1. INTRODUCTION

1.1. PROJECT DESCRIPTION

Administrator registers the student details in the application. He enters the unique registration number for each student, providing personal information such as address, parent details, contact details and academic information such as dept, degree details and any other essential information like parent’s mobile no and parent’s mail id. Through this information, administrator can send the essential information to the student’s parent on time. Admin can also edit, modify and view the details of students who are registered in this application.

Administrator can also register staff details in this project. He will enroll the information about staffs such as staff id, name, gender, mail id and their mobile no to store into the database. Theses information will be retrieved and used by administrator when staffs may wish to get the gate pass from administration. Admin can also edit, modify and view the details of staffs who are registered in this application.

In this module, administrator will assign the staff as a class teacher and that staff is declared as a responsible person to the particular class and particular section. During gate pass process for the student, admin will send the details of the student and his gate pass details to the appropriate staff using staff’s mail id. It will be used to get the knowledge of that student movement by class teacher.

After giving request to get gate pass by student manually, then administrator will do respective process using this application. At the moment, admin will give the gate pass details for a student such as student’s register number, parent mail id, staff’s mail id, date, time and purpose of leaving from the institute. After that, this information will be sent to his parent’s mail id and class teacher’s mail id to provide notification about the student. Admin can view gate pass details for all the students.

After giving request to get gate pass by staff manually, then administrator will do respective process using this application. At the moment, admin will give the gate pass details for a staff such as staff id, mail id, date, time and purpose of leaving from the institute. Then, those details will be sent to staff’s mail id for getting notifications.

2. SYSTEM ANALYSIS

###### **INTRODUCTION**

System Analysis and Design, is the process of gathering and interpreting facts, diagnosing problem and using the information to recommend improvement to the system. Before development of any project can be pursued, a system study is conducted to learn the details of the current business solution. Information gathered through the study forms the basis for creating alternative design strategies. Virtually all organizations are systems that interact with their environment through receiving input and producing output.

It is a management technique used in designing a new system, improving an existing system or solving problem. System analysis does not guarantee that the user will derive an ideal solution to a problem. This depends solely on the way one design a system to exploit the potential in the method. To put it in another way, creativity is as much as must pre-design the study and problem solving process and evaluate every successive step in the system analysis.

Taking all these factors into account and with the knowledge of the inter-relationship between the various fields and section and their potential interactions, they are consider for developing the whole system in and integrated manner, this project is developed to meet all the criteria in the

The management technique is also helps us in develop and design of the new system or to improve the existing system.

The following Objectives are kept in mind:

* Identify the customer’s need.
* Evaluate the system concept for feasibility.
* Perform economic and technical analysis.
* Allocate functions to hardware, software, people, database and other system elements.
* Establish cost and schedule constraints
* Create a system definition that forms the foundation for all subsequent engineering work.

Identification of the need:

In this, there are certain expressions that are being used in the development of the project. And, it is used to identify our needs or source in the project.

* Defining a problem
* Finding the various need for the problem
* Formalizing the need
* Relating the need

Thus, it is the first step for system development life cycle.

**Initial Investigation**

It is one way of handling the project, it is used to know about the user request and the modification of the system should be done.

The user’s request for this project is as follows:

1. Assigning separate work area for different users.
2. Nature of the work
3. Regular update and delete of record
4. Regular calculation of Net Asset Value
5. Supplying the data with the time required.

The user request identifies the need for change and authorizes the initial investigation. It may undergo several modifications before it become a written commitment. Once approved the activities are carried out into action. The proposal, when approved, it initiates a detailed user-oriented specification of system performance and analysis of the feasibility of the evaluating alternative candidate systems with a recommendation of the best system for the job.

###### **Feasibility Study**

The objective of the feasibility study is not only to solve the problem but also to acquire a sense of its scope. The reason for doing this is to identify the most beneficial project to the organization.

There are three aspects in the feasibility study:

1. Technical Feasibility
2. Financial Feasibility
3. Operating Feasibility

**Technical Feasibility**

The Technical feasibility is the study of the software and how it is included in the study of our project. Regarding this there are some technical issues that should be noted they are as follows:

* Is the necessary technique available and how it is suggested and acquired?
* Does the proposed equipment have the technical capacity to hold the data required using the new system?
* Will the system provide adequate response that is made by the requester at an periodic time interval
* Can this system be expanded after this project development
* Is there a technique guarantees of accuracy, reliability in case of access of data and security

The technical issues are raised during the feasibility study of investigating our System. Thus, the technical consideration evaluates the hardware requirements, software etc. This system uses JSP as front end and Oracle as back end. They also provide sufficient memory to hold and process the data. As the company is going to install all the process in the system it is the cheap and efficient technique.

This system technique accepts the entire request made by the user and the response is done without failure and delay. It is a study about the resources available and how they are achieved as an acceptable system. It is

an essential process for analysis and definition of conducting a parallel assessment of technical feasibility.

Though storage and retrieval of information is enormous, it can be easily handled by Oracle. As the oracle can be run in any system and the operation does not differ from one to another. So, this is effective.

###### **Economical Feasibility**

An organization makes good investment on the system. So, they should be worthful for the amount they spend in the system. Always the financial benefit and equals or less the cost of the system, but should not exceed the cost.

* The cost of investment is analyzed for the entire system
* The cost of Hardware and Software is also noted.
* Analyzing the way in which the cost can be reduced

Every organization want to reduce there cost but at the same time quality of the Service should also be maintained. The system is developed according the estimation of the cost made by the concern. In this project, the proposed system will definitely reduce the cost and also the manual work is reduced and speed of work is also increased.

###### **Operational Feasibility**

Proposed project will be beneficial only when they are turned into an information system and to meet the organization operating requirements. The following issues are considered for the operation:

* Does this system provide sufficient support for the user and the management?
* What is the method that should be used in this project?
* Have the users been involved in the planning and development of the projects?
* Will the proposed system cause any harm, bad result, loss of control and accessibility of the system will lost?

Issues that may be a minor problem will sometimes cause major problem in the operation. It is the measure of how people can able to work with the system. Finding out the minor issues that may be the initial problem of the system. It should be a user-friendly environment. All these aspect should be kept in mind and steps should be taken for developing the project carefully.

Regarding the project, the system is very much supported and friendly for the user. The methods are defined in an effective manner and proper conditions are given in other to avoid the harm or loss of data. It is designed in GUI interface, as working will be easier and flexible for the user.

They are three basic feasibility studies that are done in every project.

2.1. EXISTING SYSTEM

In existing system, most of the organization and educational institutes using the named method like ledger in keeping track of all the students and staffs permission details and records during their class time in each of the department. Problems may be raised when at certain point of time the number of students getting the gate pass and leave from the institute for various reason for the department increasing and unable to manage and messed up. Manual process leads to face many problems such as maintaining records, informing to the parents of respective students immediately.

2.2. PROPOSED SYSTEM

In proposed system, it reduces the paperwork and to maintain the document in electronic form using ASP.Net and SQL Server database. It maintains the records and data about all students, staffs details as well as gate pass information accurately and consistent records on gate usage. Once the students got the gate pass for some reason, this project will inform to the appropriate parents immediately through their mail id automatically. It also eliminates the duplicity of the pass and allows the verified student to cross the premises. In the earlier system, there were a lot of duplicities done by the students which are going to be removed by using this system

3. SYSTEM REQUIREMENT SPECIFICATION

3.1. HARDWARE SPECIFICATION

The above Hardware specifications were used in both Server and Client machines when developing.

Processor : Intel(R) Core(TM) i3

Processor Speed : 3.06 GHz

Ram : 2 GB

Hard Disk Drive : 250 GB

CD-ROM Drive : Sony

Monitor : “17” inches

Keyboard : TVS Gold

Mouse : Logitech

3.2. SOFTWARE SPECIFICATION

SERVER

Operating System : Windows 7

Technology Used : Microsoft ASP.NET

Database : Sql Server

Database Connectivity : ActiveX Data Object (ADO)

Web Server : Internet Information Server

Browser : Internet Explorer 6.0

CLIENT

Operating System : Windows 7

Browser : Internet Explorer 6.0

3.2.1 DESCRIPTION OF SOFTWARE

### **Introduction to ASP.NET**

# ASP.NET, the next version of ASP, is a programming Framework that is used to create enterprise – class web applications. The enterprise class web applications are accessible on a global basis loading to efficient information management. However, the advantages that ASP.NET offers make it more than just next version of ASP.NET.

ASP.NET is integrated with visual studio.Net, which provides a GUI designer, a rich toolbox and a fully integrated debugger. This allows the development of applications in a what you see is what you get (WYSIWYG) MANNER.

The .NET Framework is a common environment for building, deploying, and running Web Services and Web Applications. The .NET Framework contains common class libraries - like ADO.NET, ASP.NET and Windows Forms - to provide advanced standard services that can be integrated into a variety of computer systems.

The .NET Framework is language neutral. Currently it supports C++, C#, Visual Basic, JScript (The Microsoft version of JavaScript) .The new Visual Studio.NET is a common development environment for the new .NET Framework. It provides a feature-rich application execution environment, simplified development and easy integration between a number of different development languages.

Unique Features Of .Net Environment

1. Internet Inside
2. Common Language support
3. Common Class Libraries
4. Common Language Runtime
5. Garbage Collection.
6. Cross Language Reference
7. Web Services

INTERNET INFORMATION SERVICE 6.0

IIS 6.0 has strong support for more programming to take place on the server, to allow the new Web Applications to run in any browser on any platform.

## **ASP.NET**

* ASP.NET is a server side scripting technology that enables scripts (embedded in web pages) to be executed by an Internet server.
* ASP.NET is a Microsoft Technology
* ASP.NET stands for Active Server Pages
* ASP.NET is a program that runs inside IIS
* IIS stands for Internet Information Services
* IIS comes as a free component with Windows 2005
* IIS is also a part of the Windows NT 4.0 Option Pack
* The Option Pack can be downloaded from Microsoft PWS is a smaller - but fully functional - version of IIS PWS can be found on your Windows 95/98 CD.
* ASP.NET 3.0 is the latest version of ASP.NET, but there will never be an ASP.NET 4.0 version.
* ASP.NET is the next generation ASP.NET, but it's not an upgraded version of ASP.NET. ASP.NET is an entirely new paradigm for server-side ASP.NET script
* ASP.NET is a part of the new .NET (dotnet) Framework. Microsoft spent three years rewriting ASP.NET from the ground up, and ASP.NET is not fully backward compatible with ASP.NET 3.0.
* ASP.NET has better language support, a large set of new controls and XML based components, and better user authentication.
* ASP.NET provides increased performance by running compiled code.
* ASP.NET code is not fully backward compatible with ASP.NET.
* ASP.NET is a server side programming language.
* ASP.NET is an object oriented programming language.
* Active Server Pages - ASP.NET
* ASP.NET is the latest version of ASP.NET. It includes Web Services to link applications, services and devices using HTTP, HTML, XML and SOAP.

NEW IN ASP.NET

1. New Language Support
2. Programmable Controls
3. Event Driven Programming
4. XML Based Components
5. User Authentication
6. User Accounts and Roles
7. High Scalability
8. Compiled Code
9. Easy Configuration
10. Easy Deployment
11. Includes ADO .NET

## **WEB SERVICES**

* Web services are small units of code
* Web services are designed to handle a limited set of tasks
* Web services use XML based communicating protocols
* Web services are independent of operating systems
* Web services are independent of programming languages .
* .NET Web Services
* Web services are small units of code built to handle a limited task.
* Small Units of Code
* Web services are small units of code designed to handle a limited set of tasks.

## **XML BASEDWEB PROTOCOLS**

Web services use the standard web protocols HTTP, XML, SOAP, WSDL, and UDDI.

## **HTTP**

HTTP (Hypertext Transfer Protocol) is the World Wide Web standard for communication over the Internet.

## **XML**

XML (extensible Markup Language) is a well-known standard for storing, carrying, and

Exchanging data.

## **SOAP**

Simple Object Access Protocol is a lightweight platform and language neutral

Communication protocol that allows programs to communicate via standard Internet

HTTP

## **WSDL**

WSDL (Web Services Description Language) is an XML-based language used to define web services and to describe how to access them.

## **UDDI**

Universal Description, Discovery and Integration is a directory service where businesses can register and search for web services.

UDDI is a public registry, where one can publish and inquire about web services.

## **INDEPENDENT OF OPERATING SYSTEM**

Since web services use XML based protocols to communicate with other systems, web services are independent of both operating systems and programming languages.

