

DEJAR Mania

Asst.Prof. Mutyala Venugopal ^[1], Asst.prof.Nandyala Ravikumar ^[2]
 Konolla Siva Ramakrishna ^[3],
 Asst.Prof. Chikkavarau Gnanendra Rao ^[4], Asst.Prof. K.Sowjanya ^[5]
 M.phil Mathamatics Andhra University
 NUZVID,,Krishna(dist).
 Andhra Pradesh –India

ABSTRACT

This paper is aimed at developing a web based Leave Management Tool, which is of importance to either an organization or a college. The Déjàr Mania is an Internet based application that can be accessed throughout the organization or a specified group/Department.

This system can be used to automate the workflow of leave applications and their approvals. The periodic crediting of leave is also automated. There are features like notifications, cancellation of leave, automatic approval of leave, report generators etc. in this Tool.

The main objective of the proposed system is to decrease the paperwork and help in easier record maintenance by having a particular centralized Database System, where Leaves and Notices are maintained. The proposed system automates the existing system. It decreases the paperwork and enables easier record maintenance. It also reduces chances of Data loss.

Keywords:- Leave For Employee ,Database Storage

I. INTRODUCTION

Actually This paper is aimed at developing a web based Leave Management Tool, which is of importance to either an organization or a college. The Déjàr Mania is an Internet based application that can be accessed throughout the organization or a specified group/Department.

EXISTING SYSTEM

In existing system every college follows manual procedure in which faculty should enter in time and out time in a record book. At the end of each month Head of department will calculate leaves of every faculty member which is a time taking process and there are chances of losing records.

Drawbacks:

At present, the requirements of the system are done manually and hence there are number of limitations.

- Usage of Paper work
- Record maintained in a Attendance register
- Requires manual submission
- Time Consuming
- Chances of Data Loss
- No Automation

1.2 PROPOSED SYSTEM:

In proposed system chances of losing data is not possible because data is maintained in the form of database. Every day attendance, leaves and notices information is updated in to database using a user friendly GUI. This will reduce work for Head of departments. This application is an

online application which makes more flexible to access information.

ADVANTAGES:

1. Eliminates the use of paper leave application forms
2. Is cost efficient
3. Provides GUI and Digitalization
4. Easy to maintain in Centralized Database
5. Leave applications can be submitted via network
6. Leave application can be approved easily and notified to respective staff
7. Both the leave applicant as well as the approver can view the balance leave and past leave applications
8. Calculate the number of leaves taken on monthly/yearly basis.

II. LITERATURE SURVEY

2.1 TECHNIQUE

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 a technically feasible.

2.2 FEASIBILITY STUDY:

Feasibility studies aim to objectively and rationally uncover the strengths and weaknesses of the existing system or proposed venture. In its simplest term, the two criteria to judge feasibility are cost required and value to be attained. As such, a well-designed feasibility study should provide historical background of the paper. Generally, feasibility studies precede technical development and paper implementation. The assessment of feasibility study is based on the following factors:

- 1) Technical Feasibility
- 2) Economic Feasibility
- 3) Operational Feasibility

2.3 TECHNICAL FEASIBILITY:

Generally, feasibility studies precede technical development and paper implementation. The assessment is based on a system requirement in terms of Input, Processes, Output, Fields, Programs, and Procedure. This can be quantified in terms of volumes of data, trends, frequency of updating, etc., in order to estimate whether the new system will perform adequately or not.

Technological feasibility is carried out to determine the capability, in terms of software, hardware, personnel and expertise, to handle the completion of the paper. When writing a feasibility report the following should be taken to consideration:

- A brief description of the business
- The part of the business being examined
- The human and economic factor
- The possible solutions to the problems

2.4 ECONOMICAL FEASIBILITY:

Economic analysis is the most frequently used method for evaluating the effectiveness of a new system. More commonly known as cost/benefit analysis, the procedure is to determine the benefits and savings that are expected from a candidate system and compare them with costs. An entrepreneur must accurately weigh the cost versus benefits before taking an action

2.5. OPERATIONAL EASIBILITY:

Operational feasibility is a measure of how well a proposed system solves the problems, and takes advantage of the opportunities identified during scope

definition and how it satisfies the requirements identified in the requirements analysis phase of system development. The operational feasibility of the system can be checked as it solves the problems and reduces the complications occurring in the paper-pencil test.

