

A Project Report on

ERP System for Employee Management

Submitted in partial fulfillment of the requirements for the award
of the degree of

Bachelor of Engineering

in

Information Technology

by

Saif Inamdar(16204022)
Anam Quraishi(16204017)
Isha Owalekar(15104011)

Under the Guidance of

Prof. Kiran Deshpande
Prof. Neha Deshmukh



Department of Information Technology
A.P. Shah Institute of Technology
G.B.Road,Kasarvadavli, Thane(W), Mumbai-400615
UNIVERSITY OF MUMBAI

Academic Year 2018-2019

Approval Sheet

This Project Report entitled “**ERP System for Employee Management**” submitted by “**Saif Inamdar**”(16204022), “**Anam Quraishi**”(16204017), “**Isha Owalekar**”(15104011) is approved for the partial fulfillment of the requirement for the award of the degree of **Bachelor of Engineering** in **Information Technology** from **University of Mumbai**.

(Prof. Neha Deshmukh)
Co-Guide

(Prof. Kiran Deshpande)
Guide

Prof. Kiran Deshpande
HOD, Department of Information Technology

Place:A.P.Shah Institute of Technology, Thane
Date:

CERTIFICATE

This is to certify that the project entitled ***“ERP System For Employee Management”*** submitted by ***“Saif Inamdar” (16204022), “Anam Quraishi” (16204017), “Isha Owalekar” (15104011)*** for the partial fulfillment of the requirement for award of a degree ***Bachelor of Engineering in Information Technology.***, to the University of Mumbai, is a bonafide work carried out during Academic Year 2018-2019.

(Prof. Neha Deshmukh)
Co-Guide

(Prof. Kiran Deshpande)
Guide

Prof. Kiran Deshpande
HOD, Department of Information Technology

Dr. Uttam D.Kolekar
Principal

External Examiner(s)

1.

2.

Place: A.P. Shah Institute of Technology, Thane

Date:

Declaration

We declare that this written submission represents our ideas in our own words and where others' ideas or words have been included, We have adequately cited and referenced the original sources. We also declare that We have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in our submission. We understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.

Saif Inamdar (16204022)

Anam Quraishi (16204017)

Isha Owalekar (15104011)

Date:

Abstract

Staff record plays a significant role in an organization as it provides information needed to manage their employee performance. Many institutes are still using the conventional methods which are merely paper-based and are used to record the data of their employees. This often results in downright, waste of time in generating reports or searching for employee records. These inadequacies in the conventional method are characteristically eluded for the justification in developing an ERP System for Employee Management. The motive behind developing this system is to get employee availability information at click. This is a web-based system which will manage their availability and workload in the organization dynamically. The registered people are employees, HODs and Principal of the particular organization. Higher authorities can view availability and location of an employee as per their timetable. Using this system the higher authorities such as HOD and Principal can conduct activities in the premises along with selected faculty members and notifying them through Email/SMS for the same. Employees can share their workload dynamically over the web-system which affects their availability in the Institute. When an employee desires to take a leave he/she can apply through the system along with workload sharing with another employee. This system will help the institute to improve overall performance by eliminating paperwork and dynamic management of employee availability.

Contents

1	Introduction	1
1.1	What is ERP System for Employee Management?	1
1.2	Problem Definition	1
1.3	Proposed System	2
1.4	Objective	3
1.4.1	General Objective	3
1.4.2	Specific Objective	3
2	System Analysis	5
2.1	Study of Current System	5
2.2	Problems and Weakness of Current System	5
2.3	Requirement of a new System	5
2.4	System Planning	5
2.4.1	Waterfall Model	6
3	System Design	7
3.1	Design	7
3.2	Attribute of good design	7
3.3	System Architecture	8
3.4	Class Diagram	10
3.5	Activity Diagram	11
3.5.1	Activity Diagram for Faculty	11
3.5.2	Activity Diagram for HOD	12
3.5.3	Activity Diagram for Principal	13
3.6	Use Case Diagram	14
3.7	Sequence Diagram	15
3.7.1	Sequence Diagram for login	15
3.7.2	Sequence Diagram for Faculty	15
3.7.3	Sequence Diagram for HOD	16
4	Database Design	17
4.1	Data Dictionary	17
5	Project Planning and Execution	20
5.1	Technology Stack	20
5.2	Software specifications	21
5.3	Gantt Chart	22

6	User Manual	23
6.1	Registration Module	23
6.2	Dashboard	24
6.3	Statistics	25
6.4	Notifications	25
6.5	Workload Requests	26
6.6	Leave Requests	26
6.7	Faculty Availability	27
6.8	Dynamic Workload	27
6.9	Conduct Activity	28
6.10	Time Table	28
6.11	User Requests	29
6.12	User profile	29
6.13	Reset	30
6.14	Code for Dynamic Workload	32
6.15	Workload Approval	33
6.16	Leave Requests	34
6.17	Statistics	35
6.18	Time Table	36
7	Test Report	37
7.1	Test Scenario 1	37
7.2	Test Scenario 2	38
7.3	Test Scenario 3	38
7.4	Test Scenario 4	39
7.5	Test Scenario 5	39
7.6	Test Scenario 6	40
7.7	Test Scenario 7	40
8	Conclusion	41
	Bibliography	42
	Publication	44

List of Figures

1.1	Designations in an Educational Institute	3
2.1	Waterfall Model	6
3.1	Employee Login Flowchart	8
3.2	Higher Authority Flowchart	8
3.3	Class Diagram for ERP System	10
3.4	Activity Diagram for Faculty	11
3.5	Activity Diagram for HOD	12
3.6	Activity Diagram for Principal	13
3.7	Use Case Diagram for ERP System	14
3.8	Sequence Diagram for login	15
3.9	Sequence Diagram for Faculty	15
3.10	Sequence Diagram for HOD	16
4.1	List of Tables	17
4.2	Timetable	17
4.3	Conduct Activity	18
4.4	Dynamic Duty	18
4.5	Leave Application	19
4.6	List of users	19
5.1	Gantt Chart	22
5.2	Task	22
6.1	Registration Module	23
6.2	OTP	24
6.3	Dashboard	24
6.4	Statistics	25
6.5	Notifications	25
6.6	Workload Requests	26
6.7	Leave Requests	26
6.8	Faculty Availability	27
6.9	Dynamic Workload	27
6.10	Conduct Activity	28
6.11	Time Table	28
6.12	User Requests	29
6.13	User profile	29
6.14	Reset	30

6.15	Reset for new academic year	30
6.16	Reset for 2nd half	31
6.17	Dynamic Workload	32
6.18	Workload Approval	33
6.19	Leave Requests	34
6.20	Statistics	35
6.21	Time Table	36
7.1	Invalid Credentials	37
7.2	Valid Credentials	38
7.3	User not found	38
7.4	Dashboard	39
7.5	Apply for Leave	39
7.6	Dynamic Workload	40
7.7	Time-Table	40

Chapter 1

Introduction

The project idea focuses on problems faced in the institute with an adequate workforce. An institute faces many problems which are observed while managing workload between employees when some employees are on leave. Because of such unavailability of employees, managing their workload becomes a tedious task. It also becomes a laborious work for the higher authority to evaluate employees performance based on their leave records and dynamic workload. Therefore, we have proposed a system which can tackle such problems of an institute. The old practice of hand-written applications between the employees and higher authority regarding leaves, workload sharing and conducting activities will be taken over by this web system. This system will display all the records of employees regarding their schedule, leaves and workload to the higher authority. Such display of statistics will make the job easier for the higher authority to make a decision regarding employees leave requests, workload sharing and load adjustments.

