vbInformation, "Error"

End If

End Function

Function Cancel()

On Error GoTo hand:

ADO.Recordset.Cancel

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Description,vbInformation, "Error"

End If

End Function

Function FillList()

ADO.Refresh

ADO.Recordset.MoveFirst

ADO.Recordset.MoveLast

ADO.Refresh

LstFld.Clear

While Not ADO.Recordset.EOF

LstFld.AddItem ADO.Recordset.Fields(0).Value

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Function

Subject Detail

```
Private Sub BtnUp_Click(Index As Integer)
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```

```
    ElseIf Index = 3 Then
```

```
        Save
```

```
        FillList
```

```
    ElseIf Index = 4 Then
```

```
        Cancele
```

```
        FillList
```

```
    ElseIf Index = 5 Then
```

```
        Unload Me
```

```
    End If
```

End Sub

Private Sub BtnUp_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X As Single, Y As Single)

 If Button = 1 Then

 BtnUp(Index).Visible = False

 lblBtn(Index).FontBold = True

 lblBtn(Index).ForeColor = vbWhite

 End If

End Sub

Private Sub BtnUp_Mouseup(Index AS Integer, Button As Integer, Shift AS Integer, X As Single, Y As Single)

 If Button = 1 Then

 BtnUp(Index).Visible = False

 lblBtn(Index).FontBold = True

 lblBtn(Index).ForeColor = vbWhite

 End If

 If Index = 0 Then

 AddNew

 ElseIf Index = 1 Then

 Modify

FillList

ElseIf Index = 2 Then

Delete

FillList

ElseIf Index = 3 Then

Save

FillList

ElseIf Index = 4 Then

Cancle

FillList

ElseIf Index = 5 Then

Unloadd Me

End If

End Sub

Private Sub Form_Load()

lblFld(12).Caption = "Date : " + Str(Date)

FillList

End Sub

```
Private Sub lblBtn_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X  
As Single, Y As Single)
```

```
    BtnUp(Index).Visible = False
```

```
    lblBtn(Index).FontBold = True
```

```
    lblBtn(Index).ForeColor = vbWhite
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```

```
    ElseIf Index = 3 Then
```

```
        Save
```

```
        FillList
```

```
    ElseIf Index = 4 Then
```

```
        Cancele
```

```
        FillList
```

```
    ElseIf Index = 5 Then
```

Unload Me

End If

End Sub

Private Sub LstFld_Click()

ADO.Refresh

While Not ADO.Recordset.EOF

If LstFld.Text = ADO.Recordset.Fields(0).Value Then

Exit Sub

End If

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Sub

Private Sub Time1_Timer()

IbIFld(11).Caption = "Time:" + Str(Time)

End Sub

Function AddNew()

On Error GoTo hand:

ADO.Recordset.AddNew

hand :

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Fuction

Function Modify()

On Error GoTo hand :

ADO.Recordset.Update

ADO.Refresh

hand:

If Err.Number<>0 Then

MsgBox Err.Description, vbInformation, "Error",

End If

End Function

Function Save()

On Error GoTo hand:

ADO.Recordset.Save

ADO.Refresh

hand:


```
If Err.Number<>0 Then
```

```
    MsgBox Err.Description, vbInformation, "Error"
```

```
End If
```

```
End Function
```

```
Function Delete()
```

```
    On Error GoTo hand:
```

```
        ADO.Recordset.Delete
```

```
        ADO.Refresh
```

```
hand:
```

```
    If Err.Number<> 0 Then
```

```
        MsgBox Err.Description, vbInformation, "Error"
```

```
    End If
```

```
End Function
```

```
Function Cancel()
```

```
    On Error GoTo hand:
```

```
        ADO.Recordset.Cancel
```

```
        ADO.Refresh
```

```
hand:
```

```
    If Err.Number<> 0 Then
```

```
        MsgBox Err.Description, vbInformation, "Error"
```

End If

End Function

Function FillList()

ADO.Refresh

ADO.Recordset.MoveFirst

ADO.Recordset.MoveLast

ADO.Refresh

LstFld.Clear

While Not ADO.Recordset.EOF

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Fuction



Hostel Details

```
Private Sub BtnUp_Click(Index As Integer)
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```

