

INTRODUCTION

1. INTRODUCTION

1.1. PROBLEM DEFINITION

We have developed a project for computerization of Leave Management System which deals with leave records and attendance system. In computerized system database can be stored for a longer duration and can be viewed at any time.

A leave management system is a web-based application system which is used to maintain and monitor leave records of faculties over the internet/intranet.

1.2. Leave Management System Introduction

Leave Management System enables you to get the status of the leave, view daily attendance record, and to apply for leave.

In the existing system of paper work related to leave management, leaves are maintained using the attendance register for staff. The staff needs to submit their leaves manually to their respective authorities. This increases the paperwork & maintaining the records becomes tedious. Maintaining notices in the records also increases the paperwork. 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. This module intelligently adapts to HR policy of the management & allows employees and their line manager's to manage leaves and replacements for better scheduling of workload.

The application basically contains the given modules:

Module:-

- 1) STAFF MODULE: It consists of two types of faculties a) Teaching b) Non-teaching.
- 2) HOD MODULE: It consists of Head of the Department/ Manager Body which works the same as Staff Module.
- 3) PRINCIPAL MODULE: This module has the highest authorities in the system, they can take corrective action against HOD, Staff.

GATHERING INFORMATION

2. LITERATURE SURVEY

2.1 Literature survey

We have developed a project named “Leave Management System”.it is related to the process of keeping all leave details of every faculty. In this system the administrator has to login into the system using username and password technique.

Then the administrator takes attendance of each and every faculty by using attendance form. It can be fill all the criteria related with Leave management.

This system is a computerized system. The manual system maintain the record of each faculty and generate their information manually.it is very difficult and more time consuming process and also it does not guaranteed about the accuracy, security of the data because any third person can access the data, in the manual system and once losses the data cannot be get back as backup of the data is not possible in the manual system. This can degrade the performance of the system. There were many systems that demands computerization but were not being covered by the manual system.

In the manual system the user cannot get fast services, but by computerization it is possible. To solve all the problem above computerization is necessary.

2.2. Proposed System

The proposed system automates the existing sytem.it decreases the paper work and helps making easier record by having database for leave maintenance.

The Leave Management System reduces the over work and time duration. It provides an easy, maintained and systematic environment for higher panel (HOD/Principal) for the approval of leave.

The Leave Management system will provide a connection between the user and the database it will allow faculties and principal to manage their leaves and mark their leaves dates.

2.3. Benefits of the System

1. Reduces paperwork, saves time, options to customize.
2. Access from anywhere.
3. Information at a central place.
4. Accurate and Precise Information.

SOFTWARE REQUIREMENT SPECIFICATION

3. SOFTWARE REQUIREMENTS and SPECIFICATION

3.1. Introduction

This document aims to define the overall software requirement for “**Leave Management System**” for calculating, monitoring leaves and applying for leaves.

3.1.1. Purpose

The purpose of developing Leave management system is to computerized the traditional way of taking attendance and requesting leaves. Another purpose for developing this software is to generate the report automatically and the faculties will be able to see their leave status.

3.1.2. Scope

The scope of the project is the system on which the software is installed, i.e. the project is developed as a web based application, and it will work for a particular institute.

3.1.3. Overview

The rest of SRS document describes the various system requirement features and functionalities in detail.

3.2. OVERALL DESCRIPTION

The system works as per leave request given by Staff, HOD to Principal.

3.2.1. User classes and characteristics

The LEAVE MANAGEMENT SYSTEM is designed to benefit the user. With the help of this software the user can monitor, calculate leaves and generate report automatically.

3.2.2. Design and implementation constraints

The leave which is requested by user is send to the HOD, then HOD take the respective action on the respective leave.

3.2.3. User Documentation.

User manual for administrator

3.3. EXTERNAL INTERFACE REQUIREMENTS

3.3.1. Hardware interfaces

- a) Intel(R) Core™ i3 2.00 GHz (Min).
- b) True color video card-800X600, 24-Bit color.
- c) 1 GB RAM (Min).
- d) Microsoft Windows® 7 and Above.
- e) Secured Browser, IAS Server.

3.3.2. Software interfaces

- a) Any windows based 32 /64 bit operating system.
- b) .NET for coding / developing the application.
- c) MS SQL Server2008 for the management of the database.

3.4. SYSTEM FEATURES

3.4.1. Functional requirements

It defines the fundamental actions that must take place in the software for accepting and processing the inputs and generating the outputs.

3.4.2. Validity checks

The application should be able to check the validity for inputs.

3.4.3. Action/ Result

The required information about the Leave is as follows

- a) Leave request must be send to Principal.
- b) At the end must generate the report.

3.5 OTHER NON FUNCTIONAL REQUIREMENTS

3.5.1. Safety requirements

- a) System backup should be maintained.
- b) The System should display proper error messages in case of an error.
- c) The system should have login and logout facility.

3.5.2. Security requirements

- a) The administrator should have the username and password.
- b) The Principal, HOD, Staff should have the log-in.
- c) Do not expose your password to anyone.

SYSTEM DESIGN

4. SYSTEM and DESIGN:

The current application is being developed by taking the 3-tier architecture as a prototype. The 3-tier architecture is the most common approach used for web applications today. In the typical example of this model, the web browser acts as the client, IIS handles the business logic, and a separate tier MS-SQL Server handles database functions.

Although the 3-tier approach increases scalability and introduces a separation of business logic from the display and database layers, it does not truly separate the application into specialized, functional layers. For prototype or simple web applications, the 3-tier architecture may be sufficient. However, with complex demands placed on web applications, a 3-tiered approach falls short in several key areas, including flexibility and scalability. These shortcomings occur mainly because the business logic tier is still too broad- it has too many functions grouped into one tier that could be separated out into a finer grained model.

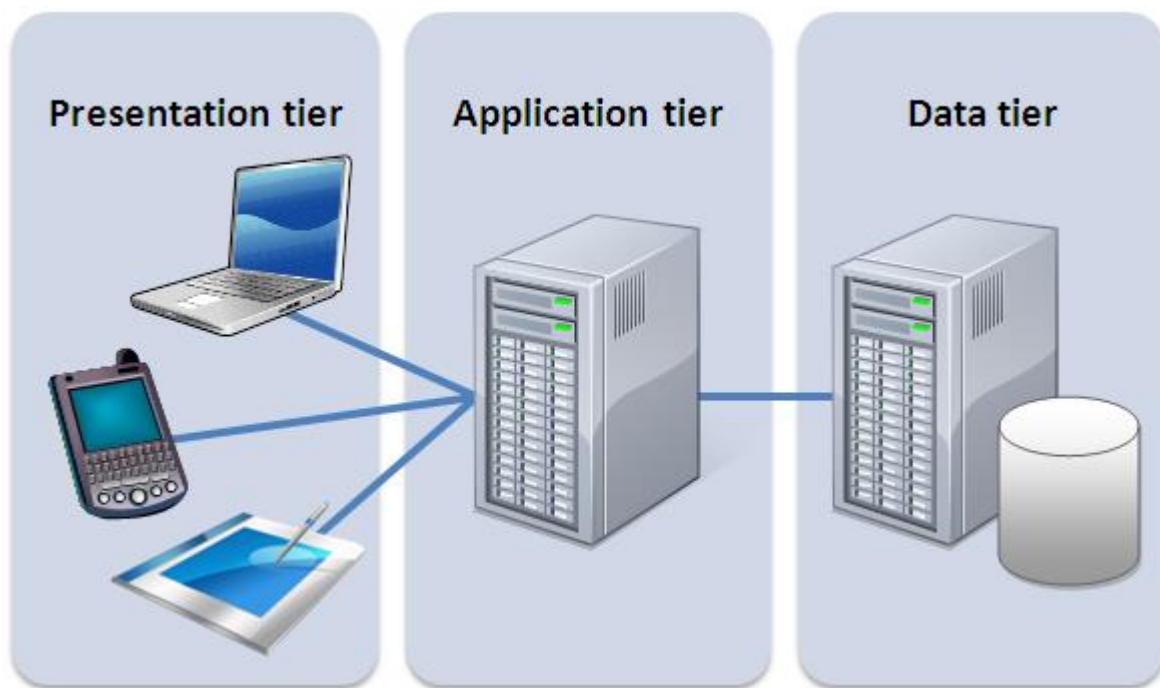


Figure 4: Three-Tier Architecture

- a) Tier 1: the client contains the presentation logic, including simple control and user input validation. This application is also known as a thin client. The client interface is developed using ASP.Net Server Controls and HTML controls in some occasions
- b) Tier 2: the middle tier is also known as the application server, which provides the business processes logic and the data access. The business logic/ business rules can be

written either with C#.Net or VB.Net languages. These business runtimes will be deployed as DLL's in IIS web server.

- c) Tier 3: the data server provides the business data. MS-SQL server acts as Tier-3, which is the database layer.

These are some of the advantages of three-tier architecture:

- It is easier to modify or replace any tier without affecting the other tiers.
- Separating the application and database functionality means better load balancing.
- Adequate security policies can be enforced within the server tiers without hindering the clients.

The proposed system can be designed perfectly with the three tier model, as all layers are perfectly getting set as part of the project. In the future, while expanding the system, in order to implement integration touch points and to provide enhanced user interfaces, the n-tier architecture can be used.

4.1. PROJECT OVERVIEW:

-Fig.4.1 gives brief overview of project showing main element in project functioning.

-This is a web based application.

-This is a .NET project.