1.1 What is ERP System for Employee Management?

ERP System for Employee Management is an online Web based system which will take care of leave and availability of Employees along with that, dynamic workload is taken into consideration. This portal will represent information and Statistics in diagrams as well as in tables, forms and is easy to use. It integrates many features in one portal which are mostly proprietary and not available in one single system. As Employees are hassled by the manual paper based system this online web based system will provide a hassle free way to manage their activities in an Institute.

1.2 Problem Definition

Following are the problems faced by the educational organizations regarding availability of employees:-

- Consider a scenario where an employee applies for a leave and the higher authority is having a tough time finding past leave applications of an employee to calculate his eligibility for the leave.

- Consider a scenario where the higher authority is trying to physically locate an employee in the organization.
- Consider a scenario where an employee who is supposed to be on a leave in future has to share workload with any colleague.
- Consider a scenario where the higher authority has to keep handwritten or printed records of each employees past leaves.
- Consider a scenario where the higher authority wants to know the total no. of leaves taken by the employees in this particular month/year. It will result in lot of time being invested just to get that simple statistic.
- Consider a scenario where the higher authority wants to check which employee(s) are available on a particular time-slot/day just to schedule some extra work or a special event/activity.
- Consider a scenario where the higher authority wants to analyse the performance of an employee for appraisals considering leaves and participation in extra activities.

Considering the above scenarios, we have proposed a web system which will dynamically manage such problems to improve the overall efficiency and performance of the organization.

1.3 Proposed System

This system will be web-based which will provide 24x7 availability and usability. It will be developed using various web technologies. Different interfaces for Employees, HODs and Principal are designed. Broadly classified into two modules:

1. Availability Of Employees.
2. Leave Management for Employees.

This system also provides three sub-modules as follows:-

1. Principal

This is the highest authority and has much more privileges than HOD and employee. The principal can reset timetable, approve leave requests, send notifications to the staff, check leave statistics, check availability of employees.

2. HOD

The HOD can approve leaves of the employees, check statistics of the concerned department only, send notifications to the staff, check employee availability.

3. Employee (Assistant Professor, Teaching Assistant, Associate Professor)

Employees can apply for leave, adjust their workload, set timetable at the beginning of the semester, view statistics of their leaves.

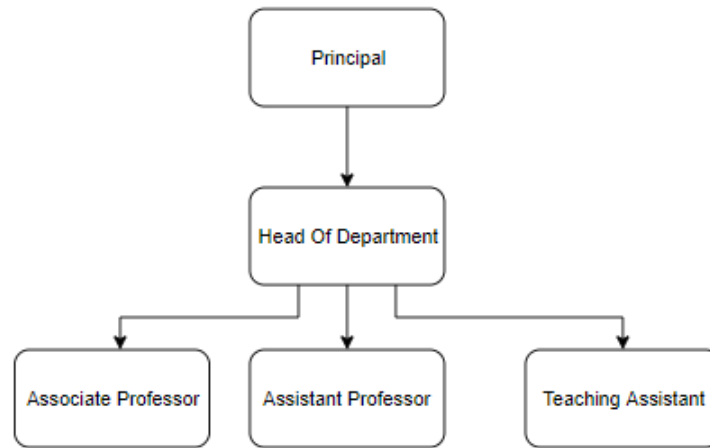


Figure 1.1: Designations in an Educational Institute

1.4 Objective

1.4.1 General Objective

Current Objective is to Develop a Web System for an Educational institute which will replace the old process of Leave and Availability Management of an Employee along with Workload Sharing between Employees and Dynamic Display of Individual Statistics.

1.4.2 Specific Objective

- To provide an interface for employees which can be used to view/edit their schedule/timetable.
- To provide an interface for employees and higher authorities which can be used to view/edit/update their profile.
- To provide a secure login interface for all users.
- To provide an interface for employees which will display their leave statistics in graphical charts and tables.
- To provide an interface for employees which can be used to apply for a leave along with description and load sharing which will be approved by the higher authority.
- To provide an interface for employee which can be used to accept/reject load adjustment requests by employees.

- To provide an interface for higher authority which can be used to view/accept/reject leave applications made by employees.
- To provide an interface for higher authorities to view statistics regarding all/individual employees.
- To develop an interface for employees to fill in their assigned dynamic activities which will affect their availability.
- To develop an interface for higher authorities to conduct activities in the institute.
- To develop an interface for higher authorities to view and accept/reject dynamic workload requests from users.
- To provide an interface for higher authority to view current location of an employee according to their schedule/timetable.
- To develop a notification system which will be used to notify employees and higher authorities via SMS and E-mail.
- To develop a GUI which displays statistics in appropriate charts for better visualization.

Chapter 2

System Analysis

2.1 Study of Current System

Current system which is orangeHRM is a subscription based proprietary SaaS web system which provides features regarding leave management, attendance records, etc. Unfortunately it is proprietary and requires extra premium charges for some essential features along with lack of dynamic workload management

2.2 Problems and Weakness of Current System

- Current system lacks the feature of dynamic workload as it provides only static workload.
- It also fails in providing integrated report generation.
- It provides only limited free features and expensive premium add-ons.
- Many features are unnecessary which is not required by the institute for any evaluation.

2.3 Requirement of a new System

To provide additional required features and functionalities which do not exist in the current system and are actually required by the institute for performance evaluation. To overcome the drawbacks of the current system which are specified above we come up with a system which integrates various such features.

2.4 System Planning

Lahman defines a methodology as a process where the activities are primarily intellectual. Typically only the end goal of the process is manifested as a physical work product. In software the analysis and design activities are normally governed by a specific methodology. Developing a software system is a complex and time-consuming process. Software engineering methodologies are the framework that tells us how we should go about developing our software system. There are a variety of methodologies available today.