```
    ElseIf Index = 3 Then
```

```
        Save
```

```
        FillList
```

```
    ElseIf Index = 4 Then
```

```
        Cancele
```

```
        FillList
```

```
    ElseIf Index = 5 Then
```

```
        Unload Me
```

```
    End If
```

```
End Sub
```

```
Private Sub BtnUp_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X  
As Single, Y As Single.
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = False
```

```
        lblBtn(Index).FontBold = True
```

```
        lblBtn(Index).ForeColor = vbWhite
```

```
    End If
```

```
End Sub
```

```
Private Sub BtnUp_MouseUp(Index As Integer, Button As Integer, Shift As Integer, X As  
Single, Y As Single)
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = True
```

```
        lblBtn(Index).FontBold = False
```

```
        lblBtn(Index).ForeColor = vbBlack
```

```
    End If
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```


Elseif Index = 2 Then

Delete

FillList

Elseif Index = 3 Then

Save

FillList

Elseif Index = 4 Then

Cancle

FillList

Elseif Index = 5 Then

Unload Me

End If

End Sub

Private Sub Form_Load()

lblFld(12).Caption = "Date :" + Str(Date)

FillList

End Sub

```
Private Sub lblBtn_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X  
As Single, Y As Single.
```

```
    BtnUp(Index).Visible = False
```

```
    lblBtn(index).FontBold = True
```

```
    lblBtn(Index).ForeColor = vbWhite
```

```
End Sub
```

```
Private Sub lblBtn_MouseUp(Index As Integer, Button As Integer, Shift As Integer, X As  
Single, Y As Single)
```

```
    BtnUp(Index).Visible = False
```

```
    lblBtn(index).FontBold = True
```

```
    lblBtn(Index).ForeColor = vbWhite
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```

```
    ElseIf Index = 3 Then
```

Save

FillList

Else If Index = 4 Then

Cancel

FillList

Else If Index = 5 Then

Unload Me

End If

End Sub

Private Sub LstFld_Click()

ADO.Refresh

While Not ADO.Recordset.EOF

Exit Sub

End If

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Sub

```
Private Sub Timer1_Timer()
```

```
    lblFld(11).Caption = "Time :" + Str(Time)
```

```
End Sub
```

```
Function AddNew()
```

```
    On Error GoTo hand :
```

```
        ADO.Recordset.AddNew
```

```
hand:
```

```
    If Err.Number<> 0 Then
```

```
        MsgBox Err.Description, vbInformation, "Error"
```

```
    End If
```

```
End Function
```

```
Function Modify()
```

```
    On Error GoTo hand:
```

```
        ADO.Recordset.UpDate
```

```
        ADO.Refresh
```

```
hand:
```

```
    If Err.Number<>0 Then
```

```
        MsgBox Err.Description,vbInformation, "Error"
```

```
    End If
```

```
End Function
```


Function Save()

On Error GoTo hand:

ADO.Recordset.Save

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Description,vbInformation, "Error"

End If

End Function

Function Delete()

On Error GoTo hand:

ADO.Recordset.Delete

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function Cancel()

On Error GoTo hand:

ADO.Recordset.Cancel

ADO.Refresh

hand :

If Err.Number<>0 Then

MsgBox Err.Description,vbInformation, "Error"

End If

End Function

Function FillList()

ADO.Refresh

ADO.Recordset.MoveFirst

ADO.Recordset.MoveLast

ADO.Refresh

LstFld.Clear

While Not ADO.Recordset.EOF

LstFld.AddItem ADO.Recordset.Fields(0).Value

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Function

Staff Details