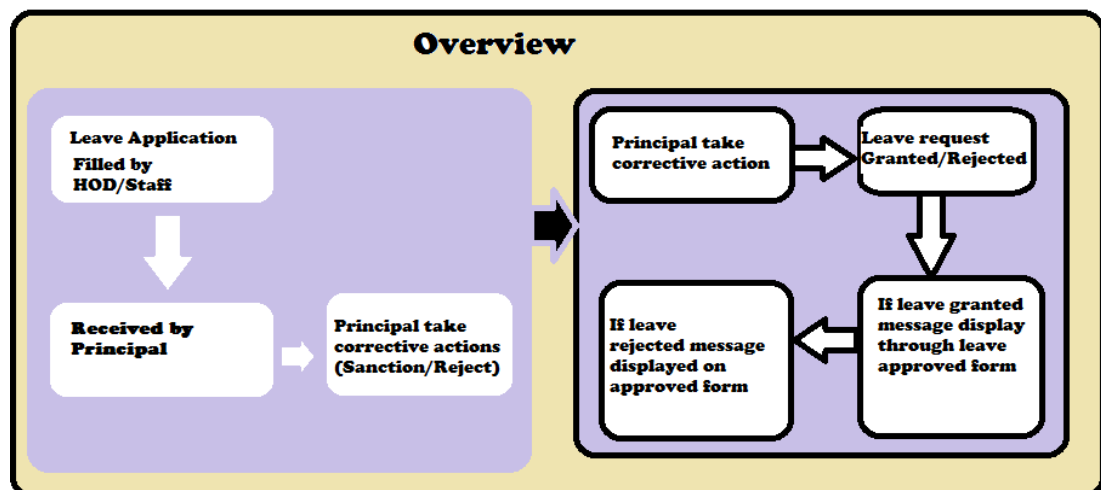


Figure 4.1: Leave Management System overview

Leave Management System consist of two major parts first is leave request through application and second is leave response to HOD,Staff . Leave application means creating an application for leave request by filling information on leave application page. After creating an application send to the HOD/Principal with the help of intranet.

Except leave request and leave response in Leave Management System contain leave report generation on the basis of leaves taken by user, also user can see his remaining leaves, sanctioned leaves, rejected leaves

PROJECT PLAN

5. PROJECT PLAN

Before developing any software system we need to plan its steps systematically. There are various methods of process of software development like waterfall model, rapid application development model, incremental model, spiral model, fourth generation techniques etc.

- Requirement analysis
- Planning
- Design & analysis
- Coding
- Testing

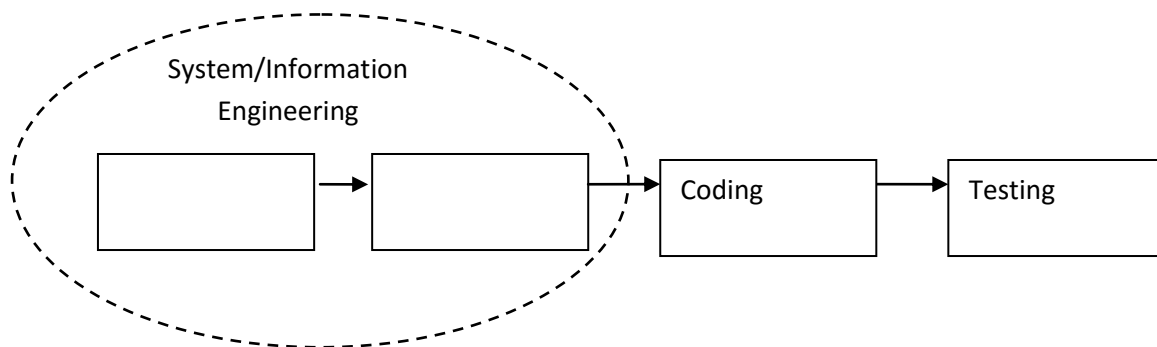


Figure 5.1 Software Development Model

This model is called as linear sequential model or waterfall model suggest a systematic sequential approach to software development that begun at the system level and progress through analysis, design, coding and testing.

5.1. PROJECT PLAN

5.1.1 System Information engineering and modeling

Because software is always a part of larger system, work begins by establishing requirements to software. This system view is essential when software must interact with other elements such as hardware people and database. System engineering and analysis encompass requirements gathering at the system level with a small amount of top level design and analysis. Information engineering encompass information gathering at the strategic business level and at the business area level.

5.1.2 Software Requirement Analysis

The requirement gathering process is intensified and focused specially on software. To understand the nature of the program to be built, the software engineer must understand the information domain of the software, as well as required function, behavior, performance and interface.

5.1.3 Design

Software design is actually a multi-step process that focuses on 4 distinct attributes of program data structures, software architecture, interface representation and procedural details. The design process translates requirement into representation of software configuration.

5.1.4 Code Generation

The design must be translated into machine readable form. The code generation step performs this task. If design is performed in detailed manner, code generation can be accomplished mechanically.

5.1.5 Testing

Once code has been generated, program testing begins. The testing process focuses on the logical internals of the software, ensuring that all statements have been tested on the functional externals that is conducting test to uncover errors and ensure that the defined input will produce actual results that agree with required results.

5.2 PHASES

5.2.1 Project Plan of our project

Table 5.2.1.1: Project plan

Phase	Task	Description
Phase 1	Analysis	Analyze the current scenario about Leave Management System
Phase 2	Literature survey	Collect raw data and elaborate on literature surveys. Study the current case.
Phase 3	Design	Assign the module and design the process flow control.
Phase 4	Implementation	Implement the code for all the modules and integrate all the modules.
Phase 5	Testing	Test the code and overall process whether the process works properly.
Phase 6	Documentation	Prepare the document for this project with conclusion and future enhancement.

5.2.2 Survey on Phase Diagram

Table 5.2.2.1: Phase Diagram

<div> <div>Date</div> <div>Phase</div> </div>	Aug 10 2016	Oct 23 2016	Nov 04 2016	Dec 30 2016	Jan 11 2017	Feb 10 2017	Mar 21 2017
Phase 1							
Phase 2							
Phase 3							
Phase 4							
Phase 5							
Phase 6							

5.3 PROJECT SCHEDULE

Table 5.3.1: Project Schedule

Task to be Accomplished	No of Days	Start Date	End Date
Understanding the concept of Leave Management System.	20	05 th August, 2016	25 th August, 2016
Studying different existing software's for Leave Managent System	15	2 ^t Sept	17 th Sept
Drafting Problem Statement and Software Requirement Specifications	08	22 st Sept	30 th Sept
Downloading research papers and studying them.	20	5 th Dec	25 th Dec
collected material as per the project requirement	15	01 st Jan2017	15 th Jan 2017
Project Report Stage – I	10	15 th January	25 th January
Implementation of 1 st module	25	25 th Jan	5 th Feb
Implementing of 2 nd module	20	5 th Feb	25 th Feb
Testing of the system	15	25 th Feb	11 th March
Documentation	10	11 th March	21 nd March

IMPLEMENTATION

6. IMPLEMENTATION:

6.1. USE CASE DIAGRAM:

There is one also separately administrator who will start up and shut down the system. The staff request for leave, an administrator can manage each and every faculties attendance.. And this attendance can be viewed by every faculty in isolated manner.