2.4.1 Waterfall Model

The Waterfall Model is a sequential software development model which is seen as flowing steadily downwards through the following phases:

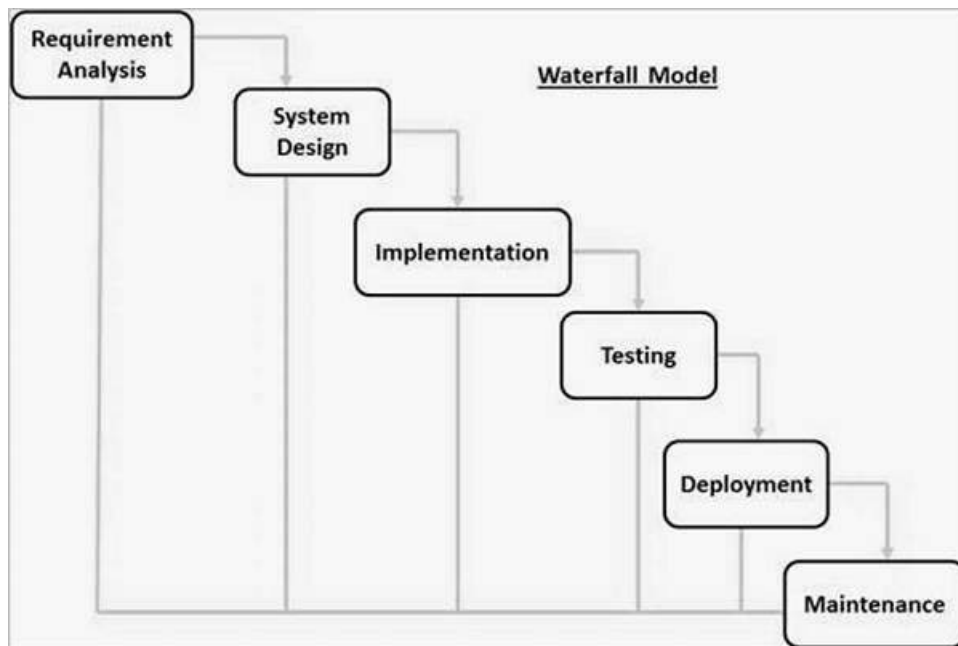


Figure 2.1: Waterfall Model

- **Requirements Analysis** : All possible requirements of the system to be developed are captured in this phase and documented in a requirement specification document.
- **Design** : The requirement specifications from first phase are studied in this phase and the system design is prepared. This system design helps in specifying hardware and system requirements and helps in defining the overall system architecture.
- **Implementation** : With inputs from the system design, the system is first developed in small programs called units, which are integrated in the next phase.
- **Testing** : All the units developed in the implementation phase are integrated into a system after testing of each unit. Post integration the entire system is tested for any faults and failures.
- **Deployment** : Once the functional and non-functional testing is done; the product is deployed in the customer environment or released into the market.
- **Maintenance** : There are some issues which come up in the client environment. To fix those issues, patches are released. Also to enhance the product some better versions are released.

We preferred using the waterfall model because of its inherent linear structure that is well-suited for the projects that work well under a milestone and date-focused paradigm. We opted for this model as we have modules followed by various sub-modules in it. Firstly, we took requirements from the educational institute then we designed a prototype and incorporated changes in it and then implemented the system, tested and deployed.

Chapter 3

System Design

3.1 Design

Design is concerned with developing a solution which meets the requirements specified. Wright states Design is the application of creativity to planning the optimum solution of a given problem and the communication of that problem to others. It is important that designers of a system understand the requirements of that system. The designer also needs to be able to represent this information through a number of modeling techniques.

3.2 Attribute of good design

There are various design strategies available to developers. In order to produce a good design, the developer must select the most appropriate approach for the system being developed. A good design may be the most efficient, the cheapest, the most maintainable, the most reliable etc. In order to achieve this, it must address the following factors:

- **Abstraction** : Abstraction is a mechanism to reduce and factor out details so that the developer can focus on a few concepts at a time. Abstraction manages complexity by emphasizing essential characteristics and suppressing implementation details allowing the developer to produce a logical model of the system which can then be transformed into a physical model.
- **Modularity** : Modularity divides the software into separate components that are integrated to solve problem requirements. Each separate component can be referred to as a module. Modularity allows systems to be designed and developed in such a way that the implementation of each module is independent of the implementation of the other modules, thus allowing the system to be maintainable.
- **Program Structure** : The program structure represents the hierarchy of control. Program structure is usually expressed as a simple hierarchy showing super-ordinate and subordinate relationships of modules.
- **Data Structure** : Data structure represents the organizations, access method, associatively and processing alternatives for problem-related information. Classic data structures include scalar, sequential, linked list and hierarchical. Data structure along with program structure makes up the software architecture.

- **Software Procedure :** Software procedure provides a precise specification of the software processing, including sequence of events, exact decision points, repetitive operations, and data organisation. Processing defined for each module must include references to all sub-ordinate modules identified by the program structure.
- **Information Hiding :** Information hiding is an attachment of modularity. It permits modules to be designed and coded without concern for the internals of other modules. Only the access protocols of a module need to be shared with the implementers of other modules. Information hiding simplifies testing and modification by localizing these activities to individual modules.

3.3 System Architecture

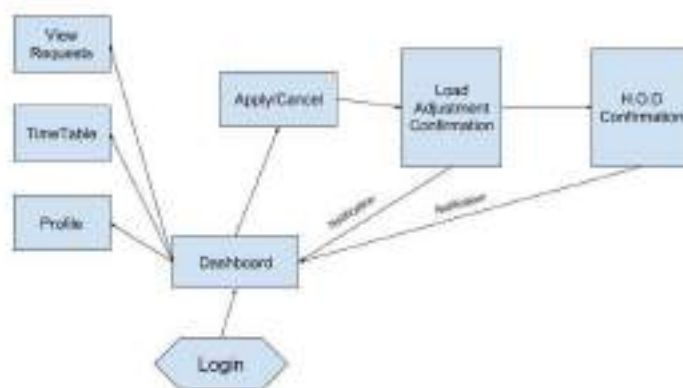


Figure 3.1: Employee Login Flowchart

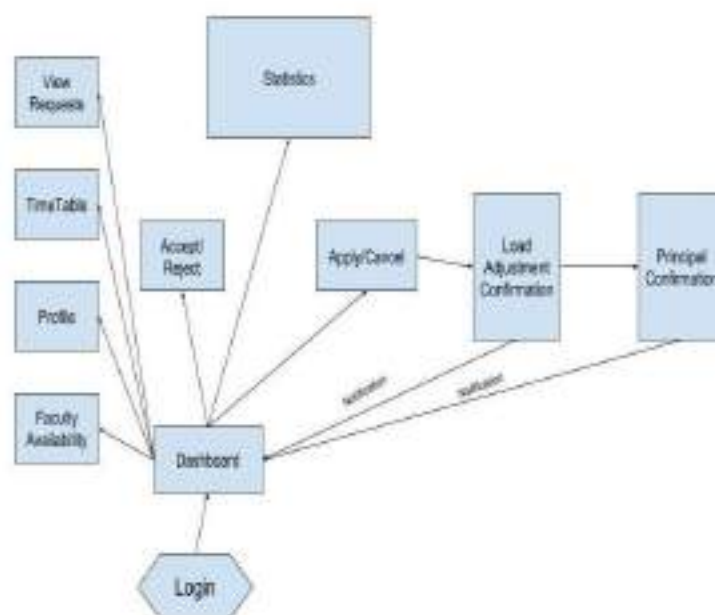


Figure 3.2: Higher Authority Flowchart

The architecture of the system described above consists of various sub-modules in it:

- **Dashboard:**It will display leave statistics of the registered employees and navigation to other sub-modules.
- **Apply/Cancel Leave:**Employees are supposed to fill a leave application form.They can also cancel the application if it is not approved by the higher authority.The application comprises four details namely:
 - Date and Time of the leave
 - Type of leave
 - Description of the leave
 - Load Adjustment
- **View Requests:**By using this sub-module, the employees are able to view pending load adjustments.They can accept/reject it.
- **Timetable:**Employees can enter timetable at the beginning of every semester.
- **View/Change Profile:**Employees can view/edit user profile.
- **HOD Confirmation:**The HOD will confirm leave requests of the employees after going through the leave application.
- **Statistics:**This sub-module will display the leave statistics of all the employees to the higher authorities.Here the HOD will be able to see the leave statistics of the concerned department and Principal will be able to see the leave statistics of the employees of all the departments.Leaves would be categorized as:
 - Earned leave
 - Sick leave
 - Casual leave
 - Compensatory off
- **Faculty Availability:**The employees with free-slots would be seen available to the higher authorities.
- **Conduct Activity:**Notifications would be sent through e-mail/SMS if department has scheduled an activity.
- **Reset:**Principal has the privilege to reset timetable at the end of every semester.

3.4 Class Diagram

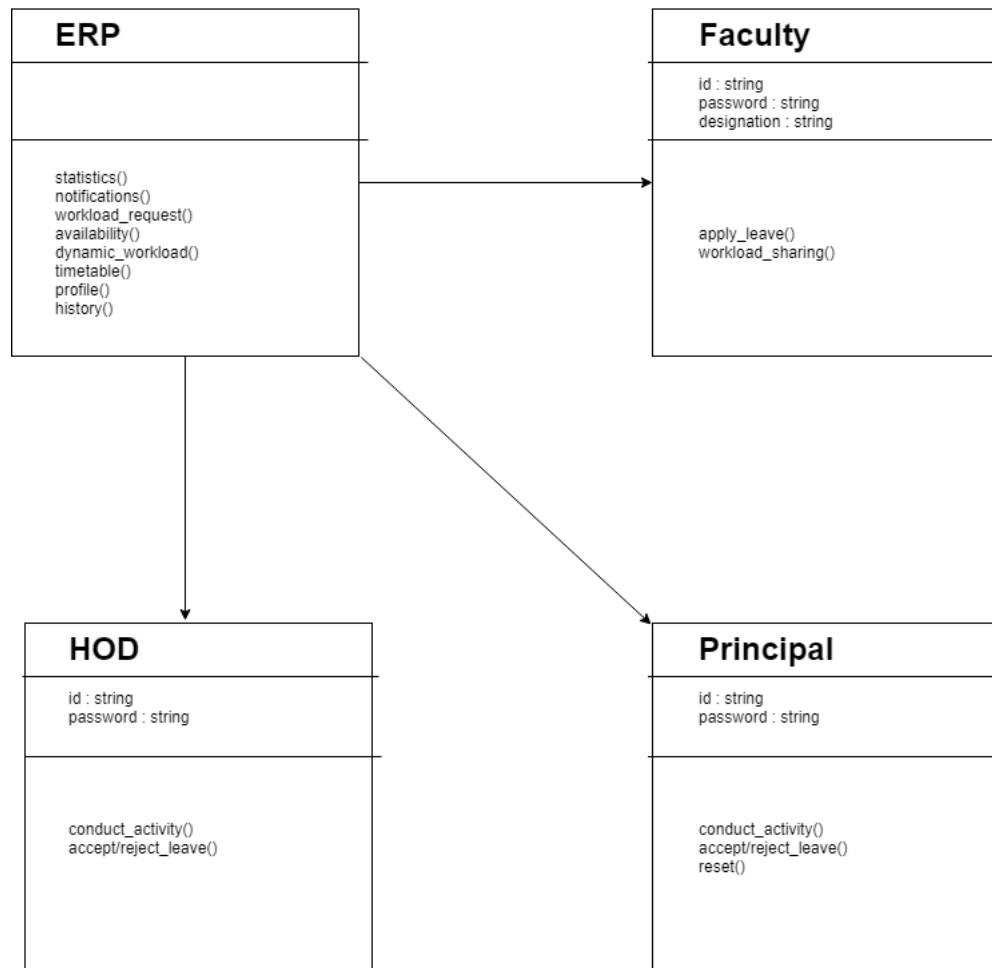


Figure 3.3: Class Diagram for ERP System

The above class diagram depicts various entities along with the variables and functions used in the system. Here ERP is the entity connected to the three interfaces of the system. Entities have the variables listed in the first row followed by the functions.

