Project

on

STUDENT ATTENDENCE CHECK

Submitted in partial fulfillment of the

requirement for the award of the degree of

Bachelor of Technology in Computer Science and Engg.

Under The Supervision of
Mr. Arjun KP
Associate Professor

Submitted By

Rajan Kumar(18SCSE1010262)
Rahul Kumar(18SCSE1010261)

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING /
DEPARTMENT OF COMPUTERAPPLICATION
GALGOTIAS UNIVERSITY, GREATER NOIDA
INDIA
DECEMBER, 2021

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING
GALGOTIAS UNIVERSITY, GREATER NOIDA

CANDIDATE'S DECLARATION

I/We hereby certify that the work which is being presented in the thesis/project/dissertation, entitled **"STUDENT ATTENDENCE CHECK"** in partial fulfillment of the requirements for the award of the Bachelor of Technology submitted in the School of Computing Science and Engineering of Galgotias University, Greater Noida, is an original work carried out during the period of month, Year to Month and Year, under the supervision of Name… Designation, Department of Computer Science and Engineering/Computer Application and Information and Science, of School of Computing Science and Engineering , Galgotias University, Greater Noida

The matter presented in the thesis/project/dissertation has not been submitted by me/us for the award of any other degree of this or any other places.

Rajan Kumar(18SCSE1010262)
Rahul Kumar(18SCSE1010261)

This is to certify that the above statement made by the candidates is correct to the best of my knowledge.

Mr. Arjun KP

Associate Professor

CERTIFICATE

The Final Thesis/Project/ Dissertation Viva-Voce examination of Rajan Kumar(18SCSE1010262), Rahul Kumar(18SCSE1010261) has been held on __________________ and his/her work is recommended for the award of Bachelor of Technology~~.~~

Signature of Examiner(s) **Signature of Supervisor(s)**

Signature of Project Coordinator **Signature of Dean**

Date: December, 2021

Place: Greater Noida

Abstract

Attendance management is important to every single organization; it can decide whether or not an organization such as educational institutions, public or private sectors will be successful in the future. Organizations will have to keep a track of people within the organization such as employees and students to maximize their performance. Managing student attendance during lecture periods has become a difficult challenge. The ability to compute the attendance percentage becomes a major task as manual computation produces errors, and wastes a lot of time. For the stated reason, an efficient Web-based application for attendance management system is designed to track student's activity in the class. This application takes attendance electronically and the records of the attendance are storing in a database. The system design using the Model, View, and Controller (MVC) architecture, and implemented using the power of Framework. JavaScript is adding to the application to improve the use of the system. MySQL used for the Application Database.The system designed in a way that can differentiate the hours of theoretical and practical lessons since the rate of them is different for calculating the percentages of the students' absence. Insertions, deletions, and changes of data in the system can do straightforward via the designed GUI without interacting with the tables. Different presentation of information is obtainable from the system. The test case of the system exposed that the system is working enormously and is ready to use to manage to attend students for any department of the University

Table of Contents

Introduction

Attendance Management System" is software developed for maintaining the attendance of the student on the daily basis in the collage. Here the staffs, who are handling the subjects, will be responsible to mark the attendance of the students. Each staff will be given with a separate username and password based on the subject they handle. An accurate report based on the student attendance is generated here. This system will also help in evaluating attendance eligibility criteria of a student. Report of the student's attendance on weekly and monthly basis is generated

Analysis can be defined as breaking up of any whole so as to find out their nature, function etc. It defines design as to make preliminary sketches of; to sketch a pattern or outline for plan. To plan and carry out especially by artistic arrangement or in a skillful wall. System analysis and design can be characterized as a set of techniques and processes, a community of interests, a culture and an intellectual orientation. The various tasks in the system analysis include the following. □ Understanding application. □ Planning. □ Scheduling. □ Developing candidate solution. □ Performing trade studies. □ Performing cost benefit analysis. □ Recommending alternative solutions. □ Selling of the system. □ Supervising, installing and maintaining the system. This system manages to the analysis of the report creation and develops manual entry of the student attendance. First design the students entry form , staff allocation and time table allocation forms. This project will helps the attendance system for the department calculate percentage and reports for eligibility criteria of examination .The application attendance entry system will provide flexible report for all students.

LITERATURE REVIEW

The purpose of developing attendance management system is to computerized the tradition way of taking attendance. Another purpose for developing this software is to generate the report automatically at the end of the session or in the between of the session.The design and implementation of the system is to provide service in institute and colleges. The system is to provide comprehensive student information system and user interface is to replace the current paper records. College Staff uploads attendance, results and college notifications through a secure, online interface using android devices. All data is thoroughly reviewed and validated on the server before actual record alteration occurs. The system plans for student user interface, allowing students to access tips and tricks as provided by their seniors. All data is stored securely on SQL servers managed by the college Administrator .

The rapid progress in mobile technology has created a new area which is knownas mobile learning. Mobile learning is the next generationof e-learning that leads attractive way of knowledge delivery especially used in teaching and learning process. With development of this Android application the student preferred to use mobile devices as technology support educational tool. This system is designed because notes dictation in the class is difficult considering semester duration, student might miss the exam and important notice displayed due to unawareness, chances of false marking of attendance is more due to more paper work done manual attendance entry, evaluation and report generation is tedious and time consuming job.

In this paper basic problem of student attendance management is defined which is traditionally taken manually by faculty. One alternative to make student attendance system automatic is provided by Computer Vision. In this paper we review the various computerized system which is being developed by using different techniques. Based on this review a new approach for student attendance recording and management is proposed to be used for various colleges or academic institutes. In[1] This mobile application will require connecting to the internet through Wi-Fi(Wireless Fidelity) technology or through GPRS(General Packet Radio Service).Lecturers will first have to sign up for this and then they can take attendance any time they wish by first logging in with the help of a smartphone to the server. After attendance has been taken lecturer will send it over to sever via GPRS. The lecturers can also enroll new students, delete information about a particular student, modify some information etc.

MODULE DETAILS

I. **Login form:** This Form is made for security purpose. So Authenticated User only Access in to this application. There are two Type of persons can enter in the project Administrator and User.

II. **Add information form:** This form we provide for student registration only and gives option to fill name of the student and password.

III. **Staff Module:** This module is designed for staff,which use mobile phone to take attendance, upload attendance result. The entered admin details are encrypted and sent to server for verification. Only after successful authentication the operations are performed.

IV. **Attendance Module**

i. Admin Registration The first step in this application is to get the teaching faculty to register. The respective person will then provide his or her phone's e-mail id and password for registration. An OTP would be then sent via e-mail address on the phone by the admin.

ii. **Admin Login**

After registering the admin is allowed to log in. He or she can now view admin homepage where there are options to take attendance, upload results

iii. **Take Attendance Here**

system will validate admin to check whether admin is applicable to take attendance for any subject which he/she selected from the application after validation is success. If admin is legitimate to take attendance and applying the operation at correct lecture time, now he can take attendance.

iv. **Upload Result**

Admin can upload student’s attendance through application. The same authentication will be performed by system as Take attendance module.

v. **Upload Final Attendance sheet**

A report is generated which has the student’s name, roll no., his or her attendance. This module categorized student according to their attendance.

DESIGN SCHEME

The program design of the system is mainly divided into four modules: fingerprint collection and comparison, establishment, reading and writing of database, network connection and the design of upper and lower computer interfaces. The overall structure of the system is shown in Figure 1. The fingerprint instrument adopts SM-2B series independent pressure feeling sensitive fingerprint identification module which launched by Hangzhou Zhongzheng company. VB and VC++ are used as

programming languages. MySQL is used as database .

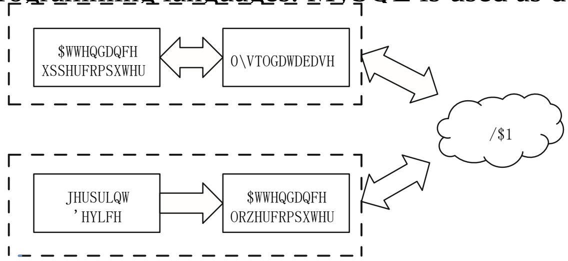


Fig. 1 The overall structure of the system

The teaching computer in the classroom is used for lower computer in the system which connects the fingerprint instrument through the USB interface. The functions of fingerprint information collection, comparison, identification, student’s attendance and teacher interface management are realized by the lower computer. The upper computer is located in the Educational Administration Office (or teaching supervisor office) of the school, which is mainly operated by the administrator to realize the query management of attendance. The communication between the lower computer and the upper computer is realized by LAN.

PROJECT DESCRIPTION

PROBLEM DEFINITION:

This system developed will reduce the manual work and avoid redundant data. By maintaining the attendance manually, then efficient reports cannot be generated. The system can generate efficient weekly,consolidate report based on the attendance. As the attendances are maintained in registers it has been a tough task for admin and staff to maintain for long time. Instead the software can keep long and retrieve the information when needed.

PROJECT OVERVIEW

Attendance Management System basically has two main modules for proper functioning

- ➢ Admin module is has rights for creating any new entry of faculty and student details.
- ➢ User has a rights of making daily attendance, generating report. Attendance report can be taken by given details of student details, date, class.

MODULE DESCRIPTION

The system should be designed in such a way that only authorized people should be allowed to access some particular modules. The records should be modified by only administrators and no one else. The user should always be in control of the application and not the vice versa.

The user interface should be consistent so that the user can handle the application with ease and speed. The application should be visually, conceptually clear.

ADMINISTRATOR MODULE:

- Student Details:

In this module deals with the allocation of roll no and personal details for new batch.It will generate of personal details of student and academic details of the students with the photos.

- Staff Details:

  - It helps to allot the subject and the subject code to the particular staffs.

  - It provides the facility to have a user name and password to the staffs .

- Time table details:

  - It will retrieve the subject information from the subject database and assign time table to the staffs.

  - It will help the admin, staff to make the entry of attendance based of the subject and period allotted to the respective staff.

- Attendance details:

  - It will be makes to the attendance database all students. Entered attendance to stored in the database subject ,period wise into the particular date.

  - It will help s to the get report of weekly and consolidate of the attendance.

Report details:

Report can be taken by daily,weekly and consolidate:

- ➢ weekly report get all hour details of attendance starting date to ending date and display the status
- ➢ Consolidate report get all student attendance details starting date to ending date status help for the eligibility criteria of the student to attend the examination.

STAFFS MODULE:

□ Attendance details:

- ➢ It assists the staff to mark attendance to the students for their subject. This will authenticate the staff before making the entry.

□ Report details:

1. weekly report get particular hour details of attendance from starting date to ending date and display the status .
2. consolidate report get all student attendance details from starting date to ending date status help for the eligibility criteria of the student to attend the examination

SYSTEM FLOW DIAGRAM:

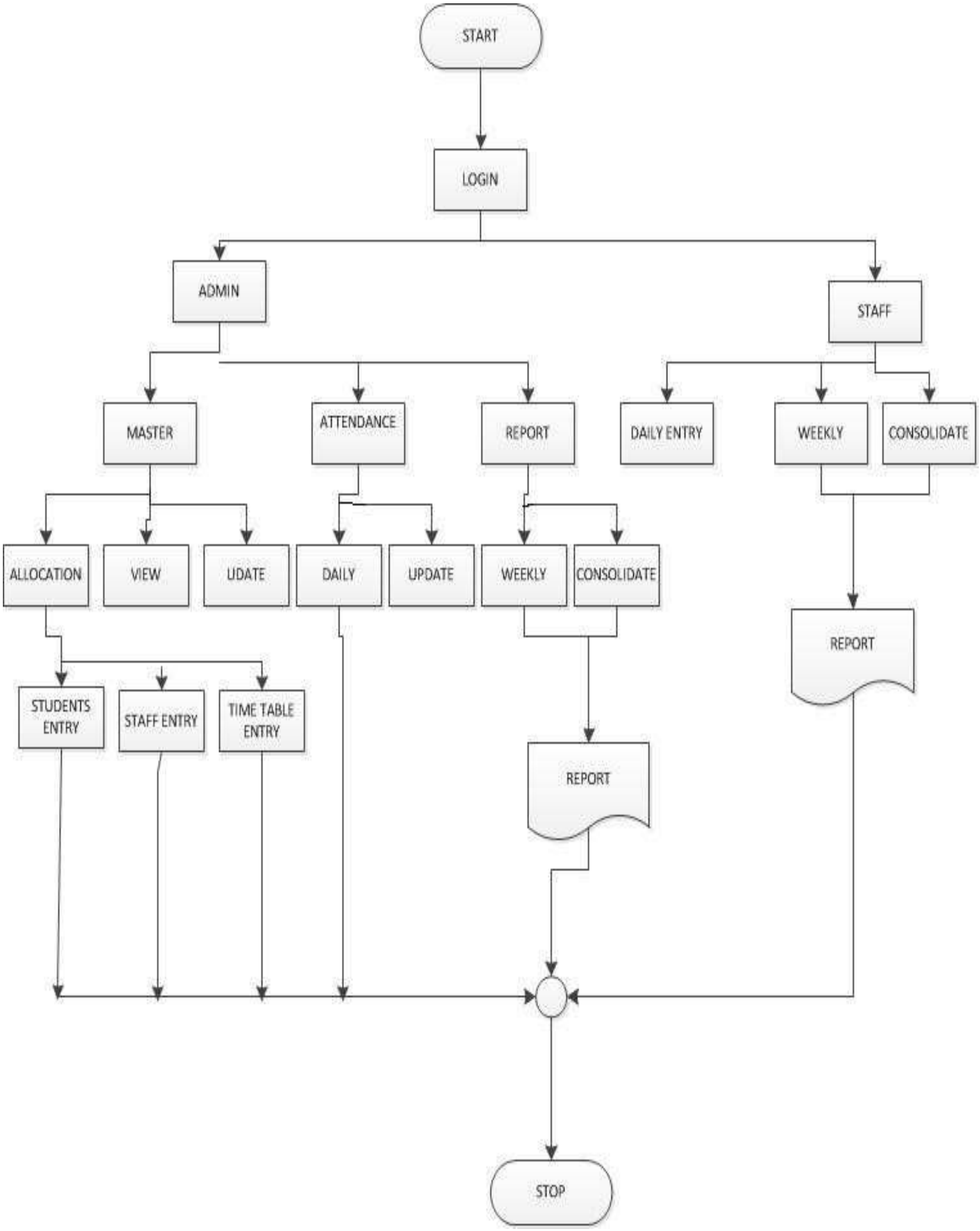


Figure -System Flow Diagram

Data Flow Diagram

DFD level 0:

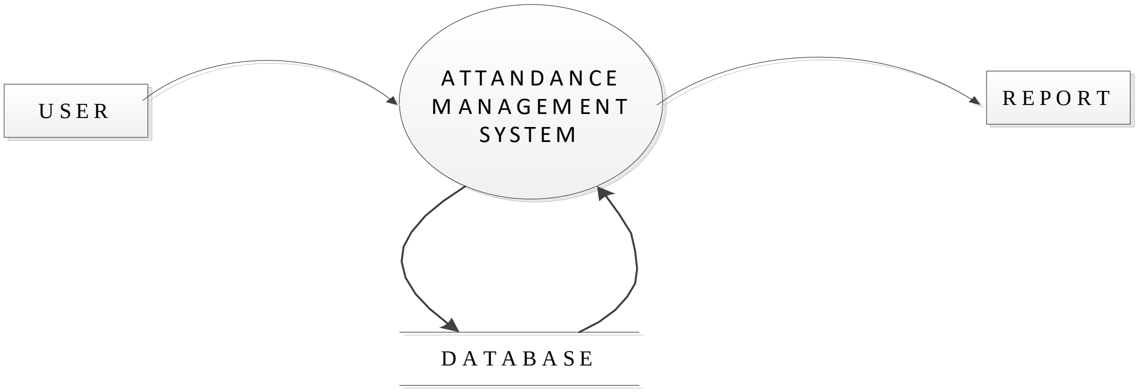


Figure -DataFlowDiagram Level1

DFD level 1:

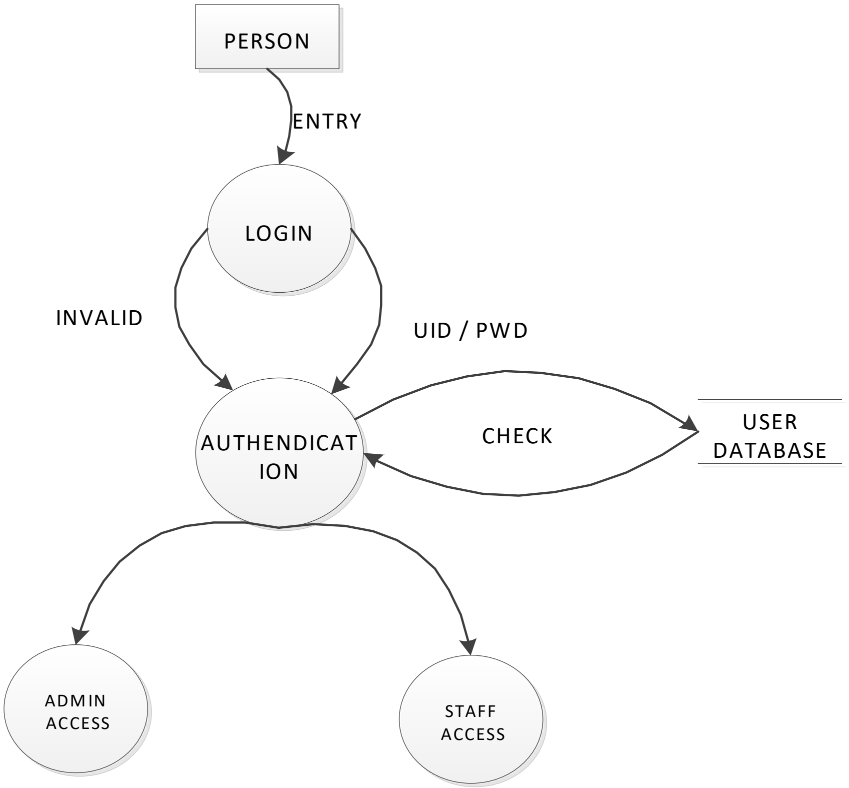


Figure -DataFlowDiagram Level1

DFD level 2:

Admin:

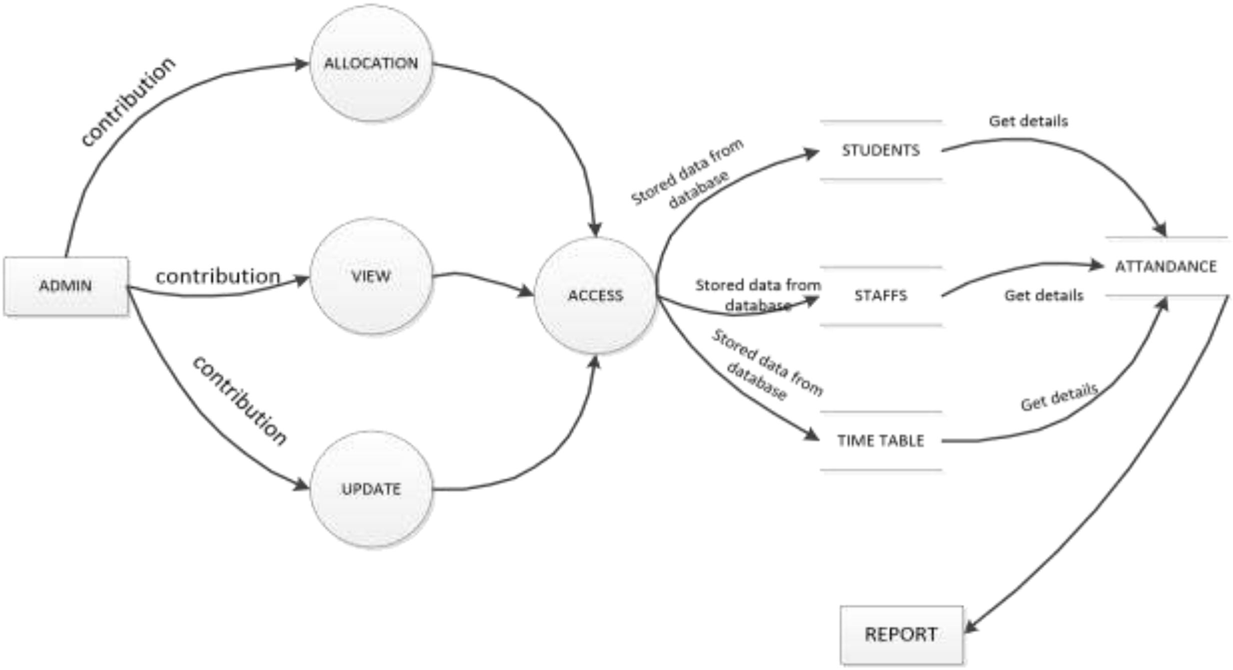


Figure -DataFlowDiagram Level2

staffs:

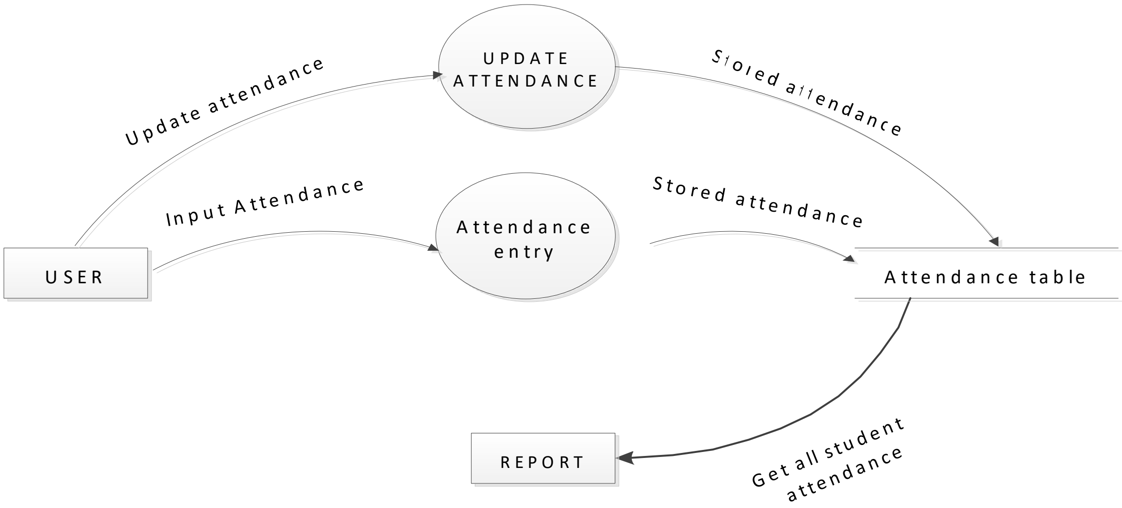


Figure -DataFlowDiagram Level2

SYSTEM DESIGN:

Entity Relationship Diagram:

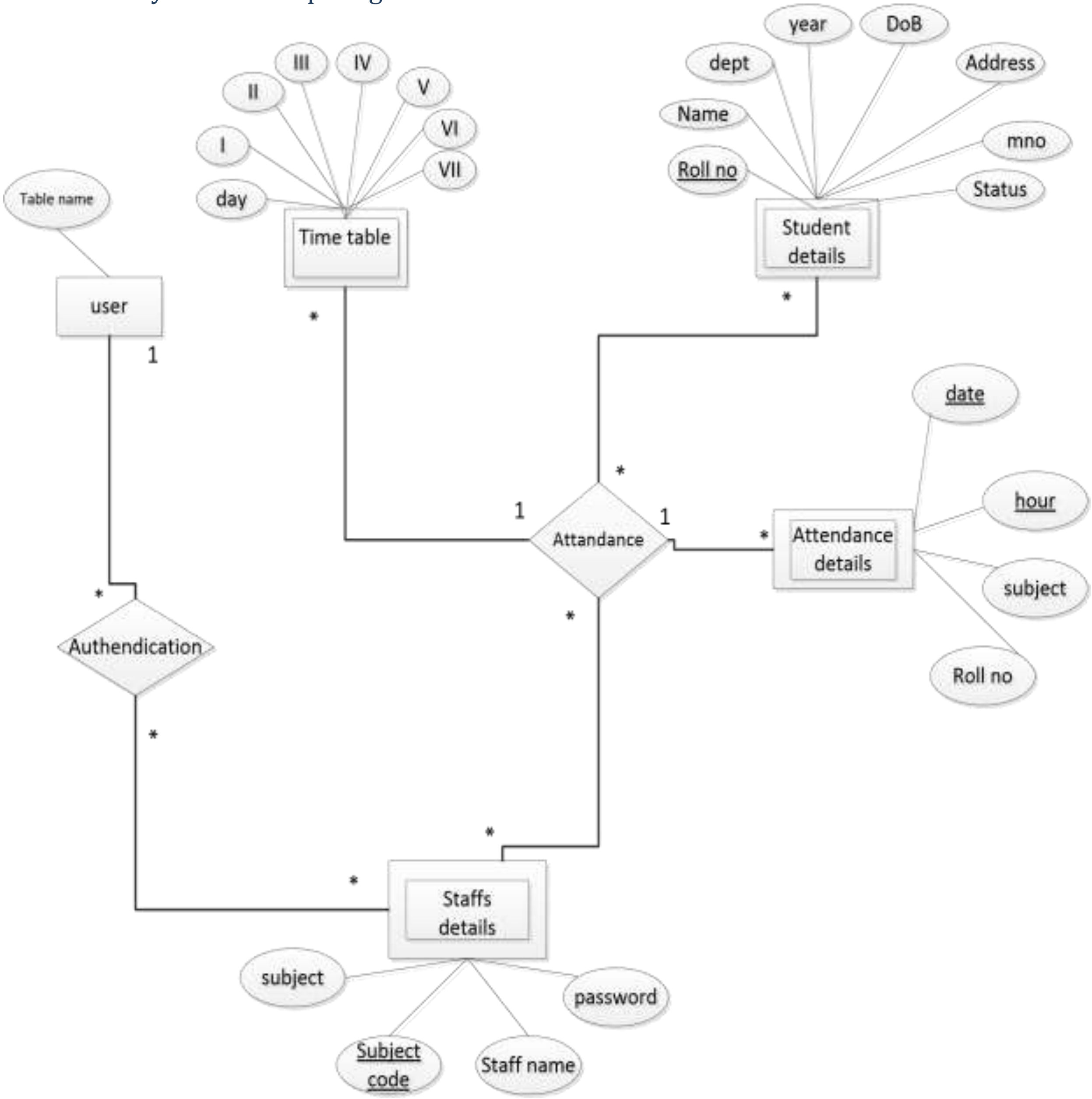


Figure -Entity Relationship Diagram

Use case Diagram:

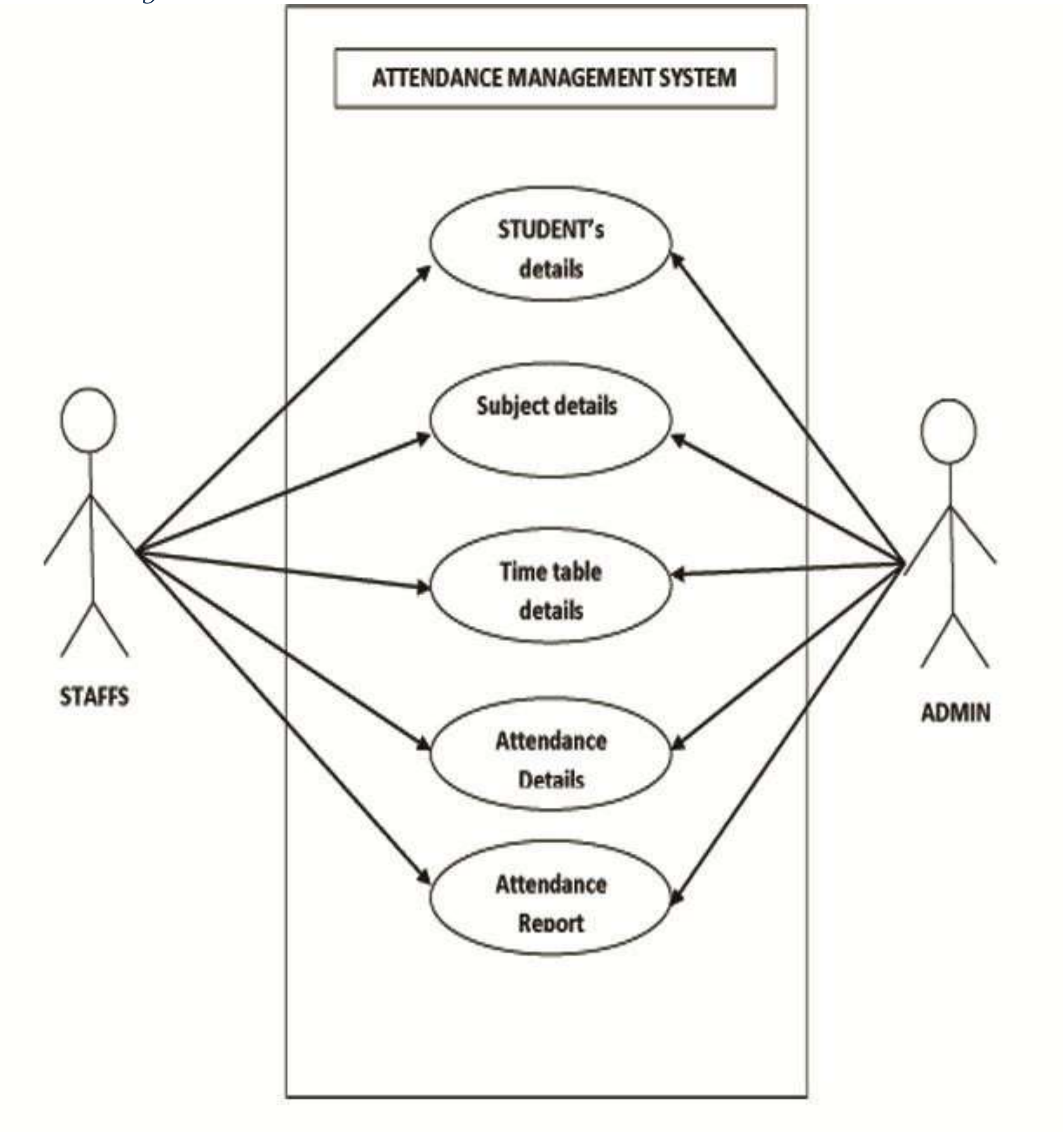


Figure: -Use case Diagram

Database Design:

LOGIN TABLE:

- To create a login details for the table.