An application calling a web service will always send its requests using XML, and get its answer returned as XML. The calling application will never be concerned about the operating system or the programming language running on the other computer.

## **BENEFITS OF WEB SERVICES**

* Easier to communicate between applications.
* Easier to distribute information to more consumers.
* Rapid development.
* Web services make it easier to communicate between different applications.
* They also make it possible for developers to reuse existing web services.
* Instead of writing new ones.

Web services can create new possibilities for many businesses because it provides an easy way to distribute information to a large number of consumers. One example could be flight schedules and ticket reservation systems.

INTRODUCTION TO SQL SERVER

The database component of Microsoft® SQL Server™ 2005 is a Structured Query Language (SQL)–based, scalable, relational database with integrated Extensible Markup Language (XML) support for Internet applications. Each of the following terms describes a fundamental part of the architecture of the SQL Server 2005 database component:

DATABASE

A database is similar to a data file in that it is a storage place for data. Like a data file, a database does not present information directly to a user; the user runs an application that accesses data from the database and presents it to the user in an understandable format.

Database systems are more powerful than data files in that data is more highly organized. In a well-designed database, there are no duplicate pieces of data that the user or application must update at the same time. Related pieces of data are grouped together in a single structure or record, and relationships can be defined between these structures and records.

When working with data files, an application must be coded to work with the specific structure of each data file. In contrast, a database contains a catalog that applications use to determine how data is organized. Generic database applications can use the catalog to present users with data from different databases dynamically, without being tied to a specific data format.

A database typically has two main parts: first, the files holding the physical database and second, the database management system (DBMS) software that applications use to access data. The DBMS is responsible for enforcing the database structure, including:

* Maintaining relationships between data in the database.
* Ensuring that data is stored correctly, and that the rules defining data relationships

are not violated.

* Recovering all data to a point of known consistency in case of system failures.

RELATIONAL DATABASE

Although there are different ways to organize data in a database, relational databases are one of the most effective. Relational database systems are an application of mathematical set theory to the problem of effectively organizing data. In a relational database, data is collected into tables (called relations in relational theory).

A table represents some class of objects that are important to an organization. For example, a company may have a database with a table for employees, another table for customers, and another for stores. Each table is built of columns and rows (called attributes and tuples in relational theory). Each column represents some attribute of the object represented by the table. For example, an Employee table would typically have columns for attributes such as first name, last name, employee ID, department, pay grade, and job title. Each row represents an instance of the object represented by the table. For example, one row in the Employee table represents the employee who has employee ID 12345.

When organizing data into tables, you can usually find many different ways to define tables. Relational database theory defines a process called normalization, which ensures that the set of tables you define will organize your data effectively.

SCALABLE

SQL Server 2005 supports having a wide range of users access it at the same time. An instance of SQL Server 2005 includes the files that make up a set of databases and a copy of the DBMS software. Applications running on separate computers use a SQL Server 2005 communications component to transmit commands over a network to the SQL Server 2005 instance. When an application connects to an instance of SQL Server 2005, it can reference any of the databases in that instance that the user is authorized to access. The communication component also allows communication between an instance of SQL Server 2005 and an application running on the same computer. You can run multiple instances of SQL Server 2005 on a single computer.

SQL Server 2005 is designed to support the traffic of the largest Web sites or enterprise data processing systems. Instances of SQL Server 2005 running on large, multiprocessor servers are capable of supporting connections to thousands of users at the same time. The data in SQL Server tables can be partitioned across multiple servers, so that several multiprocessor computers can cooperate to support the database processing requirements of extremely large systems. These groups of database servers are called federations.

Although SQL Server 2005 is designed to work as the data storage engine for thousands of concurrent users who connect over a network, it is also capable of working as a stand-alone database directly on the same computer as an application. The scalability and ease-of-use features of SQL Server 2005 allow it to work efficiently on a single computer without consuming too many resources or requiring administrative work by the stand-alone user. The same features allow SQL Server 2005 to dynamically acquire the resources required to support thousands of users, while minimizing database administration and tuning. The SQL Server 2005 relational database engine dynamically tunes itself to acquire or free the appropriate computer resources required to support a varying load of users accessing an instance of SQL Server 2005 at any specific time. The SQL Server 2005 relational database engine has features to prevent the logical problems that occur if a user tries to read or modify data currently used by others.

STRUCTURED QUERY LANGUAGE

To work with data in a database, you have to use a set of commands and statements (language) defined by the DBMS software. Several different languages can be used with relational databases; the most common is SQL. The American National Standards Institute (ANSI) and the International Standards Organization (ISO) define software standards, including standards for the SQL language. SQL Server 2005 supports the Entry Level of SQL-92, the SQL standard published by ANSI and ISO in 1992. The dialect of SQL supported by Microsoft SQL Server is called Transact-SQL (T-SQL). T-SQL is the primary language used by Microsoft SQL Server applications.

EXTENSIBLE MARKUP LANGUAGE

XML is the emerging Internet standard for data. XML is a set of tags that can be used to define the structure of a hypertext document. XML documents can be easily processed by the Hypertext Markup Language, which is the most important language for displaying Web pages.

Although most SQL statements return their results in a relational, or tabular, result set, the SQL Server 2005 database component supports a FOR XML clause that returns results as an XML document. SQL Server 2005 also supports XPath queries from Internet and intranet applications. XML documents can be added to SQL Server databases, and the OPENXML clause can be used to expose data from an XML document as a relational result set.

##### **4. SYSTEM DESIGN**

4.1. Database Design

Admin Login

False

True

Student Details

View All Student Details.Selected Registration Numberwise Update or Delete the Student Details

StudTable

Staff Details

Insert New Staff Details

View All Staff Details.Selected Staff Id wise Update or Delete the Staff Details

StaffTable

Insert New Student Details

Assign Class Teacher

Show All Staff Id.Select Any One Staff Id Class Teacher Assigning Details Successfully Inserted.

View Class Teacher Assigning Details.

StaffTable SCTTable

Report

Selected Date Wise Staff and Student Gate Pass Details

GPTable GPTable1

Student Gate Pass

Registration Number Based Gate Pass Details Successfully Inserted.Gate Pass Information Send to Parent and Class Teacher Mail.

View Registratiion Number Based Gate Pass Details.

StudTable GPTable

Staff Gate Pass

Staff Id Based Gate Pass Details Successfully Inserted.Gate Pass Information Send to School Mail.

View Staff Id Based Gate Pass Details.

StaffTable GPTable1

4.2 Table Design

Table Name: GPTable

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| SRNo | numeric(5, 0) | Foreign Key | Registration Number |
| SName | varchar(50) | Not Null | Student Name |
| GPDate | datetime | Not Null | Gate Pass Date |
| GPTime | varchar(50) | Not Null | Gate Pass Time |
| GPPurpose | varchar(350) | Not Null | Purpose |

GPTable1

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| StaffId | numeric(5, 0) | Foreign Key | Staff Id |
| StaffName | varchar(50) | Not Null | Staff Name |
| GPDate | datetime | Not Null | Gate Pass Date |
| GPTime | varchar(50) | Not Null | Gate Pass Time |
| GPPurpose | varchar(350) | Not Null | Purpose |

SCTTable

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| StaffId | numeric(10, 0) | Foreign Key | Staff Id |
| StaffName | varchar(50) | Not Null | Staff Name |
| Standard | varchar(20) | Not Null | Standard |

StaffTable

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| StaffId | numeric(5, 0) | Primary Key | Staff Id |
| StaffName | varchar(50) | Not Null | Staff Name |
| Gender | varchar(20) | Not Null | Gender |
| SMailId | varchar(50) | Not Null | Mail Id |
| SMNo | numeric(10, 0) | Not Null | Mobile Number |

StudTable

|  |  |  |  |
| --- | --- | --- | --- |
| Field Name | Data Type | Constraint | Description |
| SRNo | numeric(5, 0) | Primary Key | Registration Number |
| SName | varchar(50) | Not Null | Student Name |
| Gender | varchar(20) | Not Null | Gender |
| Standard | varchar(20) | Not Null | Standard |
| PMailId | varchar(50) | Not Null | Parent Mail id |
| PMNo | numeric(10, 0) | Not Null | Parent Mobile Number |

5. SYSTEM IMPLEMENTATION

5.1 Module Description

Student Registration

Administrator registers the student details in the application. He enters the unique registration number for each student, providing personal information such as address, parent details, contact details and academic information such as dept, degree details and any other essential information like parent’s mobile no and parent’s mail id. Through this information, administrator can send the essential information to the student’s parent on time. Admin can also edit, modify and view the details of students who are registered in this application.

Staff Registration

Administrator can also register staff details in this project. He will enroll the information about staffs such as staff id, name, gender, mail id and their mobile no to store into the database. Theses information will be retrieved and used by administrator when staffs may wish to get the gate pass from administration. Admin can also edit, modify and view the details of staffs who are registered in this application.

Assigning Class Teacher

In this module, administrator will assign the staff as a class teacher and that staff is declared as a responsible person to the particular class and particular section. During gate pass process for the student, admin will send the details of the student and his gate pass details to the appropriate staff using staff’s mail id. It will be used to get the knowledge of that student movement by class teacher.

Student Gate Pass Details

After giving request to get gate pass by student manually, then administrator will do respective process using this application. At the moment, admin will give the gate pass details for a student such as student’s register number, parent mail id, staff’s mail id, date, time and purpose of leaving from the institute. After that, this information will be sent to his parent’s mail id and class teacher’s mail id to provide notification about the student. Admin can view gate pass details for all the students.

Staff Gate Pass Details

After giving request to get gate pass by staff manually, then administrator will do respective process using this application. At the moment, admin will give the gate pass details for a staff such as staff id, mail id, date, time and purpose of leaving from the institute. Then, those details will be sent to staff’s mail id for getting notifications.

5.2 Sample Code

Admin Login

public partial class AdminLogin : System.Web.UI.Page

{

protected void Page\_Load(object sender, EventArgs e)

{

Label1.Text = "";

}

protected void LinkButton1\_Click(object sender, EventArgs e)

{

if (TextBox1.Text.Equals("Admin") && TextBox2.Text.Equals("Admin"))

Response.Redirect("StudentForm.aspx");

else

Label1.Text = "Invalid UserName or Password";

}

}

Student Details

using System.Data.SqlClient;

using System.Configuration;

public partial class StudentForm : System.Web.UI.Page

{

SqlConnection con;

SqlCommand cmd;

SqlDataReader rs;

protected void Page\_Load(object sender, EventArgs e)

{

try

{

Label1.Text = "";

con = new SqlConnection(ConfigurationManager.ConnectionStrings["connection"].ConnectionString);

con.Open();

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void LinkButton1\_Click(object sender, EventArgs e)

{

try

{

cmd = new SqlCommand("select srno from studtable where srno=@srno", con);

cmd.Parameters.AddWithValue("srno", TextBox1.Text);

rs = cmd.ExecuteReader();

bool b = rs.Read();

rs.Close();

cmd.Dispose();

if (b)

{

Label1.Text = "Registration Number Already Found.....";

return;

}

cmd = new SqlCommand("insert into studtable values(@srno,@sname,@gender,@standard,@pmailid,@pmno)", con);

cmd.Parameters .AddWithValue ("srno",TextBox1 .Text );

cmd.Parameters .AddWithValue ("sname",TextBox2 .Text );

cmd.Parameters .AddWithValue ("gender",DropDownList1 .SelectedItem .Text );

cmd.Parameters .AddWithValue ("standard",DropDownList2 .SelectedItem .Text );

cmd.Parameters .AddWithValue ("pmailid",TextBox3 .Text );

cmd.Parameters .AddWithValue ("pmno",TextBox4 .Text );

cmd.ExecuteNonQuery ();

cmd.Dispose ();

Label1 .Text ="Student Details Inserted.....";

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void LinkButton2\_Click(object sender, EventArgs e)

{

TextBox1.Text = "";

TextBox2.Text = "";

DropDownList1.SelectedIndex = 0;

DropDownList2.SelectedIndex = 0;

TextBox3.Text = "";

TextBox4.Text = "";

TextBox1.Focus();

}

}

Staff Details

using System.Data.SqlClient;

using System.Configuration;

public partial class StaffForm : System.Web.UI.Page

{

SqlConnection con;

SqlCommand cmd;

SqlDataReader rs;

protected void Page\_Load(object sender, EventArgs e)

{

try

{

Label1.Text = "";

con = new SqlConnection(ConfigurationManager.ConnectionStrings["connection"].ConnectionString);

con.Open();

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void LinkButton1\_Click(object sender, EventArgs e)

