ONLINE ATTENDANCE SYSTEM

A PROJECT REPORT

Submitted by

CHUNCHU.MANOJ **723920104015**

MADDELA.PRAVEEN **723920104037**

TOTAMSETTY.SAICHARAN **723920104059**

In partial fulfilment for the award of the degree

Of

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

ARJUN COLLEGE OF TECHNOLOGY

COIMBATORE- 642 120

ANNA UNIVERSITY: CHENNAI 600 025

MAY 2023

BONAFIDE CERTIFICATE

Certified that this Report titled "**ONLINE ATTENDANCE SYSTEM"** is the bonafide work of **CHUNCHU.MANOJ (723920104015), TOTAMSETTY.SAICHARAN (723920104059), MADDELA.PRAVEEN (723920104037),** who carried out the work under my supervision. Certified further that to the best of my knowledge the work reported here in does not form part of any other project work on the basis of which a degree or award was conferred on an earlier occasion on this or any other candidate.

SIGNATURE   SIGNATURE

Ms.M.ESTHER MAGTHALENE ANNE MCA.,B.Ed.,   **Mr.S.SATHEESH M.E.,**

SUPERVISOR   **HEAD OF THE DEPARTMENT**

Assistant professor,   Associate professor,

Department of   Department of

Computer Science and Engineering,   Computer Science and Engineering,

Arjun College of Technology ,   Arjun College of Technology ,

Coimbatore -642120   Coimbatore -642120

Submitted for the university project viva-voce held on_____

INTERNAL EXAMINER   **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

We owe our since and heartfelt thanks to our chairman **Thiru. R.SURIYANARAYANAN,** and also, we extend our profound thanks to our Secretary **Dr. R. SURESH KUMAR M.E., MBA., Ph.D**., for their exuberance in motivating young minds.

Our deepest gratitude and thanks to our motivator and Principal **Dr.C.UTHAYAKUMAR M.Tech.,MBA., Ph.D**., who always helping us whenever we approach him during the course of our Project.

We would also like to express our profound thanks to our Head of the Department **Mr. S. SATHEESH M.E**., Assistant Professor, Department of Computer Science & Engineering, whose thoughtful words, advise and help to complete our project successfully.

We would also like to express our profound thanks to our Project Coordinator **Ms.R.LATHA PRIYADHARSHINI M.E.,**Associate Professor Department of Computer Science and Engineering, whose thoughtful words, advise and help to complete our project successfully.

Our sincere gratitude and unplumbed thanks to our beloved Project guide **Ms.M.ESTHER MAGTHALENE ANNE MCA.,B.Ed.,** Assistant Professor, Department of Computer Science and Engineering, for her constant encouragement, Valuable Guidance and constructive criticism in making this Project a successful one.

We express our sincere thanks to all **Faculty Members and Skilled Assistants** of Computer Science and Engineering Department and our lovable **Friends** for their help and wishes for the successful completion of this Project.

Finally, yet importantly, we would like to express our indebtedness to our beloved **Parents** for their affectionate blessing cooperation at all stages of this academic venture and also our well wishers.

ABSTRACT

Attendance Management System is an innovative application designed to maintain and managed the attendance of students of all the CSE departments B TECH in College Management Education. This project keeps the attendance details of all students and which allows us to create flexible and configurable attendance policies to manage proper time and scheduling among the students, faculties and admin. The basic idea behind this project is to monitor activity of Studentattendance. Student attendance management system deals with the maintenance of thestudent's attendance details. It is generates the attendance of the student on basis of presence in class. It is maintained on the daily basis of their attendance. the staffs willbe provided with the separate username & password to make the student's status. Thestaffs handling the particular subjects responsible to make the attendance for all students. Only if the student present on that particular period, the attendance will becalculated. The students attendance reports based on weekly and consolidate will be generated.

TABLE OFCONTENTS

5.4   LIST OF FIGURES

5.7 LIST OF TABLES

CHAPTER 1

INTRODUCTION

1.1 OBJECTIVE:

"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 thestudents. Each staff will be given with a separate username and password based on thesubject 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. Our Project is online attendance management system and the objectivesare Reduces admin work by integrating the details of the students of allthe three departments into a single database. Manual work for information retrieval on attendance becomes lessas the work becomes digitized. Easy access for students because they can view their attendance andmake up for the shortage of attendance accordingly. It is also time saving as manual work is less. There is less chance of error. It eliminates duplicate data entry in time and attendance entries.

CHAPTER 2

SYSTEM ANALYSIS

2.1 INTRODUCTION

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 askillful 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.

2.2 EXISTING SYSTEM

The Existing system is a manual entry for the students. Here the attendance will be carried out in the hand written registers. It will be a tedious job to maintain the record for the user. The human effort is more here. The retrieval of the information is not as easy as the records are maintained in the hand written registers.