FIELDS	DATATYPE	CONTRAINTS	DESCRIPTION
Tablename	varchar(20)	primarykey	Stored number of tables from login

Table: -Login Table

Staffs Table:

- To create username and password for the staff details.

FIELDS	DATATYPE	CONSTRAINTS	DESCRIPTION
Scode	varchar(20)	primarykey	Define separate subject code id
ssname	Varchar(15)	NotNull	Short subject name (ex:cpp)
sname	Varchar(20)	NotNull	Staffs name
Password	Varchar(20)	NotNull	Staff login password

Table: –Student details Table

Student table:

- To create table for Student personal details for our department.

FIELDS	**DATATYPE**	**CONSTRAINTS**	**DESCRIPTION**
Rollno	Varchar(15)	Primarykey	Student rollnumber
Name	Varchar(20)	NotNull	Student name
Dept	Varchar(30)	NotNull	Department name
Year	Number	NotNull	Batch year
DOB	Varchar(20)	NotNull	Student date of birth
ADDRESS	Varchar(20)	NotNull	Student permanent address
MNO	Varchar(20)	NotNull	Student mobile number
EID	Varchar(30)	NotNull	Student E-mail id
CSTATUS	Varchar(20)	NotNull	Student status for dayscholler/Hosteller

Table: –Staff Details Table

Time table:

- To create the subject time table for a particular class.

FIELDS	**DATATYPE**	**CONSTRAINTS**	**DESCRIPTION**
Day	Varchar(20)	Primarykey	Days insert (ex:Monday)
I	Varchar(20)	NotNull	set the period for 1 particular subject
II	Varchar(20)	NotNull	set the period for 2 particular subject
III	Varchar(20)	NotNull	set the period for 3 particular subject
IV	Varchar(20)	NotNull	set the period for 4 particular subject
V	Varchar(20)	NotNull	set the period for 5 particular subject
VI	Varchar(20)	NotNull	set the period for 6 particular subject
VII	Varchar(20)	NotNull	set the period for 7 particular subject

Table: -Time Table

Attendance table:

- To create attendance details for particular class .

FIELDS	**DATATYPE**	**CONSTRAINTS**	**DESCRIPTION**
Dates	Date	Primarykey	Enter day by day attendance
Hour	Number	primarykey	Set particular hour only
Subject	Varchar(15)	NotNull	Particular Subject
Rollno (1 to 60)	Varcahar(20)	NotNull	Enter Present absent details in particular student(ex:M11MCA001)

Table: -Attendance Table

INPUT DESIGN

Input design is part of overall system design that requires special attention designing input data is to make the data entered easy and free from **errors**. The input forms are designed using the controls available in .NET framework. Validation is made for each and every data that is entered. Help information is provided for the users during when the customer feels difficult.

Input design is the process of converting the user originated inputs to a computer based format. A system user interacting through a workstation must be able to tell the system whether to accept the input to produce reports. The collection of input data is considered to be most expensive part of the system design. Since the input has to be planned in such a manner so as to get relevant information, extreme care is taken to obtain pertinent information

This project first will entered to the input of allocation forms it will be created on student details form and subject entry form, time table form .it will helps to calculate subject wise attendance system. next one if u want any verification on your data’s also available in details show forms. Attendance to entered single subject wise or all subject wise attendance system available in this project.

OUTPUT DESIGN

Output design this application "**Student Attendance management system"** generally refers to the results and information that are generated by the system for many end-users; output is the main reason for developing the system and the basis on which they evaluate the usefulness of the application.

The output is designed in such a way that it is attractive, convenient and informative. Forms are designed with various features, which make the console output more pleasing.

As the outputs are the most important sources of information to the users, better design should improve the system's relationships with us and also will help in decision making. Form design elaborates the way output is presented and the layout available for capturing information.

One of the most important factors of the system is the output it produces. This system refers to the results and information generated. Basically the output from a computer system is used to communicate the result of processing to the user.

Attendance management system to show the report subject wise attendance maintaining by staffs. Taken as a whole report obtain on a administrator privileges only. this forms will show weekly report and consolidate report generated date, batch, and class wise to our end user. we want to change our report to convert Excel format .if you want change any modification.

SYSTEM TESTING

Introduction

Once source code has been generated, software must be tested to uncover (and correct) as many errors as possible before delivery to customer. Our goal is to design a series of test cases that have a high likelihood of finding errors. To uncover the errors software techniques are used. These techniques provide systematic guidance for designing test that

(1) Exercise the internal logic of software components, and
(2) Exercise the input and output domains of the program to uncover errors In program function, behavior and performance.

Steps: Software is tested from two different perspectives:

(1) Internal program logic is exercised using ―White box‖ test case design Techniques.

(2) Software requirements are exercised using ―block box‖ test case Design techniques.

In both cases, the intent is to find the maximum number of errors with the Minimum amount of effort and time.

Testing Methodologies:

A strategy for software testing must accommodate low-level tests that are necessary to verify that a small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements. A strategy must provide guidance for the practitioner and a set of milestones for the manager. Because the steps of the test strategy occur at a time when deadline pressure begins to rise, progress must be measurable and problems must surface as early as possible. Following testing techniques are well known and the same strategy is adopted during this project testing.

Unit testing:

Unit testing focuses verification effort on the smallest unit of software design-the software component or module. The unit test is white-box oriented. The unit testing implemented in every module of student attendance management System. by giving correct manual input to the system ,the datas are stored in database and retrieved. If you want required module to access input or get the output from the End user. any error will accrued the time will provide handler to show what type of error will accrued .

System testing:

System testing is actually a series of different tests whose primary purpose is to fully exercise the computer-based system. Below we have described the two types of testing which have been taken for this project. it is to check all modules worked on input basis .if you want change any values or inputs will change all information. so specified input is must.

Performance Testing

Performance testing is designed to test the run-time performance of software within the context of an integrated system. Performance testing occurs throughout all steps in the testing process. Even at the unit level, the performance of an individual module may be assessed as white-box tests are conducted.

This project reduce attendance table, codes. it will generate report fast.no have extra time or waiting of results .entered correct data will show result few millisecond. just used only low memory of our system. Automatically do not getting access at another software. Get user permission and access to other applications.

Test cases

Test case is an object for execution for other modules in the architecture does not represent any interaction by itself. A test case is a set of sequential steps to execute a test operating on a set of predefined inputs to produce certain expected outputs. There are two types of test cases:-*manual* and *automated.* A manual test case is executed manually while an automated test case is executed using automation.

In system testing, test data should cover the possible values of each parameter based on the requirements. Since testing every value is impractical, a few values should be chosen from each equivalence class. An equivalence class is a set of values that should all be treated the same.

Ideally, test cases that check error conditions are written separately from the functional test cases and should have steps to verify the error messages and logs. Realistically, if functional test cases are not yet written, it is ok for testers to check for error conditions when performing normal functional test cases. It should be clear which test data, if any is expected to trigger errors.

TEST CASE:

Agent and admin login form

Sno	**Test case id**	**Test case name**	**Test case desc**	**Step**	**Expected result**	**Actual Result**	**Test case status pass/fail**
1	Login admin	Validate login	To verify that login name on login page	Enter the login name and password and click submit button	Login successful or an error message “In valid login or password” must be displayed	Login successful	Pass
2	Login Staff	Validate login	To verify that login name on login page	Enter the login name and password and click submit button	Login successful or an error message “In valid login or password” must be displayed	Login successful	Pass
3	Password	Validate password	To verify that password on login page	Enter password and login name click submit button	An error message “password invalid” must be displayed	An error message “password invalid” must be displayed	fail

MASTER form

Sno	Test case id	Test case name	Test case desc	Step	Expected result	Actual Result	Test case status pass/fail
1	Create sudent details	Validate allocation form	To allocate separate roll no for the students	Nothing entered and click submit button	An error message student name not equal to null must be displayed	Inserted succesful	Pass
2	Create staff details	Validate allocation form	To allocate separate subject username password for the staffs	Nothing entered and click submit button	An error message staff details password,username not equal to null must be displayed	Inserted succesful	Pass
3	Create time table	Validate allocate period form	To verify that data stored on database	Nothing entered and click submit button	An error message not click not allocation subject table not equal to null must be displayed	Inserted succesful	Pass
4	View	Check details of all data	To verify that data stored on database	generated	An error message return null will be displayed	An error message return null will be displayed	fail

Report form

Sno	**Test case id**	**Test case name**	**Test case desc**	**Step**	**Expected result**	**Actual Result**	**Test case status pass/fail**
1	Weekly report	Validate class attendance form	To select that source and destination	Nothing entered and click submit button	An error message on not selected	Retrived data successful	Pass
2	Consolidate report	Validate class attendance form	To select that depart on and time	Nothing entered and click submit button	An error message on not selected	Retrived data successful	Pass

SYSTEM IMPLEMENTATION

Purpose

System implementation is the important stage of project when the theoretical design is tuned into practical system. The main stages in the implementation are as follows:

- Planning
- Training
- System testing and
- Changeover Planning

Planning is the first task in the system implementation. At the time of implementation of any system people from different departments and system analysis involve. They are confirmed to practical problem of controlling various activities of people outside their own data processing departments.

The line managers controlled through an implementation coordinating committee. The committee considers ideas, problems and complaints of user department, it must also consider:

- The implication of system environment
- Self selection and allocation for implementation tasks
- Consultation with unions and resources available
- Standby facilities and channels of communication

Student Attendance management system will implement student details ,staff handle subjects details, separate login details ,time table details. It will used to entered subject wise attendance .This application elaborate attendance table generate weekly, consolidate report provide to the End user. Mostly this application will calculate date wise attendance .To select starting date to end date generate reports at the time of activities.

SYSTEM MAINTENANCE

Software maintenance is far more than finding mistakes. Provision must be made for environment changes, which may affect either the computer, or other parts of the computer based systems. Such activity is normally called maintenance. It includes both the improvement of the system functions and the corrections of faults, which arise during the operation of a new system.

It may involve the continuing involvement of a large proportion of computer department recourses. The main task may be to adapt existing systems in a changing environment.

Back up for the entire database files are taken and stored in storage devices like flash drives, pen drives and disks so that it is possible to restore the system at the earliest. If there is a breakdown or collapse, then the system gives provision to restore database files. Storing data in a separate secondary device leads to an effective and efficient maintains of the system. The nominated person has sufficient knowledge of the organization's computer passed based system to be able to judge the relevance of each proposed change.

CONCLUSION AND FUTURE ENHANCEMENT

Conclusion

To conclude, Project Data Grid works like a component which can access all the databases and picks up different functions. It overcomes the many limitations incorporated in the attendance.

- Easy implementation Environment
- Generate report Flexibly

Scope for future development

The project has a very vast scope in future. The project can be implemented on intranet in future. Project can be updated in near future as and when requirement for the same arises, as it is very flexible in terms of expansion. With the proposed software of database Space Manager ready and fully functional the client is now able to manage and hence run the entire work in a much better, accurate and error free manner. The following are the future scope for the project.

- Discontinue of particular student eliminate potential attendance.
- Bar code Reader based attendance system.
- Individual Attendance system With photo using Student login.

APPENDICES

Source code:

LOGIN:

```
Imports System.Data

Public Class login
    Dim con As New ADODB.Connection
    Dim rs, rs1 As New ADODB.Recordset
    Public str, temp1, temp2, temp3, temp4 As String
    Dim i As Integer
    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles
Button1.Click        rs = New
ADODB.Recordset        rs1 = New
ADODB.Recordset
        If String.Equals(TextBox1.Text, "Admin") Or String.Equals(TextBox1.Text,
"admin") Or String.Equals(TextBox1.Text, "ADMIN") And
String.Equals(TextBox2.Text, "Admin") Or String.Equals(TextBox2.Text, "admin") Or
String.Equals(TextBox2.Text, "ADMIN")
Then            temp4 = "MDIParent2"
TextBox1.Text = ""
            TextBox2.Text = ""
            MDIParent2.Show()

Me.Hide()
i = 1        Else
            Try
                str = "select * from logintable"
                rs.Open(str, con, ADODB.CursorTypeEnum.adOpenDynamic,
ADODB.LockTypeEnum.adLockPessimistic)
                rs.MoveFirst()
        While (rs.EOF <> True)
          str = "select * from " & rs.Fields("tablename").Value & ""
rs1.Open(str, con, ADODB.CursorTypeEnum.adOpenDynamic,
ADODB.LockTypeEnum.adLockPessimistic)
          While (rs1.EOF <> True)
            If String.Equals(rs1.Fields("sname").Value, TextBox1.Text) And
String.Equals(rs1.Fields("pass").Value, TextBox2.Text)
Then                   temp1 = rs1.Fields("sname").Value
temp2 = rs1.Fields("scode").Value                         temp3
= rs1.Fields("ssname").Value
              temp4 = "MDIParent1"
TextBox1.Text = ""                              TextBox2.Text
= ""

              MDIParent1.Show()
              Me.Hide()
i = 1             Exit
While                End If
rs1.MoveNext()
End While
rs1.Close()
          rs.MoveNext()

        End While


        If i = 0 Then
          MsgBox("LOGIN NOT VAILD")
        End If
      Catch ex As Exception
        MsgBox(ex.ToString)
      End Try
    End If



  End Sub

  Private Sub Form6_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles MyBase.Load
con = New ADODB.Connection

    If (con.State = ConnectionState.Open) Then
con.Close()
    End If

    con.Open("driver={microsoft ODBC for
Oracle};server=test;uid=M11MCA20;pwd=M11MCA20;")
    End Sub

    Private Sub Button2_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button2.Click
        End

    End Sub
End Class
```