{

try

{

cmd = new SqlCommand("select staffid from stafftable where staffid=@staffid", con);

cmd.Parameters.AddWithValue("staffid", TextBox1.Text);

rs = cmd.ExecuteReader();

bool b = rs.Read();

rs.Close();

cmd.Dispose();

if (b)

{

Label1.Text = "Staff Id Already Found.....";

return;

}

cmd = new SqlCommand("insert into stafftable values(@staffid,@staffname,@gender,@smailid,@smno)", con);

cmd.Parameters.AddWithValue("staffid", TextBox1.Text);

cmd.Parameters.AddWithValue("staffname", TextBox2.Text);

cmd.Parameters.AddWithValue("gender", DropDownList1.SelectedItem.Text);

cmd.Parameters.AddWithValue("smailid", TextBox3.Text);

cmd.Parameters.AddWithValue("smno", TextBox4.Text);

cmd.ExecuteNonQuery();

cmd.Dispose();

Label1.Text = "Staff Details Inserted.....";

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void LinkButton2\_Click(object sender, EventArgs e)

{

TextBox1.Text = "";

TextBox2.Text = "";

DropDownList1.SelectedIndex = 0;

TextBox3.Text = "";

TextBox4.Text = "";

TextBox1.Focus();

}

}

‘

Class Teacher Assigning Details

using System.Data.SqlClient;

using System.Data;

using System.Configuration;

public partial class SetClassTeacher : System.Web.UI.Page

{

SqlConnection con;

SqlCommand cmd;

SqlDataReader rs;

SqlDataAdapter adp;

DataTable dt;

void bindgrid()

{

adp = new SqlDataAdapter("select \* from stafftable where staffid not in (select staffid from scttable)", con);

dt = new DataTable();

adp.Fill(dt);

GridView1.DataSource = dt;

GridView1.DataBind();

}

protected void Page\_Load(object sender, EventArgs e)

{

try

{

Label1.Text = "";

con = new SqlConnection(ConfigurationManager.ConnectionStrings["connection"].ConnectionString);

con.Open();

if (!IsPostBack)

bindgrid();

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void GridView1\_RowCommand(object sender, GridViewCommandEventArgs e)

{

try

{

if (e.CommandName == "ss")

{

TextBox1.Text =GridView1 .Rows [int.Parse (e.CommandArgument .ToString ())].Cells [0].Text ;

TextBox2.Text =GridView1 .Rows [int.Parse (e.CommandArgument .ToString ())].Cells [1].Text ;

}

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void LinkButton1\_Click(object sender, EventArgs e)

{

try

{

string staffid = "", standard = "";

cmd = new SqlCommand("Select \* from scttable where staffid=@staffid", con);

cmd.Parameters.AddWithValue("staffid", TextBox1.Text);

rs = cmd.ExecuteReader();

if (rs.Read())

{

standard = rs["standard"].ToString();

rs.Close();

cmd.Dispose();

}

else

{

rs.Close();

cmd.Dispose();

standard = "no";

}

if (standard.Equals("no") == false)

{

Label1.Text = "StaffId :" + TextBox1.Text + "<Br>Already " + standard + " Class Teacher.<br>So Record Not Inserted......";

return;

}

cmd = new SqlCommand("Select \* from scttable where standard=@standard", con);

cmd.Parameters.AddWithValue("standard", DropDownList1.SelectedItem .Text );

rs = cmd.ExecuteReader();

if (rs.Read())

{

staffid = rs["staffid"].ToString();

rs.Close();

cmd.Dispose();

}

else

{

rs.Close();

cmd.Dispose();

staffid = "no";

}

if (staffid.Equals("no") == false)

{

Label1.Text = "Standard :" + DropDownList1.SelectedItem.Text + "<Br>Class Teacher Id :" + staffid + ". <br>So,Record Not Inserted....";

return;

}

cmd = new SqlCommand("insert into scttable values(@staffid,@staffname,@standard)", con);

cmd.Parameters.AddWithValue("staffid", TextBox1.Text);

cmd.Parameters.AddWithValue("staffname", TextBox2.Text);

cmd.Parameters.AddWithValue("standard", DropDownList1.SelectedItem.Text);

cmd.ExecuteNonQuery();

cmd.Dispose();

bindgrid();

Label1.Text = "Class Treacher Details Successfully Inserted......";

TextBox1.Text = "";

TextBox2.Text = "";

DropDownList1.SelectedIndex = 0;

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

}

Student Gate pass Details

using System.Data.SqlClient;

using System.Configuration;

using System.Data;

public partial class StudentGatePassDetails : System.Web.UI.Page

{

SqlConnection con;

SqlCommand cmd;

SqlDataAdapter adp;

DataTable dt;

SqlDataReader rs;

protected void Page\_Load(object sender, EventArgs e)

{

try

{

Label1.Text = "";

con = new SqlConnection(ConfigurationManager.ConnectionStrings["connection"].ConnectionString);

con.Open();

LinkButton2.Visible = false;

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void LinkButton1\_Click(object sender, EventArgs e)

{

try

{

cmd = new SqlCommand("select \* from gptable where srno=@srno and gpdate=@gpdate", con);

cmd.Parameters.AddWithValue("srno", TextBox1.Text);

cmd.Parameters.AddWithValue("gpdate", DateTime.Now.ToString("dd-MMM-yyyy"));

rs = cmd.ExecuteReader();

bool b = rs.Read();

rs.Close();

cmd.Dispose();

if (b)

{

Label1.Text = "Registration Number :" + TextBox1.Text + "<Br>Date :" + DateTime.Now.ToString("dd-MMM-yyyy") + "<br>Already Gate Pass Details Inserted....";

return;

}

DetailsView1.Visible = false;

adp=new SqlDataAdapter ("Select \* from studtable where srno=@srno", con);

adp.SelectCommand.Parameters.AddWithValue("srno", TextBox1.Text);

dt = new DataTable();

adp.Fill(dt);

if (dt.Rows.Count == 0)

{

Label1.Text = "Invalid Registration Number.....";

return;

}

DetailsView1.Visible = true;

DetailsView1.DataSource = dt;

DetailsView1.DataBind();

if (dt.Rows.Count != 0)

{

LinkButton2.Visible = true;

}

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void LinkButton2\_Click(object sender, EventArgs e)

{

try

{

Response.Redirect("StudentGatePassDetails1.aspx?SRNo=" + TextBox1.Text);

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

}

using System.Data.SqlClient;

using System.Configuration;

using System.Net.Mail;

public partial class StudentGatePassDetails1 : System.Web.UI.Page

{

SqlConnection con;

SqlCommand cmd;

SqlDataReader rs;

protected void Page\_Load(object sender, EventArgs e)

{

try

{

Label1.Text = "";

con = new SqlConnection(ConfigurationManager.ConnectionStrings["connection"].ConnectionString);

con.Open();

if (!IsPostBack)

{

if (Request.QueryString.Get("SRNO") != null)

{

TextBox1.Text = Request.QueryString.Get("SRNO");

cmd = new SqlCommand("Select \* from studtable where srno=@srno", con);

cmd.Parameters.AddWithValue("srno", TextBox1.Text);

rs = cmd.ExecuteReader();

string standard = "";

if (rs.Read())

{

TextBox2.Text = rs["sname"].ToString();

TextBox3.Text = rs["pmailid"].ToString();

standard = rs["standard"].ToString();

rs.Close();

cmd.Dispose();

}

else

{

rs.Close();

cmd.Dispose();

Label1.Text = "Record Not Found.Check StudTable.....";

return;

}

cmd = new SqlCommand("select smailid from stafftable where staffid=(select staffid from scttable where standard=@standard)", con);

cmd.Parameters.AddWithValue("standard", standard);

rs = cmd.ExecuteReader();

if (rs.Read())

{

TextBox4.Text = rs["smailid"].ToString();

rs.Close();

cmd.Dispose();

}

else

{

rs.Close();

cmd.Dispose();

Label1.Text = "Record Not Found.Check StaffTable.....";

return;

}

TextBox5.Text = DateTime.Now.ToString("dd-MMM-yyyy");

TextBox6.Text = DateTime.Now.ToString("hh:mm:ss tt");

}

}

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

void ParentMailCoding(string emailid, string mess, string semailid, string pword)

{

MailMessage m = new MailMessage();

m.From = new MailAddress(semailid);

m.To.Add(emailid);

m.Subject = "Gatepass Details:";

m.IsBodyHtml = true;

string sub = "From :" + m.From +"<br>Registration Number :" + TextBox1 .Text +"<br>Student Name :"+TextBox2 .Text + "<br>Date :" + TextBox5 .Text +"<Br>Time :"+ TextBox6 .Text + "<br>Purpose :" + mess ;

m.Body = sub;

SmtpClient s = new SmtpClient();

s.Host = "smtp.gmail.com";

s.Credentials = new System.Net.NetworkCredential(semailid, pword);

s.EnableSsl = true;

s.Send(m);

}

void StaffMailCoding(string emailid, string mess, string semailid, string pword)

{

MailMessage m = new MailMessage();

m.From = new MailAddress(semailid);

m.To.Add(emailid);

m.Subject = "Gatepass Details:";

m.IsBodyHtml = true;

string sub = "From :" + m.From + "<br>Registration Number :" + TextBox1.Text + "<br>Student Name :" + TextBox2.Text + "<br>Date :" + TextBox5.Text + "<Br>Time :" + TextBox6.Text + "<br>Purpose :" + mess;

m.Body = sub;

SmtpClient s = new SmtpClient();

s.Host = "smtp.gmail.com";

s.Credentials = new System.Net.NetworkCredential(semailid, pword);

s.EnableSsl = true;

s.Send(m);

}

protected void LinkButton1\_Click(object sender, EventArgs e)

{

try

{

cmd = new SqlCommand("select \* from gptable where srno=@srno and gpdate=@gpdate", con);

cmd.Parameters.AddWithValue("srno", TextBox1.Text);

cmd.Parameters.AddWithValue("gpdate", TextBox5.Text);

rs = cmd.ExecuteReader();

bool b = rs.Read();

rs.Close();

cmd.Dispose();

if (b)

{

Label1.Text = "Registration Number :" + TextBox1.Text + "<br>Date :" + TextBox5.Text + "<br> Already Inserted the Gate Pass Details.....";

return;

}

cmd = new SqlCommand("insert into gptable values(@srno,@sname,@gpdate,@gptime,@gppurpose)", con);

cmd.Parameters .AddWithValue ("srno",TextBox1 .Text );

cmd.Parameters .AddWithValue ("sname",TextBox2 .Text );

cmd.Parameters .AddWithValue ("gpdate",TextBox5 .Text );

cmd.Parameters .AddWithValue ("gptime",TextBox6 .Text );

cmd.Parameters .AddWithValue ("gppurpose",TextBox7 .Text );

cmd.ExecuteNonQuery ();

cmd.Dispose ();

Label1 .Text ="Gate Pass Details Successfully Inserted......";

ParentMailCoding (TextBox3 .Text , TextBox7 .Text , "customerproject404nf@gmail.com", "ssiaptech");

ParentMailCoding(TextBox4.Text, TextBox7.Text, "customerproject404nf@gmail.com", "ssiaptech");

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

}

View Date wise Gate Pass Details

using System.Data.SqlClient;

using System.Configuration;

using System.Data;

public partial class ViewDateWiseGatePassDetails : System.Web.UI.Page

{

SqlConnection con;

SqlDataAdapter adp;

DataTable dt;

protected void Page\_Load(object sender, EventArgs e)

{

try

{

Label1.Text = "";

con = new SqlConnection(ConfigurationManager.ConnectionStrings["connection"].ConnectionString);

con.Open();

GridView1.Visible = false;

GridView2.Visible = false;

}

catch (Exception ex)

{

Label1.Text = ex.Message;

}

}

protected void Calendar1\_SelectionChanged(object sender, EventArgs e)

{

try

{

DateTime dt1 = Calendar1.SelectedDate;

GridView1.Visible = true;

GridView2.Visible = true;

adp = new SqlDataAdapter("Select \* from gptable1 where gpdate=@gpdate", con);

adp.SelectCommand.Parameters.AddWithValue("gpdate", dt1.ToString("dd-MMM-yyyy"));

dt = new DataTable();

adp.Fill(dt);

GridView1.DataSource = dt;

GridView1.DataBind();

adp = new SqlDataAdapter("Select \* from gptable where gpdate=@gpdate", con);

adp.SelectCommand.Parameters.AddWithValue("gpdate", dt1.ToString("dd-MMM-yyyy"));

dt = new DataTable();

adp.Fill(dt);