```
Private Sub BtnUp_Click(Index As Integer)
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```

```
    ElseIf Index = 3 Then
```

```
        Save
```

```
        FillList
```

```
    ElseIf Index = 4 Then
```

```
        Cancele
```

```
        FillList
```

```
    ElseIf Index = 5 Then
```

```
        Unload Me
```

```
    End If
```

```
End Sub
```

```
Private Sub BtnUp_MouseDown(Index AS Integer, Button As Integer, Shift As Integer, X  
As Single, Y As Single)
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = False
```

```
        lblBtn(Index).FontBold = True
```

```
        lblBtn(Index).ForeColor = vbWhite
```

```
    End If
```

```
End Sub
```

```
Private Sub BtnUp_MouseUp(Index As Integer, Button As Integer, Shift As Integer, X As  
Single, Y As Integer)
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = False
```

```
        lblBtn(Index).FontBold = True
```

```
        lblBtn(Index).ForeColor = vbBlack
```

```
    End If
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```


Delete

FillList

Elseif Index = 3 Then

Save

FillList

Elseif Index = 4 Then

Cancel

FillList

Elseif Index = 5 Then

Unload Me

End If

End Sub

Private Sub Form_Load()

lblFld(12).Caption = "Date:" + Str(Date)

FillList

End Sub

Private Sub lblBtn_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X As Single, Y As Single)

```
BtnUp(Index).Visible = False  
  
lblBtn(Index).FontBold = True  
  
lblBtn(Index).ForeColor = vbWhite  
  
End Sub
```

```
Private Sub lblBtn_MouseUp(Index As Integer, Button As Integer, Shift As Integer, X As  
Single, Y As Single)
```

```
    BtnUp(Index).Visible = False  
  
    lblBtn(Index).FontBold = True  
  
    lblBtn(Index).ForeColor = vbBlack  
  
    If Index = 0 Then  
  
        AddNew  
  
    ElseIf Index = 1 Then  
  
        Modify  
  
        FillList  
  
    ElseIf Index = 2 Then  
  
        Delete  
  
        FillList  
  
    ElseIf Index = 3 Then  
  
        Save
```

FillList

ElseIf Index = 4 Then

Cancle

FillList

ElseIf Index = 5 Then

Unload Me

End If

End Sub

Private Sub LstFld_Click()

ADO.Refresh

While Not ADO.Recordset.EOF

If LstFld.Text = ADO.Recordset.Fields(0).Value Then

Exit Sub

End If

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Sub



```
Private Sub Timer1_Timer()
```

```
    lblFld(11).Caption = "Time:" + Str(Time)
```

```
End Sub
```

```
Function AddNew()
```

```
    On Error GoTo hand:
```

```
        ADO.Recordset.AddNew
```

```
hand:
```

```
    If Err.Number<>0 Then
```

```
        MsgBox Err.Description, vbInformation, "Error"
```

```
    End If
```

```
End Function
```

```
Function Modify()
```

```
    On Error GoTo hand:
```

```
        ADO.Recordset.Update
```

```
        ADO.Refresh
```

```
hand:
```

```
    If Err.Number<>0 Then
```

```
        MsgBox Err.Description,vbInformation, "Error"
```

```
    End If
```

```
End Function
```

Function Save()

On Error GoTo hand:

ADO.Recordset.Save

ADO.Refresh

hand:

If Err.Number<>0 Then

MsgBox Err.Description, vbInformation, " Error"

End If

End Function

Function Delete()

On Error GoTo hand:

ADO.Recordset.Delete

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Deshcription, vbInformation, "Error"

End If

End Function

Function Cancel()

On Error GoTo hand:

ADO.Recordset.Cancel

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function FillList()

ADO.Refresh

ADO.Recordset.MoveFirst

ADO.Recordset.MoveLast

ADO.Refresh

LstFld.Clear

While Not ADO.Recordset.EOF

LstFld.AddItem ADO.Recordset.Fields(0).Value

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Function

Classes Detail

```
Private Sub BtnUp_Click(Index As Integer)
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```

```
    ElseIf Index = 3 Then
```

```
        Save
```

```
        FillList
```

```
    ElseIf Index = 4 Then
```

```
        Cancele
```

```
        FillList
```

```
    ElseIf Index = 5 Then
```

```
        Unload Me
```

```
    End If
```

```
End Sub
```



```
Private Sub BtnUp_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X  
As Single, Y As Single)
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = False
```

```
        lblBtn(Index).FontBold = True
```

```
        lblBtn(Index).ForeColor = vbWhite
```

```
    End If
```

```
End Sub
```

```
Private Sub BtnUp_MouseUp(Index As integer, Button As Integer, Shift As Integer, X As  
Single, Y As Single)
```

```
    If Button = 1 Then
```

```
        BtnUp(Index).Visible = False
```

```
        lblBtn(Index).FontBold = True
```

```
        lblBtn(Index).ForeColor = vbBlack
```

```
    End If
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

Elseif Index = 2 Then

Delete

FillList

Elseif Index = 3 Then

Save

FillList

Elseif Index = 4 Then

Cancle

FillList

Elseif Index = 5 Then

Unload Me

End If

End Sub

Private Sub Form_Load()

IbIFld(12).Caption = "Date:" + Str(Date)

FillList

End Sub