2.6. CONCLUSION OF FEASIBILITY STUDY:

- 1) Technical Feasibility:
The system can be implemented using computer software & hardware.
- 2) Economic Feasibility:
The System implementation is economical if implemented for large/medium organization which can bear the cost of maintaining computer and server cost.
- 3) Operational Feasibility:
The system efficiently operates & reduces manual computation and time of processing, reducing cost of paperwork and human errors.

III. PRESENT WORK

3.1 PROPOSED SYSTEM:

In proposed system chances of losing data is not possible because data is maintained in the form of database. Every day attendance, leaves and notices information is updated in to database using a user friendly GUI. This will reduce work for Head of departments. This application is an online application which makes more flexible to access information.

Advantages:

1. Decreases of paper work
2. Easier record maintenance
3. Chances of reducing data loss
4. Saving time

3.2 SYSTEM REQUIREMENTS SPECIFICATION:

PRODUCT FUNCTIONALITY

Some major product functionalities of the system are as follows:

- Information about the employee/student/staff attendance.
- Check for leave availability.
- Maintain employee leave record.
- Display notices.

USERS AND CHARACTERISTICS

Primary users of the system will be employees working in company /students /staff, manager, HOD, Admin. Very little technical expertise is required for reading the outputted data since it is in graphical/tabular form.

LEVEL OF USAGE

Educational level of LMS computer software – Low
Experience of LMS software – None
Technical Expertise – Little

SPECIFIC REQUIREMENTS

External Interface Requirements

User Interfaces

The User Interface Screens are described in table

Table 1: Leave Management User Interface Screens

Software Interfaces

The system shall interface with an Oracle or Access database.

To implement the paper we have chosen HTML language for its more interactive and easy to understand support

Communications Interfaces

This System supports Google chrome and Mozilla Firefox web browsers.

This System involves FAQ forms for the requesting information, queries and problems etc.

Functional Requirements

- System will keep Employee records
- System provides Information about the leave approval and leave availability.
- Keep staff record.
- Keep notices record.
- Displays leave history.

Software requirements

Database : SQL Server2008
Operating System : Windows, Linux
Technology :Microsoft Visual Studio.NET

Hardware Requirements

Hard disk : 40GB
Processor : Intel (Pentium)

RAM : 1GB

3.3 DEFINITIONS, ACRONYMS & ABBREVIATIONS:

- 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. TCP/IP uses several protocols, the two main ones being TCP and IP.

IV. MODULE DESCRIPTION

The paper is done in 3 modules:

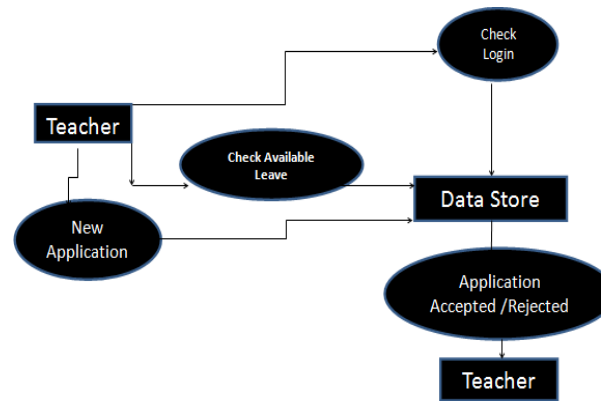
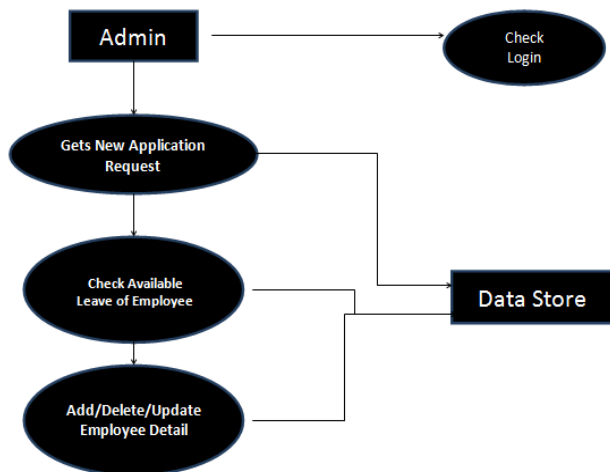
1. Admin Module
2. HOD Module
3. Staff Module

4.1 Admin Module

The Admin is responsible for any help required for the functioning of this software. He will be single point of contact for all leave related information for all employee of the organization. He has no authority of accepting or rejecting the application request.