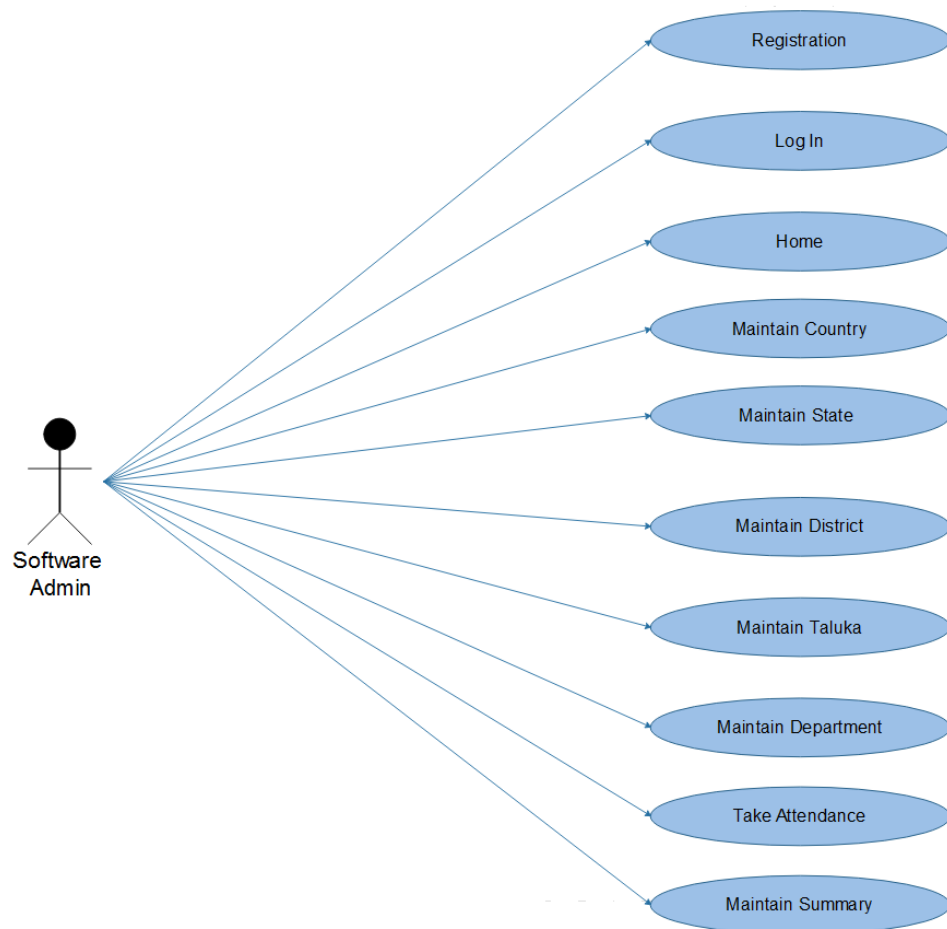


Figure 6.1.1 Use case diagram For Admin

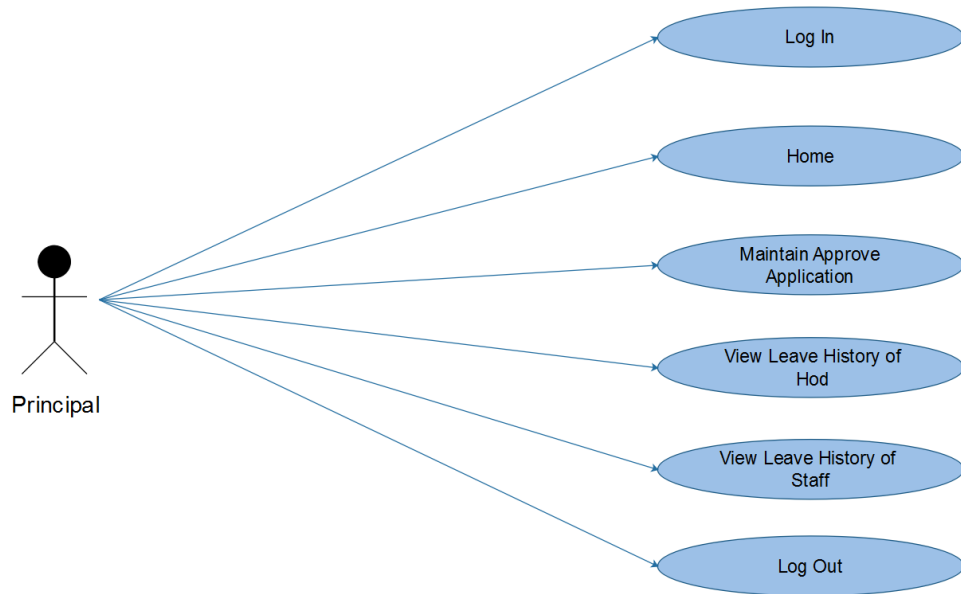


Figure 6.1.2 Use case diagram For Principal

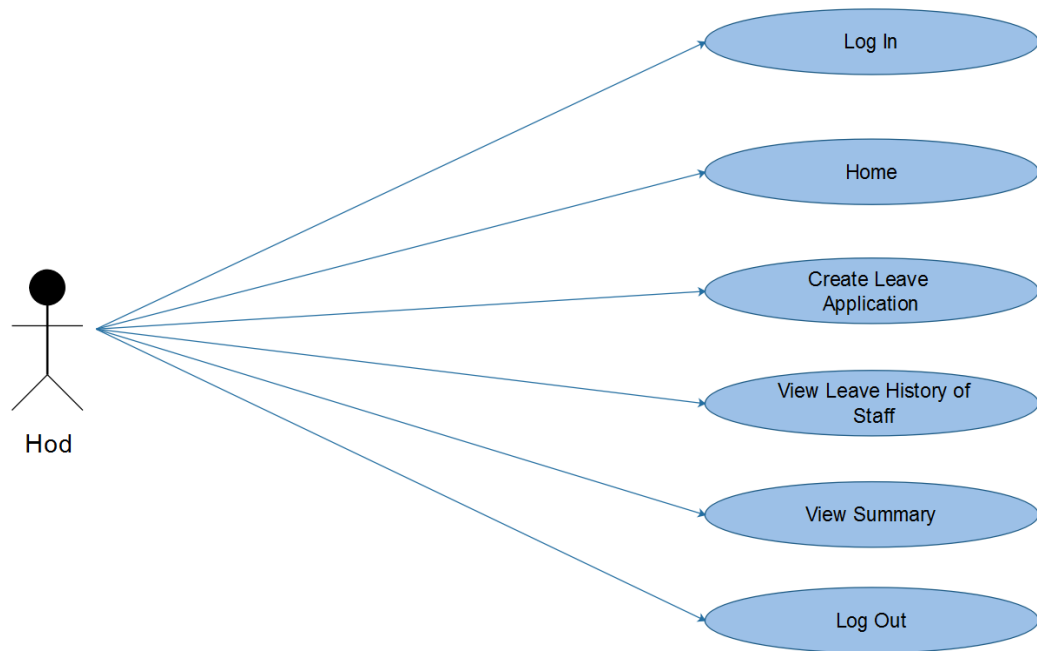


Figure 6.1.3 Use case diagram For HOD

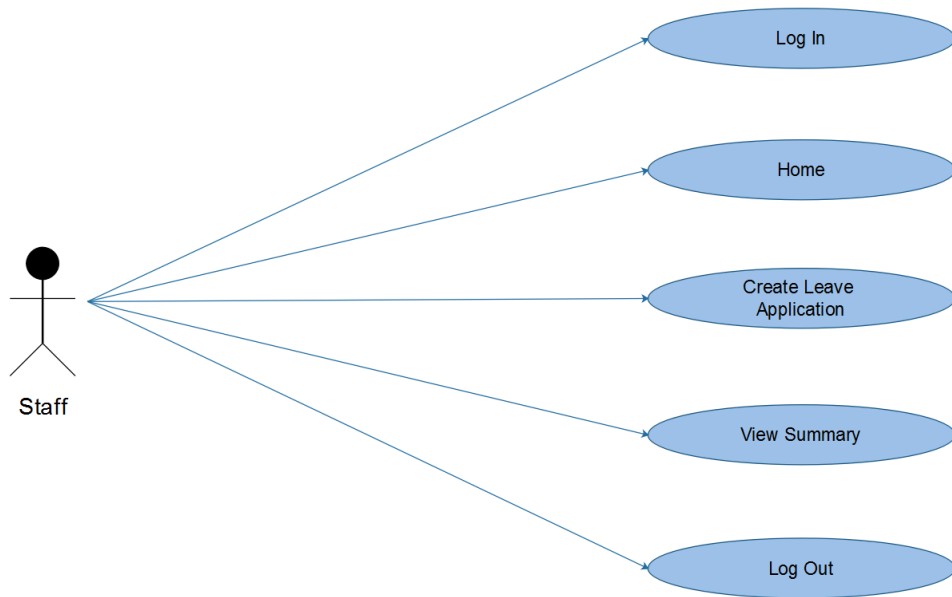


Figure 6.1.4 Use case diagram For Staff

6.2. CLASS DIAGRAM

There are four classes Admin, Staff, Principal and HOD .the staff and HOD class is dependent on Principal class. Whether the Admin, Principal are independent class refer Figure 6.2.

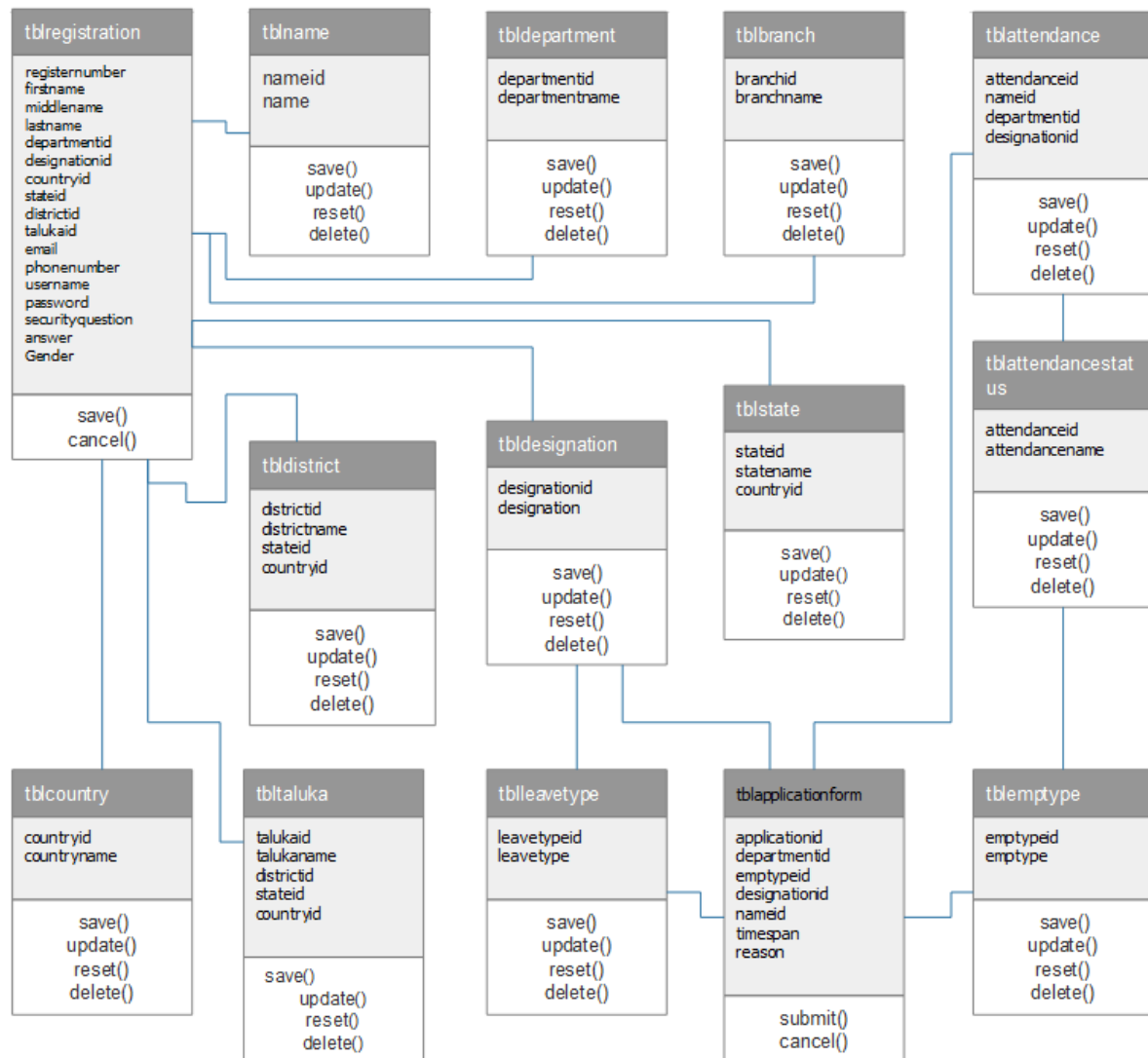


Figure 6.2 Class diagram

6.3. SEQUENCE DIAGRAM

Sequence Diagrams Represent the objects participating the interaction horizontally and time vertically. A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart. diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

