College Management System

By:- Gopi D. Gajjar
Every work that completes successfully stands on the constant encouragement, goodwill and support of the people. Through this, I would like to express our gratitude to all those who have contributed to accomplish my project successfully. I would also like to thank Mr. Keyur zala and all other faculties of our department for being a constant source of inspiration right from seeking project to the successful completion of project and also guiding us throughout our project.

My obligation remains to all those people and friends who have directly or indirectly helped us in successful completion of our project. No amount of words written here will suffice for our sense of gratitude towards all of them.
Index

Acknowledgment .............................................................. 2
Project Profile .............................................................. 4
Introduction ........................................................................ 5
Overview of Tools ............................................................ 6
Data flow Diagram ............................................................ 21
Data Dictionary ................................................................. 23
Screen Layouts ................................................................. 28
Requirements ..................................................................... 38
Limitations & future Analysis ............................................ 39
**Project Profile**

Developed By: - Gopi D. Gajjar  
Project on: - College Management System  
Sem-year: - 6th sem-2012  
Guidance By: - Mr. Keyur Zala  
Front end: - ASP.NET 2.0  
Back end: - Microsoft SQL Server 2005  
Submitted to: - Shree M.N. Virani Science College
Good management helps organization to lead among competitor. Management made our work silky, made our work trouble-free and speedy. Today’s life is so busy and organizations have less time for maintaining their important work. So, to make life easy for organizations we make a small web service for College Management System. It gives facility to do personal & professional work online easily and speedy.

College Management System is a full-featured application designed specifically for planning, managing, and scheduling College’s important dates and times. It runs in the background, sits in the system tray, and makes it nearly impossible to forget record, entry, scheduling, meetings, and other important events.

Using this web based application organization can explain rationale, roles, responsibility for reviews, performance management cycle, managing weak performance, confidentiality. The main objective is user convenience College Management System.com provides easy user interface, simple processing, and easy to use application.
Overview of Tools

:: Front-End ::

ASP.NET is a unified Web development model that includes the services necessary for you to build enterprise-class Web applications with a minimum of coding. ASP.NET is part of the .NET Framework, and when coding ASP.NET applications you have access to classes in the .NET Framework. You can code your applications in any language compatible with the common language runtime (CLR), including Microsoft Visual Basic, C#, JScript.NET, and J#. These languages enable you to develop ASP.NET applications that benefit from the common language runtime, type safety, inheritance, and so on.

ASP.NET includes:

- A page and controls framework
- The ASP.NET compiler
- Security infrastructure
- State-management facilities
- Application configuration
- Health monitoring and performance features
- Debugging support
- An XML Web services framework
- Extensible hosting environment and application life cycle management
- An extensible designer environment
ASP.NET:

ASP.NET provides a unified Web development model that includes the services necessary for you to build enterprise-class Web applications. ASP.NET is part of the .NET Framework and enables you to take full advantage of the features of the common language runtime, such as type safety, inheritance, language interoperability, and versioning.

The topics in this section provide you with conceptual information about how ASP.NET works and procedural topics that show you how to write code that take advantage of the power of ASP.NET to create dynamic Web applications.

Why is ASP.Net 2.0 proffered?

The Microsoft .NET Framework version 2.0 includes significant enhancements to ASP.NET in virtually all areas. ASP.NET has been improved to provide out-of-the-box support for the most common Web application situations. You will find that you can get Web sites and pages up and running more easily and with less code than ever before. At the same time, you can add custom features to ASP.NET to accommodate your own requirements.

Specific areas in which ASP.NET has been improved are:

- **Productivity**: You can easily and quickly create ASP.NET Web pages and applications using new ASP.NET server controls and existing controls with new features. New areas such as membership, personalization, and themes provide system-level functionality that would normally require extensive developer coding. Core development scenarios, particularly data, have been addressed by new data controls, no-code binding, and smart data-display controls.
• **Flexibility and extensibility:** Many ASP.NET features are extensible so that you can easily incorporate custom features into applications. The ASP.NET provider model, for example, provides pluggable support for different data sources.

• **Performance:** Features such as recompilations, configurable caching, and SQL cache invalidation allow you to optimize the performance of your Web applications.

• **Security:** It is now easier than ever to add authentication and authorization to your Web applications.

• **Hosting:** ASP.NET includes new features that make it easier to manage a hosting environment and create more opportunities for hosters to add value.