This application requires correct feed on input into the respective field. Suppose the wrong inputs are entered, the application resist to work. so the user find it difficult to use.

2.3 PROPOSED SYSTEM

To overcome the drawbacks of the existing system, the proposed system has beenevolved. This project aims to reduce the paper work and saving time to generate accurate results from the student's attendance. The system provides with the best user interface.

The efficient reports can be generated by using this proposed system.

1. **Advantages of Proposed System**
   - ➢ It is trouble-free to use.
   - ➢ It is a relatively fast approach to enter attendance
   - ➢ Is highly reliable, approximate result from user
   - ➢ Best user Interface
   - ➢ Efficient reports

2.3.1 FEASIBILITY STUDY

Feasibility analysis begins once the goals are defined. It starts by generating broadpossible solutions, which are possible to give an indication of what the new system should look lime. This is where creativity and imagination are used. Analysts must think up new ways of doing things- generate new ideas. There is no need to go into the detailedsystem operation yet. The solution should provide enough information to make reasonable estimates about project cost and give users an indication of how the new system will fit into the organization.

It is important not to exert considerable effort at this stage only to find out that the project is not worthwhile or that there is a need significantly change the original goal.

2.3.2 Economically Feasibility

It is cost effective in the sense that has eliminated the paper work completely. The system is also time effective because the calculations are automated which are made at the end of the month or as per the user requirement.

2.3.3 Technical feasibility

The technical requirement for the system is economic and it does not useany other additional Hardware and software. Technical evaluation must also assess whether the existing systems can be upgraded to use the new technology and whether the organization has the expertise to use it.

Install all upgrades framework into the .Net package supported widowsbased application. this application depends on Microsoft office and intranet service ,database. Enter their attendance and generate report to excel sheet.

2.3.4 Operational Feasibility

The system working is quite easy to use and learn due to its simple but attractive interface. User requires no special training for operating the system. Technical performance include issues such as determining whether the system can provide the right information for the Department personnel student details, and whether the system can be organized so that it always delivers this information at the right place and on time using intranet services. Acceptance revolves around thecurrent system and its personnel.

CHAPTER 3

SYSTEM SPECIFICATION

3.1 HARDWARE REQUIREMENTS (Minimum Requirement)

- ➢ **Minimum RAM:-**1GB
- ➢ **Hard Disk:-**128 GB
- ➢ **Processor:-**Intel Pentium 4(1.50 GHZ) or above

3.2 SOFTWARE REQUIREMENTS (minimum Requirement)

- ➢ **Operating system :**Windows XP
- ➢ **Front-End Language :** Html, Css, JavaScript
- ➢ **Back-End :** Php
- ➢ **Back-End Connectivity:** Php, MySql

CHAPTER 4

SOFTWARE DESCRIPTION

4.1 PACKAGE - VISUAL STUDIO 2010

Microsoft Visual Studio is an integrated development environment (IDE) developed by Microsoft and first released in 1997. The environment has since undergone several significant updates and revisions. The latest version, introduced in 2019, includes various new features and improvements over previous versions.

It is among the most popular tools for developers to create, test, and deploy software applications. The product includes everything a developer needs, such as a code editor, a debugger, compilers, and several other tools and services to streamline the software development process.

Visual Studio supports different programming languages by means of language services, which allow the code editor and debugger to support (to varying degrees) nearly any programming language, provided a language-specific service exists.

Visual Studio also includes a web-site editor and designer that allows web pages to be authored by dragging and dropping widgets. It is used for developing VB.NET application efficiently to get input and output design easiest one. It will be run at windows application based services provide the user.

4.2 DEVELOPMENT TOOLS AND TECHNOLOGIES
VB.NET Version 10.0

The environment has since undergone several significant updates and revisions. The latest version, introduced in 2019, includes various new features and improvements over previous versions

VBC.EXE, is installed as part of the freeware .NET Framework SDK. Mono also includes a command-line VB.NET compiler. The most recent version is VB 2012, which was released on August 15, 2012.

My goal in this article is to provide you with an intro intensive, look at Visual Basic .NET and the new Microsoft®.NET platform. In order to learn what Visual Basic .NET is all about, you must first understand a fewcore aspects of the .NET platform. This article will build your knowledge of Visual Basic .NET from the ground up, so I'll begin by discussing the new programming model and the high-level architecture of the platform's execution engine called the common language runtime (CLR).

While explaining what the CLR is and how it works, I'll show a few examples using Visual Basic .NET. As you'll see, Visual Basic® has undergone a significantoverhaul to accommodate the CLR and its associated programming model. Consequently, Visual Basic .NET has many new object- oriented design features and much higher levels of type safety than previous versions of Visual Basic.either language can be used to write software that takes full advantage of theCLR .NET Framework.