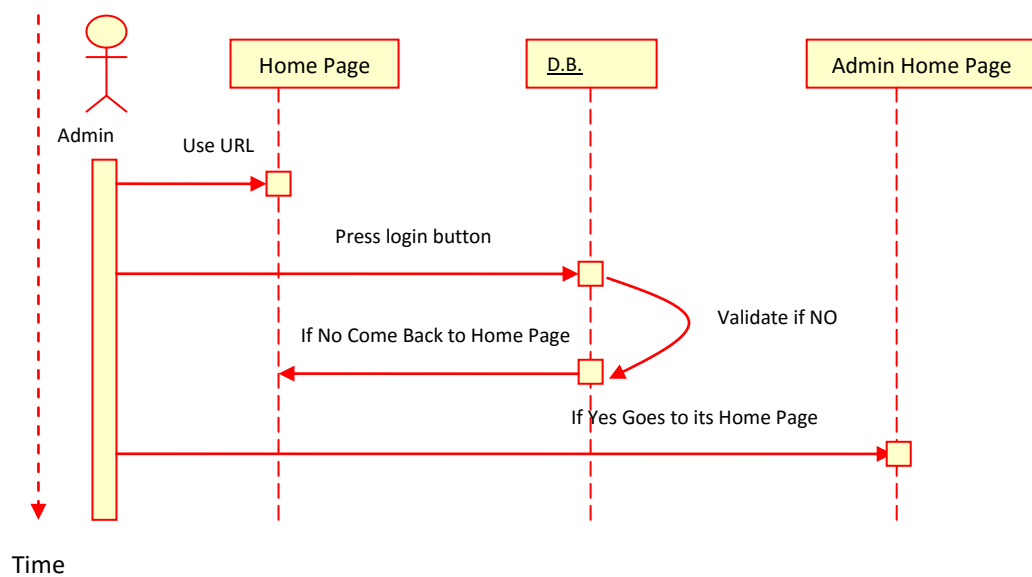


Figure 6.3.1 Sequence diagram for Admin login.

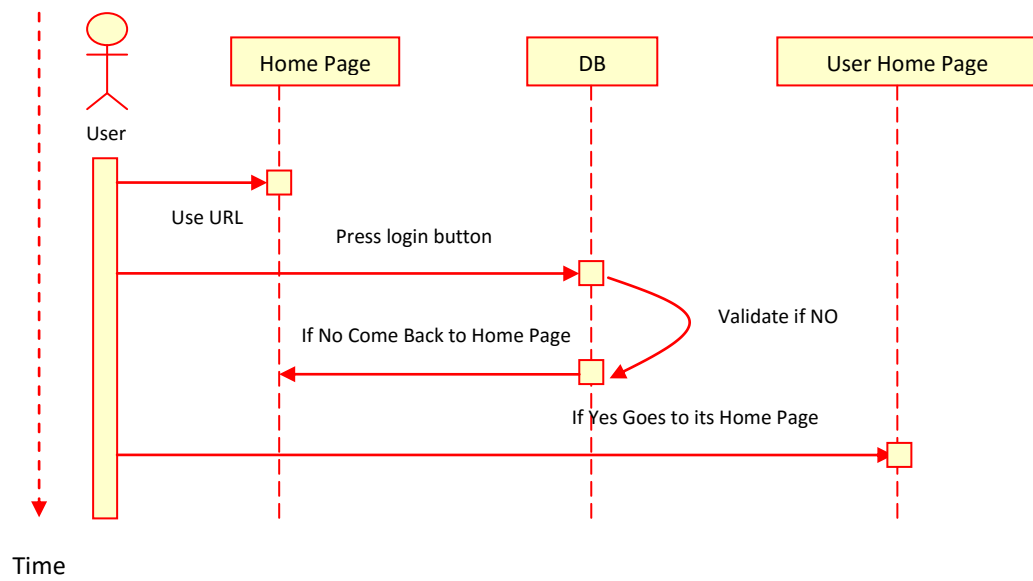


Figure 6.3.2 Sequence diagram for User .

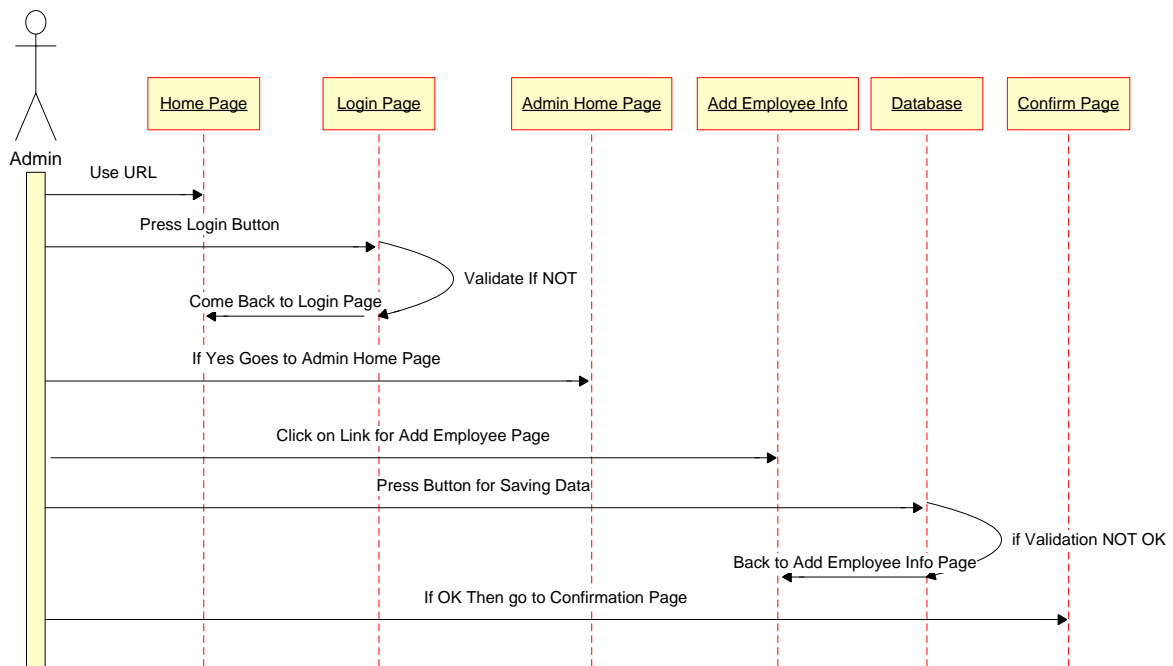
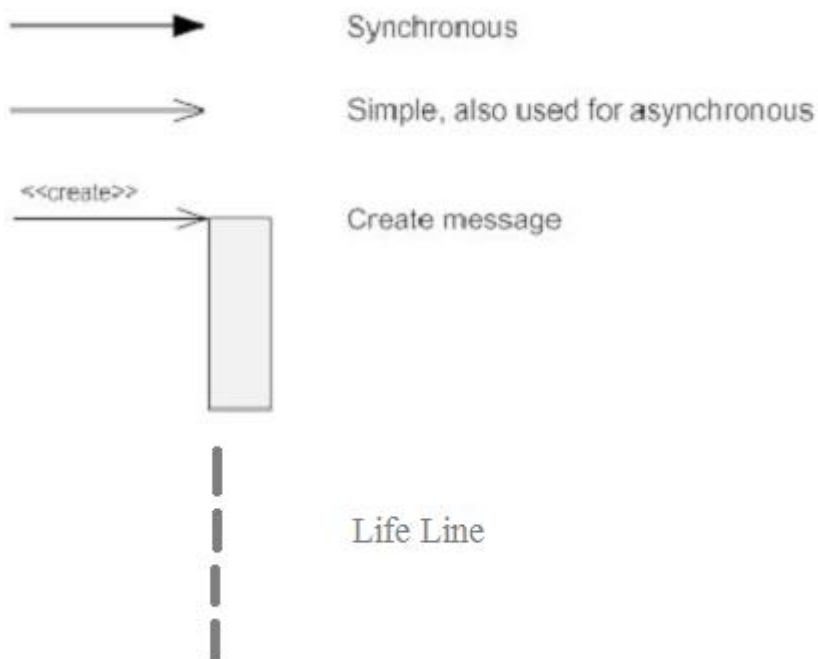


Figure 6.3.3 Sequence diagram for Admin login.



6.4. ACTIVITY DIAGRAM

Activity diagram shows the various activities that are performed. Admin can turn on and off the system. The Staff, HOD will request for leave and then principal can take corrective actions against them. The output of the leave request will show in approval form. The user can also cancel the leave request through cancelation button which is exist on Application form.