• **Completeness:** New and existing features work in concert to allow you to create end-to-end scenarios that address real-world Web development challenges.

The following sections describe changes made in different areas of ASP.NET:

1. Web Site Management
2. Page Design
3. Controls
4. Data
5. Security

1. Web Site Management:
Web site configuration is improved to include many more settings. You can easily manage application settings using the Web Site Administration Tool, which provides a wizard-like interface for setting up and maintaining your applications. The Web Site Administration Tool is particularly useful for managing remote sites (for example, sites that are hosted by an ASP.NET-compatible ISP).

If you host sites for others, you can use a new ASP.NET Microsoft Management Console (MMC) plug-in or an administrative API to manage sites and monitor their health. You can add value to your hosting site by offering controls or services that you can selectively enable or disable.

1.1 New Administration Features and Tools:

ASP.NET includes features to make Web site management easier for both Web site developers and for administrators. Configuration files include a richer set of elements that give site developers control over new features and finer control over existing ones. A new configuration API makes it possible to control configuration programmatically. New tools provide a GUI interface for configuring applications — the new Web Site Administration Tool makes it easy for Web site developers to manage their own sites using a Web-based interface (both locally and remotely), and an ASP.NET-specific MMC snap-in allows site administrators to manage complex configuration scenarios using a standard Windows server-based tool.

1.2 Reserved Folders for Special Functionality:

Web sites can include an App_Code folder into which site developers can put source code that is then compiled automatically as part of the Web site, making it unnecessary to compile components or controls before using them in a site. An App_Data folder is reserved for databases (for example, SQL Server Express Edition .mdf files). Special folders for resources contain XML-based files containing strings and other resources for localization, which are dynamically compiled into assemblies at runtime.
1.3 Recompiling Web Sites for Error Checking and Deployment

You can now precompile your Web site, which allows you to enhance performance by avoiding the overhead of dynamic compilation and allows you to catch compile-time errors. You can also precompile a site for deployment, producing a version of the site that you can easily copy to or install on a production server. Precompiling for deployment strips source code, which helps you protect your intellectual property.

1.4 Navigation

You can add site navigation to your Web sites by defining a site map (typically an XML file). You can then use new navigation controls such as the Tree View and Sitemap Path controls that can automatically create a menu or tree view of pages, or that can display a navigation path (also known as a breadcrumb) showing the current page hierarchy.

2. Page Design

New features of ASP.NET help you easily create pages that are more consistent and can offer a richer experience to users.

2.1 Consistent Layout using Master Pages

You can use master pages to create a consistent page layout for your Web site or for a group of related pages. On the master page, you define a common look; you can then create individual content pages that provide the dynamic content that is displayed using the master page as a template

2.2 Themes for Consistent Appearance
Themes allow you to create a consistent look for your site. A theme contains a collection of control skins that define property settings (such as background color and font), CSS style settings, and images to specify the appearance of a control.

2.3 New Code-Behind Model for Pages

ASP.NET continues to support the single-file model from the previous version of ASP.NET and now includes a new code-behind model. The new model allows you to use a new .NET Framework feature, partial classes, to create a code file with just the code in it that you need, such as event handlers. The end result is a clean separation of the markup from the code in a page, and a robust model for managing a page’s code.

2.4 Users-Customizable Web Pages

With Web parts, you can create modularized Web pages that users can customize extensively. A Web part incorporates a discrete piece of functionality, such as a weather report, a stock ticker, or a window for reading news. ASP.NET allows you to create your own Web parts. You can then create Web pages that include a selection of Web parts. Using the Web parts architecture, you can allow users to customize individual Web parts (for example, by typing in a ZIP code to display local information) and change the layout of Web parts. You can also allow users to add and remove Web parts, thereby giving users the ultimate flexibility in using your site.

2.5 User-Specific Values

You can provide users with a custom experience in your Web site by defining and using profile properties, which you can use to track user information (address, city), preferences (color scheme, list of stocks to follow), or any custom information required by your application (shopping cart). Once you have defined profile properties, ASP.NET automatically associates individual instances of the profile properties with each user, and you can use code to
set or get the values. ASP.NET persists property values in a data store (which you can configure), and the next time a user visits your site, ASP.NET automatically retrieves the profile property value for that user.

2.6 New Caching Features

Enhancements to caching help you increase the performance of your site and manage more precisely. You can manage caching at the Web site level using Web.config settings, rather than managing it at the page level. You can now invalidate cache contents programmatically. If you are using SQL Server 7.0 or later, you can set up a cache dependency based on a SQL Server database table so that when the table changes, the cache entry dependent on it is made invalid.