Now, let me get started by introducing the core concepts of the .NET platform

Features of .NET

- ➢ IO management
- ➢ Windows and Web Controls
- ➢ Database access
- ➢ Multithreading
- ➢ Reflections

ORACLE 10G

Oracle 10g has come with purpose of improving manageability and performance in all areas, right from the process of installation, configuration, database upgrades to application tuning, space and storage.

This Oracle version has been designed to reduce the cost of manageability and deliver high performance for all key workloads. Also various new features are provided for high- availability, including new flashback capabilities, virtualization of computing resources in Grid environment that reduce the cost of hardware and storage, security enhancement, Business intelligent solutions etc. Let have a glance at some exiting features of Oracle 10g.

- Clustering
- Grid computing
- Server manageability
- Network management
- Storage management
- Space, object transaction management
- Back up recovery management
- Reduce down time for application and database upgrades

ADO.Net

An evolutionary, more flexible successor to ADO.A system designed for connected environments. A programming model with advanced XML support A set of classes, interfaces, structures, and enumerations that manage data access from within the
.NET Framework

Data Providers

- MS SQLServer 7.0+
- Oracle
- OLE DB (old SQL & Access- Jet 4.0)
- Open Database Connectivity (ODBC)- earlier Visual Studio, Access Driver, ODBC for Oracle

CHAPTER 5

PROJECT DESCRIPTION

5.1 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 aremaintained 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.

5.2 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.

5.3 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.

5.3.1 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.

5.3.2 STAFFS MODULE

- **Attendance details**

    > ➢ It assists the staff to mark attendance to the students fortheir subject. This will authenticate the staff before making the entry.

- **Report details**

    > ➢ weekly report get particular hour details of attendance from starting date to ending date and display the status .

    > ➢ consolidate report get all student attendance details from starting date to ending date status help for the eligibilitycriteria of the student to attend the examination
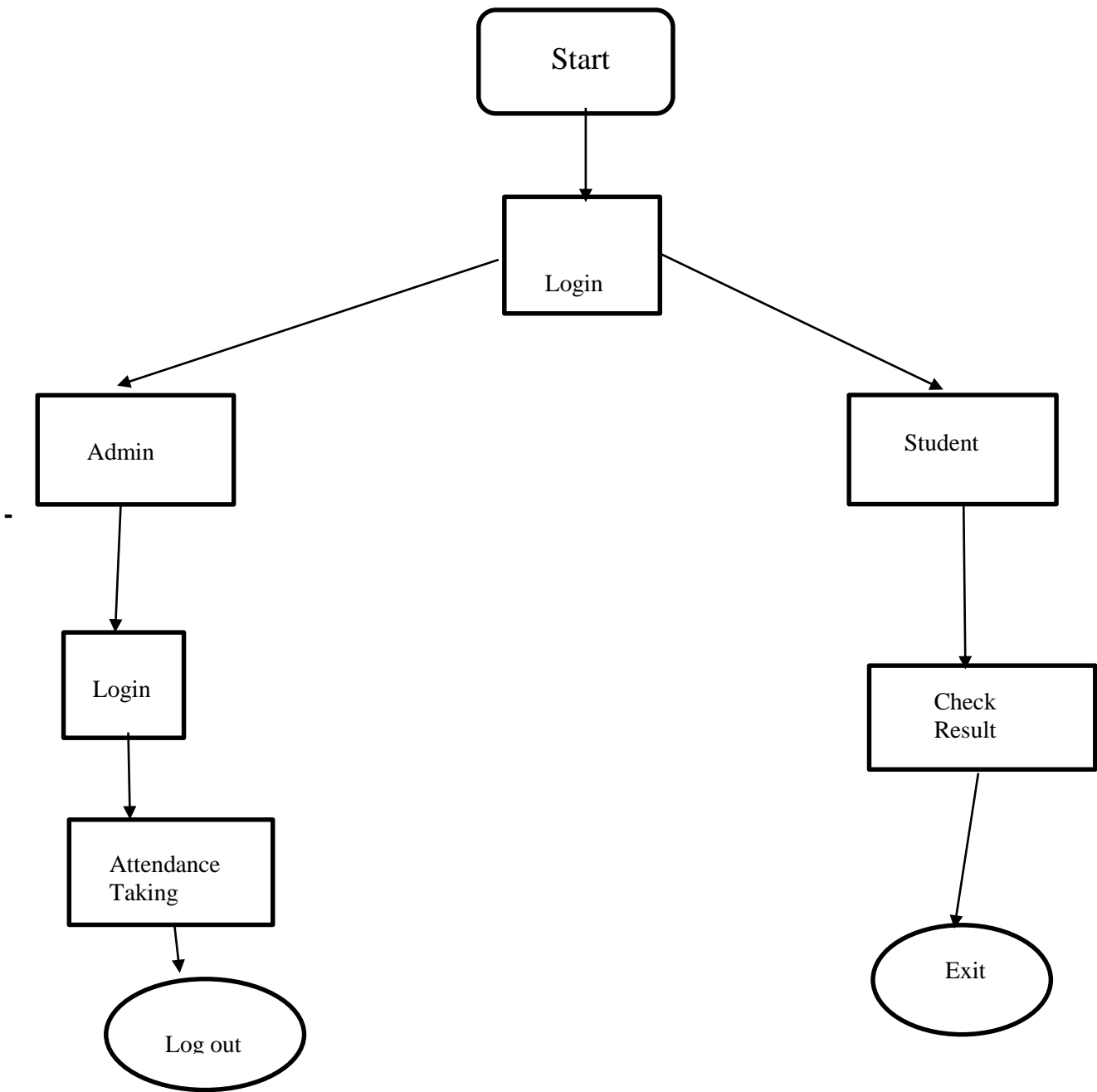
5.4 SYSTEM FLOW DIAGRAM