```
Private Sub lblBtn_MouseDown(Index As Integer, Button As Integer, Shift As Integer, X  
As Single, Y As Single)
```

```
    BtnUp(Index).Visible = False
```

```
    lblBtn(Index).FontBold = True
```

```
    lblBtn(Index).ForeColor = vbWhite
```

```
End Sub
```

```
Private Sub lblBtn_MouseUp(Index As Integer, Button As Integer, Shift As Integer, X As  
Single, Y As Single)
```

```
    BtnUp(Index).Visible = False
```

```
    lblBtn(Index).FontBold = True
```

```
    lblBtn(Index).ForeColor = vbBlack
```

```
    If Index = 0 Then
```

```
        AddNew
```

```
    ElseIf Index = 1 Then
```

```
        Modify
```

```
        FillList
```

```
    ElseIf Index = 2 Then
```

```
        Delete
```

```
        FillList
```

```
    ElseIf Index = 3 Then
```


Save

FillList

ElseIf Index = 4 Then

Cancle

FillList

ElseIf Index = 5 Then

Unload Me

End If

End Sub

Private Sub LstFld_Click()

ADO.Refresh

While Not ADO.Recordset.EOF

If LstFld.Text=ADO. Recordset.Fields(0).Value Then

Exit Sub

End If

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Sub



```

Private Sub Timer1_Timer()

    lblfld(11).Caption = "Time:" + Str(Time)

End Sub

Function AddNew()

    On Error GoTo hand:

        ADO.Recordset.AddNew

hand:

    If Err.Number <> 0 Then

        MsgBox Err.Description, vbInformation, "Error"

    End If

End Function

Function Modify()

    On Error GoTo hand:

        ADO.Recordset.Update

        ADO.Refresh

hand:

    If Err.Number <> 0 Then

        MsgBox Err.Description, vbInformation, "Error"

    End If

End function

```

Function Save()

On Error GoTo hand:

ADO.Recordset.Save

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function Delete()

On Error GoTo hand:

ADO.Recordset.Delete

ADO.Refresh

hand:

If Err.Number<> 0 Then

MsgBox Err.Description, vbInformation, "Error"

End If

End Function

Function Cancel()

On Error GoTo hand:

ADO.Recordset.Cancel

ADO.Refresh

hand:

IF Err.Number<> 0 Then

MsgBox Err.Description,vbInformation, "Error"

End If

End Fuction

Function FillList()

ADO.Refresh

ADO.Recordset.MoveFirst

ADO.Recordset.MoveLast

ADO.Refresh

LstFld.Clear

While Not ADO.Recordset.EOF

LstFld.AddItem ADO.Recordset.Fields(0).Value

ADO.Recordset.MoveNext

Wend

ADO.Refresh

End Function

About Us

