ISSN:0975-9646

Avance Bibliothek Management System

Maddirala Jagadish [#1], Palli Hemanth Kumar[*2], Pamarthi Jagadish[#3]

[#]Computer Science &Engineering, Jntuk -Kakinada University
Nuzvid,nuzvid(m.d),Krishna(dist). Vijayawada.
Computer Science &Engineering, Jntuk -Kakinada University
Vuyyuru(m.d),Krishna(dist). Vijayawada
[*] Computer Science &Engineering, Jntuk -Kakinada University
Ramanna gudam,nuzvid(md), Krishna(dist). Vijayawada

Abstract- In this paper Book management system is a place where all kind of books are available. Avance Bibliothek Management system is a web based application. This system contains list of all the books and can be accessed by remote users concurrently from anywhere in the campus by using internet. But for that users must be registered individually through their unique user id's. This system is three tier architecture. Client sends requests, on receiving the request the server processes it and extracts the data from database and sends the result back to the client. This system provides separate interface and login for librarian, students and faculties. Librarian can modify database .Users can search for books in module presented in authorized college website. They can recommend for new books by just sending messages to the librarian from anywhere in the college. They can view the issue and return dates of any book and due they have to pay. This system generates reports that can be used in analysing the library performance. Thus the management can take appropriate steps to improve the facilities.

Keywords— Bibliotheks, Clients, management, database services,

I.INTRODUCTION

Actually Avance Bibliothek is the combination of German and French which means advanced library. So, the title itself shows our intention of implementing basic existing modules of library management system to the advanced modules.

1.1 **Existing system:**

The main drawbacks of the basic library system are listed below:

The basic library management system facilitates the basic features and it provides the less interaction facility between user and the admin and it providing using the facilities at campus only and it doesn't providing the novels and self assessment schemes like very useful for relaxing in the leisure times in the campus. It took much time for queries taking long time consuming process and it doesn't supports the user at anywhere outside the campus it is only at offline statement.

- Searching of required books to the user only at campus.
- Not providing novels and self assessment schemes.
- Answering management query is a time consuming process.
- No user alerts.
- User can only use at offline level only.

1.2. **Proposed system**

The proposed system provides the advanced features as that does not provided by the basic library management system and these features are both soft copies of reference books and lectures of highly designated professionals and then providing the superior advantages which are likely to be very useful and easy to use the user by using this facilities and also providing the knowledge sharing centre through internet access by the user by using this system

- e-books and e-learning
- position finder
- buzz system
- self assessments
- providing the revised question papers – Previous Papers
- Wisdom Center

II.LITERATURE SURVEY

II.1. **Techinque:**

This involves questions such as whether the technology needed for the system exists, how difficult it will be to build, and whether the firm has enough experience using that technology. The assessment is based on an outline design of system requirements in terms of Input, Processes, Output, Fields, Programs, and Procedures. This can be qualified in terms of volumes of data, trends, frequency of updating in order to give an introduction to the technical system. The application is the fact that it has been developed on Windows XP platform and a high configuration of 1GB RAM on Intel Pentium dual core processor. This is technically feasible.

II.2. **Feasibility study:**

Feasibility Study is a preliminary study undertaken to determine and document a project's viability. The term feasibility study is also used to refer to the resulting document. These results of this study are used to make a decision whether to proceed with the project, or table it. If it indeed leads to a project being approved, it will — before the real work of the proposed project starts — be used to ascertain the likelihood of the project's success. It is an analysis of possible alternative solutions to a problem and a recommendation on the best alternative. It, for example, can decide whether an order processing be carried out by a new system more efficiently than the previous one.

III SYSTEM REQUIREMENT'S SPECIFICATION

SYSTEM ARCHITECTURE:

The application will follow three-tier architecture. In three-tier architecture application will run the traditional client/server model but from the web server. The client only displays the GUI and data but has no part in producing results.

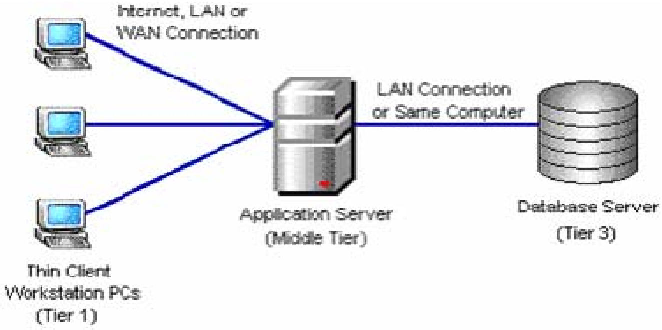


Fig. Three-tier architecture

Three-tier architecture will contain the following tiers

Client/Presentation Tier:

This tier includes all the HTML content or forms to be displayed on the client browser. It is the form which provides the user interface to end user. Programmer uses this tier to get or set the data back and forth.