- a) Responsible for addition of new employee in the system
- b) Responsible for maintaining leave record

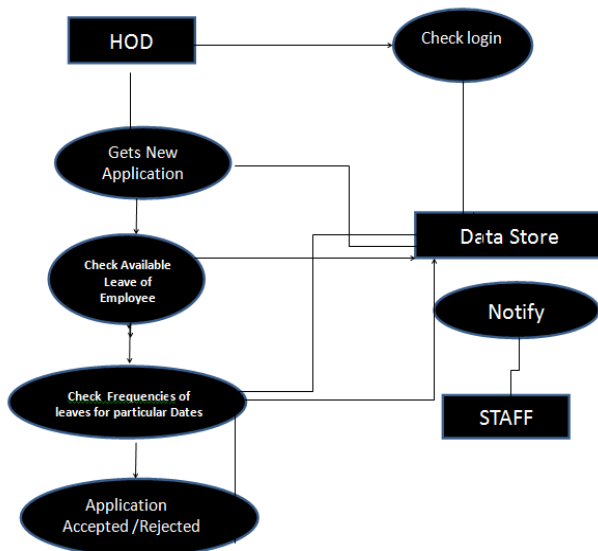
Responsible for calculation of work hours and providing leave information to the Finance Department for salary calculation



4.2 HOD Module

HOD logs into his interface window to

- Check new application request
- Check employee leave details
- Check persons on leave on particular given date/s
- Accepts/rejects leave application



4.3 Staff Module

Teaching/ Non-teaching Module: The staff member logs into his interface window here he can check:

- Personal leaves details
- Status of leave application
- Place new requests for leave

V. DESIGN

5.1 INTRODUCTION TO UML DIAGRAMS:

Systems design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. One could see it as the application of systems theory to product development. There is some overlap and synergy with the disciplines of systems analysis, systems architecture and systems engineering.

Unified Modeling Language:

UML stands for Unified Modeling Language. It is a third generation method for specifying, visualizing and documenting the artifacts of an object oriented system under development. Object modeling is the process by which the logical objects in the real world (problem space) are represented (mapped) by the actual objects in the program (logical or a mini world). This visual representation of the objects, their relationships and their structures is for the ease of understanding. This is a step while developing any product after analysis.

The goal from this is to produce a model of the entities involved in the paper which later need to be built. The representations of the entities that are to be used in the product being developed need to be designed.

Software design is a process that gradually changes as various new, better and more complete methods with a broader understanding of the whole problem in general come into existence. The UML model in diagrams

- Use case diagrams
- Class diagrams
- Sequence diagram

Use Case Diagram:

Use case diagram consists of use cases and actors and shows the interaction between them. The key points are:

- The main purpose is to show the interaction between the use cases and the actor.
- To represent the system requirement from user’s perspective.
- The use cases are the functions that are to be performed in the module.
- An actor could be the end-user of the system or an external system.

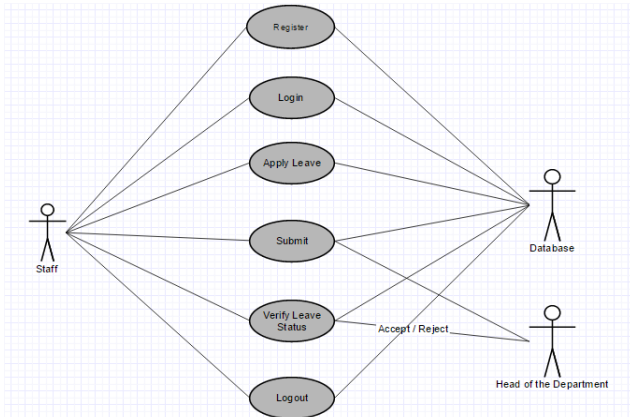


Fig:(a).use case diagram.

Class Diagram:

Class Diagram consists of the classes and the objects and the interaction between them. It mainly deals with the interaction between classes in the system, their behavior and properties of the system. Apart from classes this also provides inheritance relationships in the paper. Class diagrams consist of basically two parts: first one is the member variables and class variables and the second part consists of the total number of methods available in the class.