Figure 5.4-System Flow Diagram

5.5 Data Flow Diagram
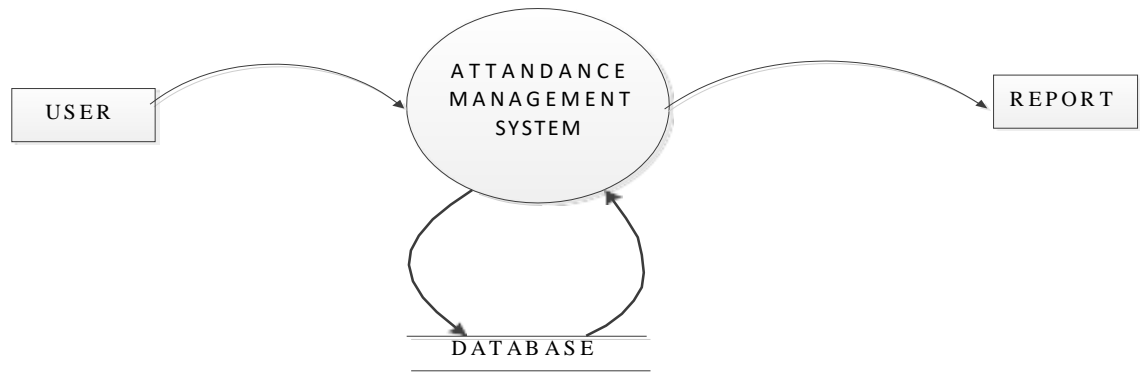
5.5.1 DFD level 0

Figure 5.5.1-DataFlowDiagram Level1

5.5.1 DFD level 1

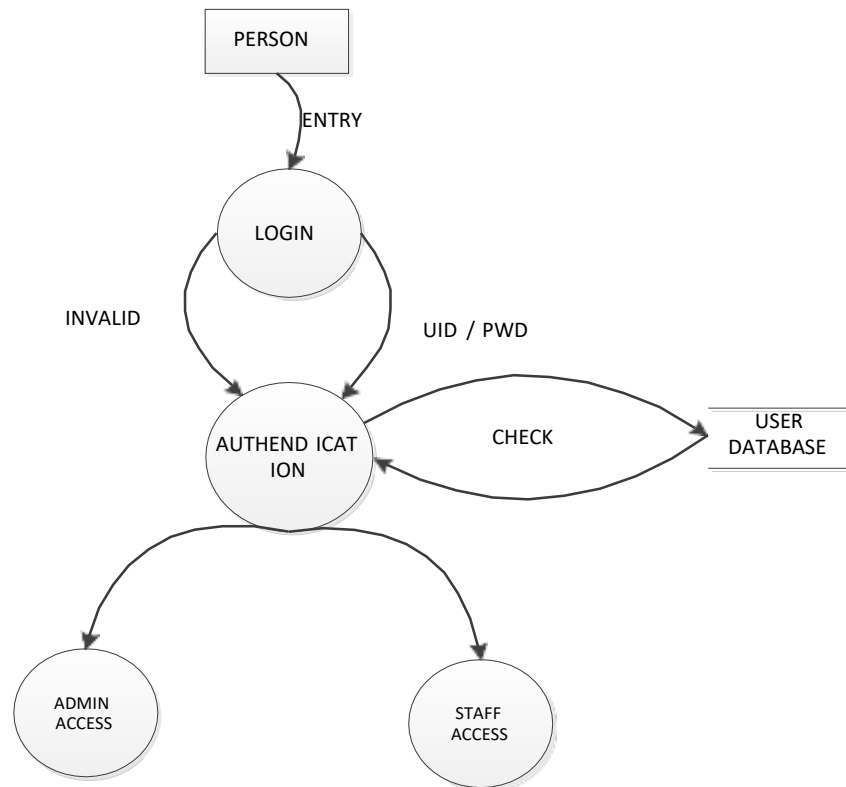
Figure 5.5.2-DataFlowDiagram Level1

5.5.1.1 Admin:
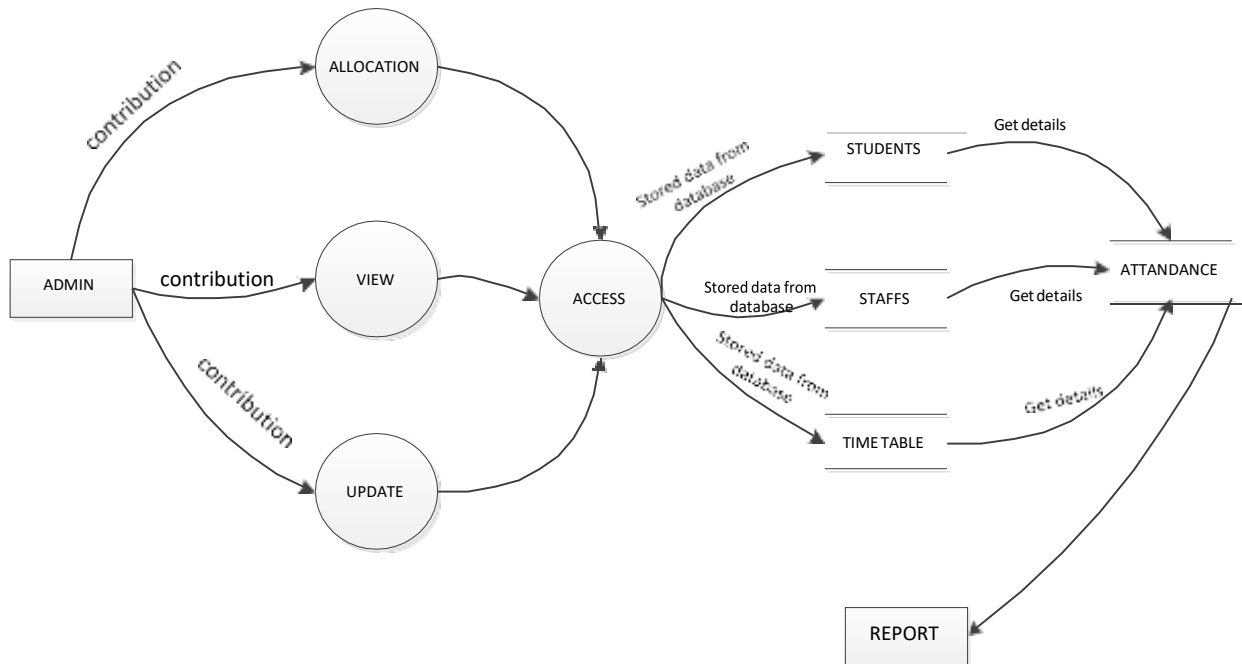
Figure 5.5.3.1-DataFlowDiagram Level2

5.5.1.2 staffs:
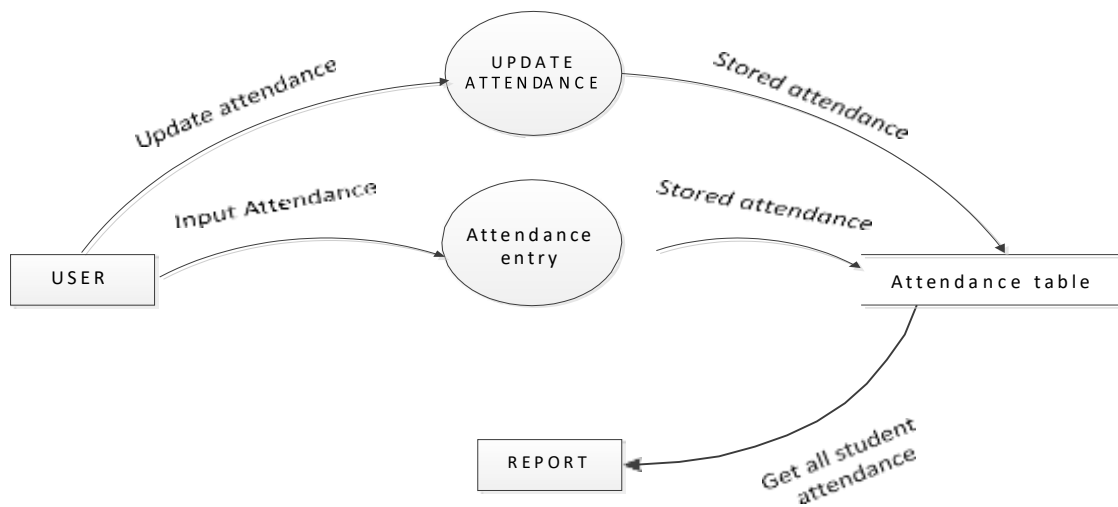
Figure 5.5.3.2-DataFlowDiagram Level2

5.6 Database Design:

5.6.1.1 LOGIN TABLE:

To create a login details for the table.