Business Logic Layer:

In the Business logic tier, the actual processing of the data and the logic behind the implementation of the application will be present. This tier can contain a class, which can be used to write the functions, and also works as a mediator between the presentation tier and data tiers.

Data Tier:

Data Tier contains methods and classes that deal with passing and storing data to the data Storage Layer. Queries or stored procedures are used to access the data from the database or to perform any operation to the database. It stores the data passed by the presentation tier.

DEFINATIONS, ACRONYMS& ABBREVATIONS:

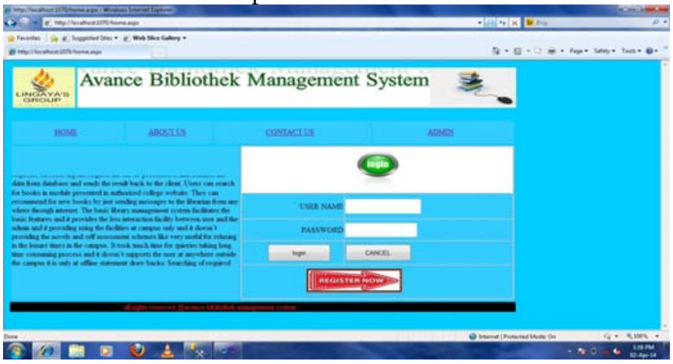
- HTML: Hypertext Markup Language is a markup language used to design static web pages.
- Asp: Active server pages are used to develop web application.
- IIS: Internet Information Service is a web server to run web application.
- VS: Visual Studio is application where we can develop application by using this IDE.
- HTTP: Hypertext Transfer Protocol is a transaction oriented client/server protocol between web browser & a Web Server.
- HTTPS: Secure Hypertext Transfer Protocol is a HTTP over SSL (secure socket layer).

TCP/IP: Transmission Control Protocol/Internet Protocol, the suite of communication protocols used to connect hosts on the Internet.

IV. MODULAR DESIGN

The modules are:

1. Registration Module .
2. Book Maintenance Module.
3. Admin Module.
4. Accounts Module.
5. Brainbooster Module.
6. Reports Module.

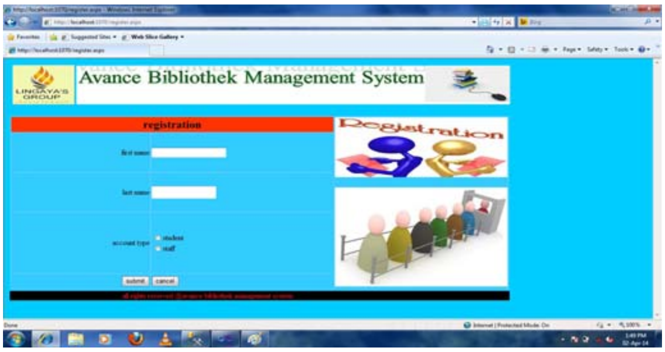


data from database and sends the result back to the client. Users can search for books in module presented in authorized college website. They can recommend for new books by just sending messages to the librarian from any where through internet. The basic library management system facilitates the basic features and it provides the less interaction facility between user and the admin and it providing using the facilities at campus only and it doesn't providing the novels and self assessment schemes like very useful for relaxing in the leisure times in the campus. It took much time for queries taking long time consuming process and it doesn't supports the user at anywhere outside the campus it is only at offline statement draw backs. Searching of required

IV.1. Registration module:

Registration module mainly contains of two types of registration

1. Student registration.
2. Staff registration.



Staff and student registration form

Firstly student registration:

If the student wants to log into the website then must registered in this student registration. It contains all the details of the students, this registration can be taken the purpose of whether the user can be trusted person or not, also it can be useful for the admin to know the complete data of the account holder.

Staff registration :

To access this web site, even faculty members also should be register here, it contains the complete data about the staff, by giving the first and last name of the user ,then this module generates the full name of the user automatically and to give a unique ID to the user.

IV.2. Book maintenance module:

This module includes the different categories of books. In simply says, this is The module that can shows the all kinds of books which are available in the web site.

These are categorized into

- TECHNICAL
- MANAGEMENT
- JOURNALS
- NOVELS
- MAGAZINES
- E-BOOKS
- E-LEARNING

TECHNICAL sub module provides the information about the technical books which are available in the library. That information can be shown in the branch wise like: CSE, IT, CIVIL, MECH, ECE, EEE.
MANAGEMENT sub module provides the information about the management books which are available in the library. This kind of information can be shown in the branch wise like HR, FIANANCE, MARKET Its NG, SYSTEMS. JOURNALS sub module provides the information about the list of journals which are available in the library. The list can be categorized into ACEDAMIC, TRADE, SCIENTIFIC. NOVELs sub module provides the information about the list of the novels which are available in the library.