Attendance Entry:

```
Public Class attentry
    Dim con As New ADODB.Connection
    Dim rs, rs1 As New ADODB.Recordset
    Dim str, dat As String
    Dim att As String
    Dim i As Integer = 1
    Dim flag As Integer = 1
    Dim chk1 As New DataGridViewCheckBoxColumn()
     Dim chk As New DataGridViewCheckBoxColumn()
    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles
Button3.Click         rs = New
ADODB.Recordset
        Try
            str = "select * from " & ComboBox1.SelectedItem & "_" &
ComboBox5.SelectedItem & ""
            rs.Open(str, con, ADODB.CursorTypeEnum.adOpenDynamic,
ADODB.LockTypeEnum.adLockOptimistic)
            rs.MoveFirst()
            DataGridView1.Rows.Clear()
            i = 1
            While (rs.EOF <> True)
                Dim row As String() = New String() {i, rs.Fields("rollno").Value,
rs.Fields("name").Value}
                DataGridView1.Rows.Add(row)
                i = i + 1
rs.MoveNext()
            End While

            rs.Close()



            DataGridView1.Columns.Add(chk)
chk.HeaderText = "PRESENT/ABSENT"
```

```
DELETEToolStripMenuItem.Click        str = "drop table " & ComboBox1.Text

        xlWorkSheet.Range("A5").Value = "BATCH:" + ComboBox1.Text + "-" +
ComboBox5.Text + "        ATTENDANCE DETAILS FROM  " +
DateTimePicker1.Value.ToString("dd-MMM-yyyy") + "  TO  " +
DateTimePicker2.Value.ToString("dd-MMM-yyyy") + "        SEMESTER:" + "-" +
ComboBox3.Text


        For Each col1 As DataGridViewColumn In DataGridView2.Columns
            If flag < 2 Then
                xlWorkSheet.Cells(6, col1.Index + 1) =
col1.HeaderText.ToString                flag = flag + 1                j = j + 1
Else                j = j + 1
                xlWorkSheet.Cells(6, j) = col1.HeaderText.ToString
                For i = 1 To
6                    j = j +
1
                    xlWorkSheet.Cells(6, j + i - 1) = "".ToString
                Next
            End If
        Next


        xlWorkSheet.Cells(6, 1) = "SNO".ToString
flag = 0
        For Each col As DataGridViewColumn In DataGridView1.Columns
            If flag < 2 Then
                xlWorkSheet.Cells(6, col.Index + 2) =
col.HeaderText.ToString                flag = flag + 1            Else
                xlWorkSheet.Cells(7, col.Index + 2) = col.HeaderText.ToString
            End If



        Next

        For i = 1 To DataGridView1.Rows.Count -
1            xlWorkSheet.Cells(i + 7, 1) =
i.ToString            flag = 0
            For j = 0 To DataGridView1.ColumnCount - 1
                Dim vv As String
                If DataGridView1(j, i - 1).Value Is Nothing
Then                    vv = "Niet ingevuld"                Else
                    vv = DataGridView1(j, i -
1).Value.ToString                    xlWorkSheet.Cells(i +
7, j + 2) = vv                    If flag < 2 Then
```

```
                        xlWorkSheet.Columns(j + 2).ColumnWidth = 15

'xlWorkSheet.Columns.Merge(2)
flag = flag + 1                     Else
                        xlWorkSheet.Columns(j + 2).ColumnWidth = 1
                    End If
                End If
                ProgressBar1.Value = (i / DataGridView1.Rows.Count) * 100

            Next

        Next

        xlWorkSheet.Range("A1:AS1").Merge()
xlWorkSheet.Range("A2:AS2").Merge()
xlWorkSheet.Range("A3:AS3").Merge()
xlWorkSheet.Range("A4:AS4").Merge()
xlWorkSheet.Range("A5:AS5").Merge()

        xlWorkSheet.Range("D6:J6").Merge()
xlWorkSheet.Range("K6:Q6").Merge()
xlWorkSheet.Range("R6:X6").Merge()
xlWorkSheet.Range("Y6:AE6").Merge()
xlWorkSheet.Range("AF6:AL6").Merge()
xlWorkSheet.Range("AM6:AS6").Merge()

        xlWorkBook.Activate()
        xlWorkBook.SaveAs("D:\export.xls")

        xlWorkBook.Close()
xlApp.Quit()
        Panel1.Visible = False


        MsgBox("You can find your report at " & "D:\export.xls")

    End Sub

    Private Sub HOMEToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles HOMEToolStripMenuItem.Click
        MDIParent2.Show()
        Me.Close()

    End Sub
End Class
```

Consolidate Report:

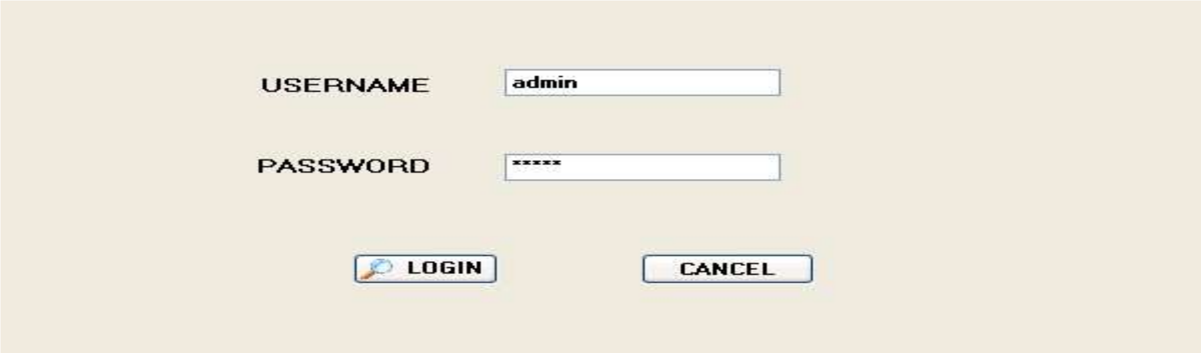
```
Imports Microsoft.Office.Interop
Public Class consli

    Dim con As New ADODB.Connection
    Dim rs, rs1 As New ADODB.Recordset
    Dim str, dat As String
    Dim i, j, k, diff, count1 As New Integer
    Dim pre_hours(100), tot_hours(100), ab_hours(100) As Integer
    Dim tot_day(100), pre_day(100), ab_day(100) As Double

    Private Sub Button3_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button3.Click
DataGridView1.Rows.Clear()
        rollno()
        daycalc()

    End Sub

    Private Sub rollno()
        DataGridView1.Rows.Clear()
        Try
            str = "select * from " & ComboBox1.SelectedItem & "_" &
ComboBox5.SelectedItem & ""
            rs.Open(str, con, ADODB.CursorTypeEnum.adOpenDynamic,
ADODB.LockTypeEnum.adLockOptimistic)

rs.MoveFirst()
i = 0
            While (rs.EOF <> True)
                Dim row As String() = New String() {rs.Fields("rollno").Value,
rs.Fields("name").Value}
                DataGridView1.Rows.Add(row)
                DataGridView1.Rows(i).HeaderCell.Value = (i +
1).ToString                rs.MoveNext()                i = i + 1            End
While           rs.Close()
        Catch ex As Exception
            MsgBox(ex.ToString)
        End Try

    End Sub



    Private Sub adconsoli_Load(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles
MyBase.Load        con = New
ADODB.Connection        rs = New
ADODB.Recordset
con.Open("driver={microsoft ODBC
for
Oracle};server=test;uid=M11MCA20;pwd=M11MCA20;")


        Label8.Text = login.temp1
        Label11.Text = login.temp2
        Label10.Text = login.temp3

    End Sub

    Private Sub daycalc()

        Dim pre_hours(100), tot_hours(100), ab_hours(100) As Integer
        Try
            Dim d As Date
            d =
DateTimePicker1.Value.Date
Dim d1 As Date            d1 =
DateTimePicker2.Value.Date
            diff = DateDiff(DateInterval.Day, d, d1)

            j = 2

            While diff >= 0

                Try
                    str = "Select * from " & ComboBox1.SelectedItem & "_" &
ComboBox5.Text & "_" & ComboBox2.SelectedItem & "_" &
ComboBox3.SelectedItem & "_att where(days='" & d.Date.ToString("dd-MMM-yyyy")
& "' and subject='" & Label10.Text & "')order by hour asc "
                    rs.Open(str, con, ADODB.CursorTypeEnum.adOpenDynamic,
ADODB.LockTypeEnum.adLockPessimistic)
                    rs.MoveFirst()

                    Dim temp(100), temp1(100) As Integer

                    Dim flag1(100) As Integer

                    While (rs.EOF <> True)
                        Dim rcount As Integer
= 0                           Dim count As
Integer = 3                         k = 0
```

```
                        While (rs.Fields.Count > count)
                            If String.Equals(rs.Fields(count).Value, "P") Or
String.Equals(rs.Fields(count).Value, "O") Then
                                pre_hours(k) = pre_hours(k) + 1
                            ElseIf String.Equals(rs.Fields(count).Value, "A") Then
                                ab_hours(k) = ab_hours(k) + 1
End If
                            tot_hours(k) = tot_hours(k) + 1


                            rcount = rcount
+ 1                               count =
count + 1                                   k = k
+ 1                           End While
j = j + 1

                        rs.MoveNext()
                    End While

                    For Me.i = 0 To DataGridView1.Rows.Count - 1
                        tot_day(i) = tot_day(i) + 1
Next
                    d = DateAdd(DateInterval.Day, 1,
d)                      diff = diff - 1
rs.Close()
                Catch ex As Exception
                    d = DateAdd(DateInterval.Day, 1,
d)                      diff = diff - 1
rs.Close()
                End Try

            End While


        Catch ex As Exception
            MsgBox(ex.ToString)
        End Try



        i = 0
        While (i < DataGridView1.Rows.Count - 1)
            DataGridView1.Rows(i).Cells(2).Value = pre_hours(i)
            DataGridView1.Rows(i).Cells(3).Value = ab_hours(i)
            DataGridView1.Rows(i).Cells(4).Value = tot_hours(i)

            DataGridView1.Rows(i).Cells(5).Value = Math.Round((pre_hours(i) /
tot_hours(i) * 100), 2)
```

```
            i = i + 1

        End While

    End Sub

    Private Sub HOMEToolStripMenuItem_Click(ByVal sender As System.Object,
ByVal e As System.EventArgs) Handles HOMEToolStripMenuItem.Click
        MDIParent1.Show()
        Me.Close()


    End Sub

    Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles Button1.Click
        Panel1.Visible = True
        ProgressBar1.Minimum = 0
        ProgressBar1.Maximum = 100


        Dim xlApp As Excel.Application
        Dim xlWorkBook As Excel.Workbook
        Dim xlWorkSheet As Excel.Worksheet
        Dim misValue As Object = System.Reflection.Missing.Value
        Dim i As Integer
        Dim j As Integer


        xlApp = New Excel.Application

        xlWorkBook = xlApp.Workbooks.Add(misValue)

        xlWorkSheet = xlWorkBook.Sheets("sheet1")

        xlWorkSheet.Cells(1, 1) = "Dr.Mahalingam College of Engineering & Technology
".ToString
        xlWorkSheet.Cells(2, 1) = "NPT -MCET Campus, Udumalai Road -
Makkinaickenpatti - Pollachi".ToString        xlWorkSheet.Cells(3, 1) = "Phone :
04259-236030 Fax : 04259-236070".ToString        xlWorkSheet.Cells(4, 1) = "E-
Mail : principal@drmcet.ac.in Web Site :
www.mcet.in".ToString

        xlWorkSheet.Range("A5").Value = "BATCH:" + ComboBox1.Text + "-" +
ComboBox5.Text + "        ATTENDANCE DETAILS FROM  " +
DateTimePicker1.Value.ToString("dd-MMM-yyyy") + "  TO  " +
DateTimePicker2.Value.ToString("dd-MMM-yyyy") + "        SEMESTER:" + "-" +
ComboBox3.Text


        For Each col As DataGridViewColumn In DataGridView1.Columns
xlWorkSheet.Cells(6, col.Index + 1) = col.HeaderText.ToString
        Next
        For i = 1 To DataGridView1.Rows.Count - 1
            For j = 0 To DataGridView1.ColumnCount - 1
                Dim vv As String
                If DataGridView1(j, i - 1).Value Is Nothing Then
                    vv = "Niet ingevuld"
Else

                    vv = DataGridView1(j, i - 1).Value.ToString

                    xlWorkSheet.Cells(i + 6, j + 1) = vv


                End If


            Next
            ProgressBar1.Value = (i / DataGridView1.Rows.Count) * 100
        Next

        xlWorkBook.Activate()

xlWorkBook.SaveAs("D:\Consolidate.xls")
xlWorkBook.Close()        xlApp.Quit()
Panel1.Visible = False
        MsgBox("You can find your report at " & "D:\Consolidate.xls")
    End Sub
End Class
```

Screen Shots:

LOGIN:



USERNAME admin

PASSWORD *****

Figure: -login

ADMIN HOME PAGE:

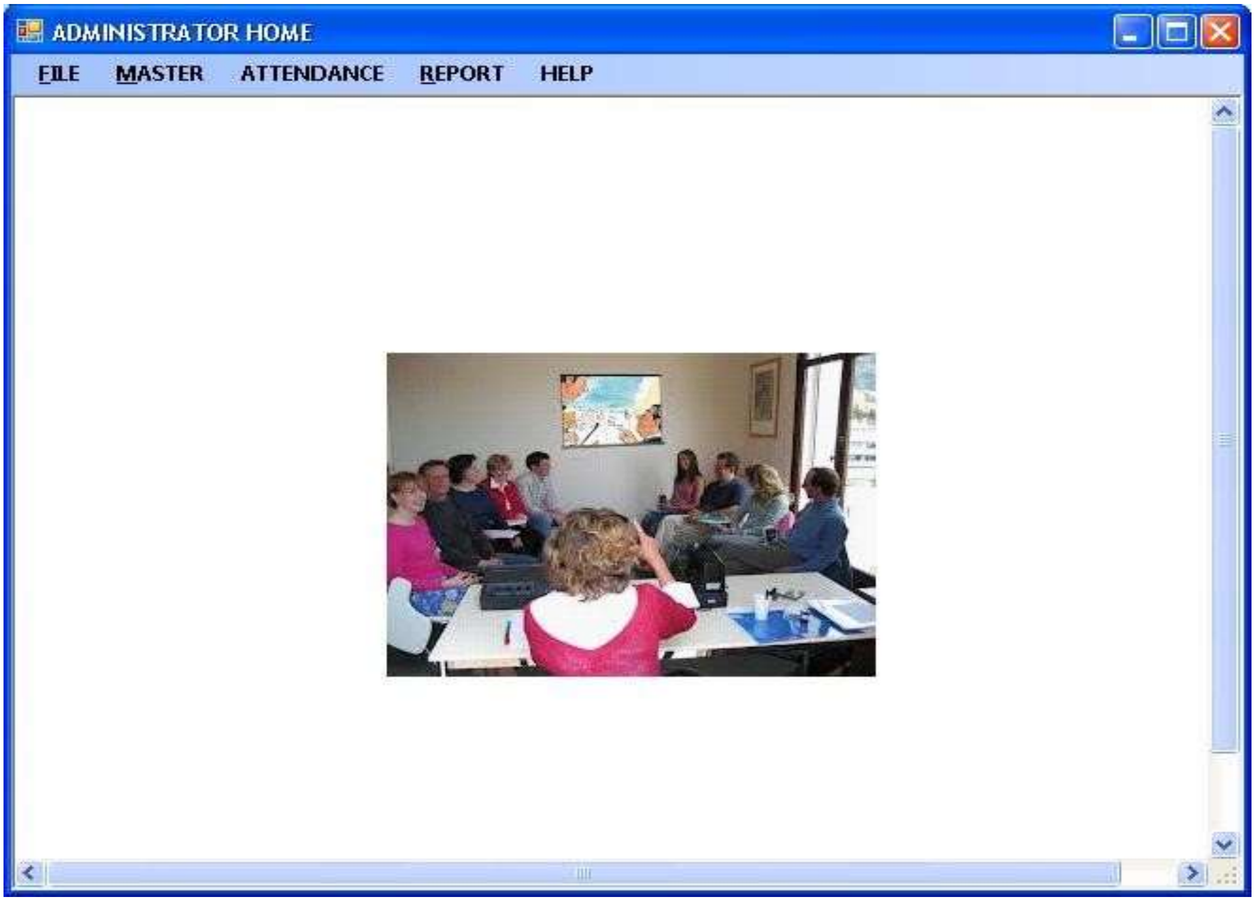
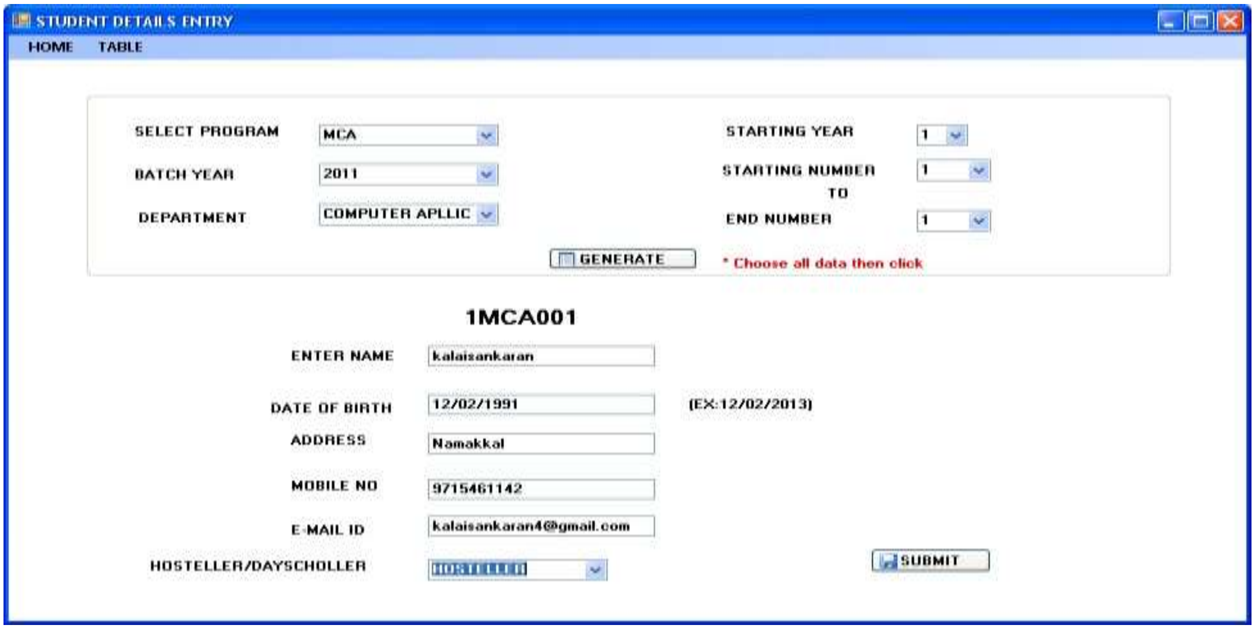


Figure : admin home page

STUDENT DETAILS:



SELECT PROGRAM: MCA

BATCH YEAR: 2011

DEPARTMENT: COMPUTER APLLIC

STARTING YEAR: 1

STARTING NUMBER: 1

TO

END NUMBER: 1

1MCA001

ENTER NAME: kalaisankaran

DATE OF BIRTH: 12/02/1991 (EX:12/02/2013)

ADDRESS: Namakkal

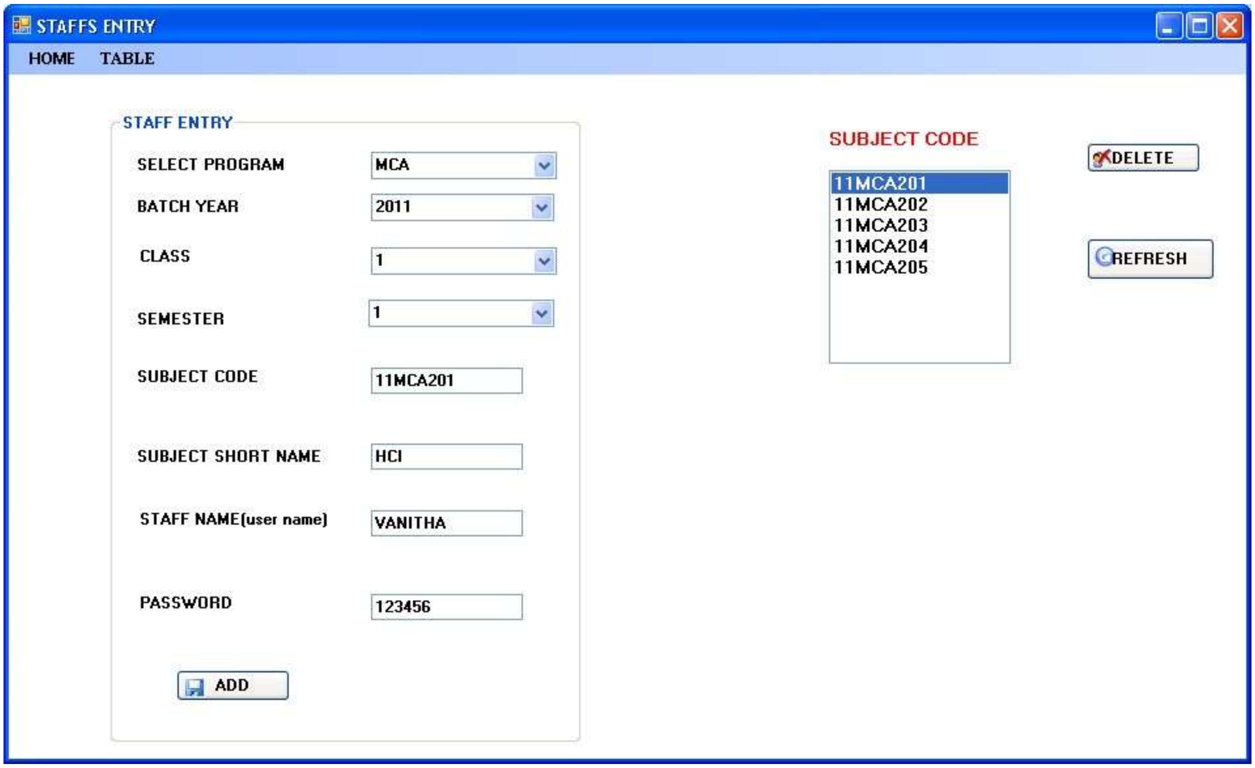
MOBILE NO: 9715461142

E-MAIL ID: kalaisankaran4@gmail.com

HOSTELLER/DAYSCHOLLER: HOSTELLER

Figure: student details Entry

Staffs:



SELECT PROGRAM: MCA

BATCH YEAR: 2011

CLASS: 1

SEMESTER: 1

SUBJECT CODE: 11MCA201

SUBJECT SHORT NAME: HCI

STAFF NAME(user name): VANITHA

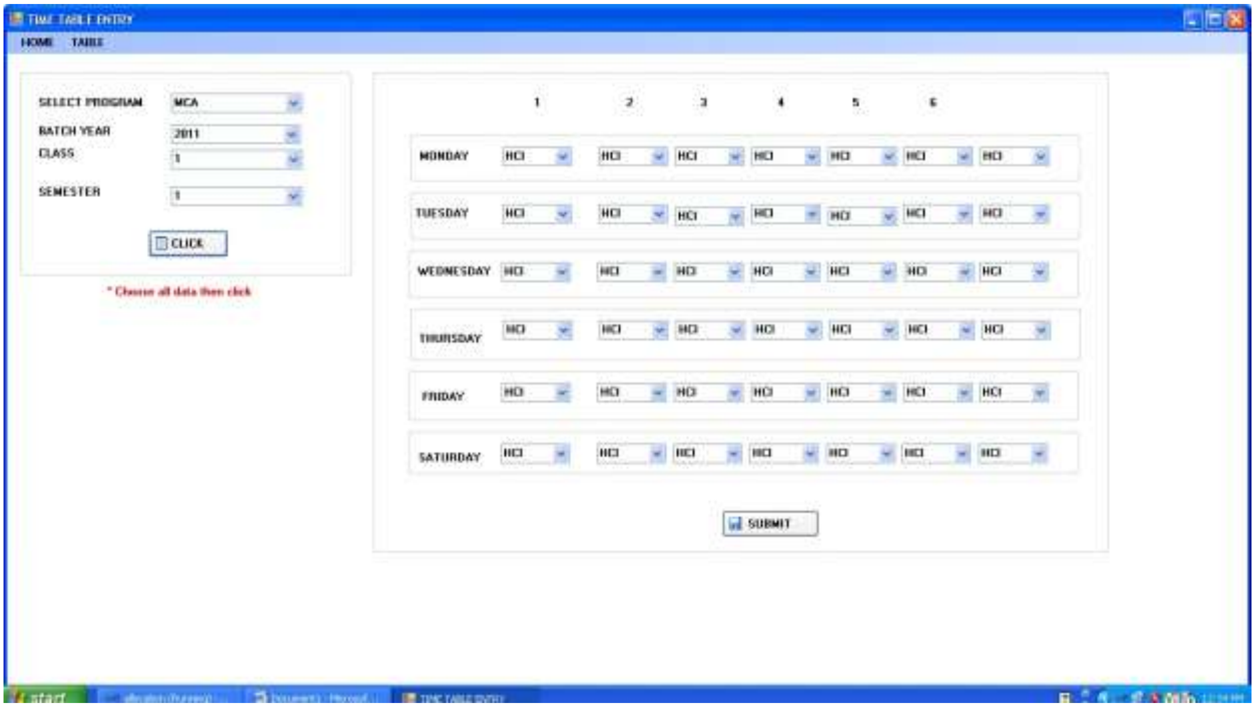
PASSWORD: 123456

SUBJECT CODE

11MCA201
11MCA202
11MCA203
11MCA204
11MCA205

Figure: Staffs insertion

TIME TABLE:



SELECT PROGRAM: MCA

BATCH YEAR: 2011

CLASS: 1

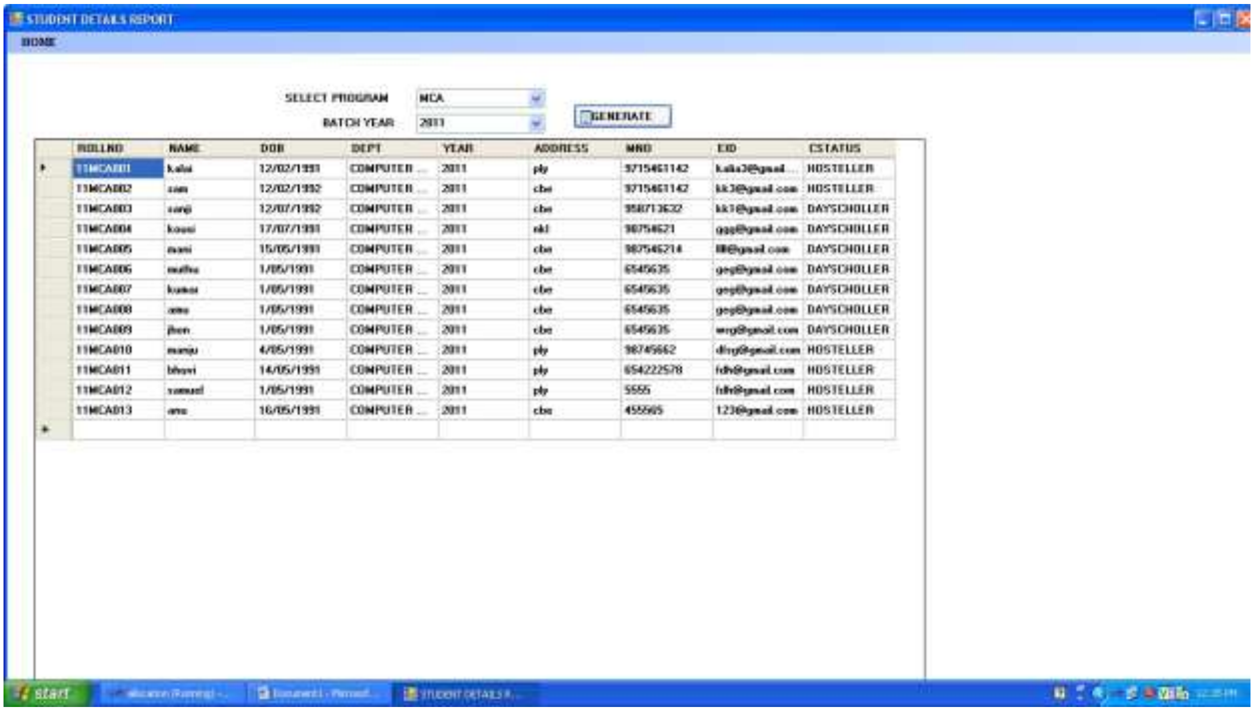
SEMESTER: 1

* Choose all data then click

	1	2	3	4	5	6	
MONDAY	HCI	HCI	HCI	HCI	HCI	HCI	HCI
TUESDAY	HCI	HCI	HCI	HCI	HCI	HCI	HCI
WEDNESDAY	HCI	HCI	HCI	HCI	HCI	HCI	HCI
THURSDAY	HCI	HCI	HCI	HCI	HCI	HCI	HCI
FRIDAY	HCI	HCI	HCI	HCI	HCI	HCI	HCI
SATURDAY	HCI	HCI	HCI	HCI	HCI	HCI	HCI