2.7 Enhanced Client-Side Functionality

ASP.NET pages make it easier to add client-side functionality to pages to enhance the run-time experience for users. You can more easily bind client-side event handlers to controls and add better support for accelerator keys. For multi-page forms, you can use the new Wizard control, or you can take advantage of the new cross-page-posting facility in Web pages.

3. Control

ASP.NET features many improvements to Web server controls, including both enhancements to existing controls and a selection of new controls.

3.1 General Control Improvements

A major area of enhancement in ASP.NET 2.0 is in the controls you use to create ASP.NET Web pages. General improvements to controls include:
• By default, controls now generate markup that is compatible with the XHMTL 1.1. Standard.

• All controls support adaptive rendering, and can emit markup that is appropriate for the requesting browser.

• All data controls can use either the new data-binding model with data source controls, or can continue to use the model used in earlier versions of ASP.NET.

• All controls support themes and skins so that you can customize their appearance using an ASP.NET theme.

• You can use device filtering with many control properties, specifying different property values for different devices. When the control is rendered, the appropriate property value is set based on the requesting browser.

• Validation controls can now be grouped, which allows you to selectively enable validation for some controls.

• Improvements to individual controls such as the Label control, List Box control, and others to add functionality often requested by developers.

3.2 New Controls

ASP.NET offers a greater selection of controls to help you create fully featured Web pages much more quickly. You can now take advantage of the following controls:
• **Navigation.** You can add navigation paths (also known as breadcrumbs) to pages with the *Sitemap Path* control and display a site map using the *Tree View* control.

• **Security.** You can authenticate users with a suite of login controls that allow you to get and validate user credentials, display custom output for logged-in users, and more.

• **Web Parts.** A new set of Web Part controls allow you to create portal pages that users can personalize at run time. For example, you can create a Web Part that allows a user to enter a custom value such as a postal code and get localized weather reports.

• **Client behavior.** New controls provide ASP.NET server control functionality for tasks that previously was handled only by HTML elements. These include the *FileUpload* control, *Image Map* control, and the *Hidden Field* control.

  The new Wizard controls allow you to create a page with multiple panels that step the user through multi-part forms.

  In addition, existing ASP.NET controls have been enhanced with new features. All controls now support the new data-binding model, themes, and personalization. You can also use all controls to create pages for devices; controls now automatically support adaptive rendering and device filtering.
4. Data

ASP.NET includes substantially improved support for working with data in your Web applications.

4.1 Data Source Controls

For binding data to controls on Web pages, you can now use data source controls, which encapsulate connections, query commands, and parameters into a single control. ASP.NET includes data source controls that work with a variety of back-end data sources, including Microsoft SQL Server, Microsoft Access, XML files, Web services, FrontPage site maps, and business objects that return data sets. All data source controls support the same basic object model, giving you a consistent way to work with data regardless of its source.

Data source controls can automatically fetch data and manage it when the page runs. You no longer need to write code to execute commands and manage datasets for common data scenarios. However, if your application requires it, you still have access to the lower-level data functions exposed by ADO.NET.

To pass parameters to data source controls, you can configure the controls to draw parameters values from other controls, Session state, query strings, or cookies; in addition, you can set parameter values programmatically.

4.2 Middle-Tier Data Access

By using the new Object Data Source control, you can easily add data access to a page that is based on a middle-tier business object. The
Object Data Source exposes the same binding interface to controls on the page, but instead of performing direct database access, it invokes methods on a component that you specify.

### 4.3 Data Display Controls

ASP.NET also includes enhanced support for displaying and updating data with controls on Web pages. All controls can now use data source controls as data source instead of working directly against a dataset or other store. You can also take advantage of the following new controls that are specifically built to make data access easier:

- The Grid View, Details View, Form View controls to display and edit data. (The Grid View control supersedes the Data Grid control from previous versions of ASP.NET.)

- The Tree View control to display hierarchical information from XML files, sitemap files, and relational data sources.

- The Sitemap Path and Menu controls to provide data-bound support for navigation.

### 4.4 XML Support

You can use XML data in a variety of ways in ASP.NET. An XML data source control exposes XML data to be used either as hierarchical or tabular data. You can bind a Tree View control to XML data to provide a hierarchical view for users, or you can bind a list control such as the Grid View control to display XML in a traditional way.