IV.3. Accounts module:

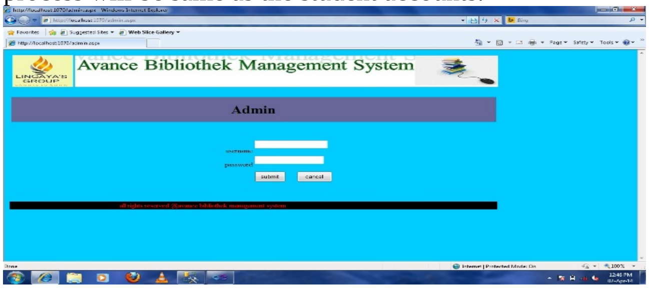
Accounts module consists of two types of accounts

1.Student accounts
2. staff accounts

It's main purpose is to display the total issuing and returning of books by the both students and staff it shows the information of all the books issued or returned from the library and information includes book name, book id ,author name, user phone number , Address, and type of entry and date of entry

IV.4. Admin module:

Admin module is nothing but the administration module. This module contains **.**all the rights regarding the books. The permissions can be approved through this module. In this module there are two phases. One was issue book and other was return book. If the student can get the book from the library then in the admin module will be update the issue book in the user account. If the student can submit the book to the librarian, then the admin will be a\update the student account in return books. For the staff also this process will be same as the student accounts.


http://localhost:1070/admin.aspx - Windows Internet Explorer
http://localhost:1070/admin.aspx
Bing
Favorites
Suggested Sites
Web Slice Gallery
http://localhost:1070/admin.aspx
Page
Safety
Tools
LINGAYA'S GROUP
Avance Bibliothek Management System
Admin
username
password
submit
cancel
all rights reserved @avance bibliothek managemant system
Done
Internet | Protected Mode: On
100%
12:46 PM

IV.5. Brain booster module:

Brain booster is the module that can be useful for the students to prepare The competitive examinations. This module contains the sub modules those are self assessments, wisdom center and mind games. Self assessments are the self tests conducting for the competitive exams, wisdom center includes the e-paper and current affairs for the students to develop the skills in general knowledge. Mind games are more useful for the relaxation.

IV.6. Reports module:

This module contains the data about the issue book, return book and issue date, return date. In this module has extra feature that is to send the request to admin for the required book.

V. DESIGN

5.1 UNIFIED MODELING LANGUAGES

Definition:

UML is a general-purpose visual modelling language that is used to specify, visualize, construct, and document the artifacts of the software system. UML is a method for describing the system architecture in detail using the blueprint.UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems.UML is a very important part of developing objects oriented software and the software development process.UML uses mostly graphical notations to express the design of software projects.

UML is a language: It will provide vocabulary and rules for communications and function on conceptual and physical representation. So it is modeling language.

UML Specifying:

Specifying means building models that are precise, unambiguous and complete. In particular, the UML address the specification of all the important analysis, design and implementation decisions that must be made in developing and displaying a software intensive system.

UML Visualization:

The UML includes both graphical and textual representation. It makes easy to visualize the
system and for better understanding.

UML Constructing:

UML models can be directly connected to a variety of programming languages and it is sufficiently expressive and free from any ambiguity to permit the direct execution of models.

Building blocks of UML:

- The vocabulary of the UML encompasses 3 kinds of building blocks.
- Things.
- Relationships.
- Diagrams.

Things:

Things are the data abstractions that are first class citizens in a model. Things are of 4 types
Structural Things, Behavioural Things, Grouping Things, An notational Things

Relationships:
Relationships tie the things together. Relationships in the UML are Dependency, Association, Generalization, Specialization

UML Diagrams:
A diagram is the graphical presentation of a set of elements, most often rendered as a connected graph of vertices (things) and arcs (relationships).

There are two types of diagrams, they are followed by

- Structural Diagrams.
- Behavioural Diagrams.

V.1. Structural Diagrams:

The UML's four structural diagrams exist to visualize, specify, construct and document the static aspects of a system. I can View the static parts of a system using one of the following diagrams. Structural diagrams consists of Class Diagram, Object Diagram, Component Diagram, Deployment Diagram.

V.2 Behavioural Diagrams:

The UML's five behavioural diagrams are used to visualize, specify, construct, and document the dynamic aspects of a system. The UML's behavioural diagrams are roughly organized around the major ways which can model the dynamics of a system Behavioural diagrams consists of Use case Diagram, Sequence Diagram, Collaboration Diagram, State chart Diagram, Activity Diagram.

VI.UML DIAGRAMS

VI.1 Use-Case diagram:

A use case is a set of scenarios that describing an interaction between a user and a system. A use case diagram displays the relationship among actors and use cases. The two main components of a use case diagram are use cases and actors.