Figure : time table

VIEW STUDENTS DETAILS:



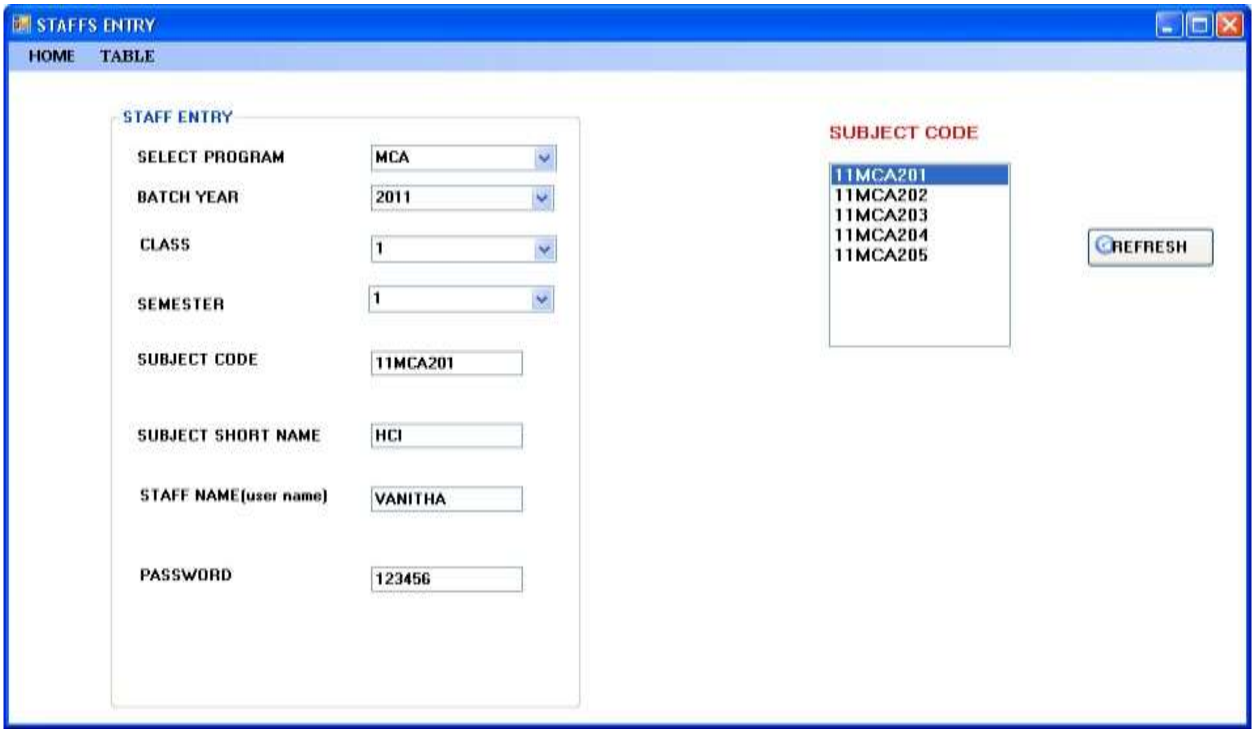
SELECT PROGRAM MCA

BATCH YEAR 2011

ROLLNO	NAME	DOB	DEPT	YEAR	ADDRESS	MNO	EID	CSTATUS
11MCA001	kalai	12/02/1991	COMPUTER ...	2011	ply	9715461142	kalai3@gmail...	HOSTELLER
11MCA002	sam	12/02/1992	COMPUTER ...	2011	cbe	9715461142	kk3@gmail.com	HOSTELLER
11MCA003	sanji	12/07/1992	COMPUTER ...	2011	cbe	958713632	kk1@gmail.com	DAYSCHOLLER
11MCA004	kousi	17/07/1991	COMPUTER ...	2011	nkl	98754621	ggg@gmail.com	DAYSCHOLLER
11MCA005	mani	15/05/1991	COMPUTER ...	2011	cbe	987546214	lll@gmail.com	DAYSCHOLLER
11MCA006	mathu	1/05/1991	COMPUTER ...	2011	cbe	6545635	geg@gmail.com	DAYSCHOLLER
11MCA007	kumar	1/05/1991	COMPUTER ...	2011	cbe	6545635	geg@gmail.com	DAYSCHOLLER
11MCA008	anu	1/05/1991	COMPUTER ...	2011	cbe	6545635	geg@gmail.com	DAYSCHOLLER
11MCA009	jhon	1/05/1991	COMPUTER ...	2011	cbe	6545635	wrg@gmail.com	DAYSCHOLLER
11MCA010	manju	4/05/1991	COMPUTER ...	2011	ply	98745662	dfsg@gmail.com	HOSTELLER
11MCA011	bhuvi	14/05/1991	COMPUTER ...	2011	ply	654222578	fdh@gmail.com	HOSTELLER
11MCA012	samuel	1/05/1991	COMPUTER ...	2011	ply	5555	fdh@gmail.com	HOSTELLER
11MCA013	anu	16/05/1991	COMPUTER ...	2011	cbe	455565	123@gmail.com	HOSTELLER

Figure: view students details

Staffs details:

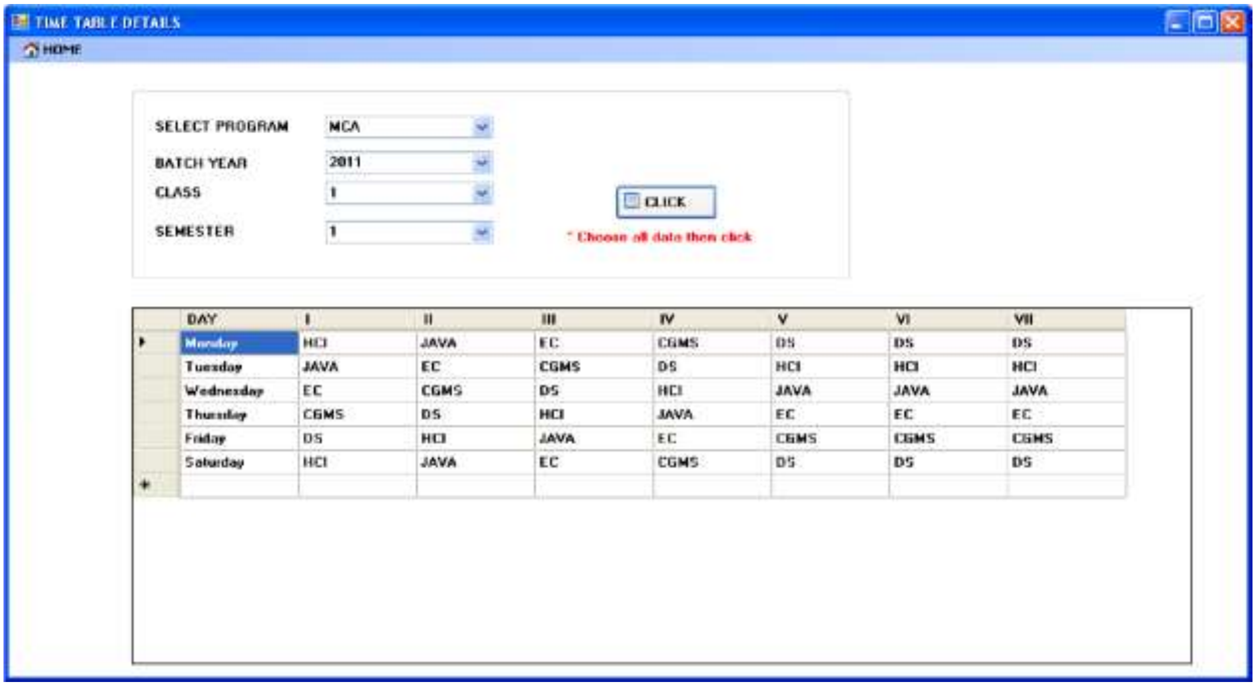


SELECT PROGRAM	MCA
BATCH YEAR	2011
CLASS	1
SEMESTER	1
SUBJECT CODE	11MCA201
SUBJECT SHORT NAME	HCI
STAFF NAME(user name)	VANITHA
PASSWORD	123456

11MCA201
11MCA202
11MCA203
11MCA204
11MCA205

Figure: -Staffs details

TIME TABLE:



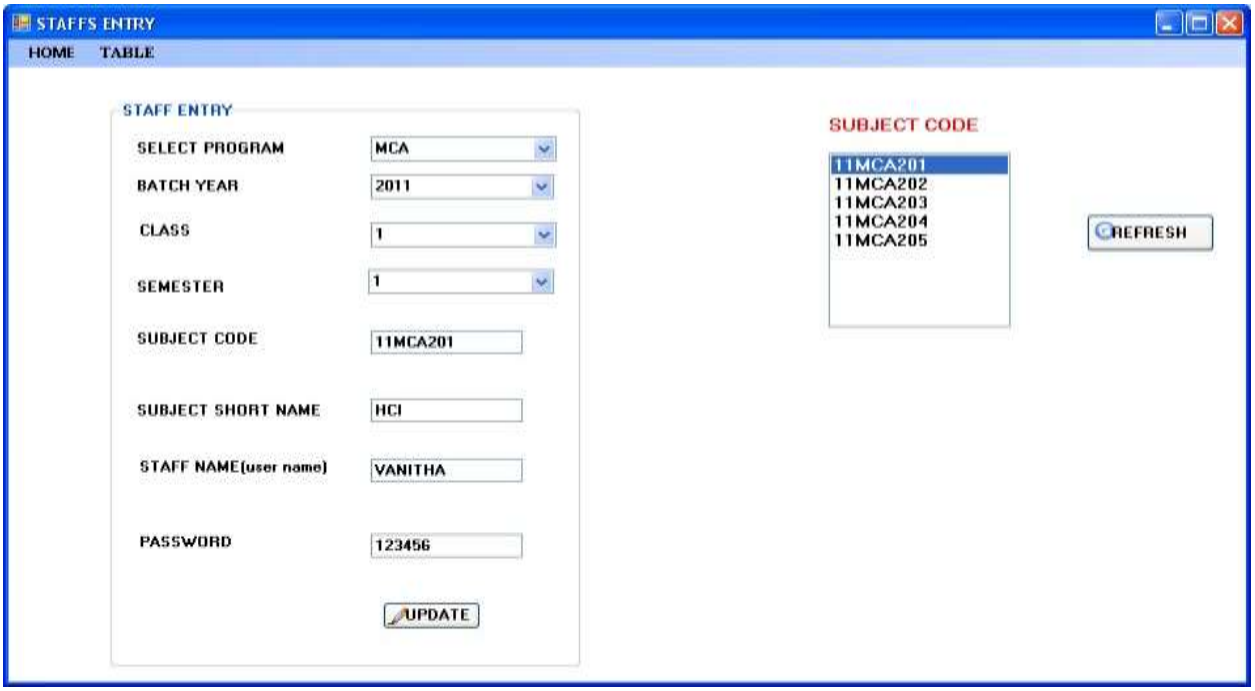
SELECT PROGRAM	MCA
BATCH YEAR	2011
CLASS	1
SEMESTER	1

* Choose all data then click

DAY	I	II	III	IV	V	VI	VII
Monday	HCI	JAVA	EC	CGMS	DS	DS	DS
Tuesday	JAVA	EC	CGMS	DS	HCI	HCI	HCI
Wednesday	EC	CGMS	DS	HCI	JAVA	JAVA	JAVA
Thursday	CGMS	DS	HCI	JAVA	EC	EC	EC
Friday	DS	HCI	JAVA	EC	CGMS	CGMS	CGMS
Saturday	HCI	JAVA	EC	CGMS	DS	DS	DS

Figure: time table details

UPDATE:



STAFF ENTRY

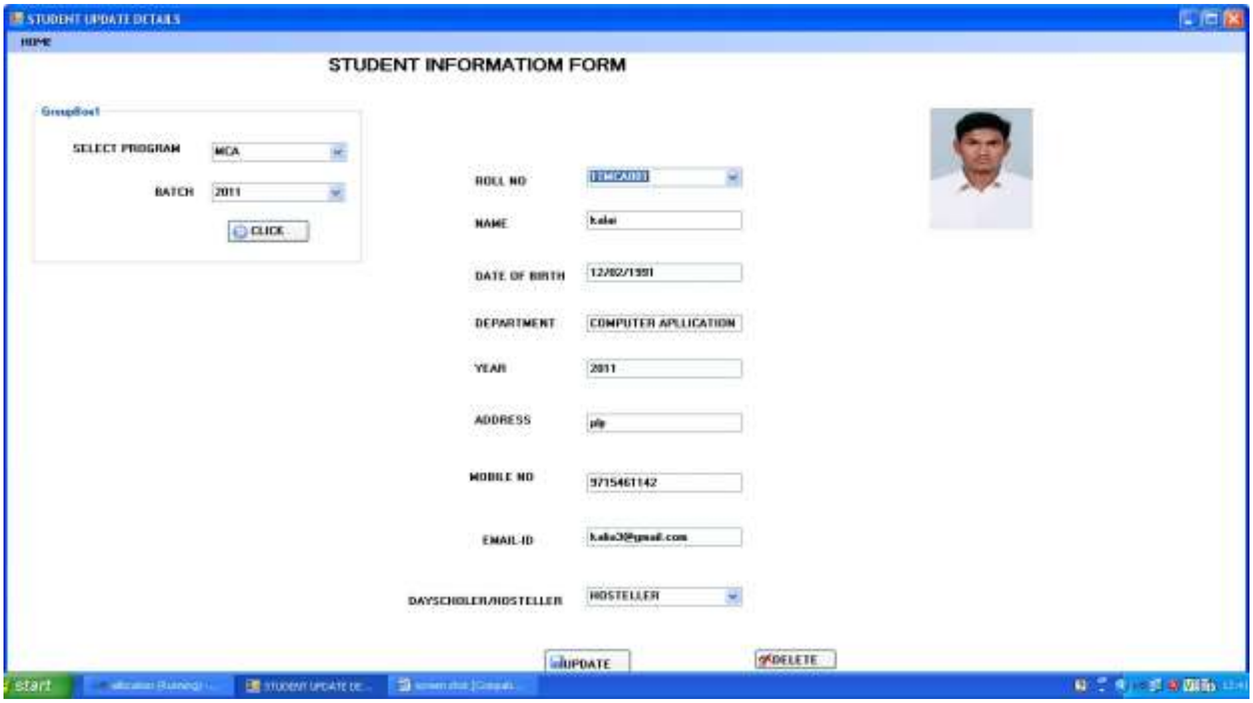
SELECT PROGRAM	MCA
BATCH YEAR	2011
CLASS	1
SEMESTER	1
SUBJECT CODE	11MCA201
SUBJECT SHORT NAME	HCI
STAFF NAME(user name)	VANITHA
PASSWORD	123456