### 4.5 Connection String Storage
To enhance Web site security, you can store connection strings in a dedicated section of the configuration file.

5. SECURITY

New ASP.NET features provide built-in support for authenticating and authorizing users. ASP.NET membership manages authentication, providing facilities for validating user credentials and helping users manage their passwords.

The membership service provides APIs that you can call programmatically to create new users, validate credentials, and get user information. To simplify authentication even more, you can use the new login controls, which work with the ASP.NET membership to perform a variety of tasks, often with no extra code required. The Login control prompts users for credentials and validates them. The Password Recovery control provides various options for helping users either change or remember their password. You can display user information with the Login Name control, and present a Login or Logout button using the Login Status control. The Login View control allows you to mark content in a page that is visible only to authenticated users.

ASP.NET role management helps you manage authorization, providing high-level ways for you to define and check roles for users.

Both the membership and role management systems use the new ASP.NET provider model, which separates the functionality of the systems from the data store that supports them. For example, membership allows you to store user information in Microsoft SQL Server or Microsoft Access. If these default options are not suitable for your requirements, you can create your own provider that accepts calls from membership and then fulfills them using your custom data logic.
Microsoft’s C#.Net

An ASP.NET page is restricted to code written in a single programming language. Currently, ASP.NET supports Visual Basic, C#, and J#. The default language is Visual Basic, but any other language can be declared as the default language for the page by placing a directive at the top of the page, such as the following:

```csharp
<%@Language="C#"%>
```

The language can also be declared in a `<script language= ...>` block, as in the example provided in Code Blocks in ASP.NET. If different languages are declared in separate script blocks on the same page, an error will be thrown.

Although only one language can be used on a single page, any user controls that you use on a page can be written in different languages, including Visual C++. This allows applications to take advantage of function libraries in different languages.

Visual Basic Scripting Edition (VBScript) is not supported, but VBScript syntax is very similar to Visual Basic 2005 syntax, so any required changes to existing code should be minor.

:: Back-End ::

Microsoft SQL Server is a set of components that work together to meet the data storage and analysis needs of the largest Web sites and enterprise data processing systems. Microsoft SQL Server 2005 data is stored in databases. The data in a database is organized into the logical components visible to users. A database is also physically implemented as two or more files on disk.

Each instance of SQL Server has four system databases (Login, Registered and Rem) and one or more user databases (Grp and Mbr). Some
organizations have only one user database, containing all the data for their organization. Some organizations have different databases for each group in their organization, and sometimes a database used by a single application. For example, an organization could have one database for sales, one for payroll, one for a document management application, and so on. Sometimes an application uses only one database; other applications may access several databases.

It is not necessary to run multiple copies of the SQL Server database engine to allow multiple users to access the databases on a server. An instance of the SQL Server Standard or Enterprise Edition is capable of handling thousands of users working in multiple databases at the same time.

Database Architecture:

Microsoft SQL Server 2005 data is stored in databases. The data in a database is organized into the logical components visible to users. A database is also physically implemented as two or more files on disk.

When using a database, you work primarily with the logical components such as tables, views, procedures, and users. The physical implementation of files is largely transparent. Typically, only the database administrator needs to work with the physical implementations.
Data Flow Diagram

Context Diagram

Level1 Diagram
Process 1.1

Process 2.1
Data Dictionary

Admin_Login

<table>
<thead>
<tr>
<th>Login</th>
<th>Name</th>
<th>Type</th>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Int</td>
<td>P.K.</td>
<td></td>
<td>Auto generated ID</td>
</tr>
<tr>
<td>Username</td>
<td>Varchar(50)</td>
<td></td>
<td></td>
<td>User Name</td>
</tr>
<tr>
<td>Password</td>
<td>Varchar(50)</td>
<td></td>
<td></td>
<td>Password of User</td>
</tr>
</tbody>
</table>

This table is used to maintain the information of the user’s login.

Dept

<table>
<thead>
<tr>
<th>Department</th>
<th>Name</th>
<th>Type</th>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Int</td>
<td>P.K.</td>
<td>Auto generated</td>
<td>Dept ID</td>
</tr>
<tr>
<td>ImgUrl</td>
<td>varchar(50)</td>
<td></td>
<td></td>
<td>Show Path to Choose Image</td>
</tr>
<tr>
<td>Name</td>
<td>varchar(50)</td>
<td></td>
<td></td>
<td>Department Name</td>
</tr>
<tr>
<td>Designation</td>
<td>Varchar(50)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Degree</td>
<td>Varchar(50)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
This table is used to maintain the information of the various Departments.