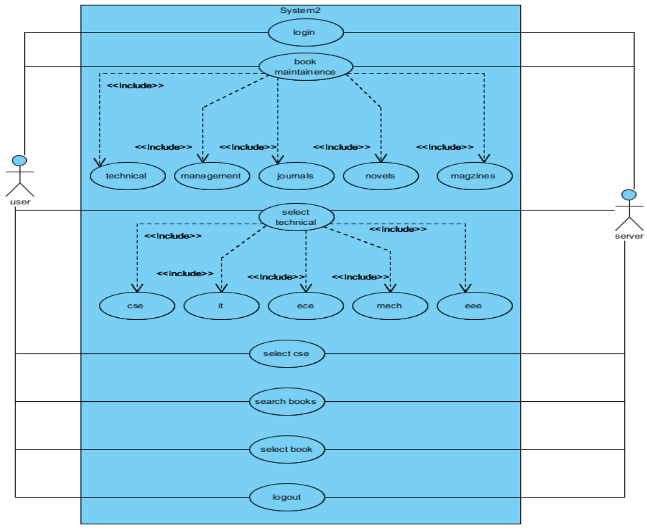


Fig:Usecase Diagram

VI.2. Class Diagram:

Class diagrams are widely used to describe the types of objects in a system and their relationships. Class diagrams model class structure and contents using design elements such as classes, packages and objects.

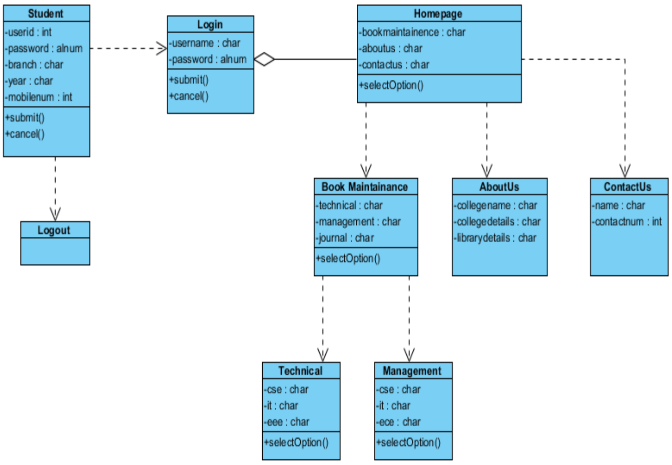


Fig: Class Diagram.

VI.3. Deployment Diagram

A deployment diagrams shows the hardware of your system and the software in those hardware. Deployment diagrams are useful when your software solution is deployed across multiple machines with each having a unique configuration. Below is an example deployment diagram.

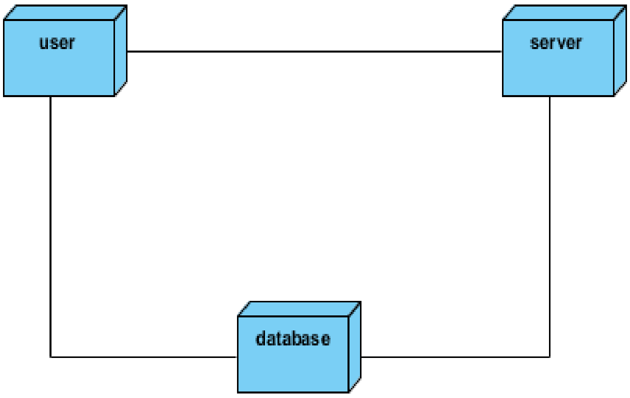


Fig:Deployment Diagram

VII. IMPLEMENTATION

VII.1 .Net Frame Work:

Microsoft .NET Framework:

The .NET Framework is a new computing platform that simplifies application development in the highly distributed environment of the Internet. The .NET Framework is designed to fulfil the following objectives:

- To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
- To provide a code-execution environment that minimizes software deployment and versioning conflicts.
- To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party.
- To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.

- To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.
- To build all communication on industry standards to ensure that code based on the .NET Framework can integrate with any other code. The .NET Framework has two main components: the common language runtime and the .NET Framework class library. The common language runtime is the foundation of the .NET Framework. You can think of the runtime as an agent that manages code at execution time, providing core services such as memory management, thread management, and remoting, while also enforcing strict type safety and other forms of code accuracy that ensure security and robustness. In fact, the concept of code management is a fundamental principle of the runtime. Code that targets the runtime is known as managed code, while code that does not target the runtime is known as unmanaged code. The class library, the other main component of the .NET Framework, is a comprehensive, object-oriented collection of reusable types that you can use to develop applications ranging from traditional command-line or graphical user interface (GUI) applications to applications based on the latest innovations provided by ASP.NET, such as Web Forms and XML Web services. The .NET Framework can be hosted by unmanaged components that load the common language runtime into their processes and initiate the execution of managed code, thereby creating a software environment that can exploit both managed and unmanaged features. The .NET Framework not only provides several runtime hosts, but also supports the development of third-party runtime hosts.