GridView2.DataSource = dt;

GridView2.DataBind();

}

catch (Exception ex)

{

Label1.Text = ex.Message;

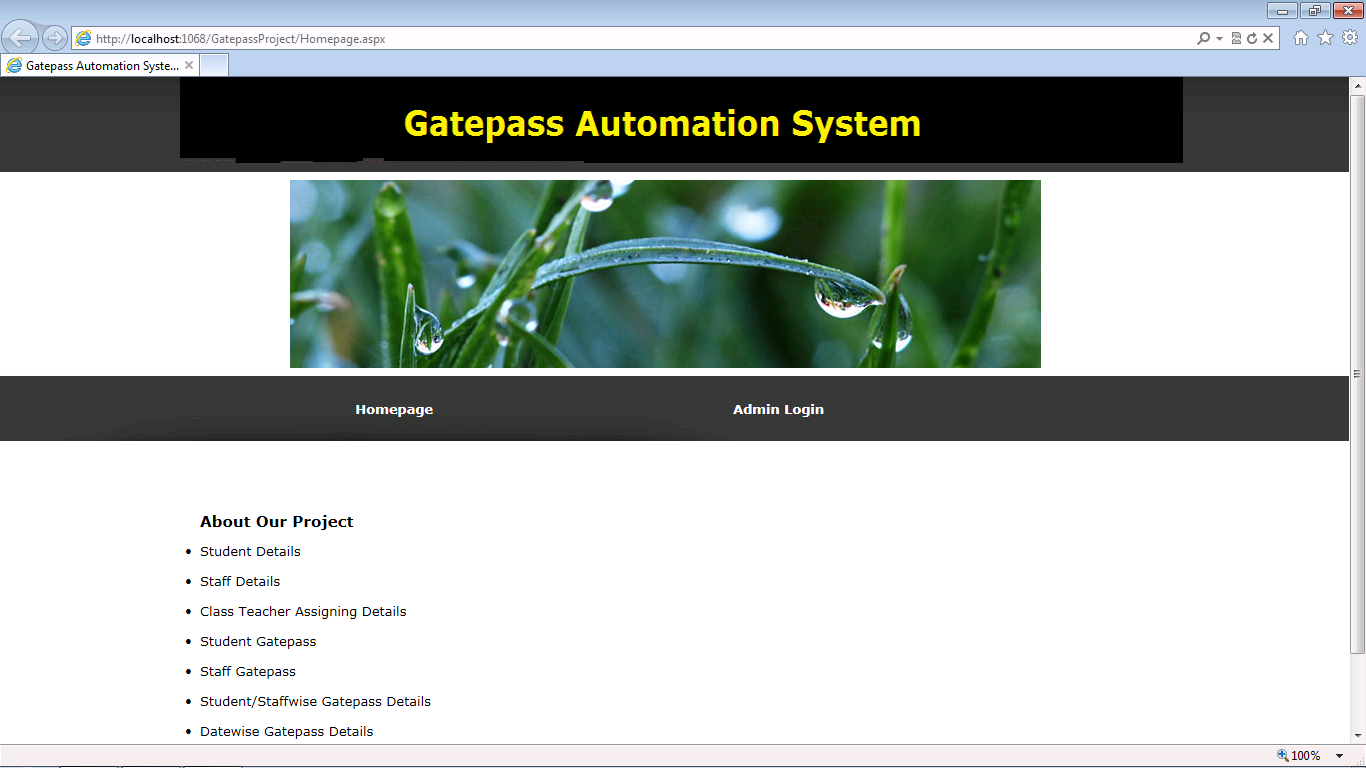
}

}

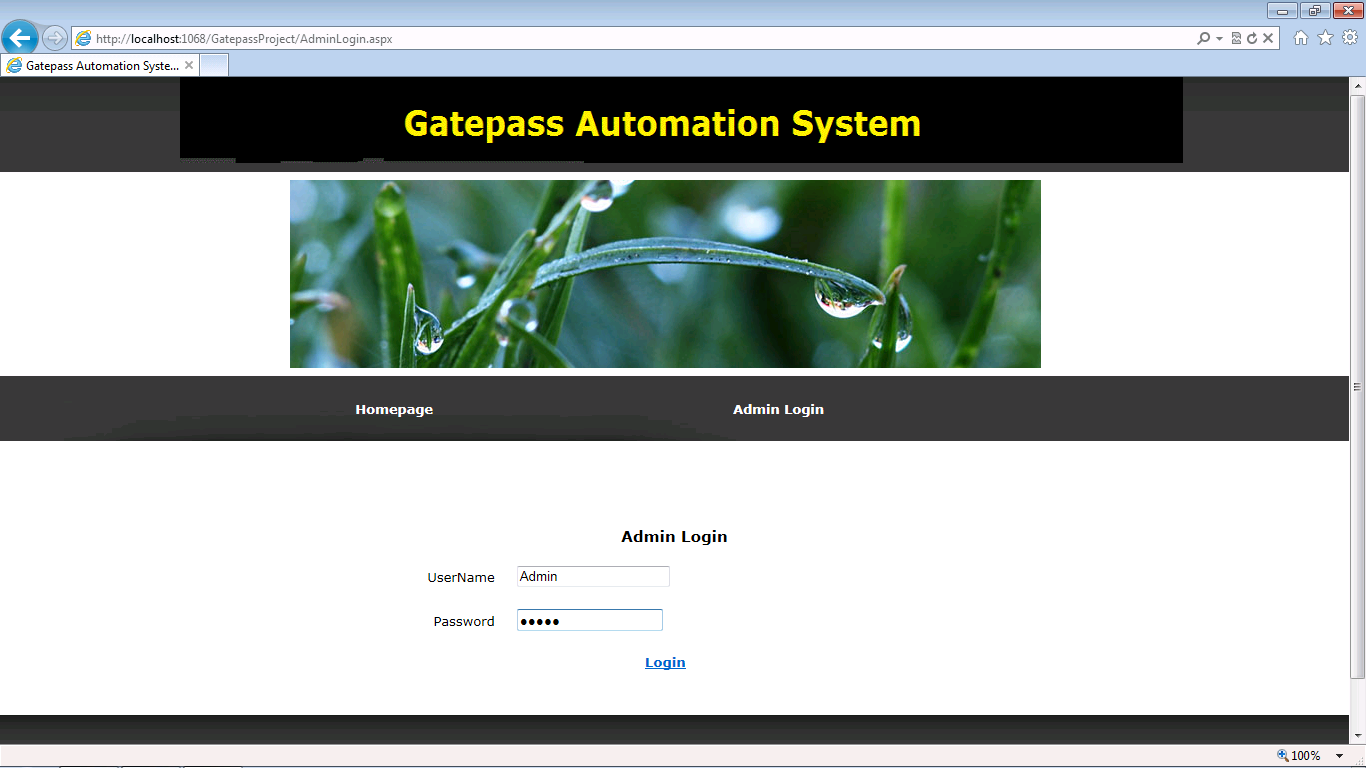
}

5.3 Sample Output

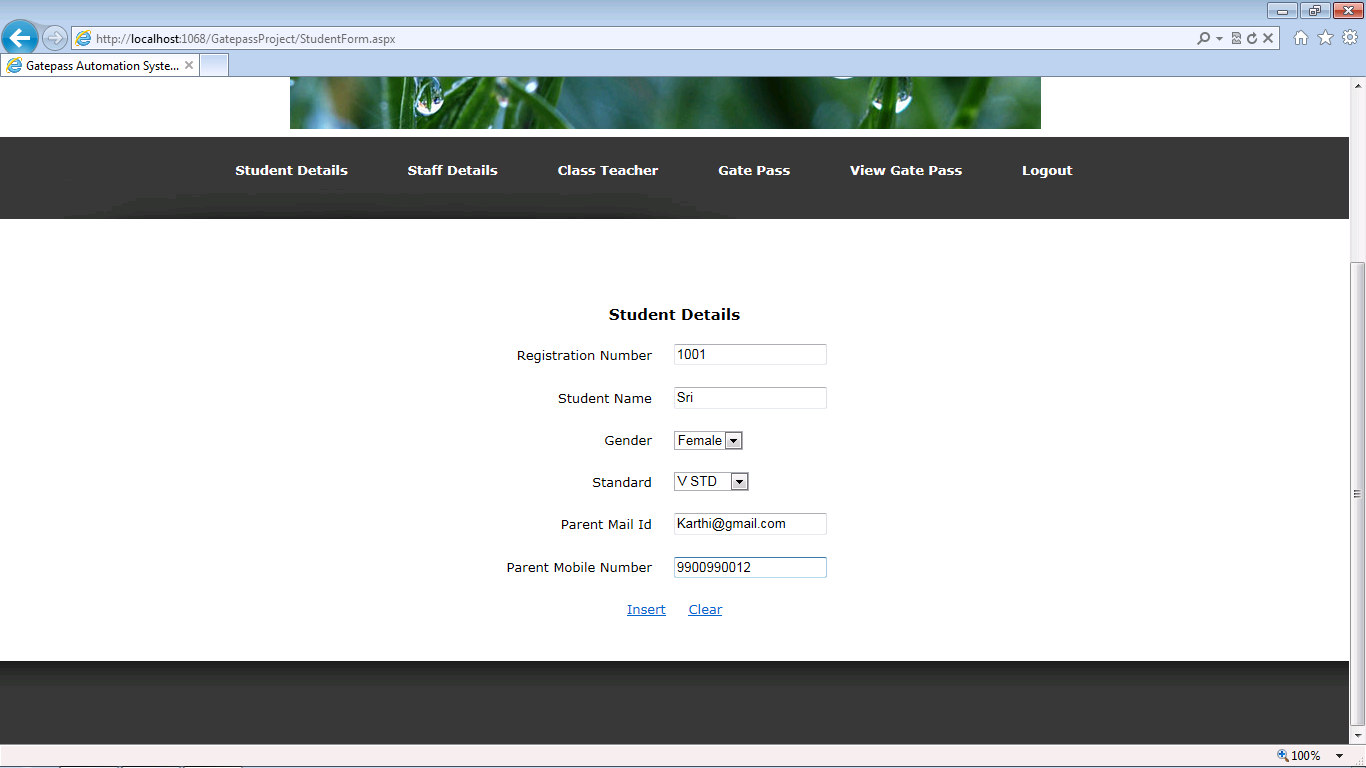
Homepage



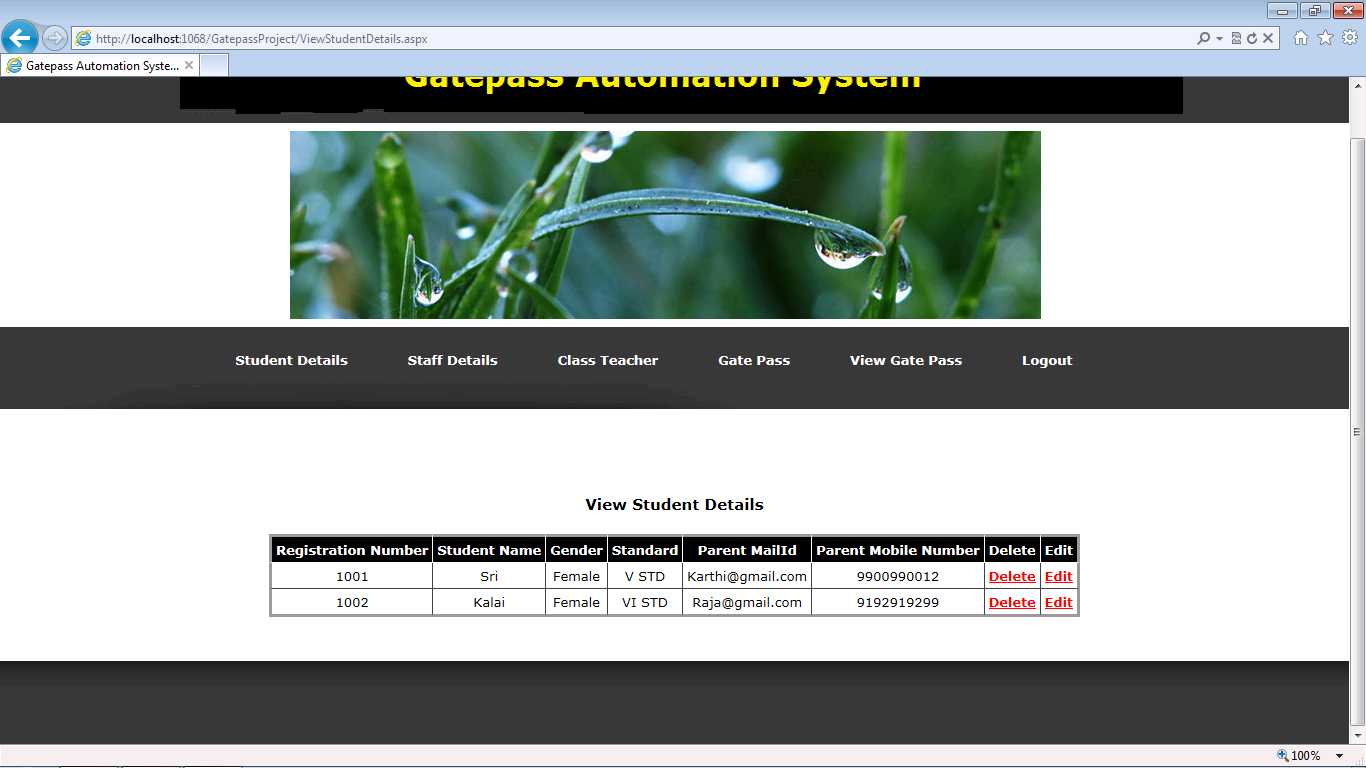
Admin Login



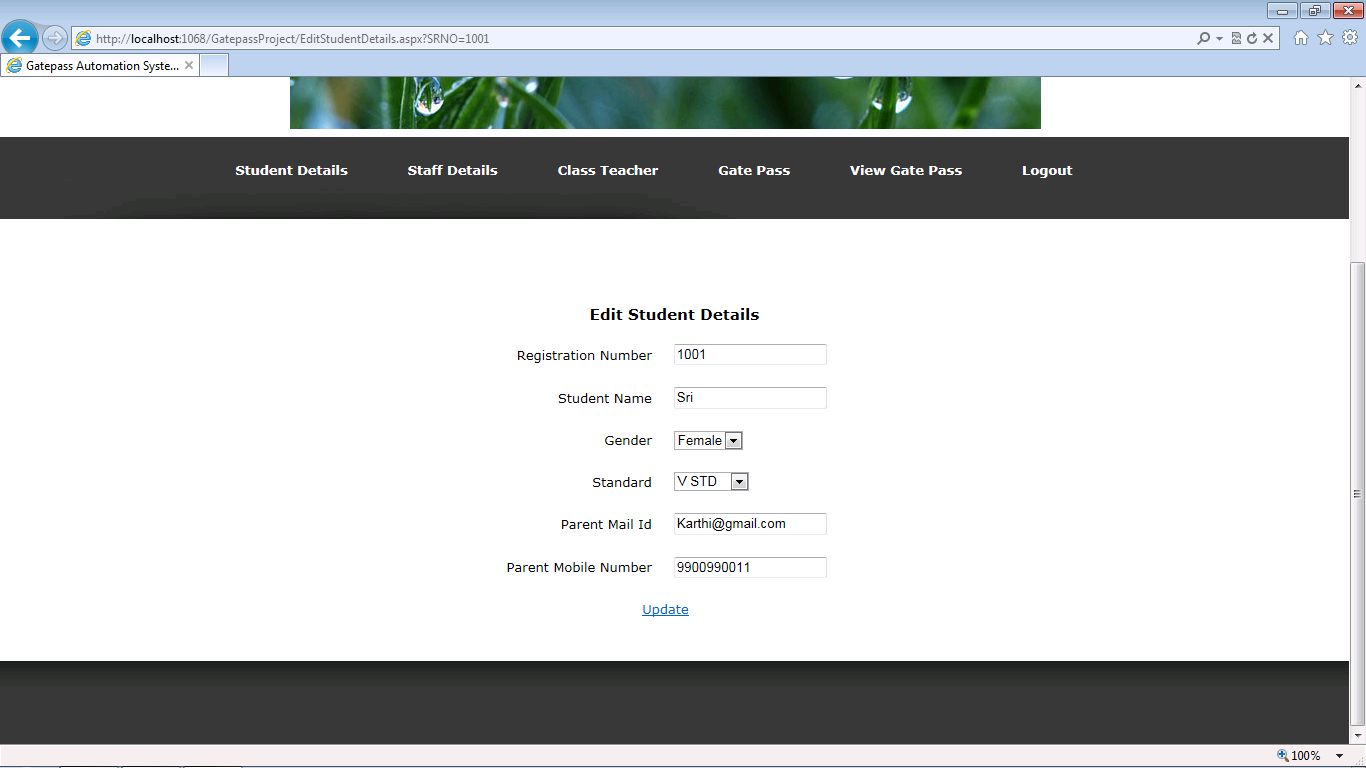
Insert Student Details



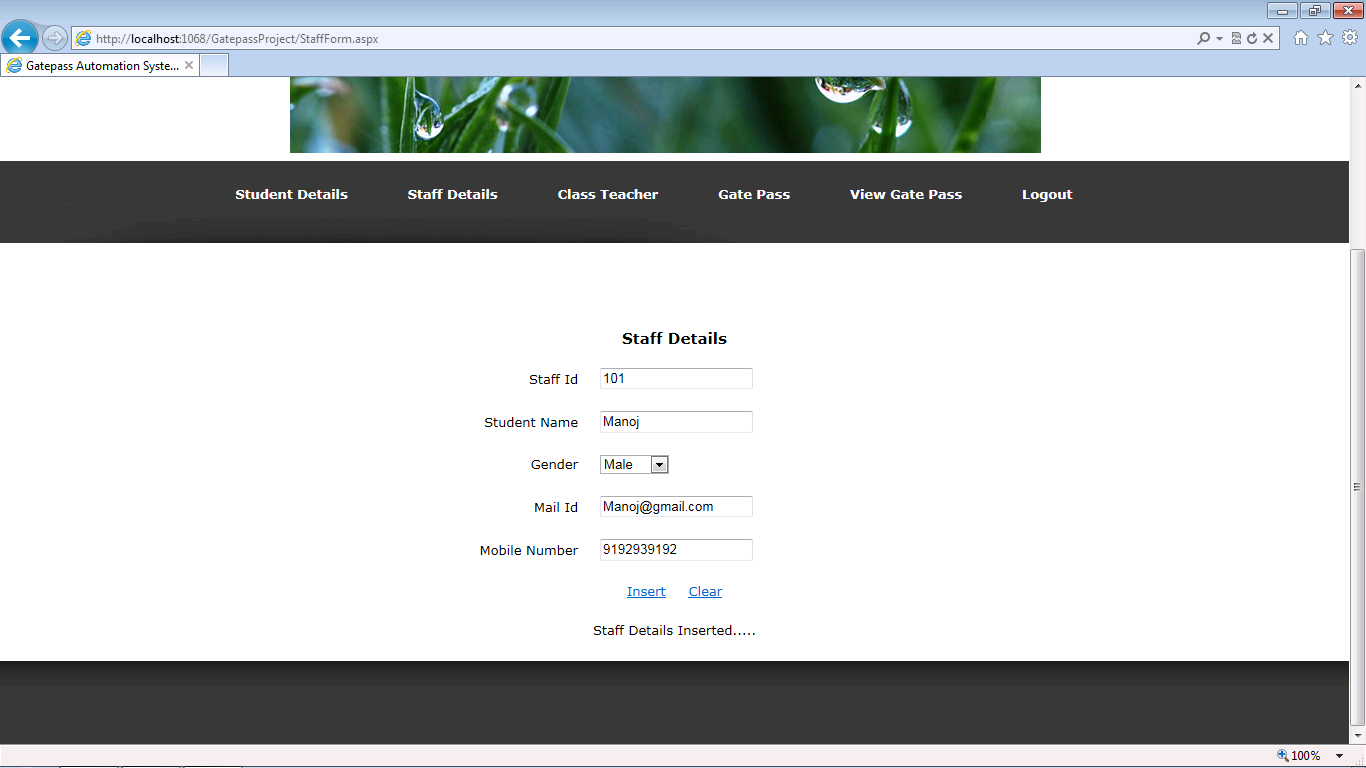
View Student Details



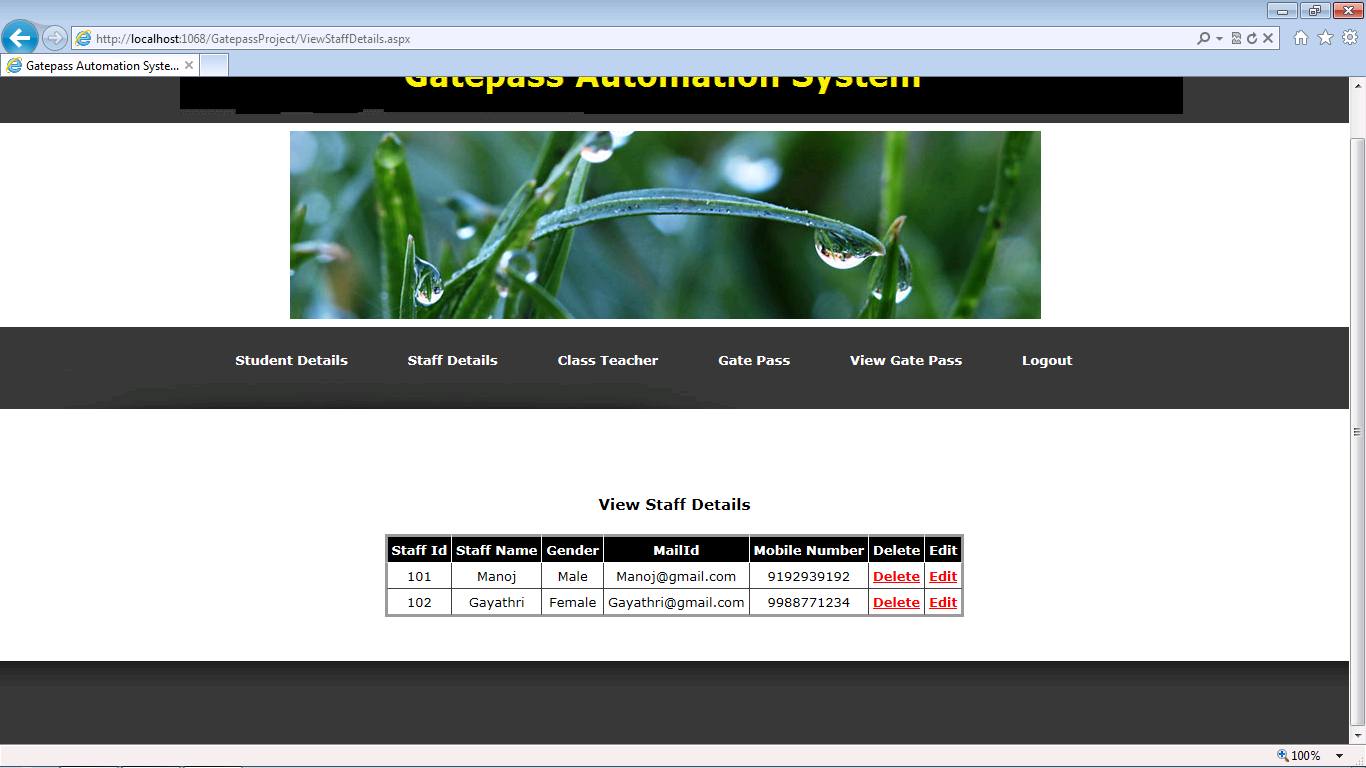
Edit Student Details



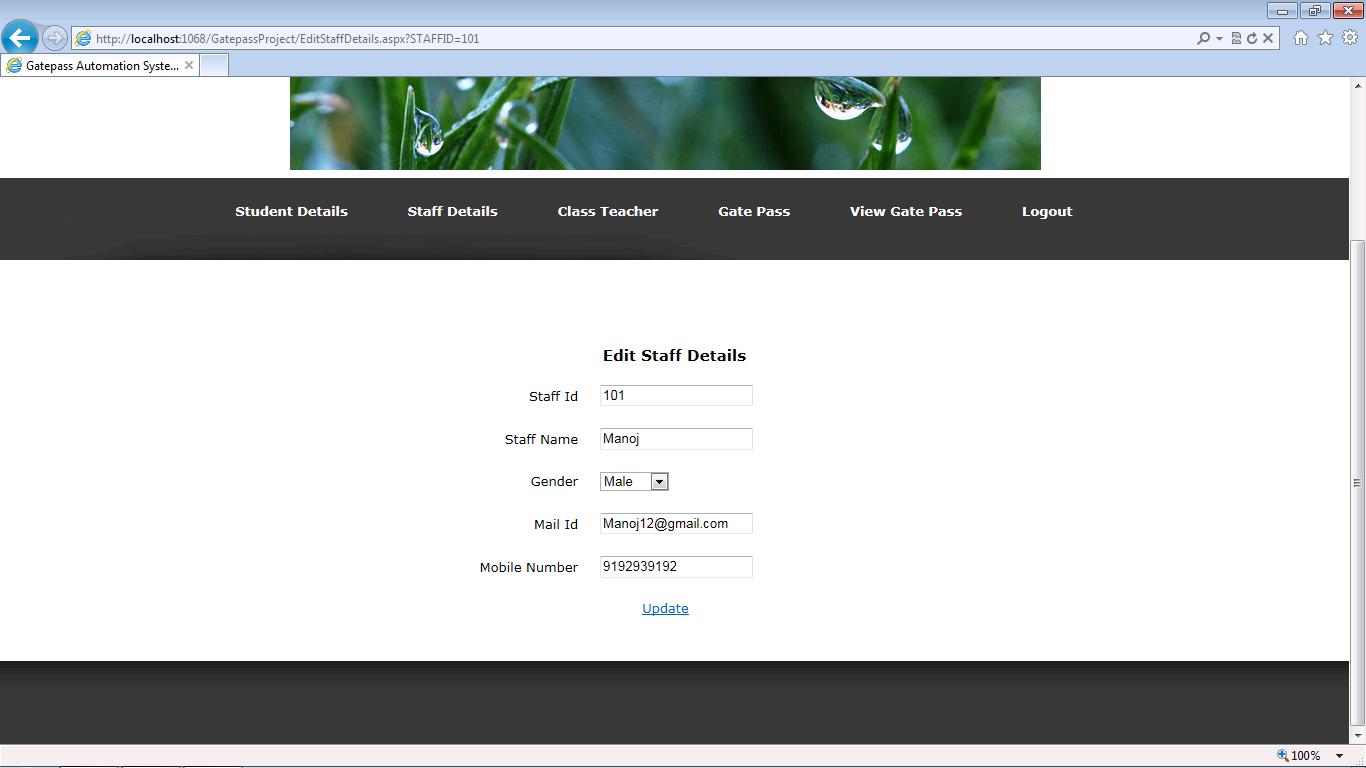
Insert Staff Details



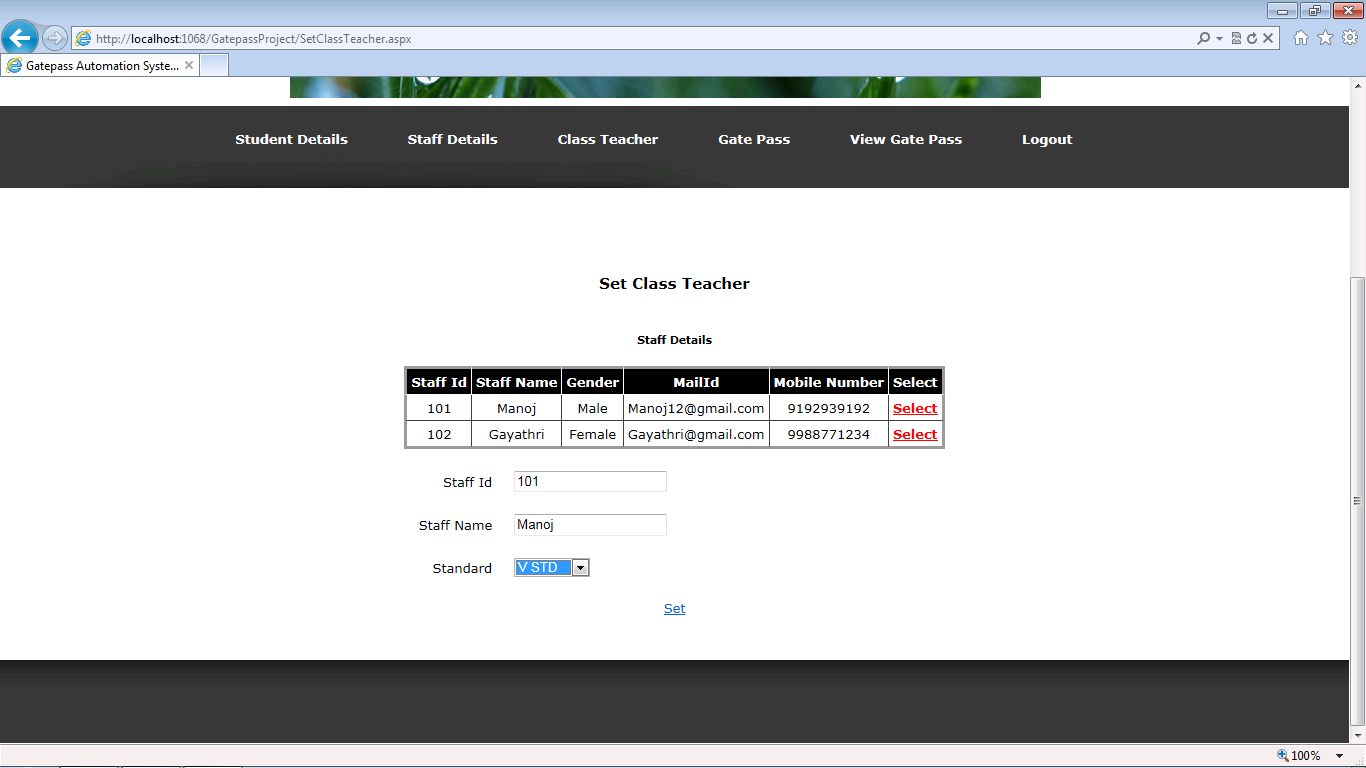
View Staff Details



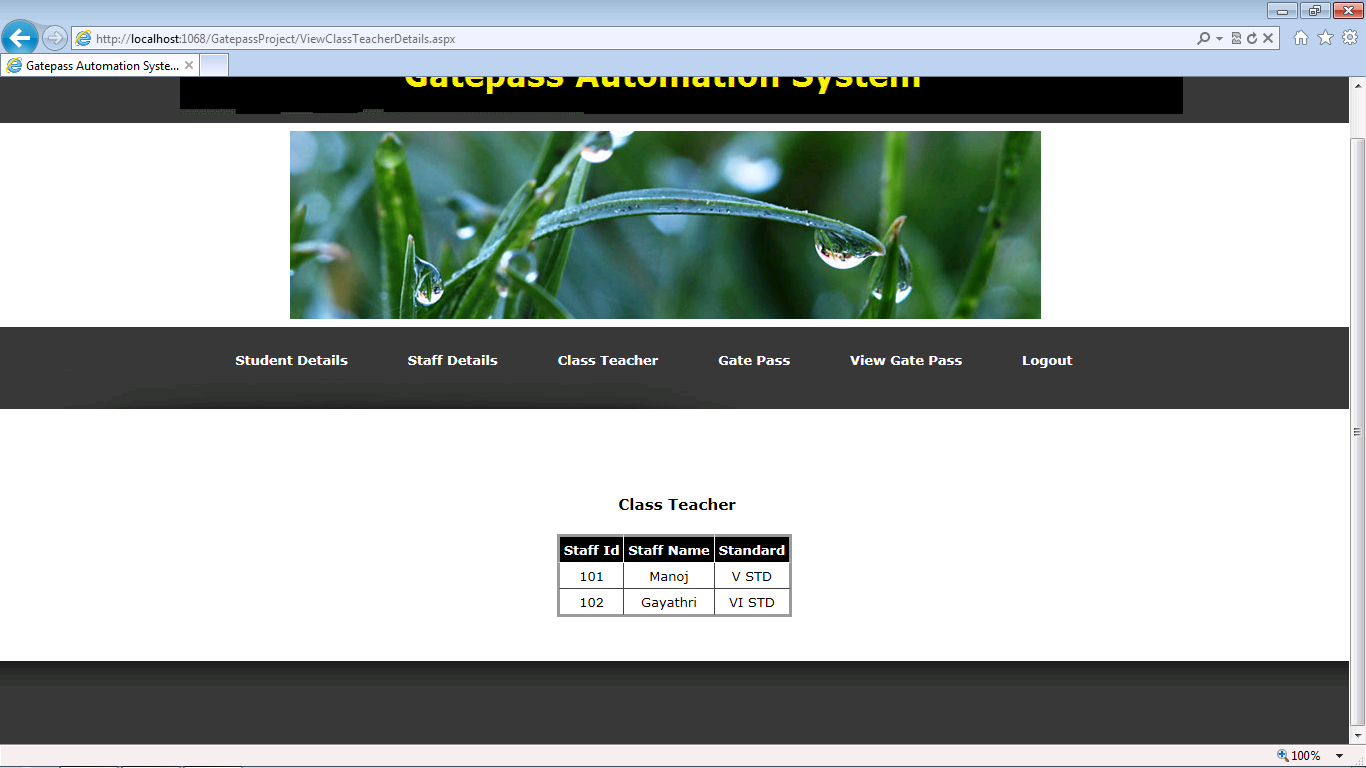
Edit Staff Details



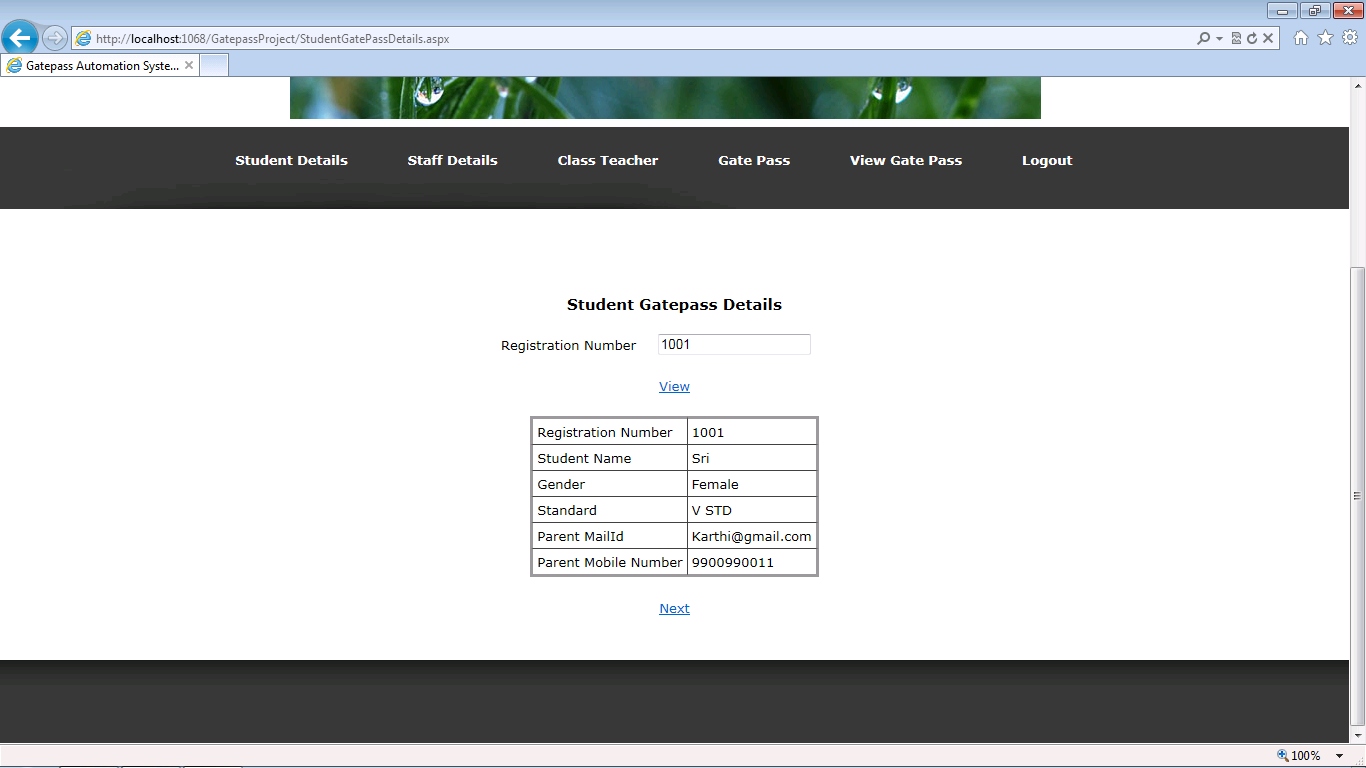
Assign Class Teacher



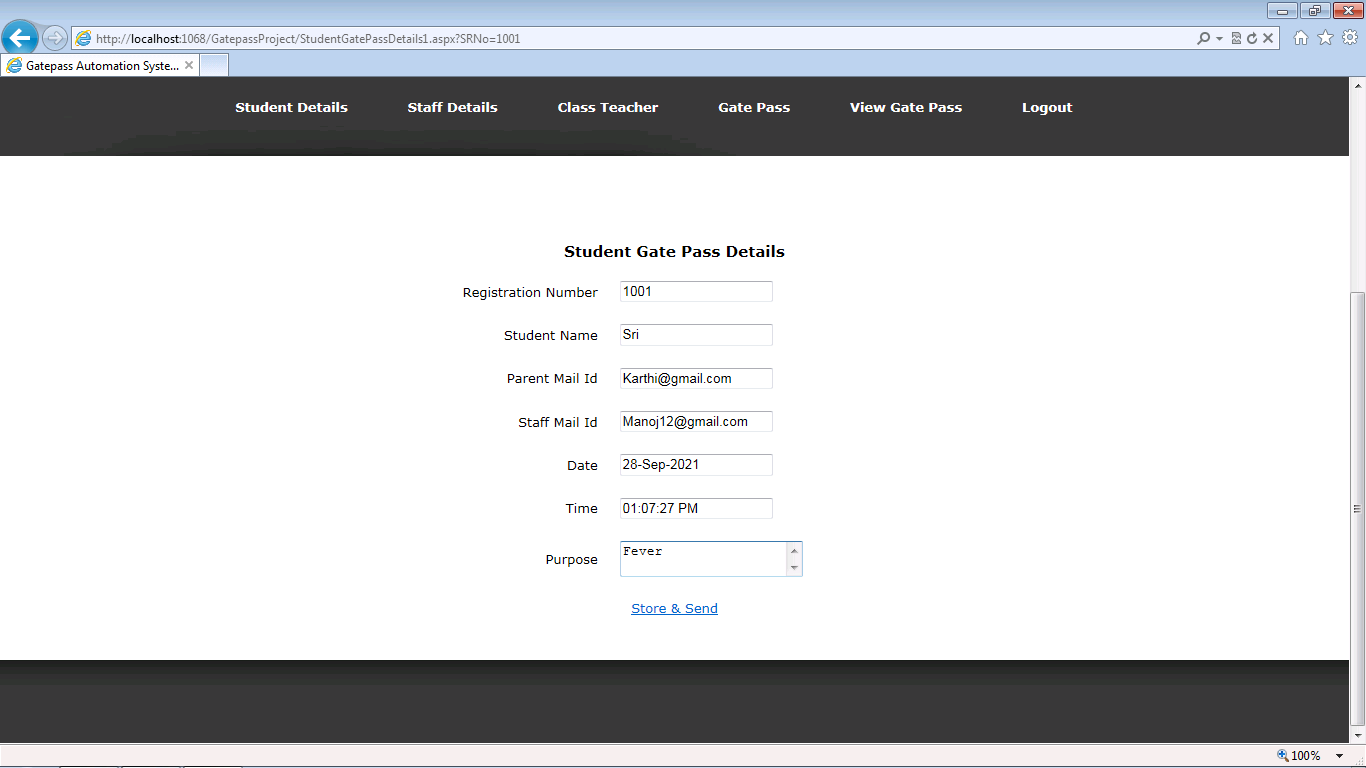
View Class Teacher Details



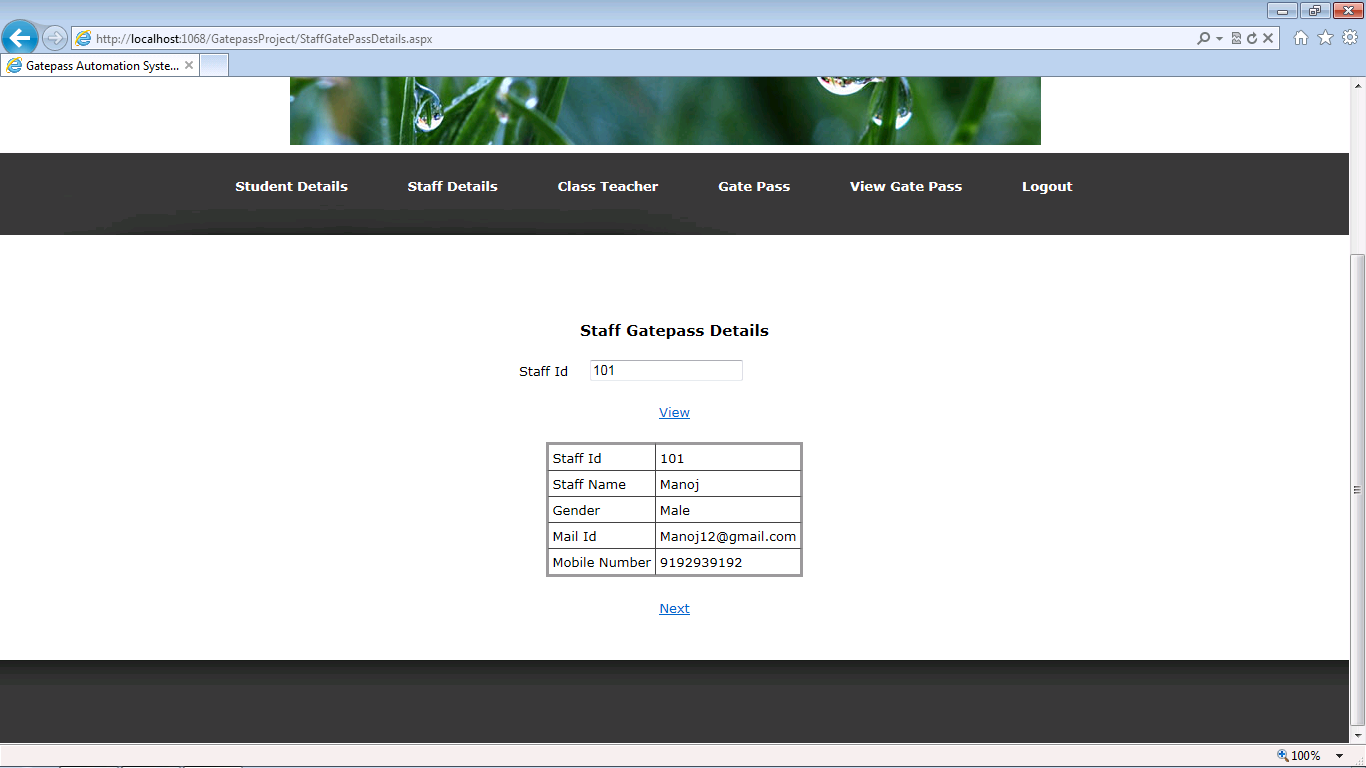
Student Gate Pass Details



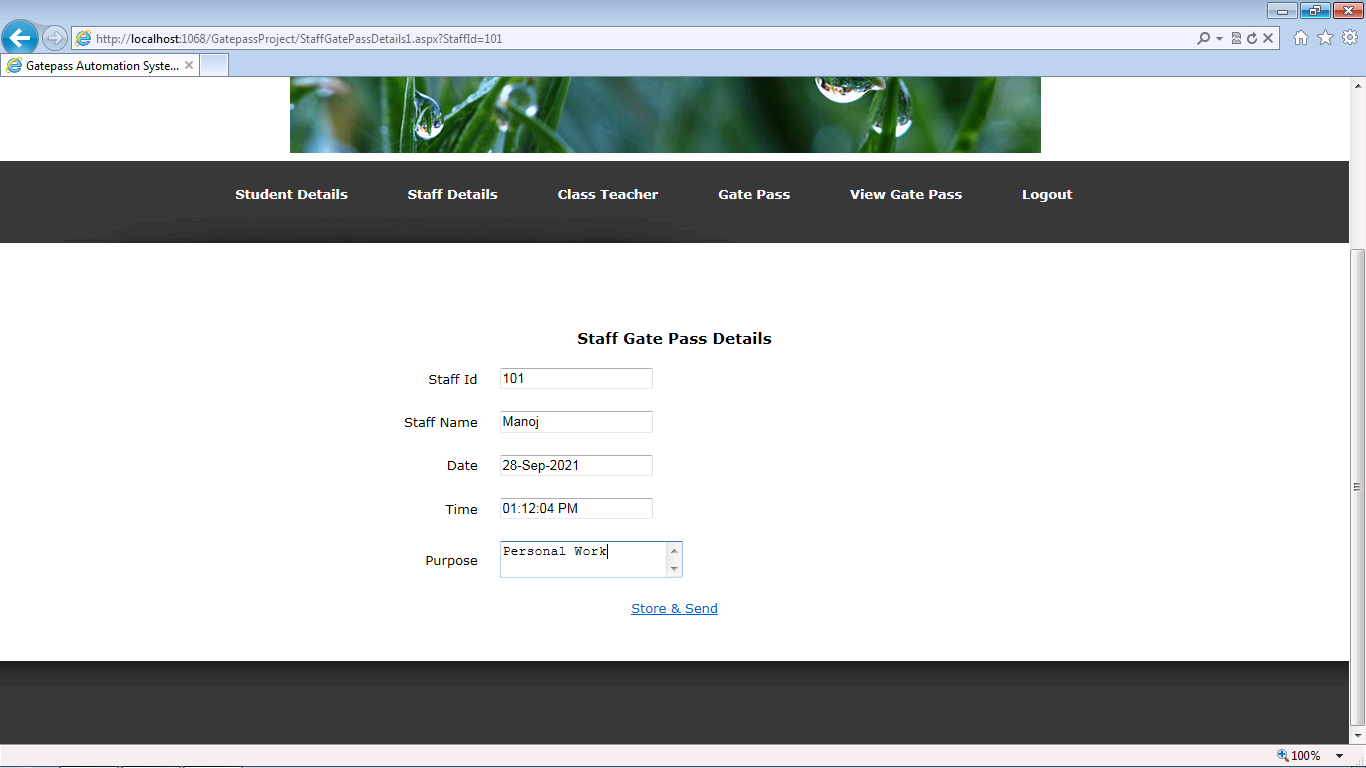
Student Gate Pass Details



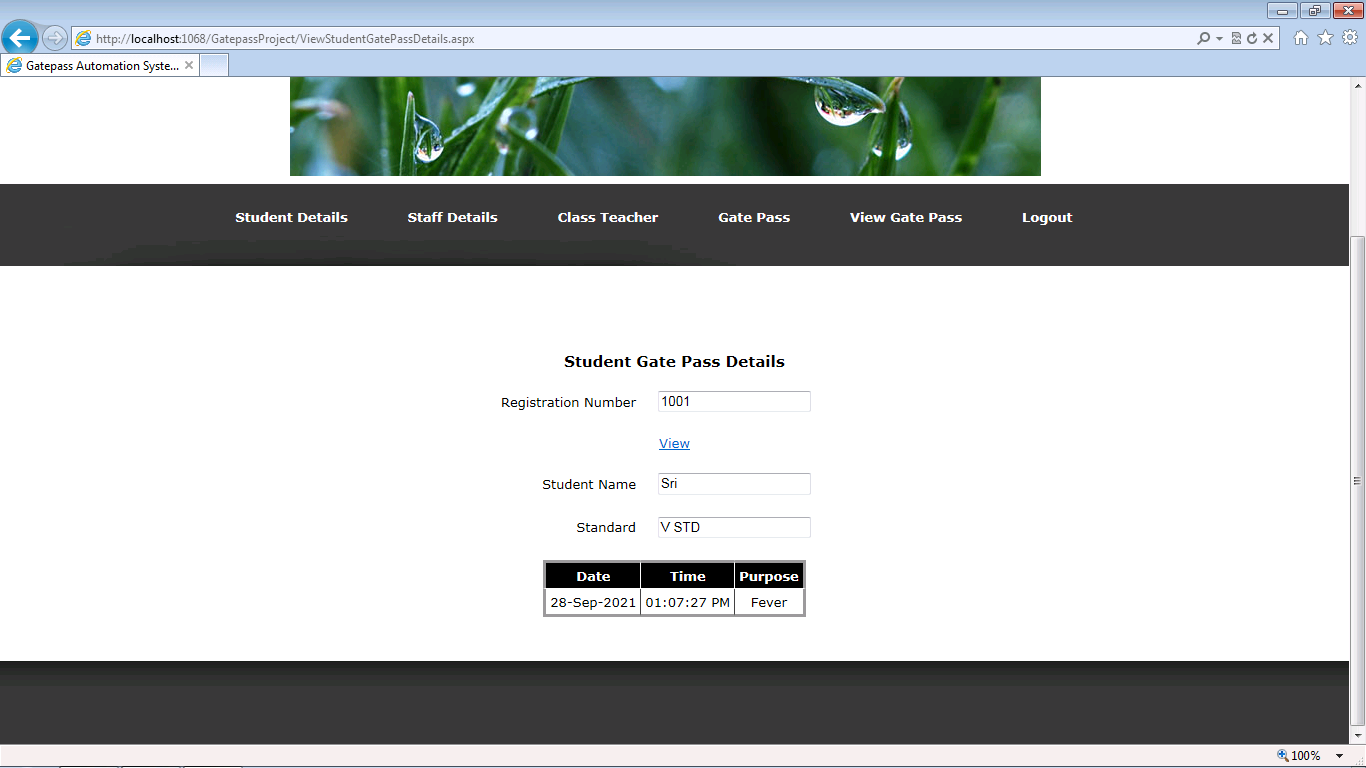
Staff Gate pass Details



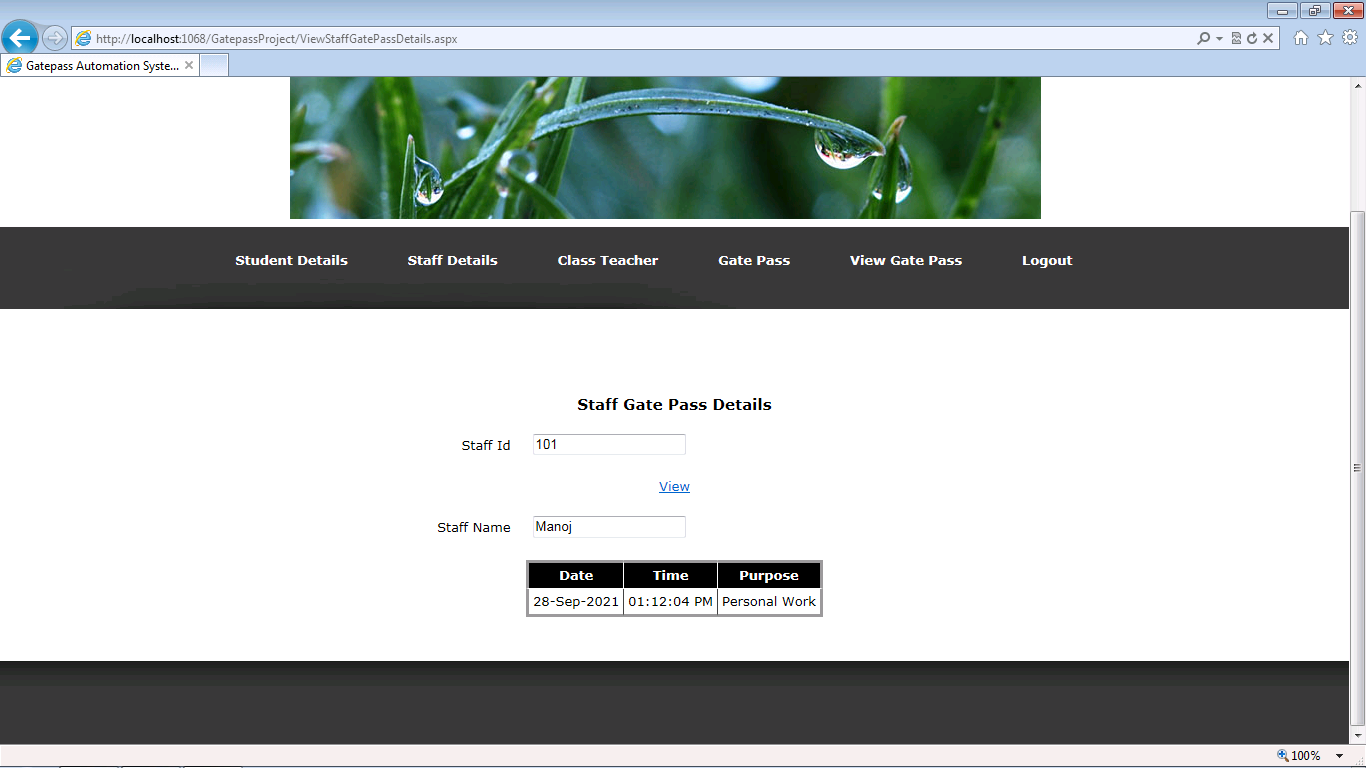
Staff Gate Pass Details



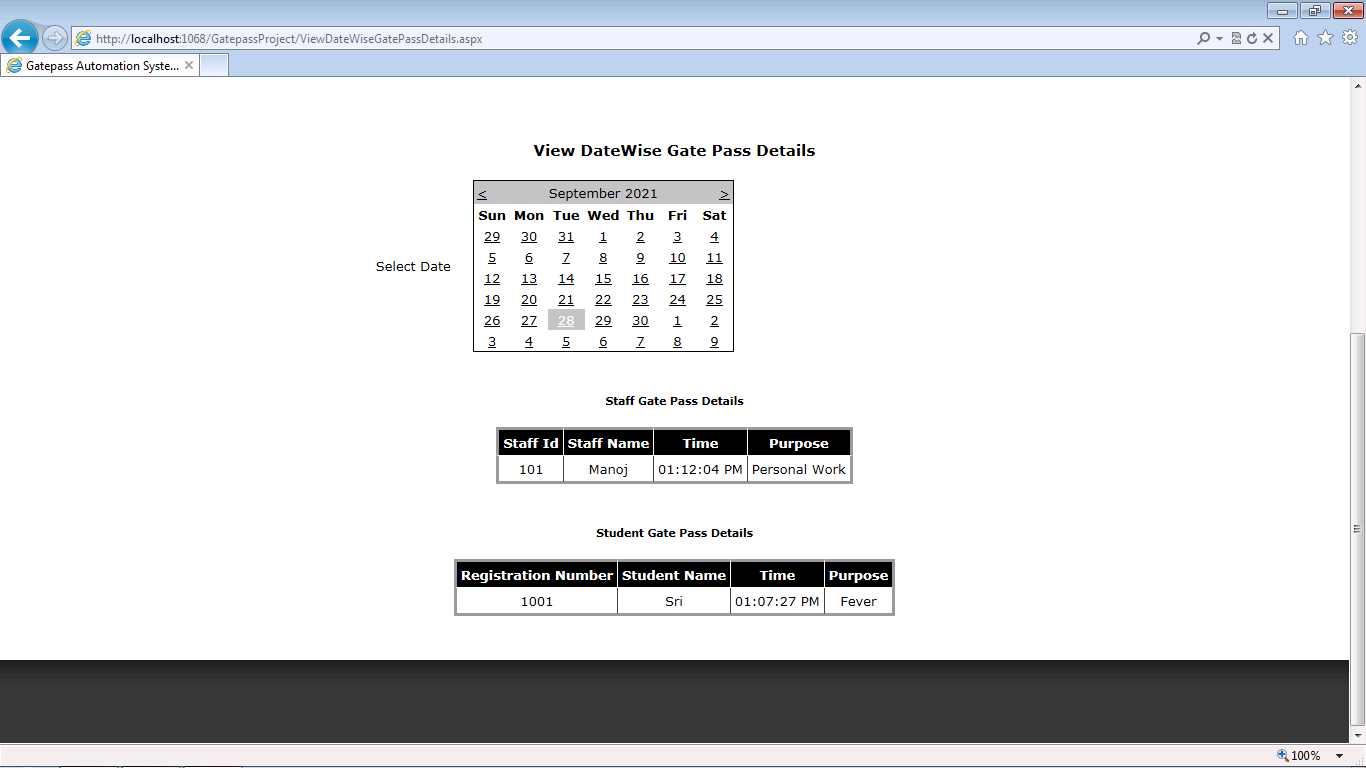
Student Gate Pass Details



Staff Gate Pass Details



Date wise Gate Pass Details



6. SYSTEM TESTING