3.5 Activity Diagram

3.5.1 Activity Diagram for Faculty

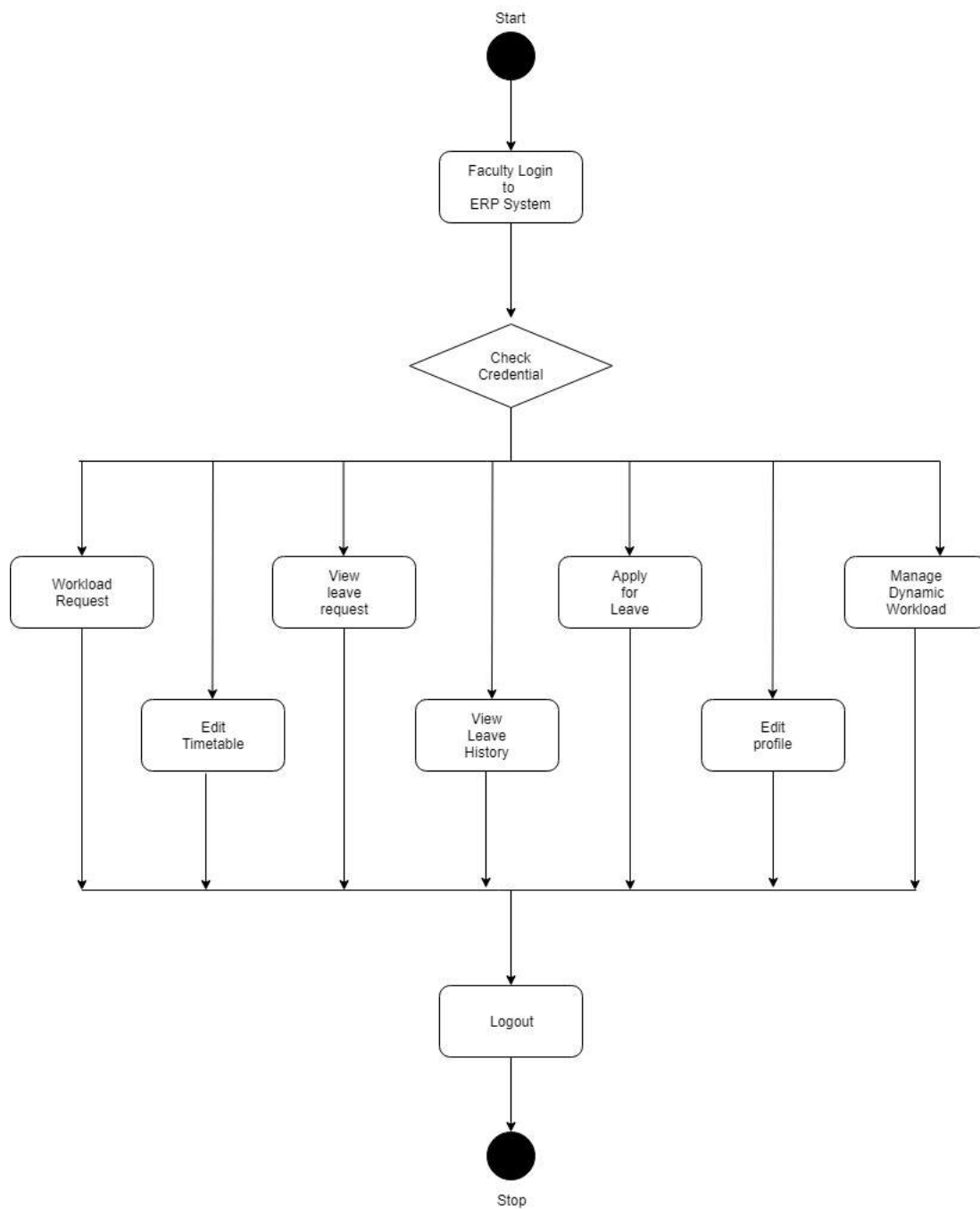


Figure 3.4: Activity Diagram for Faculty

The above diagram depicts the activities/tasks done by the faculties. Activities such as workload requests, edit timetable, apply for leave, edit profile, manage dynamic workload are done by the faculties.

3.5.2 Activity Diagram for HOD

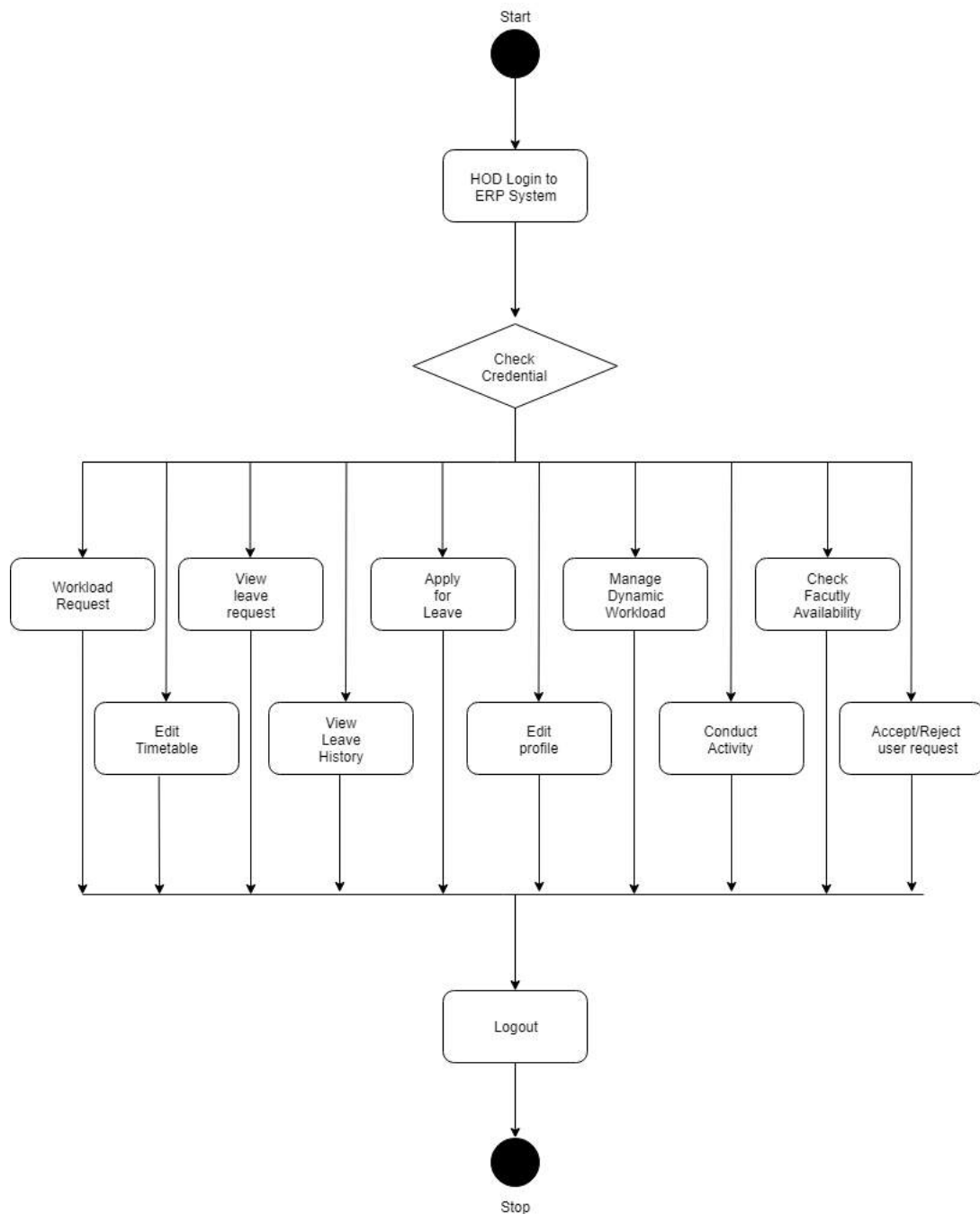


Figure 3.5: Activity Diagram for HOD

The above mentioned figure describes the tasks/activities which are done by the HODs. since HOD has more privileges than the faculties he/she will have activities similar to the faculties with add-ons of conduct activity, confirm user requests, confirm load adjustments.