For example, ASP.NET hosts the runtime to provide a scalable, server-side environment for managed code. ASP.NET works directly with the runtime to enable Web Forms applications and XML Web services, both of which are discussed later in this topic. Internet Explorer is an example of an unmanaged application that hosts the runtime (in the form of a MIME type extension). Using Internet Explorer to host the runtime enables you to embed managed components or Windows Forms controls in HTML documents. Hosting the runtime in this way makes managed mobile code (similar to Microsoft® ActiveX® controls) possible, but with significant improvements that only managed code can offer, such as semi-trusted execution and secure isolated file storage. The following illustration shows the relationship of the common language runtime and the class library to your applications and to the overall system. The illustration also shows how managed code operates within a larger architecture.

Features of the Common Language Runtime:

The common language runtime manages memory, thread execution, code execution, code safety verification, compilation, and other system services. These features are intrinsic to the managed code that runs on the common language runtime. With regards to security, managed components are awarded varying degrees of trust, depending on a number of factors that include their origin (such as the Internet, enterprise network, or local computer). This means that a managed component might or might not be able to perform file-access operations, registry-access operations, or other sensitive functions, even if it is being used in the same active application. The runtime enforces code access security. For example, users can trust that an executable embedded in a Web page can play an animation on screen or sing a song, but cannot access their personal data, file system, or network. The security features of the runtime thus enable legitimate Internet-deployed software to be exceptionally featuring rich. The runtime also enforces code robustness by implementing a strict type- and code-verification infrastructure called the common type system (CTS). The CTS ensures that all managed code is self-describing. The various Microsoft and third-party language compilers generate managed code that conforms to the CTS. This means that managed code can consume other managed types and instances, while strictly enforcing type fidelity and type safety. In addition, the managed environment of the runtime eliminates many common software issues. For example, the runtime automatically handles object layout and manages references to objects, releasing them when they are no longer being used. This automatic memory management resolves the two most common application errors, memory leaks and invalid memory references. The runtime also accelerates developer productivity. For example, programmers can write applications in their development language of choice, yet take full advantage of the runtime, the class library, and components written in other languages by other developers. Any compiler vendor who chooses to target the runtime can do so. Language compilers that target the .NET Framework make the features of the .NET Framework available to existing code written in that language, greatly easing the migration process for existing applications. While the runtime is designed for the software of the future, it also supports software of today and yesterday. Interoperability between managed and unmanaged code enables developers to continue to use necessary COM components and DLL.The runtime is designed to enhance performance. Although the common language runtime provides many standard runtime services, managed code is never interpreted. A feature called just-in-time (JIT) compiling enables all managed code to run in the native machine language of the system on which it is executing. Meanwhile, the memory manager removes the possibilities of fragmented memory and increases memory locality-of-reference to further increase performance.

Finally, the runtime can be hosted by high-performance, server-side applications, such as Microsoft® MS Access™ and Internet Information Services (IIS). This infrastructure enables you to use managed code to write your business logic, while still enjoying the superior performance of the industry's best enterprise servers that support runtime hosting.

.NET Framework Class Library:

The .NET Framework class library is a collection of reusable types that tightly integrate with the common language runtime. The class library is Object Oriented, providing types from which your own managed code can derive functionality. This not only makes the .NET

Framework types easy to use, but also reduces the time associated with learning new features of the .NET Framework. In addition, third-party components can integrate seamlessly with classes in the .NET Framework. For example, the .NET Framework collection classes implement a set of interfaces that you can use to develop your own collection classes. Your collection classes will blend seamlessly with the classes in the .NET Framework.

As you would expect from an object-oriented class library, the .NET Framework types enable you to accomplish a range of common programming tasks, including tasks such as string management, data collection, database connectivity, and file access. In addition to these common tasks, the class library includes types that support a variety of specialized development scenarios. For example, you can use the .NET Framework to develop the following types of applications and services:

- Console applications.
- Scripted or hosted applications.
- Windows GUI applications (Windows Forms).
- ASP.NET applications.
- XML Web services.
- Windows services.

For example, the Windows Forms classes are a comprehensive set of reusable types that vastly simplify Windows GUI development. If you write an ASP.NET Web Form application, you can use the Web Forms classes.

Client Application Development:

Client applications are the closest to a traditional style of application in Windows-based programming. These are the types of applications that display windows or forms on the desktop, enabling a user to perform a task. Client applications include applications such as word processors and spreadsheets, as well as custom business applications such as data-entry tools, reporting tools, and so on. Client applications usually employ windows, menus, buttons, and other GUI elements, and they likely access local resources such as the file system and peripherals such as printers .Another kind of client application is the traditional ActiveX control (now replaced by the managed Windows Forms control) deployed over the Internet as a Web page. This application is much like other client applications: it is executed natively, has access to local resources, and includes graphical elements.

In the past, developers created such applications using C/C++ in conjunction with the Microsoft Foundation Classes (MFC) or with a rapid application development (RAD) environment such as Microsoft® Visual Basic®. The .NET Framework incorporates aspects of these existing products into a single, consistent development environment that drastically simplifies the development of client applications.