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

Table:5.7.2.1 -Login Table

Staffs Table:

To create username and password for the staff details.

FIELDS	DATATYPE	CONSTRAINTS	DESCRIPTION
Scode	varchar(20)	Primary key	Define separatesubject code id
ssname	Varchar(15)	Not Null	Short subject name (ex:cpp)
sname	Varchar(20)	Not Null	Staffs name
Password	Varchar(20)	Not Null	Staff login password

Table:5.7.2.2 –Student details Tabl

5.6.1.2 Student table:

To create table for Student personal details for our department.

FIELDS	DATATYPE	CONSTRAINTS	DESCRIPTION
Roll no	Varchar(15)	Primary key	Student roll number
Name	Varchar(20)	Not Null	Student name
Dept	Varchar(30)	Not Null	Department name
Year	Number	Not Null	Batch year
DOB	Varchar(20)	Not Null	Student date of birth
ADDRESS	Varchar(20)	Not Null	Student permanent address
MNO	Varchar(20)	Not Null	Student mobile number
EID	Varchar(30)	Not Null	Student E-mail id
CSTATUS	Varchar(20)	Not Null	Student status for dayscholler/Hosteller

Table:5.7.2.3 –Staff Details Table

5.6.1.3 Time table:

To create the subject time table for a particular class.

FIELDS	DATATYPE	CONSTRAINTS	DESCRIPTION
Day	Varchar(20)	Primary key	Days insert (ex:Monday)
I	Varchar(20)	Not Null	set the period for 1 particular subject
II	Varchar(20)	Not Null	set the period for 2 particular subject
III	Varchar(20)	Not Null	set the period for 3 particular subject
IV	Varchar(20)	Not Null	set the period for 4 particular subject
V	Varchar(20)	Not Null	set the period for 5 particular subject
VI	Varchar(20)	Not Null	set the period for 6 particular subject
VII	Varchar(20)	Not Null	set the period for 7 particular subject

Table:5.7.2.4 -Time Table

5.6.1.4 Attendance table:

To create attendance details for particular class .

FIELDS	DATATYPE	CONSTRAINTS	DESCRIPTION
Dates	Date	Primary key	Enter day by day attendance
Hour	Number	Primary key	Set particular hour only
Subject	Varchar(15)	Not Null	Particular Subject
Rollno(1 to 60)	Varcahar(20)	Not Null	Enter Present absent details in particular student(ex:M11MCA001)

Table:5.7.2.5 -Attendance Table

5.6.2 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 tocalculate 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.

5.6.3 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.

CHAPTER 6

SYSTEM TESTING

6.1 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

errorsIn program function, behavior and performance.

6.2 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.

6.3 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.

6.3.1 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 .

6.3.2 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.

6.3.3 Performance Testing

Performance testing is designed to test the run-time performance of softwarewithin 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.

6.4 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 stepsto 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 isa 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:

6.4.1 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

6.4.2 MASTER form

Sno	Test case id	Test case name	Test case desc	Step	Expected result	Actual Result	Test case status pass/fail
1	Create student 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 succesfull	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 ,user name not equal to null must be displayed	Inserted succesfull	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 succesfull	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

6.4.3 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

CHAPTER 7

SYSTEM IMPLEMENTATION

7.1 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 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.

7.2 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.