**Faculty_Detail**

<table>
<thead>
<tr>
<th>Faculty</th>
<th>Type</th>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Int</td>
<td>P.K.</td>
<td>Auto generated ID of Faculty</td>
</tr>
<tr>
<td>RegNo</td>
<td>Varchar(50)</td>
<td></td>
<td>Faculty Registration Number</td>
</tr>
<tr>
<td>Name</td>
<td>Varchar(50)</td>
<td></td>
<td>Name of Faculty/User</td>
</tr>
<tr>
<td>LastName</td>
<td>Varchar(50)</td>
<td></td>
<td>Last Name of Faculty/User</td>
</tr>
<tr>
<td>Password</td>
<td>Varchar(50)</td>
<td></td>
<td>Require Password to Login as Faculty</td>
</tr>
<tr>
<td>Joining Date</td>
<td>Varchar(50)</td>
<td></td>
<td>Joining Date Of Faculty</td>
</tr>
<tr>
<td>Birth Date</td>
<td>Varchar(50)</td>
<td></td>
<td>Birth Date Of Faculty</td>
</tr>
<tr>
<td>Department</td>
<td>Varchar(50)</td>
<td></td>
<td>Department In Which Faculty Recruited</td>
</tr>
<tr>
<td>Post</td>
<td>Varchar(50)</td>
<td></td>
<td>Post On Which Faculty Recruited</td>
</tr>
<tr>
<td>Contact No</td>
<td>Varchar(50)</td>
<td></td>
<td>Mobile Number Of Faculty</td>
</tr>
<tr>
<td>Email ID</td>
<td>Varchar(50)</td>
<td></td>
<td>E-Mail ID Of Faculty</td>
</tr>
</tbody>
</table>
This table is used to maintain the information of the groups Member of Faculty.

**Gallery**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Int</td>
<td>P.K.</td>
<td>Auto generated ID for Photo</td>
</tr>
<tr>
<td>ImgUrl</td>
<td>Varchar(50)</td>
<td></td>
<td>Display Path to Choose Image</td>
</tr>
<tr>
<td>Category</td>
<td>Varchar(50)</td>
<td></td>
<td>Name of Category for which this Photo belongs to</td>
</tr>
</tbody>
</table>

This table is used to maintain the detailed information of various category photo.

**Mail**

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Int</td>
<td>P.K.</td>
<td>Auto generated ID</td>
</tr>
<tr>
<td>SuserName</td>
<td>varchar(50)</td>
<td></td>
<td>Source</td>
</tr>
</tbody>
</table>
This table is used to maintain the detailed information of mail facility.

News

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Int</td>
<td>P.K.</td>
<td>Auto generated ID for Photo</td>
</tr>
<tr>
<td>Image</td>
<td>Varchar(50)</td>
<td></td>
<td>Display Path to Choose Image</td>
</tr>
<tr>
<td>Sdesc</td>
<td>Varcahr(50)</td>
<td></td>
<td>Small Description</td>
</tr>
<tr>
<td>Fdesc</td>
<td>Varcahr(50)</td>
<td></td>
<td>Full Description</td>
</tr>
</tbody>
</table>

This table is used to maintain the detailed information of display events & news.
## Registration

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Constraint</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>Int</td>
<td>P.K.</td>
<td>Auto generated ID of Faculty</td>
</tr>
<tr>
<td>RegNo</td>
<td>Varchar(50)</td>
<td></td>
<td>Faculty Registration Number</td>
</tr>
<tr>
<td>Name</td>
<td>Varchar(50)</td>
<td></td>
<td>Name of Faculty/User</td>
</tr>
<tr>
<td>LastName</td>
<td>Varchar(50)</td>
<td></td>
<td>Last Name of Faculty/User</td>
</tr>
<tr>
<td>UserName</td>
<td>Varchar(50)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Password</td>
<td>varchar(50)</td>
<td></td>
<td>Require Password to Login as Faculty</td>
</tr>
<tr>
<td>Category</td>
<td>Varchar(50)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Class</td>
<td>Varchar(50)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Joining Date</td>
<td>Varchar(50)</td>
<td></td>
<td>Joining Date Of Faculty</td>
</tr>
<tr>
<td>Birth Date</td>
<td>Varchar(50)</td>
<td></td>
<td>Birth Date Of Faculty</td>
</tr>
<tr>
<td>Department</td>
<td>Varchar(50)</td>
<td></td>
<td>Department In Which Faculty Recruited</td>
</tr>
<tr>
<td>Post</td>
<td>Varchar(50)</td>
<td></td>
<td>Post On Which Faculty Recruited</td>
</tr>
<tr>
<td>Contact No</td>
<td>Varchar(50)</td>
<td></td>
<td>Mobile Number Of</td>
</tr>
<tr>
<td>Faculty</td>
<td>Email ID</td>
<td>VarChar(50)</td>
<td>E-Mail ID Of Faculty</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>------------</td>
<td>-------------</td>
<td>----------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Photo</td>
<td>VarChar(50)</td>
<td>Photo Link Of Faculty</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Status</td>
<td>VarChar(50)</td>
<td></td>
</tr>
</tbody>
</table>