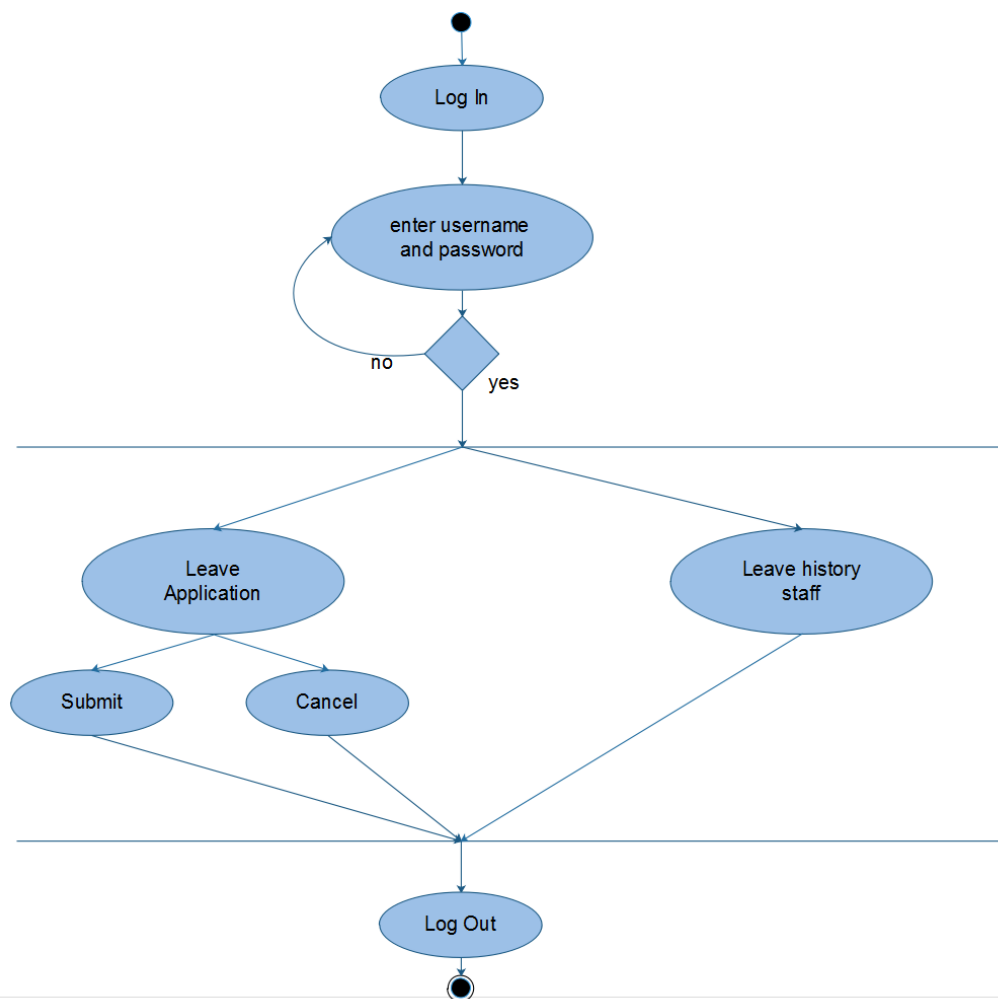


Figure 6.4.1: Activity diagram for Staff

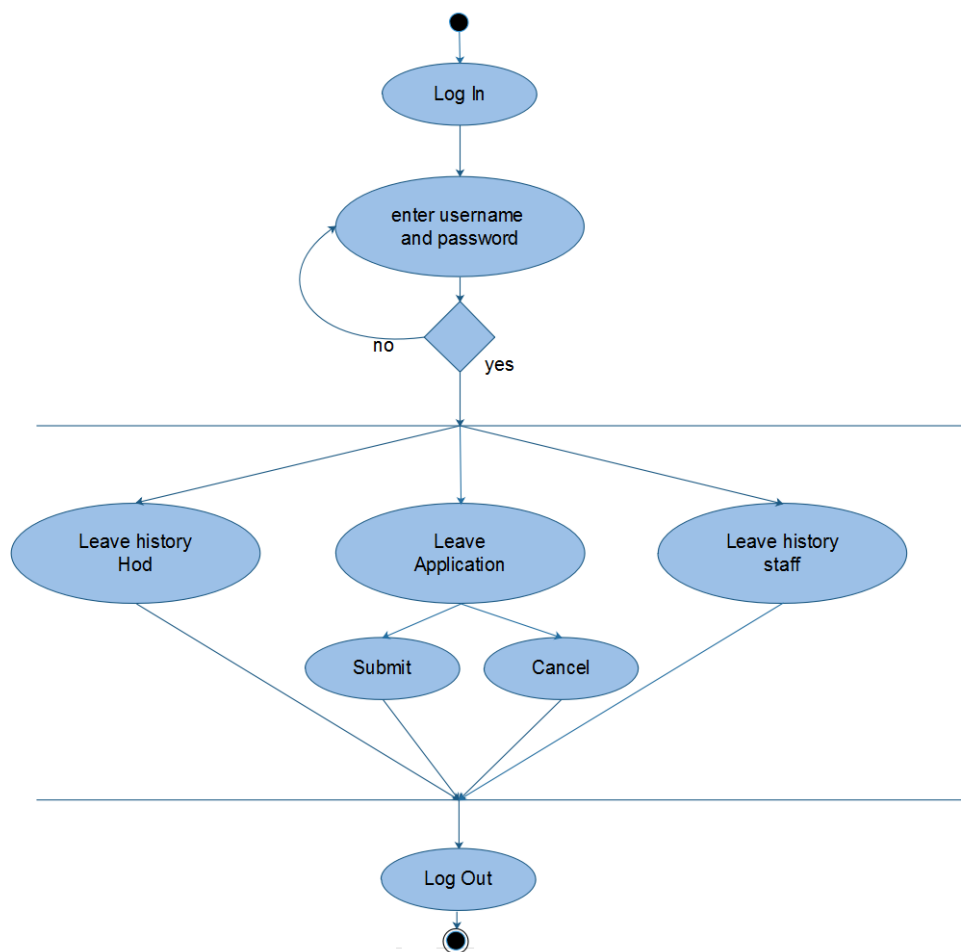


Figure 6.4.2: Activity diagram for HOD

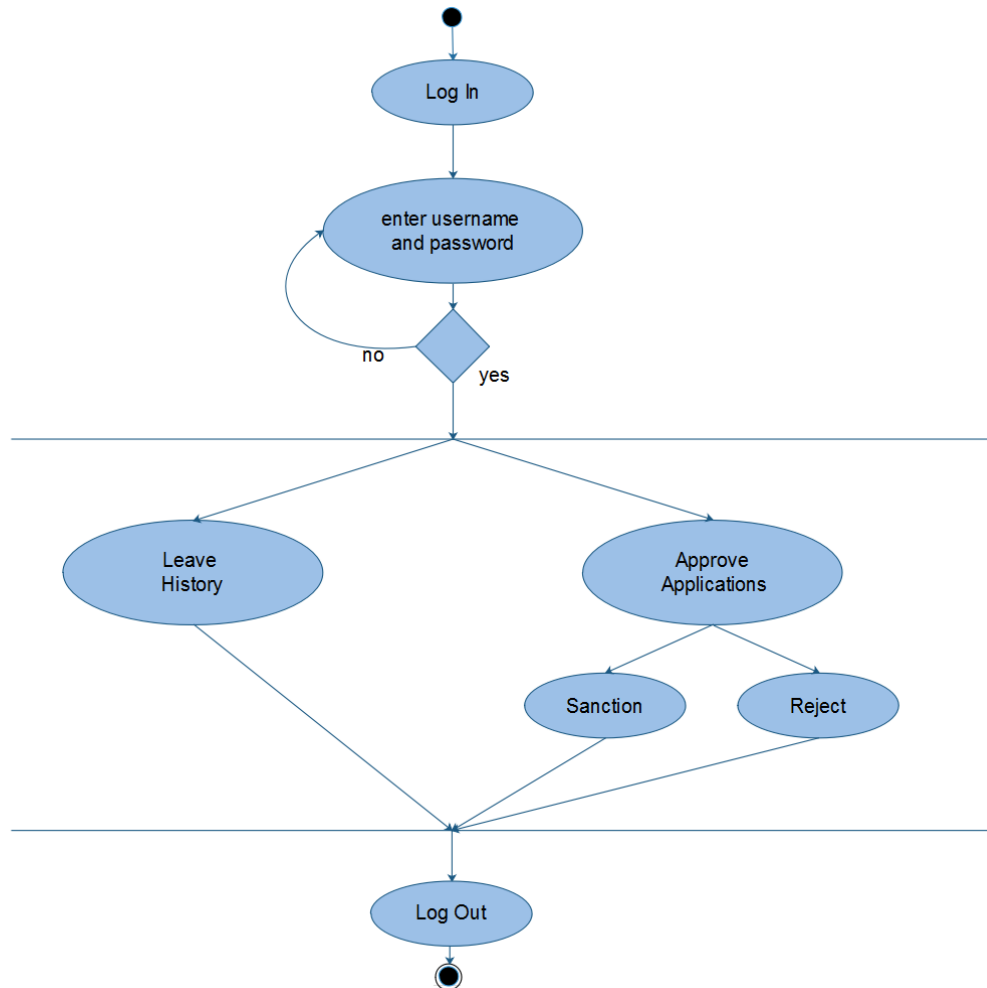


Figure 6.4.3: Activity diagram for Principal

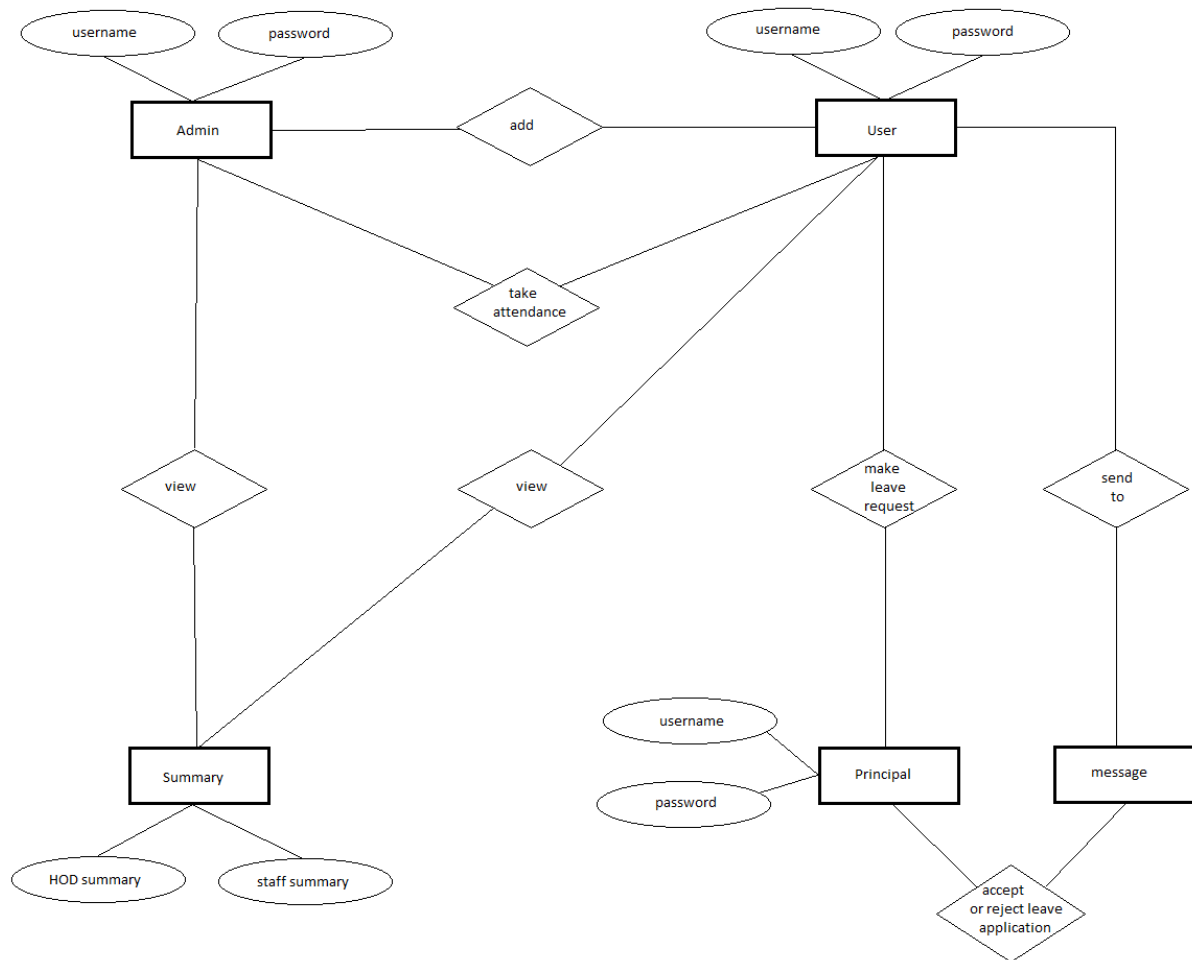
6.5. ENTITY RELATIONSHIP DIAGRAM.