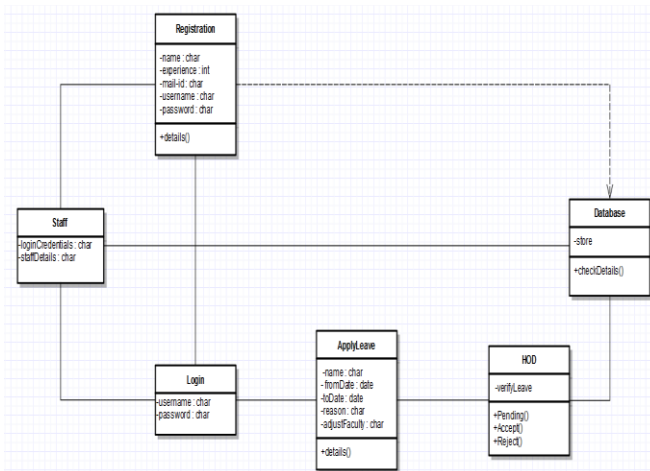


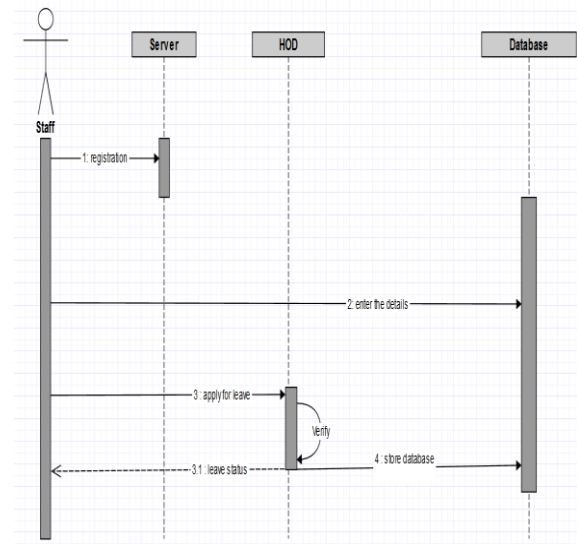
Fig:(b) Class diagram for dijarmania

Sequence Diagram:

The purpose of sequence diagram is to show the flow of functionality through a use case. In other words, we call it a mapping process in terms of data transfers from the actor through the corresponding objects.

The key points are:

- The main purpose is to represent the logical flow of data with respect to a process
- A sequence diagram displays the objects and not the classes.



5.2 BEHAVIOURAL DIAGRAMS:

What are Behavioral Diagrams?

Structural diagrams are state chart diagrams, activity diagrams, sequence diagram and collaboration diagrams.

- **Statechart diagram** represent the possible states and transitions of instances of a specific class.
- **Activity diagram** are like flowcharts. They diagram a set of possible user interactions. Individual examples of user interactions are written in use cases.
- **Sequence diagram** show example sequences of messages sent between instances. . Sequences diagram show instances in columns and messages in rows with time increasing down the length of the page.
- **Collaboration diagram** also show example sequences of messages sent between instances. But they number the messages. The diagram can be laid out any way you choose.

5.3 Activity Diagram:

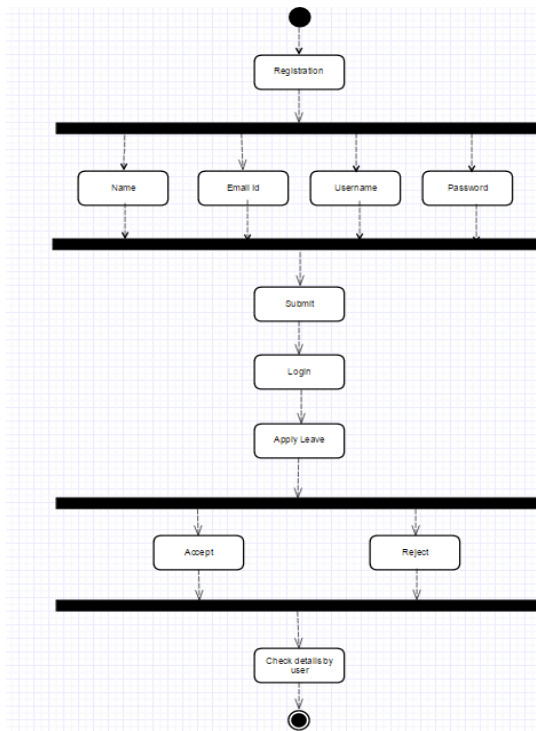


Fig:(c)Activity diagram for dijarmania
5.4.Component Diagram:

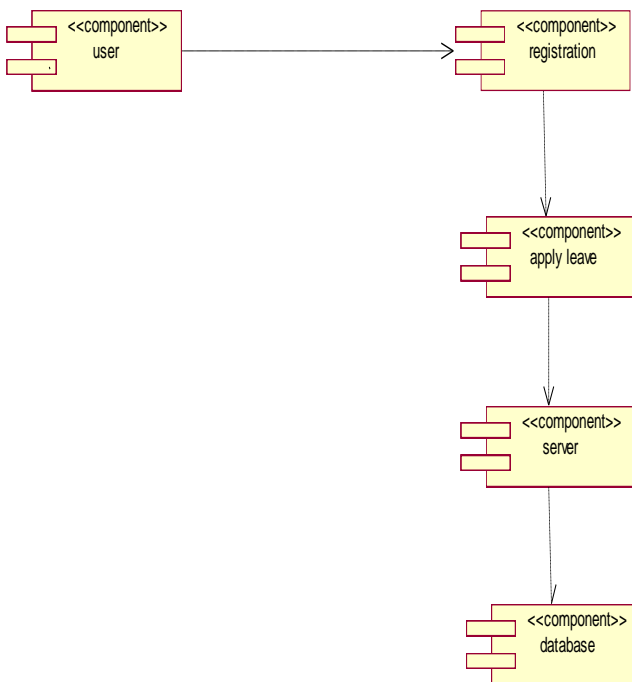


Fig: (d) Component diagram for dijar mania
5.5.Deployment Diagram:

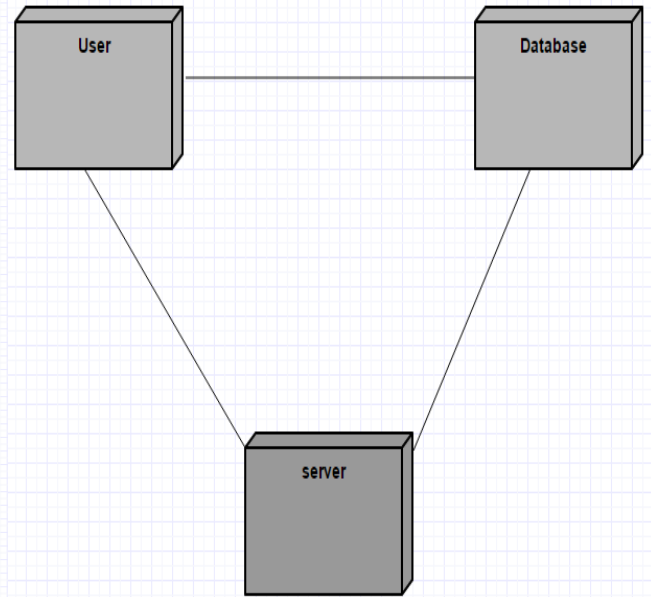


Fig: (e)Deployment diagram for dijarmania

VI. IMPLEMENTATION

6.1 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 fulfill 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 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 DLLs.

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.

6.2 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 and devices.
- Enable a new breed of web applications.

ActiveX

ActiveX is a specification developed 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 compiled 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.

6.3 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.

6.4 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 2005 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-yam" 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....

INFORMATION SUPER HIGHWAY:

A set of computer networks, made up of a large number of smaller networks, using different networking protocols. The world's largest computing network is consisting of over two million computers supporting over 20 million users in almost 200 different countries. The Internet is growing a phenomenal rate between 10 and 15 percent. So any size estimates are quickly out of date.

Internet was originally established to meet the research needs of the U.S Defense Industry. But it has grown into a huge global network serving universities, academic researches, commercial interest and Government agencies, both in the U.S and Overseas. The Internet uses TCP/IP protocols and many of the Internet hosts run the Unix Operating System.

6.5 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.

In either case the result is a plain text file that computers can easily exchange. The browser displays this text file on the client computer.