This table is used to maintain the detailed information of about the user registration.
Screen Layouts

Input/output Screen Design

Home.aspx

Education is the process by which an individual is encouraged and enabled to fully develop his or her potential; it may also serve the purpose of equipping the individual with what is necessary to be a productive member of society. Through teaching and learning the individual acquires and develops knowledge and skills.

The term education is often used to refer to formal education. However, the word’s broader meaning covers a range of experiences, from formal learning to the building of understanding and knowledge through day to day experiences. Ultimately, all that we experience serves as a form of education.

It is widely accepted that the process of education is lifelong. Studies have shown that the child already in utero is educated by the experiences it is exposed to.
Crystal College is an accredited Information Technology & Management College based in Rajkot, Gujarat, India. Founded by a group of dedicated IT & Management professionals and experienced educators, we have set a new benchmark in Information Technology & Management education.

We believe that to be successful in the Information Technology & Management Field, one needs to have knowledge of Management theory, as well as substantial firsthand knowledge of the practical aspects of Information Technology. Our philosophy is to combine these 2 elements in a theory-plus-practice approach, which balances academic learning with hands-on training.
Login.aspx
Gallery.aspx
The Crystal College Library is a hub of academic and research activities. The library has a rich collection of old and rare books, archival records, valuable reference books, important journals, educational CDs and floppies for the use of students, staff, research scholars and special borrowers.

The total number of books in Crystal College Library is around 3.5 lakh, the number of periodicals is 1.5 lakh and also there are thousands of non-book materials, including CDs and floppies. It has a good number of archival materials depicting history of Indian education, correspondences, minutes, proceedings, etc. The collection includes rare and valuable of paintings, water colours, photographs, etc., mostly drawn by renowned European painters.

The Library introduced the Web-based fully integrated multi-user library system LIBSYS 4 ( Release 5.0 ) for the complete library automation including LAN among its all branches.
News & Events.aspx
Department.aspx
Mail.aspx
LogOut.aspx
## Requirements

### Hardware requirements

<table>
<thead>
<tr>
<th>Component</th>
<th>Development</th>
<th>Deployment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer System/Processor</td>
<td>Pentium 4.1GHz processor</td>
<td>Pentium 4.1GHz processor</td>
</tr>
<tr>
<td>Hard Drive</td>
<td>80 GB</td>
<td>80 GB</td>
</tr>
<tr>
<td>RAM</td>
<td>Minimum 512 MB</td>
<td>Minimum 512 MB</td>
</tr>
<tr>
<td>Internet Connection</td>
<td>No</td>
<td>Yes</td>
</tr>
<tr>
<td>Printer</td>
<td>No</td>
<td>Yes</td>
</tr>
</tbody>
</table>

### Software requirements

<table>
<thead>
<tr>
<th>Component</th>
<th>Development</th>
<th>Deployment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating System</td>
<td>Windows XP Professional or later</td>
<td>Windows XP Professional or later</td>
</tr>
<tr>
<td>Database server</td>
<td>Microsoft SQL Server 2005</td>
<td>Microsoft SQL Server 2005</td>
</tr>
<tr>
<td>Front End Tool</td>
<td>Asp.Net 2.0</td>
<td>Internet Explorer</td>
</tr>
</tbody>
</table>
Limitations and Future Analysis

**Limitations**
- Web Application runs only on windows based operating system.
- This web application is static means not run on internet.

**Future Analysis**
- Student also edit, update their information.
- Student also made their community and mails each other.
- Student can upload their events photos.
THANK YOU