The Windows Forms classes contained in the .NET Framework are designed to be used for GUI development. You can easily create command windows, buttons, menus, toolbars, and other screen elements with the flexibility necessary to accommodate shifting business needs.

For example, the .NET Framework provides simple properties to adjust visual attributes associated with forms. In some cases the underlying operating system does not support changing these attributes directly, and in these cases the .NET Framework automatically recreates the forms. This is one of many ways in which the .NET Framework integrates the developer interface, making coding simpler and more consistent.

Unlike ActiveX controls, Windows Forms controls have semi-trusted access to a user's computer. This means that binary or natively executing code can access some of the resources on the user's system (such as GUI elements and limited file access) without being able to access or compromise other resources. Because of code access security, many applications that once needed to be installed on a user's system can now be safely deployed through the Web. Your applications can implement the features of a local application while being deployed like a Web page.

VI.2. **ASP.NET:**

ASP.NET is part of the whole. NET framework, built on top of the Common Language Runtime (also known as the CLR) - a rich and flexible architecture, designed not just to cater for the needs of developers today, but to allow for the long future we have ahead of us. What you might not realize is that, unlike previous updates of ASP, ASP.NET is very much more than just an upgrade of existing technology – it is the gateway to a whole new era of web development.

ASP.NET is a feature at the following web server releases:

- ➢ Microsoft IIS 5.0 on WINDOWS 2000 Server
- ➢ Microsoft IIS 5.1 on WINDOWS XP

ASP.NET has been designed to try and maintain syntax and run-time compatibility with existing ASP pages wherever possible. The motivation behind this is to allow existing ASP Pages to be initially migrated ASP.NET by simply renaming the file to have an extension of .aspx. For the most part this goal has been achieved, although there are typically some basic code changes that have to be made, since VBScript is no longer supported, and the VB language itself has changed.

Some of the key goals of ASP.NET were to:

- Remove the dependency on script engines, enabling pages to be type safe and compiled.
- Reduce the amount of code required to develop web applications.
- Make ASP.NET well factored, allowing customers to add in their own custom functionality, and extend/ replace built-in ASP.NET functionality.
- Make ASP.NET a logical evolution of ASP, where existing ASP investment and therefore code can be reused with little, if any, change.
- Realize that bugs are a fact of life, as ASP.NET should be as fault tolerant as possible.

Benefits of ASP.NET:

The .NET Framework includes a new data access technology named ADO.NET, an evolutionary improvement to ADO. Though the new data access technology is evolutionary, the classes that make up ADO.NET bear little resemblance to the ADO objects with which you might be familiar. Some fairly significant changes must be made to existing ADO applications to convert them to ADO.NET. The changes don't have to be made immediately to existing ADO applications to run under ASP.NET, however.ADO will function under ASP.NET. However, the work necessary to convert ADO applications to ADO.NET is worthwhile. For disconnected applications, ADO.NET should offer performance advantages over ADO disconnected record sets. ADO requires that transmitting and receiving components be COM objects. ADO.NET transmits data in a standard XML-format file so that COM marshaling or data type conversions are not required

ASP.NET has several advantages over ASP:

The following are some of the benefits of ASP.NET:

- ➢ Make code cleaner.
- ➢ Improve deployment, scalability, and reliability.
- ➢ Provide better support for different browsers .
- ➢ Enable a new breed of web applications.

ActiveX:

ActiveX is a specification develops by Microsoft that allows ordinary Windows programs to be run within a Web page. ActiveX programs can be written in languages such as Visual Basic and they are complied before being placed on the Web server.

ActiveX application, called controls, are downloaded and executed by the Web browser, like Java applets. Unlike Java applets, controls can be installed permanently when they are downloaded; eliminating the need to download them again. ActiveX's main advantage is that it can do just about anything.

This can also be a disadvantage:

Several enterprising programmers have already used ActiveX to bring exciting new capabilities to Web page, such as “the Web page that turns off your computer” and “the Web page that formats disk drive”. Fortunately, ActiveX includes a signature feature that identifies the source of the control and prevents controls from being modified. While this won't prevent a control from damaging system, we can specify which sources of controls we trust.

ActiveX has two main disadvantages:

It isn't as easy to program as scripting language or Java.

ActiveX is proprietary.It works only in Microsoft Internet Explorer and only Windows platforms.

VI.4. ADO.NET:

ADO.NET provides consistent access to data sources such as Microsoft SQL Server, as well as data sources exposed via OLE DB and XML. Data-sharing consumer applications can use ADO.NET to connect to these data sources and retrieve, manipulate, and update data.

ADO.NET cleanly factors data access from data manipulation into discrete components that can be used separately or in tandem. ADO.NET includes .NET data providers for connecting to a database, executing commands, and retrieving results. Those results are either processed directly, or placed in an ADO.NET Dataset object in order to be exposed to the user in an ad-hoc manner, combined with data from multiple sources, or remote between tiers. The ADO.NET Dataset object can also be used independently of a .NET data provider to manage data local to the application or sourced from XML.