System Testing is an important stage in any system development life cycle. Testing is a process of executing a program with the intention of finding errors. The importance of software testing and its implications with respect to software quality cannot be overemphasized. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. A good test case is one that has a high probability of finding a yet undiscovered error.

Testing is the set of activities that can be planned in advance and conducted systematically. Different test conditions should be thoroughly checked and the bugs detected should be fixed. The testing strategies formed by the user are performed to prove that the software is free and clear from errors. To do this, there are many ways of testing the system’s reliability, completeness and maintainability.

Unit Testing

In the unit testing the analyst tests the program making up a system. The software units in a system are the modules and routines that are assembled and integrated to perform a specific function. In a large system, many modules on different levels are needed.

Unit testing can be performed from the bottom up starting with the smallest and lowest level modules and proceeding one at a time. For each module in a bottom-up testing, a short program executes the module and provides the needed data.

# **Integration Testing**

Integration testing is a systematic technique for constructing the program structure while conducting test to uncover errors associate with interfacing. Objectives are used to take unit test modules and built program structure that has been directed by design.

The integration testing is performed for this Project when all the modules where to make it a complete system. After integration the project works successfully.

# **Validation Testing**

Validation testing can be defined in many ways, but a simple definition is that can be reasonably expected by the customer. After validation test has been conducted, one of two possible conditions exists.

* The functions or performance characteristics confirm to specification and are accepted.
* A deviation from specification is uncovered and a deficiency list is created.

Proposed system under consideration has been tested by using validation testing and found to be working satisfactorily.

For example, in this project validation testing is performed against user module. This module is tested with the following valid and invalid inputs for the field staff id.

##### **White Box Testing**

White box testing, sometimes called glass-box testing is a test case design method that uses the control structure of the procedural design to derive test cases. Using white box testing methods, the software engineer can derive test cases that

* Guarantee that all independent paths with in a module have been exercised at least once.
* Exercise all logical decisions on their true and false sides.
* Execute all loops at their boundaries and with in their operational bounds and
* Exercise internal data structure to assure their validity.