3.5.3 Activity Diagram for Principal

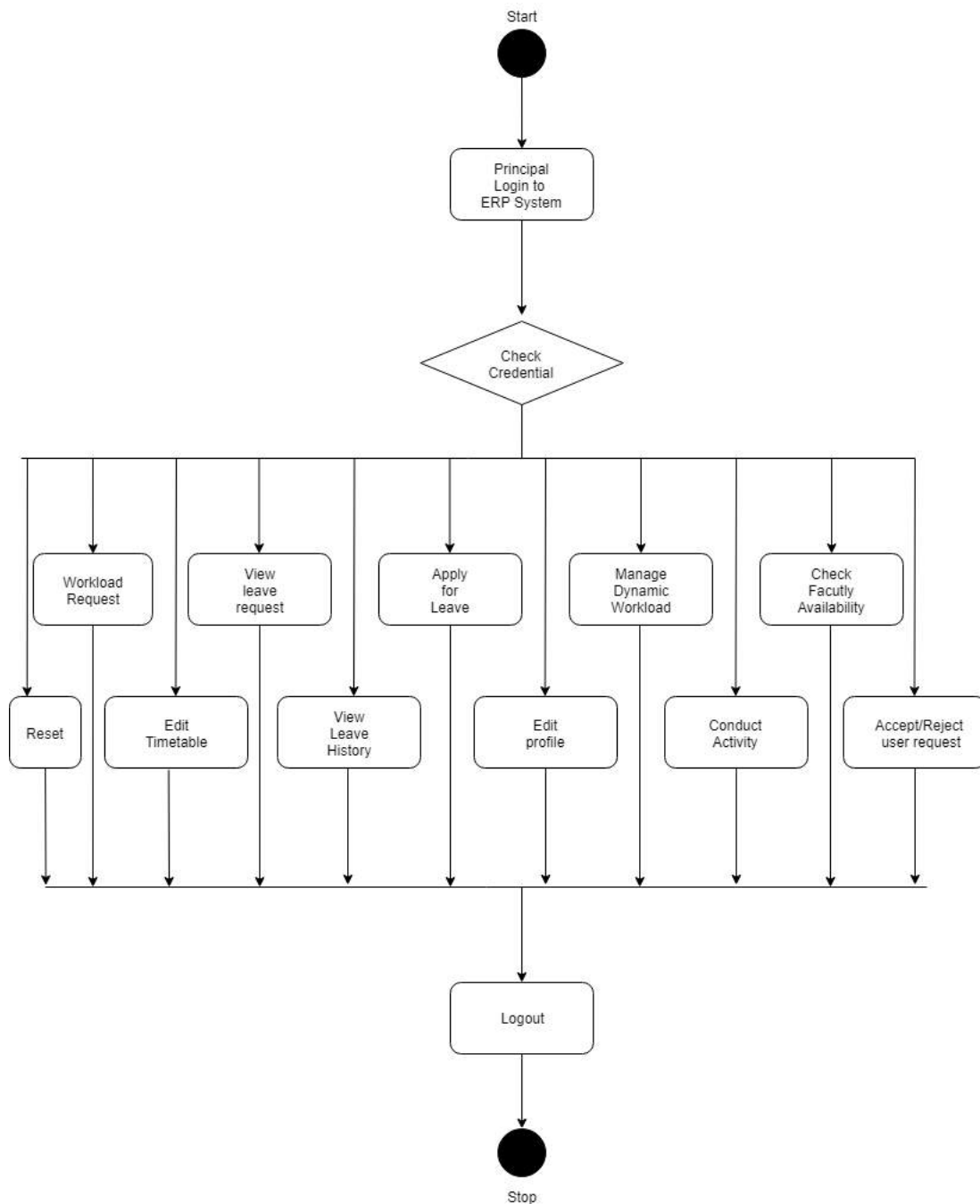


Figure 3.6: Activity Diagram for Principal

The above mentioned figure describes the tasks/activities which are done by the Principal. Since, Principal has more privileges than the faculties and HODs, he/she will have activities similar to them but with an add-on of reset timetable.

3.6 Use Case Diagram

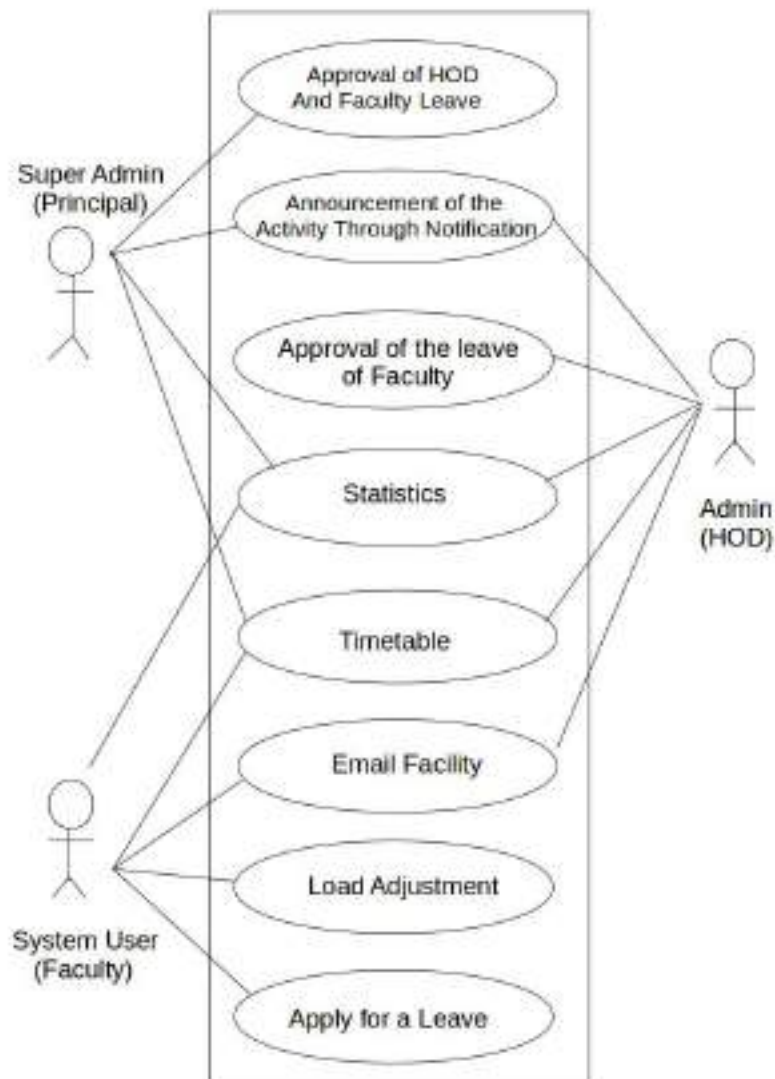


Figure 3.7: Use Case Diagram for ERP System

The above use-case diagram helps to identify any internal or external factors that may influence the system. It provides a good high level analysis from outside the system wherein the users are principal, faculty, HOD operating the system.