Figure 6.5: Entity Relationship diagram

6.6. COMPONENT DIAGRAM

User is the provider interface and system is the required interface. User gives the whole database to the system. Images are the provider interface to the morphed algorithm and morphed algorithm is required interface. It takes image from images.

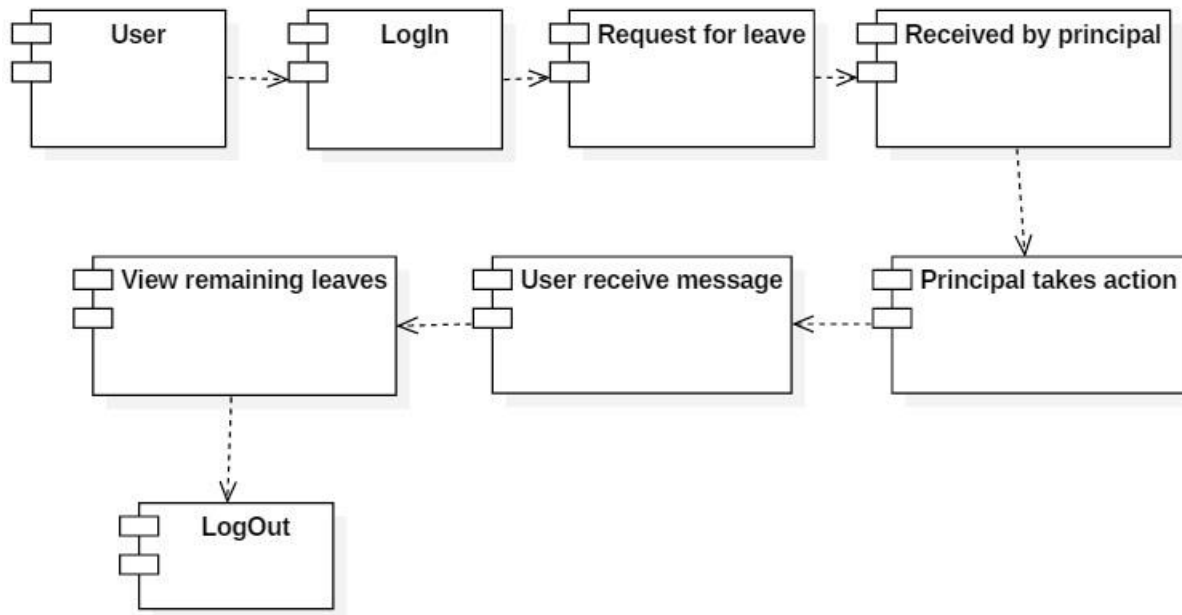


Figure 6.6: Component diagram

6.7. DFD DIAGRAM

6.7.1 Level 0 DFD:

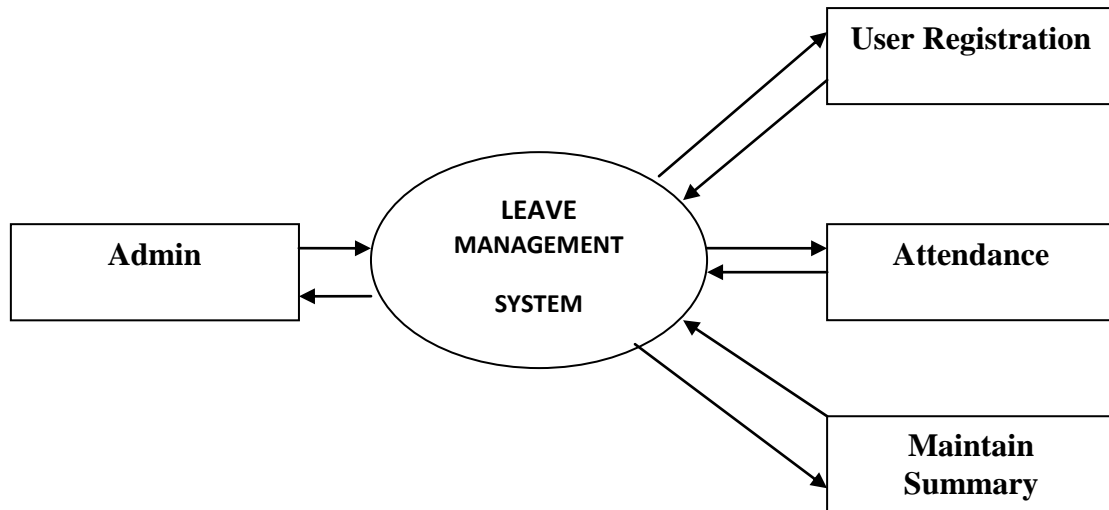


Figure 6.7.1.1: DFD 0 diagram for Admin.

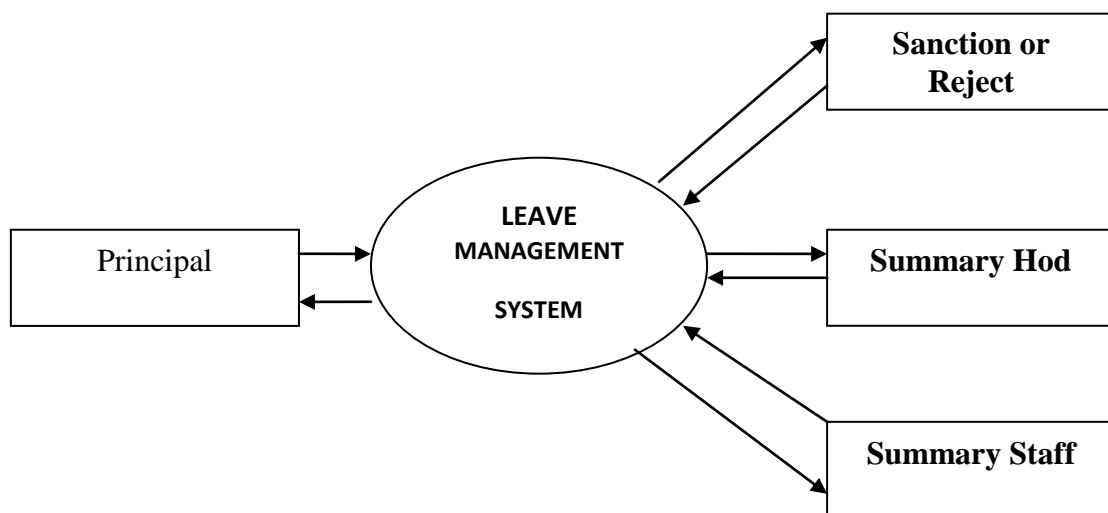


Figure 6.7.1.2: DFD 0 diagram for Principal.

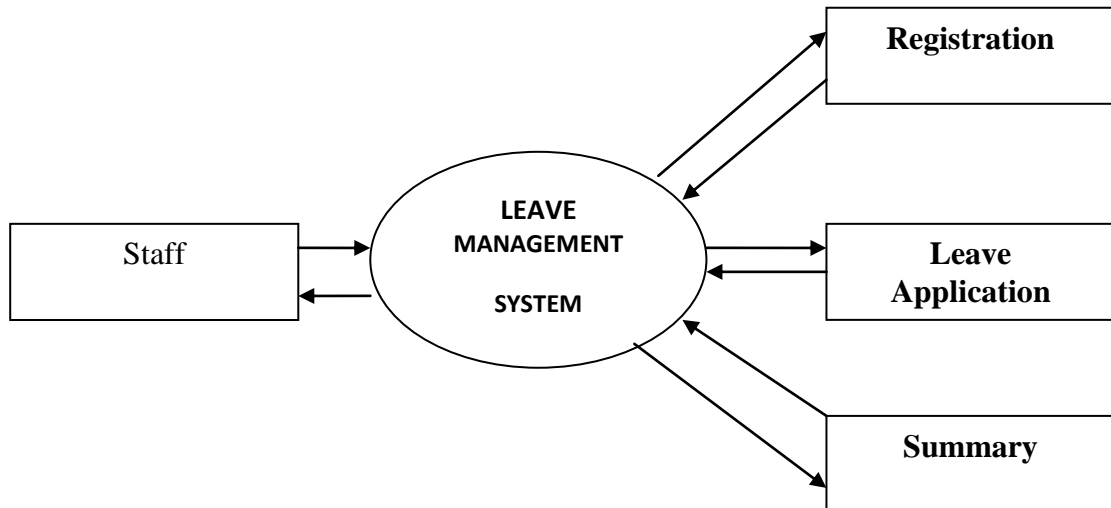


Figure 6.7.1.3: DFD 0 diagram for Principal.