```
Private Sub cmdOk_Click()
```

```
    Unload Me
```

```
End Sub
```

```
Private Sub Form_Load()
```

```
    Time1.Enabled = True
```

```
End Sub
```

```
Private Sub Timer1_Timer()
```

```
    If lblFld(11).Left >= 9700 Then
```

```
        lblFld(11).Left = 4000
```

```
    End If
```

```
    lblFld (11).Left = lblFld(11).Left +100
```

```
End Sub
```


KAMLA NEHRU POLYTECHNIC PHARMACY

Student Registration

Registration ID :	R1000
Student Name :	Mr.Javeed
DOB :	06/05/1989
Gender :	Male
12 th Percentage :	80%
12 th CET Marks	260
College Name :	Sir Sayyed College Aurangabad
Address :	Plot No A-20 , Kat Kat Gate

KAMLA NEHRU POLYTECHNIC PHARMACY

Admission Form

Registration ID :	R1000
Student ID :	S1000
Student :	Mr. Mohsin
DOB	11/02/1988
Gender:	Male
12 th Percentage :	75%
12 th CET Marks	250
College Name :	Maulana Azad College Aurangabad
10 th TC :	Yes
12 th TC :	Yes
10 th & 12 th	Yes
Domicile	Yes
Admission:	12/01/2011
Contact 1 :	9890098900
Contact 2:	9890012345
Address :	Rauzah Bagh, Beside MAC Aurangabad



KAMLA NEHRU POLYTECHNIC PHARMACY

Fees Details

Student ID :	S1000
Student Name :	Mr Mohsin
Class :	1 D-Pharmacy
Running Year	1 Year
PaidDate :	07/02/2011
Admission	1000
Tuition Fee :	2000
Exam Fee :	500
Hostel Fee :	4000
Total Fees :	75000

KAMLA NEHRU POLYTECHNIC PHARMACY

Hostel Details

Student ID :	S1000
Student :	Mr Mohsin
Class :	Pharamacy DE
Running Year :	1 St Year
Building No :	1
Room No :	125
Bed No :	45
Fan :	45
Fan :	Yes
Room Shared :	NO

KAMLA NEHRU POLYTECHNIC PHARMACY

Subject Details

Student ID :	S1000
Student :	Mr Mohsin
Class :	3 rd Year
Running Year :	1 st Year
Subject 1 :	JAVA
Subject 2 :	VB 6.0
Subject 3 :	Unix
Subject 4 :	C Programming
Subject 5 :	VB.net 2010
Subject 6 :	SQL Server 2010

KAMLA NEHRU POLYTECHNIC PHARMACY

Staff Details

LectureId	:	1
Name	:	Mr. XYZ
Qualification	:	MCA
Subjects	:	C,C++,VB,SQL
Classes	:	BCA, BCS

KAMILA NEHRU POLYTECHNIC PHARMACY

Classes Details

LectureId	:	1
Name	:	Mr. XYZ
1_Period_Time	:	8 to 9
1_Subject_Name	:	C
1_Class_Name	:	MCA
2_Period_Time	:	11 To 12
2_Subject_Name	:	C++
2_Class_Name	:	BCA
3_Period_Time	:	2 to 3
3_Subject_Name	:	VB 6.0
3_Class_Name	:	BCS

Future Work

1. User Wants to capable of Storing all types of information present in any pharmacy College.
2. All the student record can be update easily.
3. All courses detail can be store without making any kinds of duplication.
4. Records can be set with current date and time.
5. Previous record can be easily seen.

Conclusion

This project is based on the software for "Pharmacy Management System" Which contains the complete information of management of a "Pharmacy College"

This project is very convenient for use, Therefore it is user friendly. It automatically generates the daily reports, Which helps the college Administration to know the transaction taking place regularly. Another plus point of project is its ability to keep the records.

We have design our system which is very useful to any "Pharmacy College". Our software is user friendly and user will find it easy to use. When any system is implemented the user's satisfactions should be given first priority for the system. The system gets satisfaction when the is satisfied, this system has tried its best to reduce physical effort of the user of "Pharmacy college" and Satisfied them in this generation, facilities of the system has totally eliminated the chances for wrong data entry.

Our project consist of main [MD | Form] in that there are many options. To access the another form. The main form proofs a helping hand as it is allows accessing same of the form directly form using the tab control, which it return saves the time required for accessing each form individually.

The new records can be added, saved as well as it is also possible to view previous and next record and delete and cancel a record if created unnecessarily. The project contains all the requirement needed till today by any pharmacy colleges and is used by them very easily here we conclude our project by thanking our respected helpful teachers.

REFERENCE

The information we have collected for developing our project "Pharmacy Management System" in Visual Basic, We discussed the information from the reputed college. We also discussed with our guide Mrs. Naziya Madam about information of "Pharmacy Database"

The few books and modes of information, which help us in making our project, the successful are :

- Visual Basic
 - Scott Warner
 - Peter Atkins
- DBMS
 - Bipin Desai
 - Korth
- Software Engineering
 - Roger Pressmen
 - Pankaj Jalote

Experts Opinion