CHAPTER 8

APPENDICES

8.1 Source code:

LOGIN:

```php
<?php
 session_start();
 $isIndex = 0;
 if(!(array_key_exists('teacher_id',$_SESSION) && isset($_SESSION['teacher_id']))) {session_destroy();
   if(!$isIndex) header('Location: index.php');
 }
?>
<?php include 'php/node_class.php'; ?>
<html>
 <head>
  <link rel="stylesheet" href="css/style.css"/>
  <title>Teacher Dashboard</title>
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <link rel="stylesheet" href="css/bootstrap-theme.min.css">
  <script src="js/jquery.min.js"></script>
  <script src="js/bootstrap.min.js"></script>
  <script src="js/teacher.js"></script>
  <!-- Custom styles for this template -->
    <link href="navbar-fixed-top.css" rel="stylesheet">
 </head>
 <body style="background-color:pink;">
  <!-- Fixed navbar -->
   <nav class="navbar navbar-inverse navbar-fixed-top" style="background-color:yellow;">
    <div class="container" style="background-color:yellow;">
     <div class="navbar-header">
       <button type="button" class="navbar-toggle collapsed" data-toggle="collapse" data-target="#navbar" aria-expanded="false" aria-controls="navbar">
         <span class="sr-only">Toggle navigation</span>
         <span class="icon-bar"></span>
```

```
    <span class="icon-bar"></span>
    <span class="icon-bar"></span>
   </button>
   <a class="navbar-brand" href="index.php" style="color:orange">Online Attendance</a>
  </div>
  <div id="navbar" class="navbar-collapse collapse">
   <ul class="nav navbar-nav navbar-right" style="background-color:pink;">
    <li class="active"><a href="teacher.php" style="color:green; background-color:yellow;
background-color:yellow;">Dashboard</a></li>
     <li><a href="profile.php" style="color:green; background-color:yellow;">Profile</a></li>

          <li><a href="statistics.php" style="color:green;background-
color:yellow;">Statistics</a></li>
             <li><a href="logout.php" style="color:green;background-
color:yellow;">Logout</a></li>
<li><a href="index.php" style="color:green; background-color:yellow;">BACK</a></li>

     </ul>
    </div><!--/.nav-collapse -->
   </div>
  </nav></br></br></br></br>

 <div class="container" style="color:red">
  <?php
   $name = $_SESSION['name'];
   $classes = $_SESSION['classes'];
   $teacher_id = $_SESSION['teacher_id'];
   echo '<h2>Welcome , '.$name.'.</h2>';
   echo '<div class="wrapper">';
   // FOR EACH CLASS , GET IT'S INFO AND PREPARE A LINK
   $n = new Node;

   if(!$classes) {
    echo '<h3 class="no-classes">You haven\'t taken any class yet!</h3>';
   } else {
    echo '<h3 class="no-classes">Click on a class to take attendance.</h3>';
    foreach($classes as $class_id) {
```

```php
    $node = $n->retrieveObjecti($class_id,$teacher_id) or die("No such record");
    $code = $node->getCode();
    $section = $node->getSection();
    $year = $node->getYear();
    $numClasses = $node->getDays();
    $link = 'take.php?cN='.$class_id;
    echo '<div class="class">

      <button class="btn btn-danger delete-class-warning" data-toggle="modal" data-target=".delete-warning">&times;</button>
      <a class="no-decoration" href="'.$link.'">
      <div><strong>Code</strong> : <span class="code">'.$code.'</span></div>
      <div><strong>Section</strong> : <span class="section">'.$section.'</span></div>
      <div><strong>Year</strong> : <span class="year">'.$year.'</span></div>
      <div><strong>Classes</strong> : '.$numClasses.'</div>
    </div></a>';
  }
}
echo '<div class="class" data-toggle="modal" data-target=".bs-example-modal-lg" id="addClass">
  <span class="glyphicon glyphicon-plus"></span>
 </div>
</div>';
?>

</div>
<div class="modal fade bs-example-modal-lg" tabindex="-1" role="dialog" aria-labelledby="addClass" aria-hidden="true">
  <div class="modal-dialog modal-lg">
    <div class="modal-content">
     <h2 class="text-center">Add Class </h2>
     <hr>
      <div id="add_class_form">
        <select class="form-control" name="year">
        <?php foreach(range(date('Y',time()),1983) as $r) echo '<option>'.$r.'</option>'; ?>
        </select>
<select class="form-control" name="month">
        <?php foreach(range(date('M',time()),12) as $r) echo '<option>'.$r.'</option>'; ?>
```

```
        </select>
<select class="form-control" name="day">
        <?php foreach(range(date('M',time()),31) as $r) echo '<option>'.$r.'</option>'; ?>
        </select>
        <input class="form-control" name="code" placeholder=" Eg :Branch-codelike 302">
        <select class="form-control" name="section">
        <option value="-1">Choose Year</option>
        <?php foreach(range(1,4) as $r) echo '<option>'.$r.'</option>'; ?>
        </select>
        <select class="form-control" name="semester">
        <option value="-1">Choose Semester</option>
        <?php foreach(range(1,8) as $r) echo '<option>'.$r.'</option>'; ?>
        </select>
        <input class="form-control" name="start" placeholder="Starting Roll Number (Eg.
   401/CO/12)">
        <input class="form-control" name="end" placeholder="Ending Roll Number (Eg.
   465/CO/12)">
        <button class="btn btn-primary" id="add">Add Class</button>
        <button class="btn" id="cancel">Cancel</button>
      </div>
    </div>
  </div>
</div>
<div class="modal fade delete-warning" tabindex="-1" role="dialog" aria-labelledby="delete-
   warning" aria-hidden="true">
  <div class="modal-dialog modal-sm">
    <div class="modal-content">
    <h2 class="text-center"> Do you really want to delete <br> <span class="warning-
   class"></span> ?</h2>
    <hr>
    <div class="text-center">
     <p>
      Are you sure you want to delete <span class="warning-class"></span> ? <br>
      You can't undo this action.
     </p>
     <button class="btn btn-danger delete-class-code">Delete</button> <button class="btn btn-
   primary"  onclick="$('.delete-warning').modal('hide');">Cancel</button>
```

```
        </div>
      </div>
    </div>
  </div>
      <div style="text-align:center;padding:20px; ">
</div>
 </body>
      </html>
```