SUBJECT CODE

11MCA201
11MCA202
11MCA203
11MCA204
11MCA205

Figure : update details

STUDENT:



STUDENT INFORMATIOM FORM

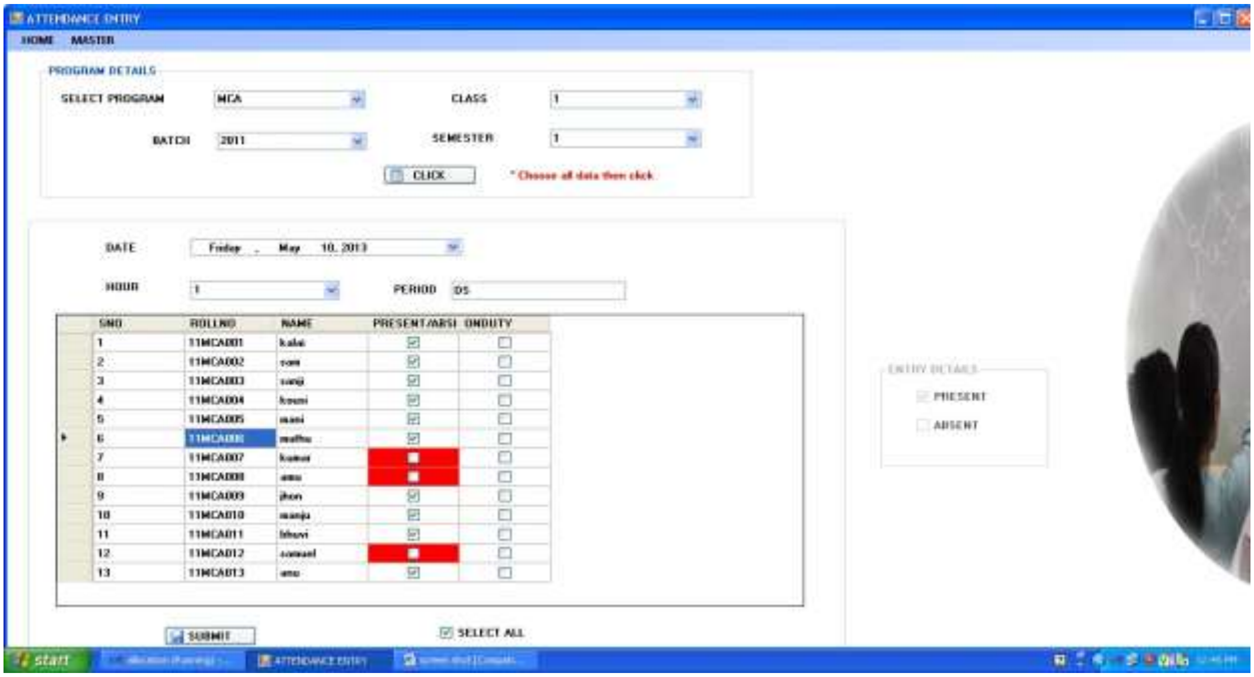
GroupBox1

SELECT PROGRAM	MCA
BATCH	2011

ROLL NO	[illegible]
NAME	kalai
DATE OF BIRTH	12/02/1991
DEPARTMENT	COMPUTER APLLICATION
YEAR	2011
ADDRESS	plr
MOBILE NO	9715461142
EMAIL-ID	kalai3@gmail.com
DAYSCHOLER/HOSTELLER	HOSTELLER

Figure: student details

ATTENDANCE ENTRY:



SELECT PROGRAM: MCA

CLASS: 1

BATCH: 2011

SEMESTER: 1

* Choose all data then click

DATE: Friday , May 10, 2013

HOUR: 1

PERIOD: DS

SNO	ROLLNO	NAME	PRESENT/ABSI	ONDUTY
1	11MCA001	kalai	☑	☐
2	11MCA002	sam	☑	☐
3	11MCA003	sanji	☑	☐
4	11MCA004	kousi	☑	☐
5	11MCA005	mani	☑	☐
6	11MCA006	muthu	☑	☐
7	11MCA007	kumar	☐	☐
8	11MCA008	amu	☐	☐
9	11MCA009	jhon	☑	☐
10	11MCA010	manju	☑	☐
11	11MCA011	bhuvi	☑	☐
12	11MCA012	samuel	☐	☐
13	11MCA013	anu	☑	☐

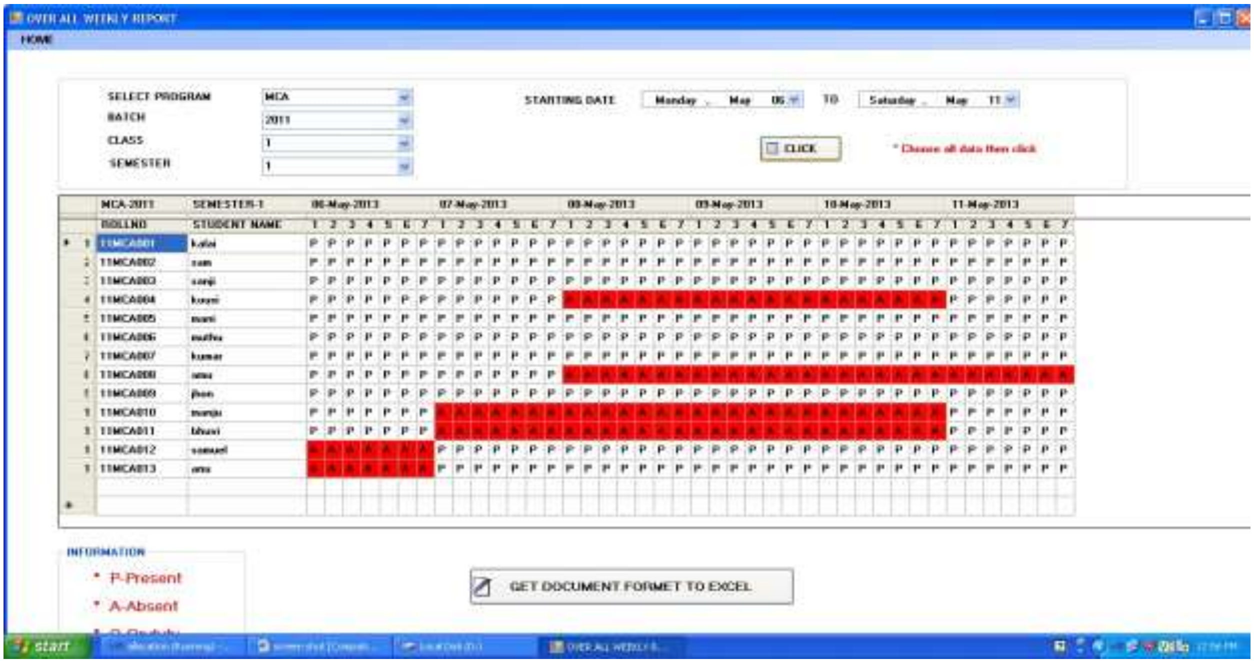
☑ SELECT ALL

☑ PRESENT

☐ ABSENT

Figure: attendance entry

WEEKLY REPORT:



SELECT PROGRAM: MCA

BATCH: 2011

CLASS: 1

SEMESTER: 1

STARTING DATE: Monday , May 06 TO Saturday , May 11

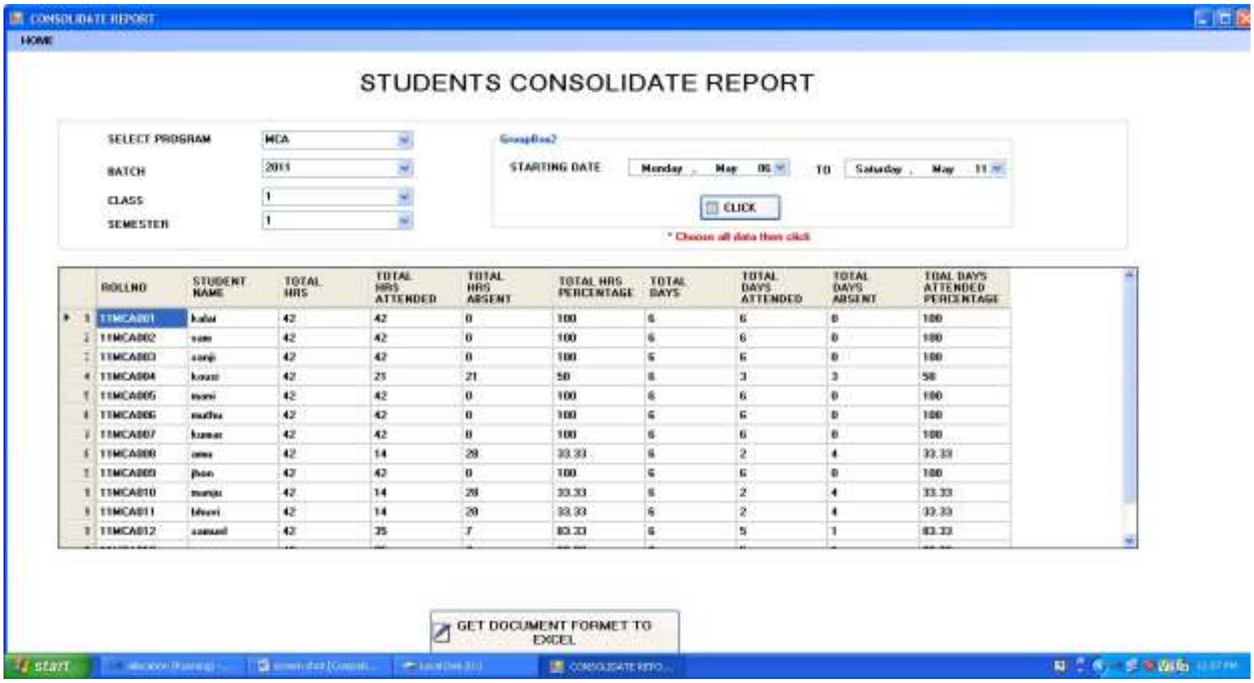
* Choose all data then click

	MCA-2011	SEMESTER-1	06-May-2013							07-May-2013							08-May-2013							09-May-2013							10-May-2013							11-May-2013						
	ROLLNO	STUDENT NAME	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7	1	2	3	4	5	6	7
1	11MCA001	kalai	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
2	11MCA002	sam	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
3	11MCA003	sanji	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
4	11MCA004	kousi	P	P	P	P	P	P	P	P	P	P	P	P	P	P	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	P	P	P	P	P	P	P
5	11MCA005	mani	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
6	11MCA006	muthu	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
7	11MCA007	kumar	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
8	11MCA008	amu	P	P	P	P	P	P	P	P	P	P	P	P	P	P	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A
9	11MCA009	jhon	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
1	11MCA010	manju	P	P	P	P	P	P	P	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	P	P	P	P	P	P	P
1	11MCA011	bhuvi	P	P	P	P	P	P	P	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	A	P	P	P	P	P	P	P
1	11MCA012	samuel	A	A	A	A	A	A	A	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P
1	11MCA013	anu	A	A	A	A	A	A	A	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P	P

- * P-Present
- * A-Absent
- * O-Onduty

Figure :weekly report

CONSOLIDATE REPORT:



STUDENTS CONSOLIDATE REPORT

SELECT PROGRAM: MCA

BATCH: 2011

CLASS: 1

SEMESTER: 1

GroupBox2

STARTING DATE: Monday , May 06 TO Saturday , May 11

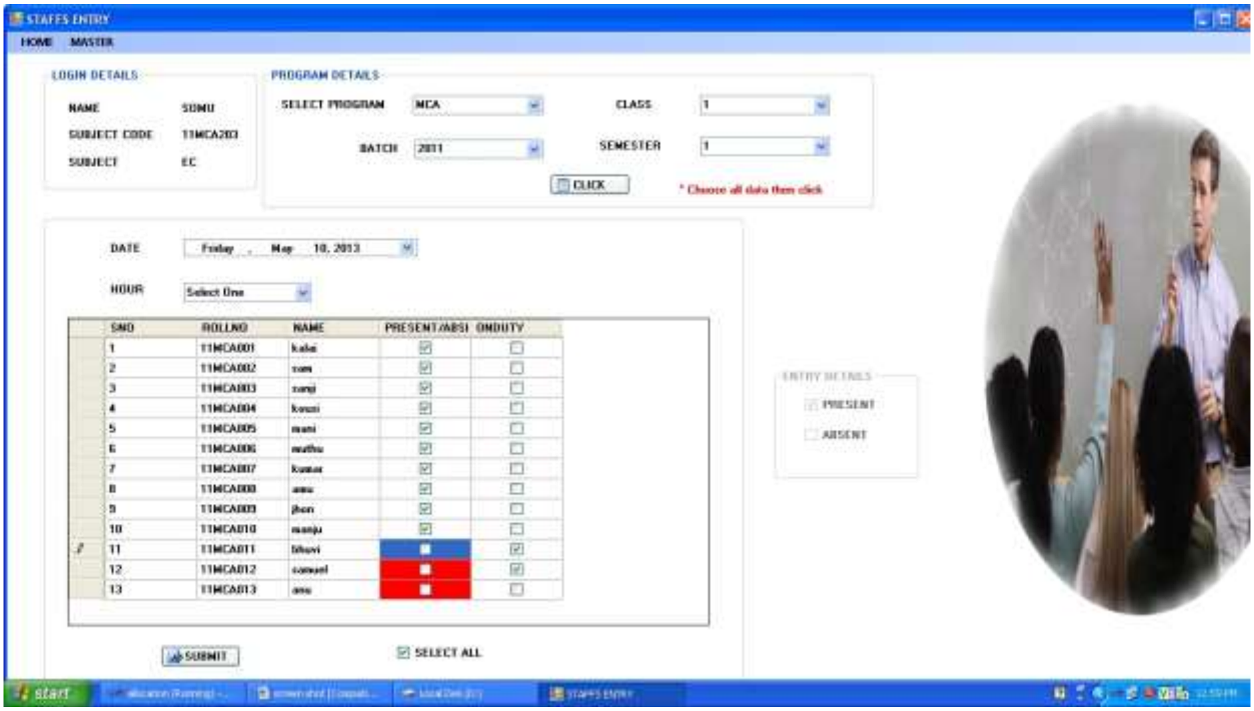
* Choose all data then click

ROLLNO	STUDENT NAME	TOTAL HRS	TOTAL HRS ATTENDED	TOTAL HRS ABSENT	TOTAL HRS PERCENTAGE	TOTAL DAYS	TOTAL DAYS ATTENDED	TOTAL DAYS ABSENT	TOAL DAYS ATTENDED PERCENTAGE
11MCA001	kalai	42	42	0	100	6	6	0	100
11MCA002	sam	42	42	0	100	6	6	0	100
11MCA003	sanji	42	42	0	100	6	6	0	100
11MCA004	kousi	42	21	21	50	6	3	3	50
11MCA005	mani	42	42	0	100	6	6	0	100
11MCA006	muthu	42	42	0	100	6	6	0	100
11MCA007	kumar	42	42	0	100	6	6	0	100
11MCA008	anu	42	14	28	33.33	6	2	4	33.33
11MCA009	jhon	42	42	0	100	6	6	0	100
11MCA010	manju	42	14	28	33.33	6	2	4	33.33
11MCA011	bhuvi	42	14	28	33.33	6	2	4	33.33
11MCA012	samuel	42	35	7	83.33	6	5	1	83.33