"Hypertext" is the jumping frog portion. A hyperlink can jump to any place within your own page(s) or literally to anyplace in the world with a 'net address (URL, or Uniform Resource Locator.) It's a small part of the html language.

VII. RESULT ALONG WITH TEST CASES

Apply Leave:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Web.UI;
using System.Web.UI.WebControls;
using System.Data;
using System.Data.SqlClient;
using System.IO;
```

```
namespace Dejar_Mania
{
    public partial class applyleave : System.Web.UI.Page
    {
```

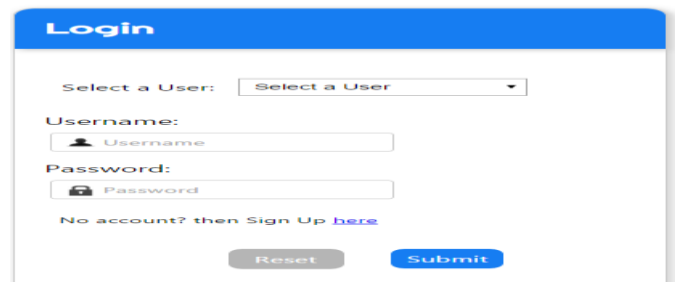
```
        protected void Page_Load(object sender, EventArgs
e)
        {

        }

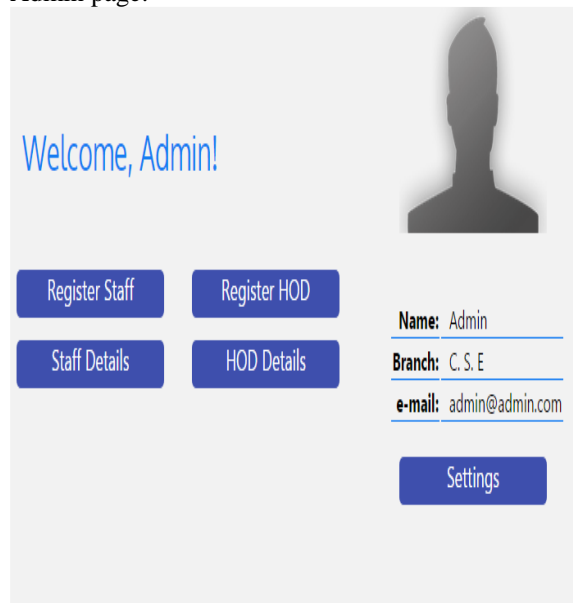
        protected void SubmitBtn_Click(object sender,
EventArgs e)
        {
            int i = 0;
            SqlConnection con = new SqlConnection("Data
Source = .;Initial Catalog =Dejar Mania ;Integrated
Security=SSPI");
            SqlCommand cmd = new SqlCommand("insert
into ApplyLeave values('" + FromDateTextbox.Text + ",
'" + ToDateTextbox.Text + ", '" + ReasonTextbox.Text +
",'" + AdjFacultyTextbox.Text + "')", con);
            con.Open();
            i = cmd.ExecuteNonQuery();
            con.Close();
        }
    }
}
```

7.2 SAMPLE SCREEN SHOTS:

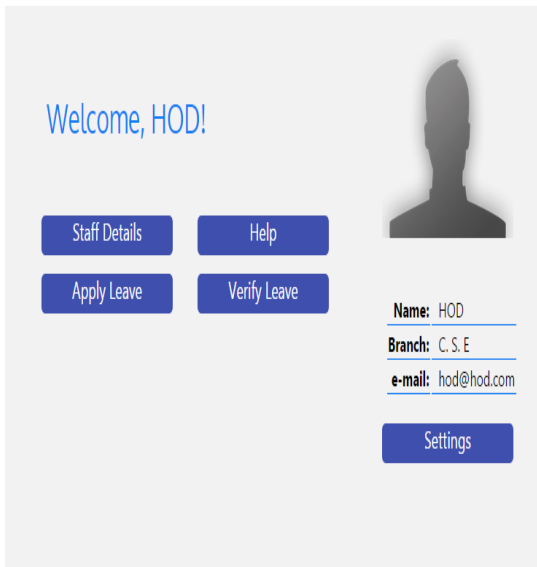
Login into your Account



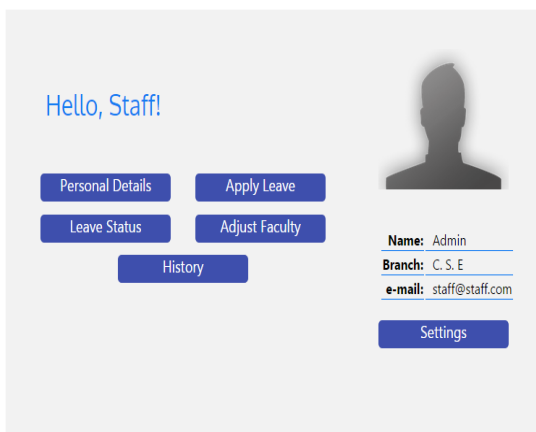
Admin page:



HOD welcome page:



Staff welcome page:



Apply leave page:

Details

From Date:

To Date:

Reason:

Adjusted Faculty:

VIII. CONCLUSION

From a proper analysis of positive points and constraints on the component, it can be safely concluded that the product is a highly efficient GUI based component. This component can be easily plugged in many other systems. Also the component is user friendly. Generally the TPO's of the Colleges has to face a lot of problems in management of the Students information. All this information has to be managed manually. So, there is a need to develop a system that can solve the mentioned problem. This software comes with just that solution.

REFERENCES

- [1] Jeff Poise (2002). Programming Microsoft .NET.
- [2] Asp.net general information and tutorials: <http://www.asp.net>
- [3] Building an ASP.NET Internet, Wrox publication Jeff Pose (2002). Programming Microsoft.Net. Microsoft Press.
- [4] Asp.net general information and tutorials: <http://www.asp.net>
- [5] Building an ASP.NET Intranet, Wrox publication
- [6] <http://msdn.microsoft.com/asp.net/>
- [7] Design Patterns C# by steven metsker.
- [8] The Unified Modeling Language- ByGrandy Booch.