Attendance Entry:

```php
<?php
  session_start();
  $isIndex = 0;
  if(!(array_key_exists('teacher_id',$_SESSION) && isset($_SESSION['teacher_id']))) {session_destroy();
    if(!$isIndex) header('Location: index.php');
  }
?>
<?php include 'php/node_class.php'; ?>
<html>
 <head>
  <link rel="stylesheet" href="css/style.css"/>
  <title>Edit Class</title>
  <link rel="stylesheet" href="css/bootstrap.min.css">
  <link rel="stylesheet" href="css/bootstrap-theme.min.css">
  <script src="js/jquery.min.js"></script>
  <script src="js/bootstrap.min.js"></script>
  <script src="js/class.js"></script>
  <style>.form-control{display:inline-block !important; width: 185px !important; margin:5px
!important;}.details{padding:5px 10px;margin-bottom:30px;border: 1px solid lightgrey;border-top:
none;}}</style>
 </head>
 <body>
  <div id="header" class="clearfix">
    <h1>Netaji Subhas Institute of Technology</h1>
    <h3>Edit Class</h3>
  </div>
  <nav class="navbar navbar-default" id="sub-menu">
    <div class="navbar-header">
```

```
      <button type="button" class="navbar-toggle" data-toggle="collapse" data-target="#bs-example-
navbar-collapse-1">
        <span class="sr-only">Toggle navigation</span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
        <span class="icon-bar"></span>
      </button>
    </div>
    <div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">
      <ul class="nav navbar-nav navbar-left">
        <li><a href="teacher.php">Home</a></li>
        <li><a href="profile.php">Profile</a></li>
        <li class="active"><a href="class.php">Classes</a></li>
        <li><a href="statistics.php">Statistics</a></li>
        <li><a href="logout.php">Logout</a></li>
      </ul>
    </div>
  </nav>
  <div class="container">
    <h2>You can edit details of your classes here. </h2>
    <?php
      $classes = $_SESSION['classes'];
      $teacher_id = $_SESSION['teacher_id'];
      if(!$classes) echo '<h4> You haven\'t taken any classes yet. </h4>';else {
        foreach($classes as $class_id) {
          $n = new Node;
          $node = $n->retrieveObjecti($class_id,$teacher_id) or die("No such record");
          $code = $node->getCode();
          $section = $node->getSection();
          $year = $node->getYear();
          $semester = $node->getSemester();

          echo '<ul class="nav nav-tabs">
                <li class="active"><a href="#"><strong>'.$code . ' ( '.$section.' ) ,
'.$year.'</strong></a></li>
                </ul>';
          echo '<div class="details" id="_'.$class_id.'_">';
          echo 'Code : <input class="form-control" name="code" value="'.$code.'" placeholder="Entercode ,
eg COE-123">';
          echo 'Year : <input class="form-control" name="year" value="'.$year.'" placeholder="Enter
Year">';
          echo 'Section : <input class="form-control" name="section" value="'.$section.'"
          </html>
```

8.2 Screen Shots:

1. LOGIN:

2. ADMIN HOME PAGE

3.STUDENT DETAILS:

4.Profile:

CHAPTER 9

CONCLUSION AND FUTURE ENHANCEMENT

9.1 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 limitationsincorporated in the attendance.

> ➤ Easy implementation Environment
> ➤ Generate report Flexibly

9.2 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.

9.2.1 Discontinue of particular student eliminate potential attendance.

9.2.2 Bar code Reader based attendance system.

9.2.3 Individual Attendance system With photo using Student login.

CHAPTER 10

REFERENCES

Books References

➤ Introducing Microsoft .NET, Second Edition author David S. Platt.

➤ Joe Mayo, "**Microsoft Visual Studio 2010: A Beginner's Guide"**, Tata McGrawHill, 2010.

➤ Alex Mackey, **"Introducing .NET 4.0: With Visual Studio 2010"**, Press, USA,2010.

➤ ASP.NET 2.0 Website Programming: Problem-Design-Solution(Programmer to Programmer) by Marco Bellinaso

➤ Professional ASP.NET 2.0 by Bill Evjen, Scott Hanselman, Farhan

➤ OReilly (E-Book) – Programming in .Net

➤ Oreille (E-Book) – Framework + Essentials.pdf

➤ Beginner ASP.NET 2.0 – Unleashed

➤ PL/SQL Programming – Evan Bayross