Figure: consolidate report details

STAFFS LOGIN:

DAILY ATTENDANCE:



LOGIN DETAILS

NAME SOMU

SUBJECT CODE 11MCA203

SUBJECT EC

PROGRAM DETAILS

SELECT PROGRAM MCA

CLASS 1

BATCH 2011

SEMESTER 1

* Choose all data then click

DATE Friday , May 10, 2013

HOUR Select One

SNO	ROLLNO	NAME	PRESENT/ABSI	ONDUTY
1	11MCA001	kalai	☑	□
2	11MCA002	sam	☑	□
3	11MCA003	sanji	☑	□
4	11MCA004	kousi	☑	□
5	11MCA005	mani	☑	□
6	11MCA006	muthu	☑	□
7	11MCA007	kumar	☑	□
8	11MCA008	amu	☑	□
9	11MCA009	jhon	☑	□
10	11MCA010	manju	☑	□
11	11MCA011	bhuvi	□	☑
12	11MCA012	samuel	□	☑
13	11MCA013	anu	□	□

ENTRY DETAILS

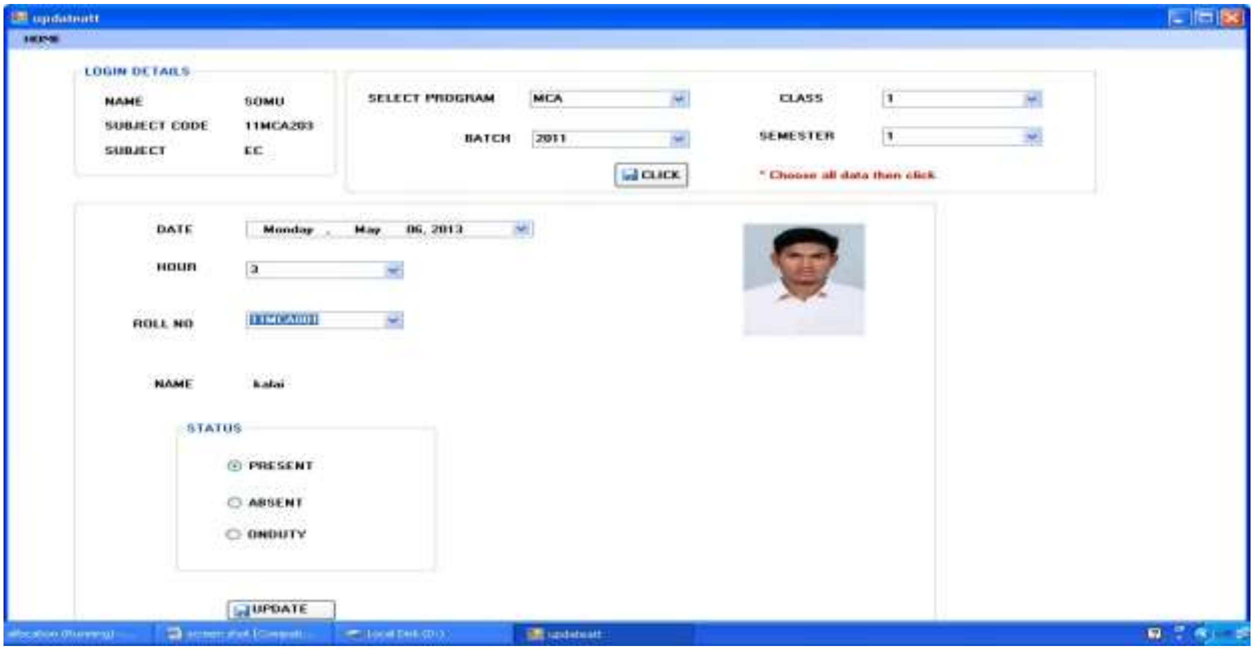
□ PRESENT

□ ABSENT

☑ SELECT ALL

Figure:daily attendance

UPDATE ATTENDANCE:



LOGIN DETAILS

NAME	SOMU
SUBJECT CODE	11MCA203
SUBJECT	EC

SELECT PROGRAM	MCA	CLASS	1
BATCH	2011	SEMESTER	1

* Choose all data then click

DATE	Monday , May 06, 2013
HOUR	3
ROLL NO	11MCA001
NAME	kalai

STATUS

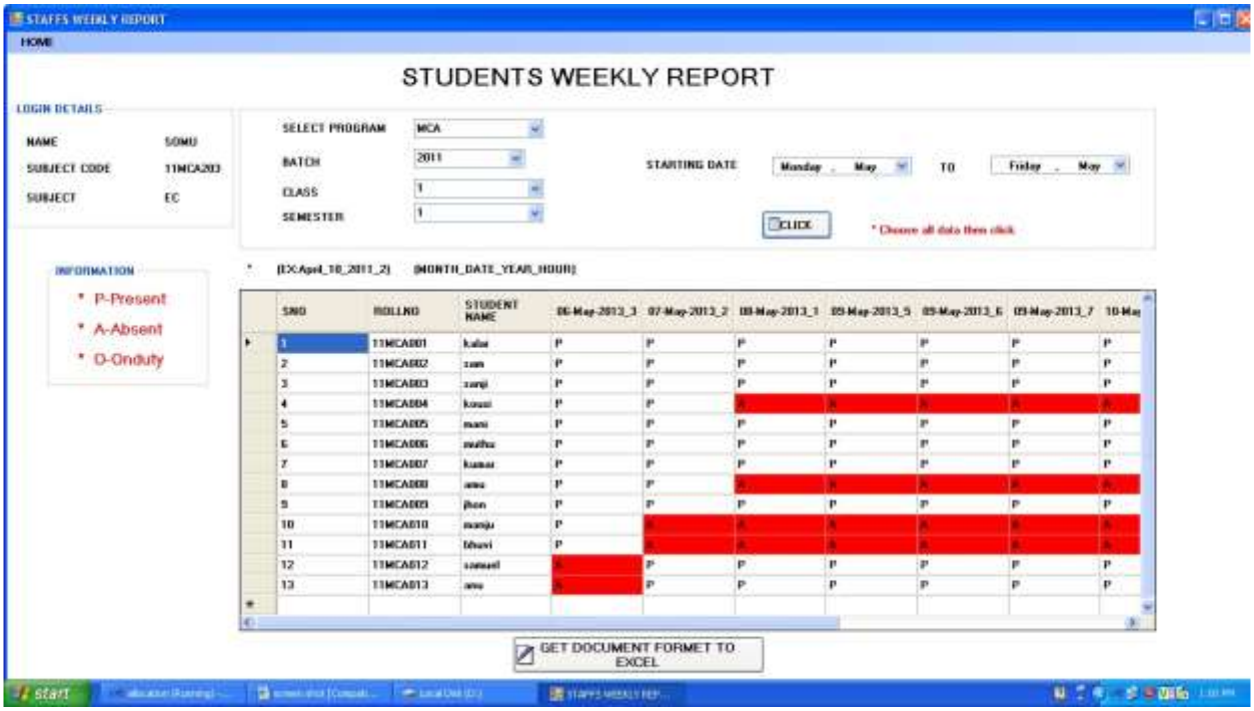
● PRESENT

○ ABSENT

○ ONDUTY

Figure: Update attendance

WEEKLY REPORT:



STUDENTS WEEKLY REPORT

LOGIN DETAILS

NAME	SOMU
SUBJECT CODE	11MCA203
SUBJECT	EC

SELECT PROGRAM	MCA
BATCH	2011
CLASS	1
SEMESTER	1

STARTING DATE Monday , May TO Friday , May

* Choose all data then click

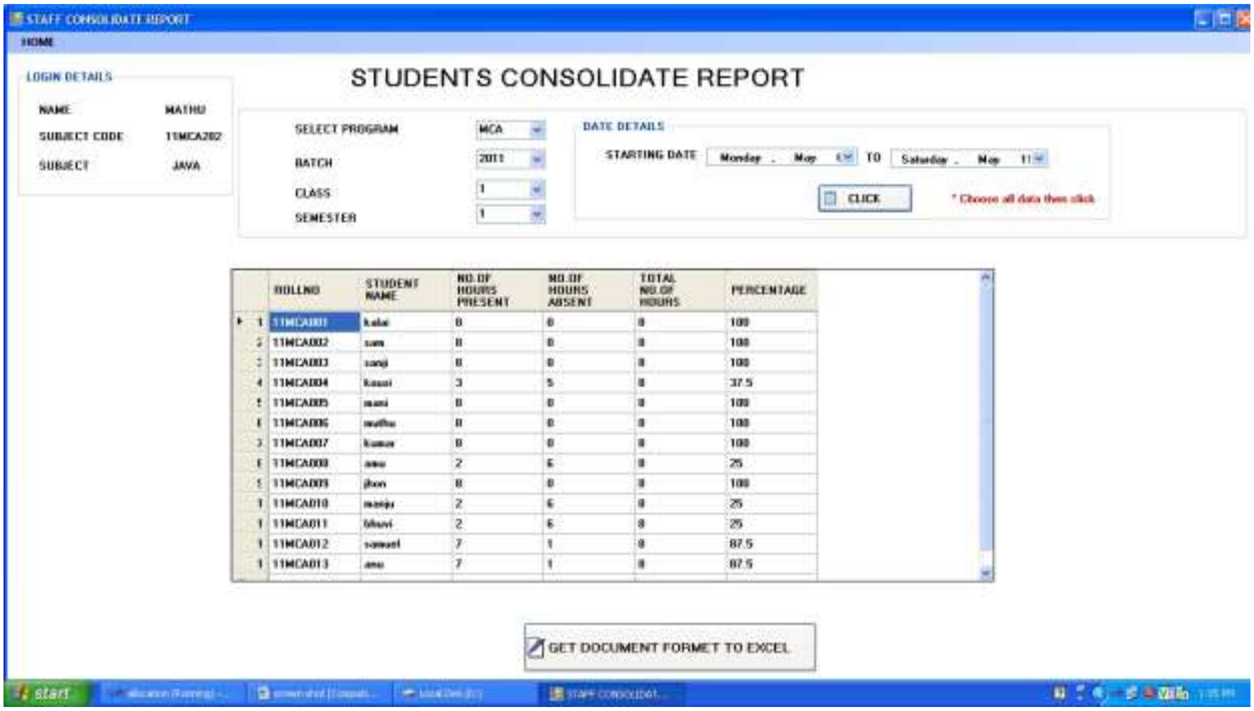
INFORMATION

- * P-Present
- * A-Absent
- * O-Onduty

* (EX:April_10_2011_2) (MONTH_DATE_YEAR_HOUR)

SNO	ROLLNO	STUDENT NAME	06-May-2013_3	07-May-2013_2	08-May-2013_1	09-May-2013_5	09-May-2013_6	09-May-2013_7	10-May
1	11MCA001	kalai	P	P	P	P	P	P	P
2	11MCA002	sam	P	P	P	P	P	P	P
3	11MCA003	sanji	P	P	P	P	P	P	P
4	11MCA004	kousi	P	P	A	A	A	A	A
5	11MCA005	mani	P	P	P	P	P	P	P
6	11MCA006	muthu	P	P	P	P	P	P	P
7	11MCA007	kumar	P	P	P	P	P	P	P
8	11MCA008	anu	P	P	A	A	A	A	A
9	11MCA009	jhon	P	P	P	P	P	P	P
10	11MCA010	manju	P	A	A	A	A	A	A
11	11MCA011	bhuvi	P	A	A	A	A	A	A
12	11MCA012	samuel	A	P	P	P	P	P	P
13	11MCA013	anu	A	P	P	P	P	P	P

figure : weekly report details Consolidate:



STUDENTS CONSOLIDATE REPORT

LOGIN DETAILS

NAME MATHU

SUBJECT CODE 11MCA202

SUBJECT JAVA

SELECT PROGRAM MCA

BATCH 2011

CLASS 1

SEMESTER 1

DATE DETAILS

STARTING DATE Monday , May 6 TO Saturday , May 11

* Choose all data then click

	ROLLNO	STUDENT NAME	NO.OF HOURS PRESENT	NO.OF HOURS ABSENT	TOTAL NO.OF HOURS	PERCENTAGE
1	11MCA001	kalai	8	0	8	100
2	11MCA002	sam	8	0	8	100
3	11MCA003	sanji	8	0	8	100
4	11MCA004	kavui	3	5	8	37.5
5	11MCA005	mani	8	0	8	100
6	11MCA006	mathu	8	0	8	100
7	11MCA007	kumar	8	0	8	100
8	11MCA008	anu	2	6	8	25
9	11MCA009	jhon	8	0	8	100
1	11MCA010	manju	2	6	8	25
1	11MCA011	bhuvi	2	6	8	25
1	11MCA012	samuel	7	1	8	87.5
1	11MCA013	anu	7	1	8	87.5

Figure: Consolidate details

REFERENCES

Books References

1. Introducing Microsoft .NET, Second Edition author David S. Platt.
2. Joe Mayo, “Microsoft Visual Studio 2010: A Beginner's Guide”, Tata McGraw Hill, 2010.
3. Alex Mackey, “Introducing .NET 4.0: With Visual Studio 2010”, Press, USA, 2010.

WEBSITES

1. http://www.msdn.net/
2. http://msdn.microsoft.com/en-us/library/orm-9780596518455-02.aspx
3. http://www.w3schools.com/asp.net/
4. http://www.cramerz.com/aspdotnet
5. http://www.dotnetspider.net/
6. http://www.stackoverflow.com
7. http://www.codeproject.com