For example in this project white box testing is performed against patient module. Without entering text if we apply it displays the message “First add record then save it” else it should be saved.

##### **Black Box Testing**

This method treats the coded module as a black box. The module runs with inputs that are likely to cause errors. Then the output is checked to see if any error occurred. This method cannot be used to test all errors, because some errors may depend on the code or algorithm used to implement the module.

7. FUTURE ENHANCEMENT

The newly developed system, in its present form, is eminently suited to the existing needs. But in order to meet the future needs, which can become progressively more complex the efficiency of the system can be improved by making some simple modifications in the programs.

The Gatepass system is still developing in many ways. The site is being added new section and new services to the visitors. In future the site will be providing the service of modifying the website to the users.

The user is given a blank website so as to design the website or web page of his own, by activating links, changing images and changing the text using editor.

In future the project will be altered by which the user is given more options to design more number of pages. This will be achieved by providing more links ,more customizing options and we may offer the user possibility of creating domain of its won

8. CONCLUSION

In this report, Therefore the offline gate pass system using department effective tool which can be used to a great extent. The system is portable and can be easily installed and used in the department. Using this application proxy is completely avoided with a pure software approach. The activity will take up most of the student’s time. It will reduce the time, effort and resources such as paper for both the parties involved in the process. Also, it will eliminate the tedious work of the teachers of maintaining different gate pass papers The system will also avoid a number of proxies in a educational institute as the teacher will have constant eye contact with the student activity.

10. BIBLIOGRAPHY

Books

1. “Microsoft ASP.NET” by Mirdula Pariha, Jeff Webb, Tata Mac Graw Hill.
2. Microsoft SQL SERVER2000 by Ray Rankings, Paul Bertucci, Paul Jensen
3. Gerry O’Brien,” Microsoft IIS 5.0 Administration”
4. Sql Server - The complete Reference-Gayle Coffman
5. Teach yourself Sql Server-Richard Way Mire
6. Asp.Net-Stephen Walther (Second Edition)
7. Asp.Net-Daniel Maharry

WEBSITES

ASP.NET

1. www.asp.net/(S(pdfrohu0ajmwt445fanvj2r3))/learn/data-access/
2. www.w3schools.com/aspnet/default.asp
3. www.411asp.net/home/sites
4. www.cristiandarie.ro/asp-net-tutorial/
5. www.asp.net-tutorials.com/basics/first-website/

SQL SERVER

1. www.functionx.com/sqlserver/
2. www.technet.microsoft.com/en-us/library/ms169620.aspx
3. www.msdn.microsoft.com/en-us/library/ms169620(SQL.90).aspx
4. www.softwaretrainingtutorials.com/ms-sql-server-2005.php