3.7 Sequence Diagram

3.7.1 Sequence Diagram for login



Figure 3.8: Sequence Diagram for login

The above sequence diagram is a general diagram which tells us about the login procedure of the employee during registration/login of the system.

3.7.2 Sequence Diagram for Faculty

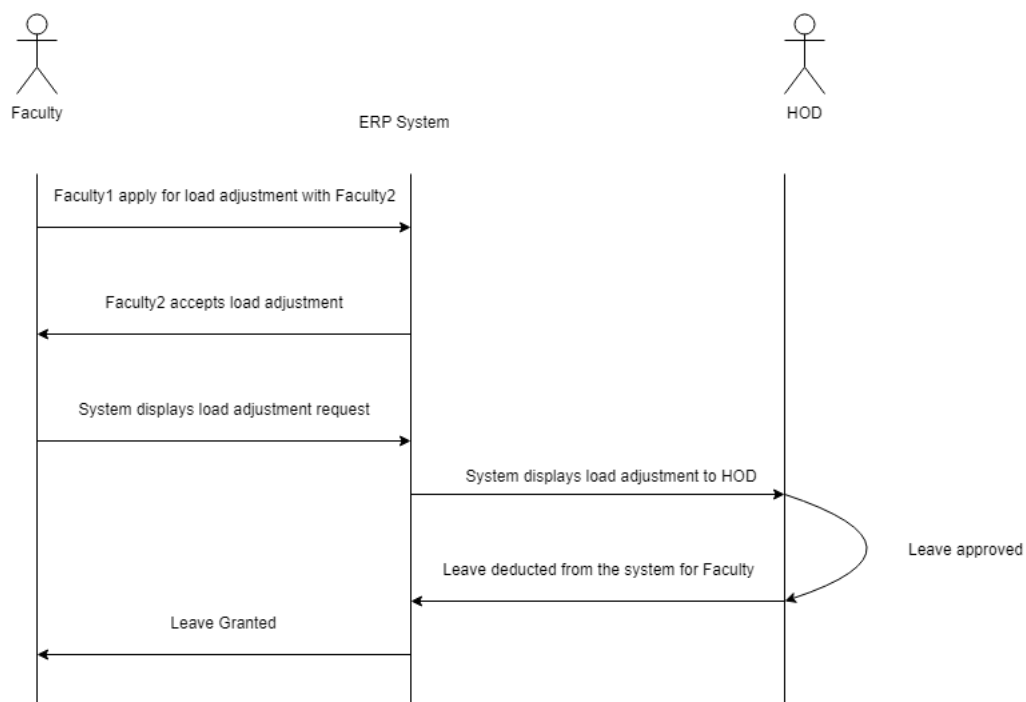


Figure 3.9: Sequence Diagram for Faculty

The above mentioned sequence diagram depicts the procedure for leave application along with the load adjustment of the faculties followed by the approval of the HOD.

3.7.3 Sequence Diagram for HOD

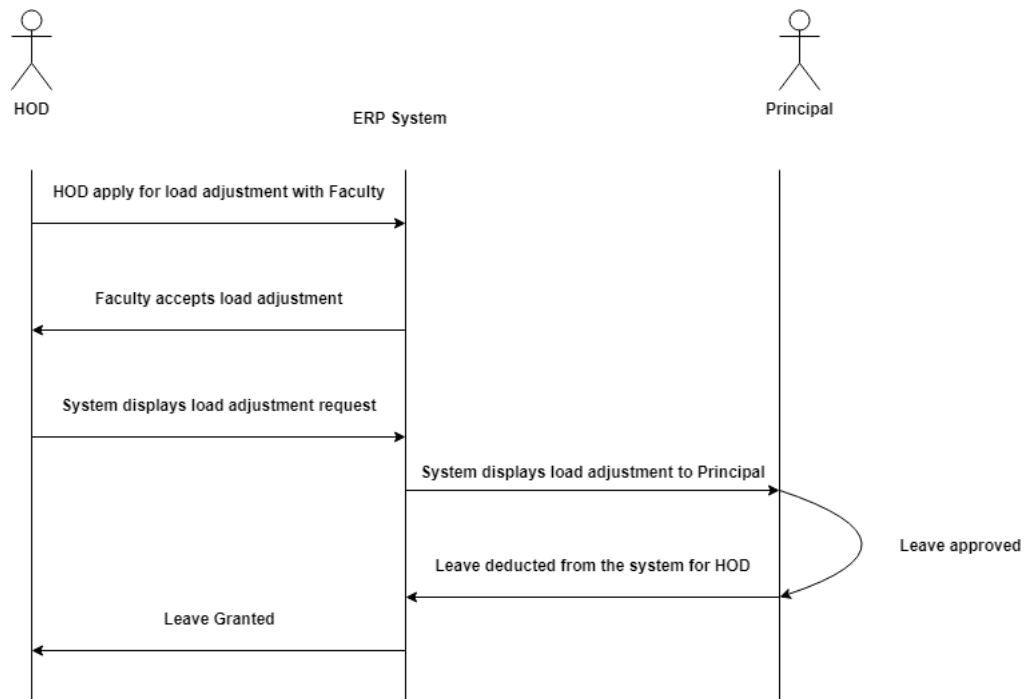


Figure 3.10: Sequence Diagram for HOD

The above mentioned sequence diagram depicts the procedure for leave application along with the load adjustment of the HOD followed by the approval of the Principal.