Why ADO.NET?

As application development has evolved, new applications have become loosely coupled based on the Web application model. More and more of today's applications use XML to encode data to be passed over network connections. Web applications use HTTP as the fabric for communication between tiers, and therefore must explicitly handle maintaining state between requests. This new model is very different from the connected, tightly coupled style of programming that characterized the client/server era, where a connection was held open for the duration of the program's lifetime and no special handling of state was required.

In designing tools and technologies to meet the needs of today's developer, Microsoft recognized that an entirely new programming model for data access was needed, one that is built upon the .NET Framework. Building on the .NET Framework ensured that the data access technology would be uniform—components would share a common type system, design patterns, and naming conventions.

ADO.NET was designed to meet the needs of this new programming model: disconnected data architecture, tight integration with XML, common data representation with the ability to combine data from multiple and varied data sources, and optimized facilities for interacting with a database, all native to the .NET Framework.

Leverage Current ADO Knowledge:

Microsoft's design for ADO.NET addresses many of the requirements of today's application development model. At the same time, the programming model stays as similar as possible to ADO, so current ADO developers do not have to start from scratch in learning a brand new data access technology. ADO.NET is an intrinsic part of the .NET Framework without seeming completely foreign to the ADO programmer.ADO.NET coexists with ADO. While most new .NET applications will be written using ADO.NET, ADO remains available to the .NET programmer through .NET COM interoperability services.

ADO.NET provides first-class support for the disconnected, n-tier programming environment for which many new applications are written. The concept of working with a disconnected set of data has become a focal point in the programming model. The ADO.NET solution for n-tier programming is the Dataset.

XML Support:

XML and data access are intimately tied—XML is all about encoding data, and data access is increasingly becoming all about XML. The .NET Framework does not just support Web standards—it is built entirely on top of them.

VI.5. **SQL SERVER 2008:**

Microsoft SQL Server 2008 is comprehensive, integrated data management and analysis software that enables organizations to reliably manage mission-critical information and confidently run today's increasingly complex business applications. SQL Server 2008 allows companies to gain greater insight from their business information and achieve faster results for a competitive advantage.

Top-10 Features of SqlServer-2008:

1. T-SQL (Transaction SQL) enhancements:

T-SQL is the native set-based RDBMS programming language offering high-performance data access. It now incorporates many new features including error handling via the TRY and CATCH paradigm, Common Table Expressions (CTE), which return a record set in a statement, and the ability to shift columns to rows and vice versa with the PIVOT and UNPIVOT commands.

2. CLR (Common Language Runtime):

   The next major enhancement in SQL Server 2005 is the integration of a .NET compliant language such as C#, ASP.NET or VB.NET to build objects (stored procedures,
   triggers, functions, etc.). This enables you to execute .NET code in the DBMS to take advantage of the .NET functionality. It is expected to replace extended stored procedures in the SQL Server 2000 environment as well as expand the traditional relational engine capabilities.

3. Service Broker:

   The Service Broker handles messaging between a sender and receiver in a loosely coupled manner. A message is sent, processed and responded to, completing the transaction. This greatly expands the capabilities of data-driven applications to meet workflow or custom business needs.

4. Data encryption:

   SQL Server 2000 had no documented or publicly supported functions to encrypt data in a table natively. Organizations had to rely on third-party products to address this need. SQL Server 2008 has native capabilities to support encryption of data stored in user-defined databases.

5. SMTP mail:

   Sending mail directly from SQL Server 2000 is possible, but challenging. With SQL Server 2005, Microsoft incorporates SMTP mail to improve the native mail capabilities. Say "see-ya" to Outlook on SQL Server!

6. HTTP endpoints

   You can easily create HTTP endpoints via a simple T-SQL statement exposing an object that can be accessed over the Internet. This allows a simple object to be called across the Internet for the needed data.

7. Multiple Active Result Sets (MARS):

   MARS allow a persistent database connection from a single client to have more than one active request per connection. This should be a major performance improvement, allowing developers to give users new capabilities when working with SQL Server. For example, it allows multiple searches, or a search and data entry. The bottom line is that one client connection can have multiple active processes simultaneously.

8. Dedicated administrator connection:

   If all else fails, stop the SQL Server service or push the power button. That mentality is finished with the dedicated administrator connection. This functionality will allow a DBA to make a single diagnostic connection to SQL Server even if the server is having an issue.

9. SQL Server Integration Services (SSIS):

   SSIS has replaced DTS (Data Transformation Services) as the primary ETL (Extraction, Transformation and Loading) tool and ships with SQL Server free of charge. This tool, completely rewritten since SQL Server 2000, now has a great deal of flexibility to address complex data movement.

10. Database mirroring:

    It's not expected to be released with SQL Server 2005 at the RTM in November, but I think this feature has great potential. Database mirroring is an extension of the native high-availability capabilities. So, stay tuned for more details.