- [9]. <http://msdn.microsoft.com/asp.net/>
- [10]. Design Patterns C# by steven metsker.
- [11]. The Unified Modeling Language- ByGrandy Booch

BIOGRAPHIES



Asst. prof. Mutyala Venu Gopal. He was born on 16-01-1983. received his M.Sc., in mathematics from Acharya Nagarjuna University, Guntur, AP, India in the year of 2005. He did M.Phil in the Specialization of Algebra in mathematics from Andhra University, Vishakhapatnam, AP and India. He worked as Asst. Professor of mathematics & HOD S & H in the Dept of Basic Science and Engineering in Sri Sarathi Institute of Engineering and Technology; Nuzvid, AP and India since 2006 to 2012. Present he is working as an asst professor of mathematics at lingays institute of management and technology, madalavari gudem. His research interests in Graph Theory, Data Structures, Lattice Theory, Formal Language, Auto mata Theory.



Mr.Nandyala Ravikumar.M.Sc---vikas p.g college. mca---marri laxman reddy institute of tech.dundigal ,rr dist,hyd. working in marri laxman reddy institute of tech.&management .



K.s.n.v.d.Sowjanya.Msc from vikas pg college visannapeta-2005.present she is worked as an Asst professor of mathematics at vikas college of engineering and technology.



Konolla Siva Rama Krishna received his B.Tech, in Computer science and engineering from Lingayas Institute of Management and technology, madhalavarigudem,AP,india in the year 2013.He is pursuing M.Tech in Lngayas Institute of mang and tech, madalavari gudem in the stream of Computer Science and Engineering was born on 18th April, 1992. He received a Bachelor Degree in Computer Science and Engineering from **Lingayas Institute of Management and Technology** under JNTUK in 2013.



Chikkavarapu Gnanendra Rao, (born on 31-1-1983 in nuzvid , Andhra Pradesh), son of father ch Krishna and mother chmangammagaru . he completed his post graduation in mathematics in P.B.Siddartha college of arts and science in Vijayawada in the year 2005 worked as a lecturer in vikas pg college (in vissannapeta for one year 2005 to 2006 for one sem only) and in nalanda educational institutions (in Vijayawada for one year from 2006 to 2007) and worked as an assistant professor of mathematics in Sir C.R.R college of engineering in Eluru for one year from 2007 to 2008 and working as assistant professor of mathematics in srisarathi institute of engineering and technology till now. He was qualified GATE 2013 with 93.71 percentile and APSET-2012 qualified person. He has nice idea of attempting the exams csir net and gate exams regularly.