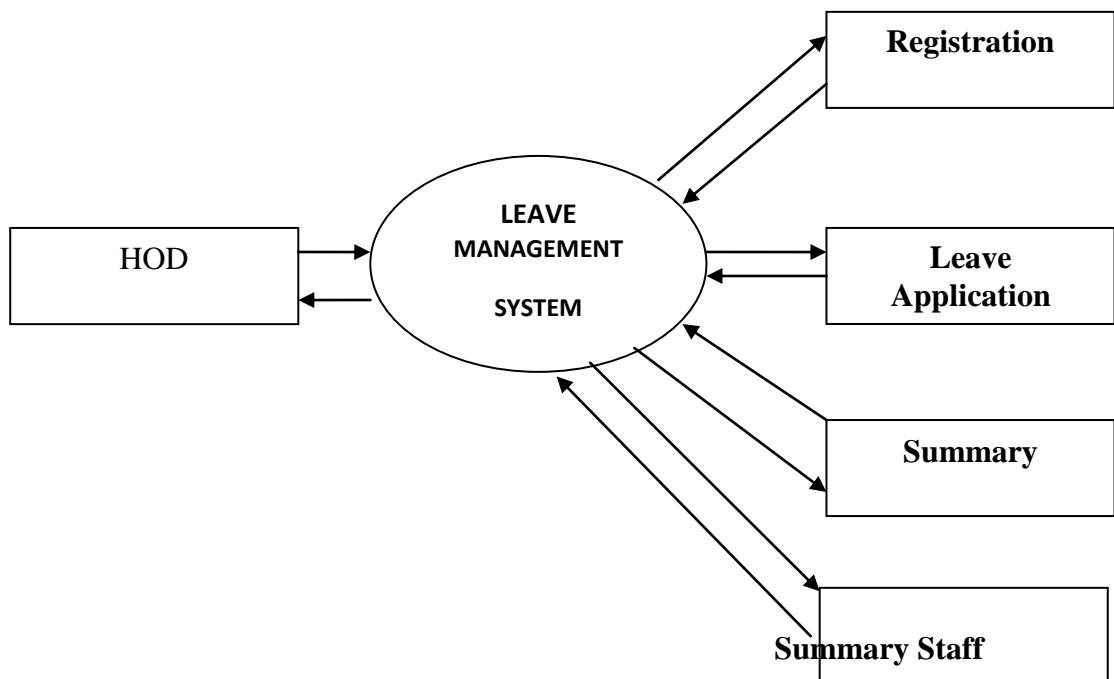


Figure 6.7.1.4: DFD 0 diagram for HOD.

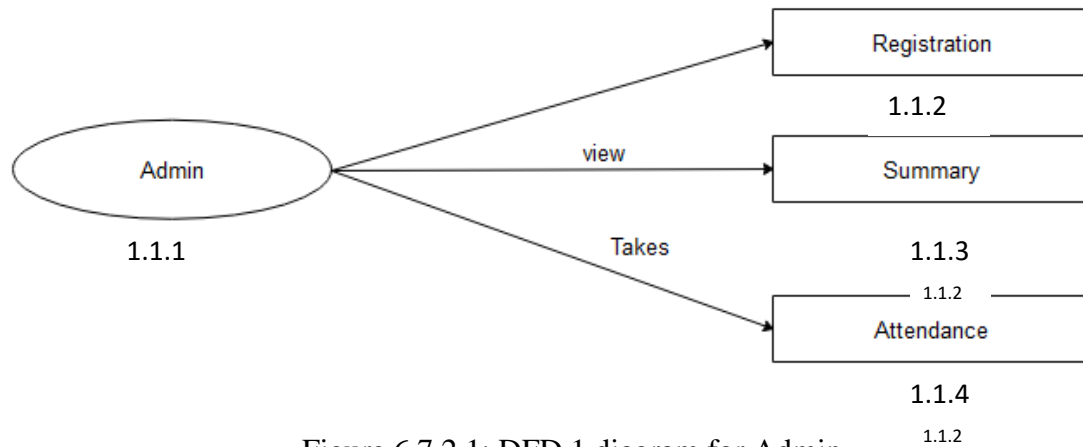
6.7.2 Level 1 DFD:

Figure 6.7.2.1: DFD 1 diagram for Admin.

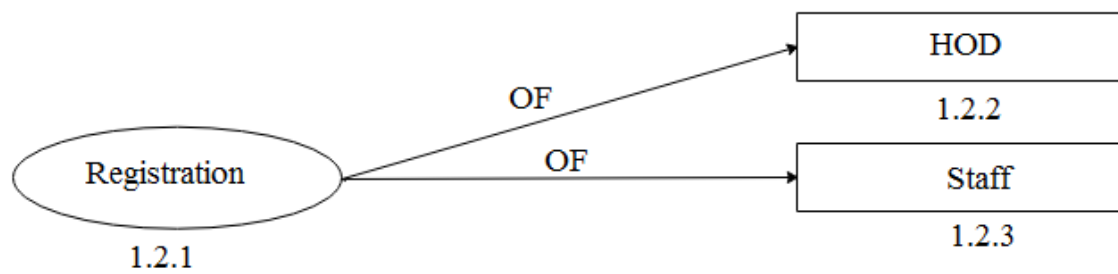


Figure 6.7.2.2: DFD 1 diagram for Registration.

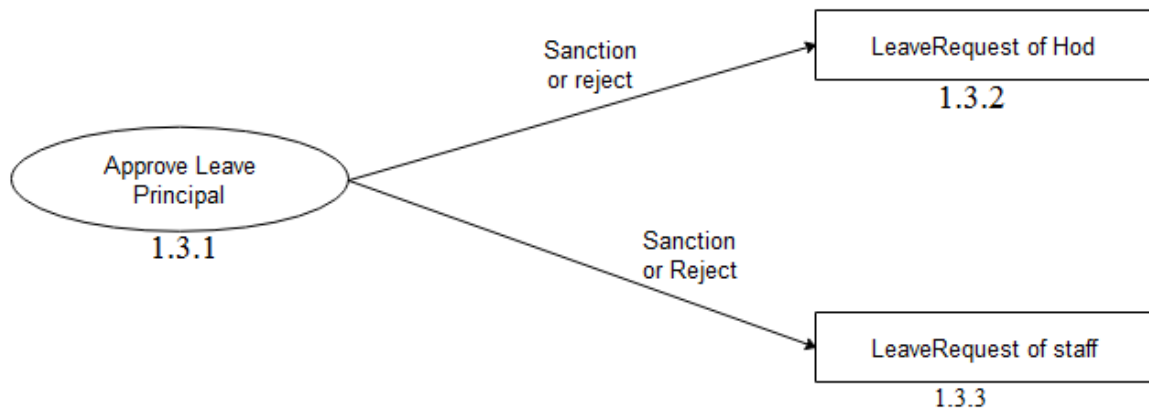


Figure 6.7.2.3: DFD 1 diagram for Approval of Application.

6.8. DEPLOYMENT DIAGRAM

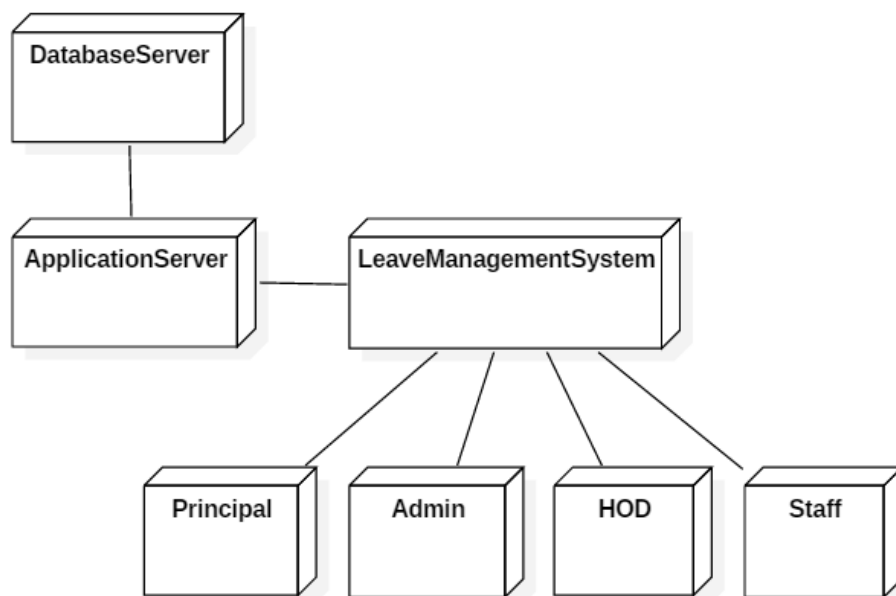


Figure 6.8: Deployment diagram

6.9. SNAPSHOTS.

6.9. 1. about us.

The screenshot shows the 'about' page of the Leave Management System. The header is teal with the system name and navigation links. The main content area is white with a blue 'about' link and a paragraph describing the system's purpose.

leave management system

Home About Master Transaction HOD Principle

[about](#)

Leave Management System is a web-based System that streamlines communication between staffs and facilitates simple yet efficient management of staffs leave. For staffs, this system allows them to easily determine their vacation availability (from any point-in-time) and request leave online. For managers, leave approval no longer involves trails of paperwork. e-leave ensures that all leaves taken and leave requests are properly monitored, where everybody can apply and view leave records without the hassle and delay of going through the admin staff - it's all at your fingertips with anytime, anywhere access...

6.9. 2. Registration Form:

The screenshot shows the 'Registration' form in the Leave Management System. The header is teal with the system name and navigation links. The form is divided into two columns with various input fields for user registration.

leave management system

Home About Master Transaction HOD Principle

Registration

Register number	<input type="text"/>	Middle Name	<input type="text"/>
First Name	<input type="text"/>	Date of Birth	<input type="text"/>
Last Name	<input type="text"/>	Merital Status	<input type="radio"/> Single <input type="radio"/> Married
Gender	<input type="radio"/> Male <input type="radio"/> Female	Designation	HOD <input type="text"/>
Department	Electronics&Telecommunication <input type="text"/>	State Name	Maharashtra <input type="text"/>
Country	India <input type="text"/>	Taluka Name	Nagar <input type="text"/>
Ditric Name	Ahmednagar <input type="text"/>	Phone No	<input type="text"/>
Email	<input type="text"/>	Password	<input type="text"/>
User Name	<input type="text"/>	Answer	<input type="text"/>
Security Question	<input type="text"/>		

Save Cancel

6.9. 3. LogIn Form.