VI.6. HTML:

HTML (Hyper Text Markup Language) is the language that is used to prepare documents for online publications. HTML documents are also called Web documents, and each HTML document is known as Web page.

A page is what is seen in the browser at any time. Each Web site, whether on the Internet or Intranet, is composed of multiple pages. And it is possible to switch among them by following hyperlinks. The collection of HTML pages makes up the World Wide Web. A web pages is basically a text file that contains the text to be displayed and references of elements such as images, sounds and of course hyperlinks to other documents. HTML pages can be created using simple text editor such as Notepad or a WYSIWYG application such as Microsoft FrontPage.

VII.CONCLUSION

The goal of the system is to bring down the work load with the increased efficiency and to speed up the activities. With this it is very easy to process course fee that is collected time to time from students who are registered and studying at franchisees.

REFERENCES

1. Abraham, J. (1996), “Computers in modernising Library Information System and Services: Perspectives of Library Automation”, International Library Movement” vol. 18, p3.
2. Abdul Azeez, T. A (2004), “Tkm College of Engineering Library Automation System”, Annals of Library & Information Studies, Vol. 51, no. 2, p5257.
3. Adanu, Theodosia S. A (2007), “Planning and Implementation of the University of Ghana Library Automation Project”, African Journal of Library, Archives & Information Science, Vol. 16, no. 2, p101108.
4. Breeding, Marshall (2009), “Library Automation in a Difficult Economy”, Computers in Libraries, Vol. 29, no. 3, p2224.
5. Breeding, Marshall (2009), “Opening up Library Automation Software”, Computers in Libraries, Vol. 29 no. 2, p2527.
6. Breeding, Marshall (2009) “The Viability of Open Source ILS” Bulletin of the American Society for Information Science & Technology, Vol. 35.no.2, pp.2025.
7. Breeding, Marshall (2009), “Next Generation Library Automation: Its Impact on the Serials Community”, Serials Librarian, Vol. 56, no. 14,p5564.
8. Bansode, Sadanand Y and Periera, Shamin (2008), “A Survey of Library Automation in College Libraries in Goa State, India”, Library Philosophy & Practice, Vol. 10, no. 2, p17.
9. Bailey, Penny (2004), “INTERNET, Interaction and Intelligence: Latest Developments in Library Automation”, Managing Information, Vol. 11, no. 3, p3838.

10. De Smet, Egbert (2009), “ABCD: a new FOSS library automation solution based onISIS”, Information Development, Vol. 25, no.1, p6167.
11. Dietz, Roland; Grant, Carl (2005), “The Disintegrating World of Library Automation”, Library Journal, Vol. 130, no. 11, p3840.
12. Falup, Mariana (2007), “Achievements and Perspectives in Library Automation and Modernization”, Philobiblon, Vol. 12, p375381.
13. Fiehn, Barbara 14 (2006), “Library Automation in K12”,Multimedia & Internet, Vol.13, no. 5, p2729.
14. Groenewegen, Hans (2004), “Four decades of library automation recollections and reflections”, Australian Library Journal, Vol. 53, no.1 1, p3953.
15. Harder (1936), “Library automation and networking in India: An overview of recent developments”, Annals of Library Science & Documentation, vol.40, no.1, p 3240.
16. Haravu, L.J (1993), “Library automation and networking in lndia: An overview of recent developments”, Annals of Library Science & Documentation, vol.40, no.1, p 3240.
17. Hane, Paula J (2008), “Advances in Digital Reading, Enterprise Search, and Library Automation”, Information Today, Vol. 25, No.10, p713.
18. Henderson, Melissa (2008), “Heresy and Misconduct: Evolution in Library Automation”, ILA Reporter, Vol. 26, no. 3, p47.
19. Husain, Shabahat and Ansari, Mehtab Alam (2007), “Library automation software packages in India: A study of the cataloguing modules of Alice for Windows, Libsys and Virtua”, Annals of Library & Information Studies, Vol. 54, no.3, p33.
20. Kimber, Richard T (1996), “Automation in Libraries”, Oxford Pergamon.

BIOGRAPHIES

Maddirala Jagadish pursuing M.Tech in LIMAT in the stream of Computer Science and Engineering was born on 12th August, 1992. He received a Bachelor Degree in Computer Science and Engineering from Sri Saradhi Institute of Engineering & Technology in 2013.

Palli Hemanth Kumar pursuing M.Tech in LIMAT in the stream of Computer Science and Engineering was born on 15th June, 1991. He received a Bachelor Degree in Electronics and Computer Engineering from Prasad V.Potluri Siddhartha Institute of Technology in 2013.

Pamarthi Jagadish pursuing M.Tech in LIMAT in the stream of Computer Science and Engineering was born on 15th June, 1991. He received a Bachelor Degree in Information Technology from Sri Saradhi Institute of Engineering & Technology in 2012.