The screenshot shows the login interface of the Leave Management System. At the top, there is a teal header bar with the system name 'leave management system' on the left and a navigation menu with links: Home, About, Master, Transaction, HOD, and Principle. Below the header is a light green horizontal bar. The main content area is white and contains the following elements:

- A 'log in' link in blue text.
- A message: 'Please enter your username and password. [Register](#) if you don't have an account.'
- A section titled 'Account Information' with two input fields: 'Username:' and 'Password:'.
- A checkbox labeled 'Keep me logged in'.
- A green 'Log In' button.

6.9.4. Attendance Form

The screenshot shows the attendance form interface of the Leave Management System. It features the same teal header bar with the system name and navigation menu as the login form. Below the header is a light green horizontal bar. The main content area is white and contains the following elements:

- A section titled 'Attendance'.
- Four input fields: 'Date', 'Name', 'Department' (a dropdown menu with '-select-' selected), and 'Designation' (a dropdown menu with '-select-' selected).
- Four green buttons: 'Save', 'Update', 'Reset', and 'Delete'.

6.9.5. Application Form.

leave management system

[Home](#) [About](#) [Master](#) [Transaction](#) [HOD](#) [Principle](#)

Application Form

Leave Date

Department

--select--

Employee Type

--select--

Designation

--select--

Employee Name

--select--

Time Span

to

Reason

Submit

Cancel

6.9. Masters Forms:

A) Master Form Name:

leave management system

[Home](#) [About](#) [Master](#) [Transaction](#) [HOD](#) [Principle](#)

Name

Name

Save

Update

Delete

Reset

Select	name	Edit
<input type="checkbox"/>	karad P M	Edit
<input type="checkbox"/>	Sayyed Noor	Edit
<input type="checkbox"/>	Bhawale ps	Edit
<input type="checkbox"/>	Inamdar Asim	Edit
<input type="checkbox"/>	Rakesh Samanta	Edit

Cancel

B) Master Form Attendance Status:

The screenshot shows the 'Attendance Status' master form. At the top, there is a header bar with the system name 'leave management system' and a navigation menu with links: Home, About, Master, Transaction, HOD, and Principle. Below the header, the form has a title 'Attendance Status' and a text input field. Underneath the input field are four green buttons: Save, Update, Delete, and Reset. Below these buttons is a table with three columns: 'Select', 'attendancename', and 'Edit'.

Select	attendancename	Edit
<input type="checkbox"/>	present	Edit
<input type="checkbox"/>	present	Edit
<input type="checkbox"/>	MedicalLeave	Edit
<input type="checkbox"/>	PersonalLeave	Edit
<input type="checkbox"/>	absent	Edit

C) Master Form Employee Type:

The screenshot shows the 'Employee Type' master form. It has a similar layout to the previous form, with a header bar containing 'leave management system' and a navigation menu. The form title is 'Employee Type' with a corresponding text input field. Below the input field are four green buttons: Save, Update, Delete, and Reset. Below these buttons is a table with three columns: 'Select', 'emplype' (note the typo), and 'Edit'.

Select	emplype	Edit
<input type="checkbox"/>	HOD	Edit
<input type="checkbox"/>	STAFF	Edit
<input type="checkbox"/>	PRICIPLE (note the typo)	Edit

Below the table, there is a green 'Cancel' button.

D) Master Form Employee Type:

leave management system

[Home](#) [About](#) [Master](#) [Transaction](#) [HOD](#) [Principle](#)

Leave Type

Leave Type

Save

Update

Reset

Delete

Select	leavetype	Edit
<input type="checkbox"/>	Medical	Edit
<input type="checkbox"/>	Personal	Edit
<input type="checkbox"/>	Casual	Edit
<input type="checkbox"/>	Relious Observation	Edit
<input type="checkbox"/>	Vacation Days	Edit

Cancel

TEST CASES

7. TEST CASES

7.1. INTRODUCTION

The project is used to generate Leave Application. This Leave Application is send to the HOD/Principal over intranet, after sending leave application HOD/Principal send response to the user.

Following are the testing goals

- a) Test the length of the password.
- b) Test the performance of the system.
- c) Test the relevance of message.
- d) To ensure proper communication between the source and destination.
- e) To ensure proper retrieval of applications.

7.2. ITEMS TO BE TESTED

- a) To test the GUI.
- b) Test the generation of appropriate report.
- c) Test the communication between Principal, HOD, Staff.

7.3. FEATURES TO BE TESTED

7.3.1. Performance

- a) Must perform efficient application and retrieval of application.
- b) Increase in the rate of morphing.

7.3.2. Functionality

- a) Proper login of end users.
- b) Taking attendance.
- c) Approval and rejection of leave application.
- d) Sending message to the respective faculty.

7.4 Test Cases (Item pass/fail criteria)

Table 7.1: Item pass/fail criteria

Test Case ID	Objective	Steps to Perform	Input Data	Expected Result	Actual result	Test Case Pass/fail
TC_01	Enter the Correct Username with Wrong Password.	Enter the correct username but with Wrong password.	“Admin” 456	It Show Message “Enter Valid Username and password.	“Enter Valid Username and password .	Pass
TC_02	Enter the Correct Username and Password.	1) Enter the correct username(“admin”) 2) Enter the correct Password(123)	“Admin” 123	It Show Message “Login successfully”	“Login successfully”	Pass
TC_03	To Check all forms are open or not.	1) Select master forms. 2) open one by one form.		All forms are open successfully	Same as Expected Result.	Pass

Leave Management System

TC_04	To take the attendance.	1)Take the attendance And save	Absent or present	Successfully attendance saved	Same as Expected Result.	Pass
TC_05	To send leave application	1)Make a request and submit it	Enter the reason and span of time	Request is sent	Same as Expected Result.	Pass
TC_06	To approve the leave request	1)approve the leave request via approve button		Leave request approved	Same as Expected Result.	Pass
TC_07	To reject the leave request	1)reject the leave request via approve button		Leave request rejected	Same as Expected Result.	Pass

PLATFORM CHOICE

8. PLATFORM CHOICE

8.1. .Net Framework:

Microsoft .net is based on the .net framework, which consists of two major components. The common language runtime (CLR) and an extensive set of framework class libraries (FCL) the CLR Defines a common programming model and a standard type system for cross-platform multi-language development.

The CLR provides a run-time environment that execution of code and provides services that improve development, deployment and run time. Code that targets the CLR is called managed code. 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 but internet-distributed or executed remotely.

- I. To provide a code-execution environment that minimizes software deployment and versioning conflicts.
- II. To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party.
- III. To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
- IV. To make the developer experience consistent across varying types of applications.

8.2. Tool for implementation:

8.2.1 Microsoft Visual Studio:

Microsoft Visual Studio is an integrated development environment (IDE) from Microsoft. It is used to develop computer programs for Microsoft Windows, as well as web sites, web applications and web services. Visual Studio uses Microsoft software development platforms such as Windows API, Windows Forms, Windows Presentation Foundation, Windows Store and Microsoft Silverlight. It can produce both native code and managed code.

Visual Studio includes a code editor supporting IntelliSense (the code completion component) as well as code refactoring. The integrated debugger works both as a source-level debugger and a machine-level debugger. Other built-in tools include a forms designer for building GUI applications, web designer, class designer, and database schema designer. It accepts plug-ins that enhance the functionality at almost every level—including adding support for source-control systems (like Subversion) and adding new toolsets like editors and visual designers for domain-specific languages or toolsets for other aspects of the software development lifecycle (like the Team Foundation Server client: Team Explorer).

8.2.1 Microsoft SQL Server:

The SQL Server consist different create table, insert, delete, update commands and also store procedures, trigger for multiple transactions. The different database connections are open and close for insertion of values in the database.

FUTURE WORK

9. FUTURE WORK

We propose the following functionality.

- a) To reduce the paper work.
- b) To increase the accuracy of leave management.

CONCLUSION

10. CONCLUSION

Leave Management System is very useful for college to maintain leave records of the teachers .this system not only maintain leave details of the staff, it also maintain the leave applications of the staff.

The higher authorities may accept or reject the applications requested by the staff. Thus this system maintain excess amount of job done by the college to maintain the leaves.

REFERENCES

11. REFERENCES

1. A field guide to Leave management systems. (2005). Retrieved November 12, 2006, from http://www.learningcircuits.org/NR/rdonlyres/BFEC9F41-66C2-42EFBE9D-E4FA0D3CE1CE/7304/LMS_fieldguide1.pdf
2. Gibbons, A. S., Nelson, J. M., & Richards, R. (2002). The nature and origin of instructional objects. In D. A. Wiley (Ed.), *the instructional use of learning objects*: Online version. Retrieved April 5, 2005, from
3. Gilhooly, K. (2001). Making e-learning effective. *Computerworld*, 35(29), 52-53.
4. Greenberg, L. (2002). LMS and LMS: What's the Difference?. *Learning Circuits* from <http://www.learningcircuits.org/2002/dec2002/greenberg.htm>
5. Hodgins, H. W. (2002). The future of learning objects. In D. A. Wiley (Ed.), *The instructional use of learning objects*: Online version. Retrieved March 13, 2005, from <http://reusability.org/read/chapters/hodgins.doc>
6. Introduction: why we need LMS, first version, and redesign. (2016). Retrieved on November 20, 2016.
7. Wiley, D. (2002). Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy. In D. A. Wiley (Ed.), *The instructional use of learning objects*: Online version. Retrieved March 13, 2015

Book references:

The complete reference SQL by James R.Groof and Paul N.Weinberg

Microsoft .NET framework 3.5 by Mike Snell,Glenn Johnson and GrandMaster

APPENDIX

Appendix:

1. **Leave Management** : It manages all the details related to teachers and also it maintain record to the teachers.
2. **.NET** : Programmers produce software by combining their own source code with
3. **GUI**: The graphical user interface is a type of user interface that allows users to interact with electronic devices through graphical icons and visual indicators such as secondary notation, instead of text-based user interfaces, typed command labels or text navigation.
4. **SQL**: stands for Structured Query Language. SQL is used to communicate with a database. According to ANSI (American National Standards Institute),it is the standard language for relational database management systems