Chapter 4

Database Design

4.1 Data Dictionary

Table	Action	Rows	Type	Collation	Size	Overhead
abcdefgh	★ Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	16 Kib	-
abcdefgh_ss	★ Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 Kib	-
academic_master	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 Kib	-
anamgaurahi	★ Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	16 Kib	-
anamgaurahi_ss	★ Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 Kib	-
civilteacher	★ Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	16 Kib	-
civilteacher_ss	★ Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 Kib	-
conduct_activty	★ Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 Kib	-
dynamic_duty	★ Browse Structure Search Insert Empty Drop	1	InnoDB	latin1_swedish_ci	16 Kib	-
lahansawekar	★ Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	16 Kib	-
lahansawekar_ss	★ Browse Structure Search Insert Empty Drop	0	InnoDB	latin1_swedish_ci	16 Kib	-
leave_applications	★ Browse Structure Search Insert Empty Drop	4	InnoDB	latin1_swedish_ci	16 Kib	-
softframar	★ Browse Structure Search Insert Empty Drop	10	InnoDB	latin1_swedish_ci	16 Kib	-
softframar_ss	★ Browse Structure Search Insert Empty Drop	8	InnoDB	latin1_swedish_ci	16 Kib	-
users	★ Browse Structure Search Insert Empty Drop	5	InnoDB	latin1_swedish_ci	16 Kib	-
15 tables	Sum	71	InnoDB	latin1_swedish_ci	452 Kib	8 K

Figure 4.1: List of Tables

#	Name	Type	Collation	Attributes	Null	Default
1	day	varchar(10)	latin1_swedish_ci		Yes	NULL
2	5_10	varchar(20)	latin1_swedish_ci		Yes	NULL
3	10_11	varchar(20)	latin1_swedish_ci		Yes	NULL
4	11_12	varchar(20)	latin1_swedish_ci		Yes	NULL
5	12_1	varchar(20)	latin1_swedish_ci		Yes	NULL
6	1_2	varchar(20)	latin1_swedish_ci		Yes	NULL
7	2_3	varchar(20)	latin1_swedish_ci		Yes	NULL
8	3_4	varchar(20)	latin1_swedish_ci		Yes	NULL
9	4_5	varchar(20)	latin1_swedish_ci		Yes	NULL

Figure 4.2: Timetable

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/> 2	activity_title	varchar(100)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 3	activity_description	varchar(400)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 4	dept	varchar(50)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 5	activity_from	datetime			Yes	NULL		
<input type="checkbox"/> 6	activity_to	datetime			Yes	NULL		
<input type="checkbox"/> 7	members	varchar(400)	latin1_swedish_ci		Yes	NULL		

Figure 4.3: Conduct Activity

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/> 2	name	varchar(100)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 3	applied_on	datetime			Yes	NULL		
<input type="checkbox"/> 4	dept	varchar(35)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 5	designation	varchar(30)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 6	type	varchar(30)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 7	duty_from	datetime			Yes	NULL		
<input type="checkbox"/> 8	duty_to	datetime			Yes	NULL		
<input type="checkbox"/> 9	description	varchar(255)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 10	alternative_arrangement	varchar(255)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 11	approval_status	varchar(40)	latin1_swedish_ci		Yes	NULL		

Figure 4.4: Dynamic Duty

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/> 2	applied_on	datetime			Yes	NULL		
<input type="checkbox"/> 3	name	varchar(100)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 4	dept	varchar(35)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 5	type	varchar(35)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 6	status	varchar(35)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 7	duration_of_leave	varchar(15)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 8	f_from	date			Yes	NULL		
<input type="checkbox"/> 9	f_to	date			Yes	NULL		
<input type="checkbox"/> 10	h_from	datetime			Yes	NULL		
<input type="checkbox"/> 11	h_to	datetime			Yes	NULL		
<input type="checkbox"/> 12	no_of_days	int(3)			No	None		
<input type="checkbox"/> 13	reason	varchar(255)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 14	selected_faculty	varchar(100)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 15	academic_workload	varchar(255)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 16	academic_arranged	varchar(255)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 17	dept_workload	varchar(255)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 18	dept_arranged	varchar(255)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 19	approval_status	varchar(40)	latin1_swedish_ci		Yes	NULL		

Figure 4.5: Leave Application

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
<input type="checkbox"/> 1	id	int(11)			No	None		AUTO_INCREMENT
<input type="checkbox"/> 2	full_name	varchar(100)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 3	email	varchar(100)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 4	password	varchar(100)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 5	mobile_no	digit(15)			No	None		
<input type="checkbox"/> 6	designation	varchar(40)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 7	dept	varchar(35)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 8	room	varchar(10)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 9	profile_pic	varchar(200)	latin1_swedish_ci		Yes	NULL		
<input type="checkbox"/> 10	ci	int(4)			Yes	NULL		
<input type="checkbox"/> 11	co	int(4)			Yes	NULL		
<input type="checkbox"/> 12	mi	int(4)			Yes	NULL		
<input type="checkbox"/> 13	complete_profile	tinyint(1)			Yes	NULL		
<input type="checkbox"/> 14	timetable	varchar(100)	latin1_swedish_ci		No	None		
<input type="checkbox"/> 15	sessions	varchar(100)	latin1_swedish_ci		Yes	NULL		

Figure 4.6: List of users

Chapter 5

Project Planning and Execution

5.1 Technology Stack

Front End :

HTML :

HTML is the standard markup language for creating Web pages.

- HTML stands for Hyper Text Markup Language.
- HTML describes the structure of Web pages using markup.
- HTML elements are the building blocks of HTML pages.
- HTML elements are represented by tags.
- HTML tags label pieces of content such as "heading", "paragraph", "table", and so on.
- Browsers do not display the HTML tags, but use them to render the content of the page.

CSS :

CSS stands for Cascading Style Sheets.

CSS describes how HTML elements are to be displayed on screen, paper, or in other media.

CSS can be added to HTML elements in 3 ways:

- **Inline** - by using the style attribute in HTML elements.
- **Internal** - by using a `<style>` element in the `<head>` section.
- **External** - by using an external CSS file.

The most common way to add CSS, is to keep the styles in separate CSS files. However, here we will use inline and internal styling, because this is easier to demonstrate, and easier for you to try it yourself.

JS :

- JavaScript is the programming language of HTML and the Web.
- JavaScript is to program the behavior of web pages.
- Web pages are not the only place where JavaScript is used. Many desktop and server programs use JavaScript. Node.js is the best known. Some databases, like MongoDB and CouchDB, also use JavaScript as their programming language.
- JavaScript was invented by Brendan Eich in 1995, and became an ECMA standard in 1997. ECMA-262 is the official name of the standard. ECMAScript is the official name of the language.

Back End :

PHP :

- PHP is an acronym for "PHP: Hypertext Preprocessor"
- PHP is a widely-used, open source scripting language
- PHP scripts are executed on the server
- PHP is free to download and use

MySQL :

MySQL is the most popular Open Source Relational SQL Database Management System. MySQL is one of the best RDBMS being used for developing various web-based software applications. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company. This tutorial will give you a quick start to MySQL and make you comfortable with MySQL programming.

5.2 Software specifications

Minimum Requirement :

Web Browser :

- Mozilla Firefox 55+
- Google Chrome 20.0

Web Server :

- PHP 7.0+

5.3 Gantt Chart

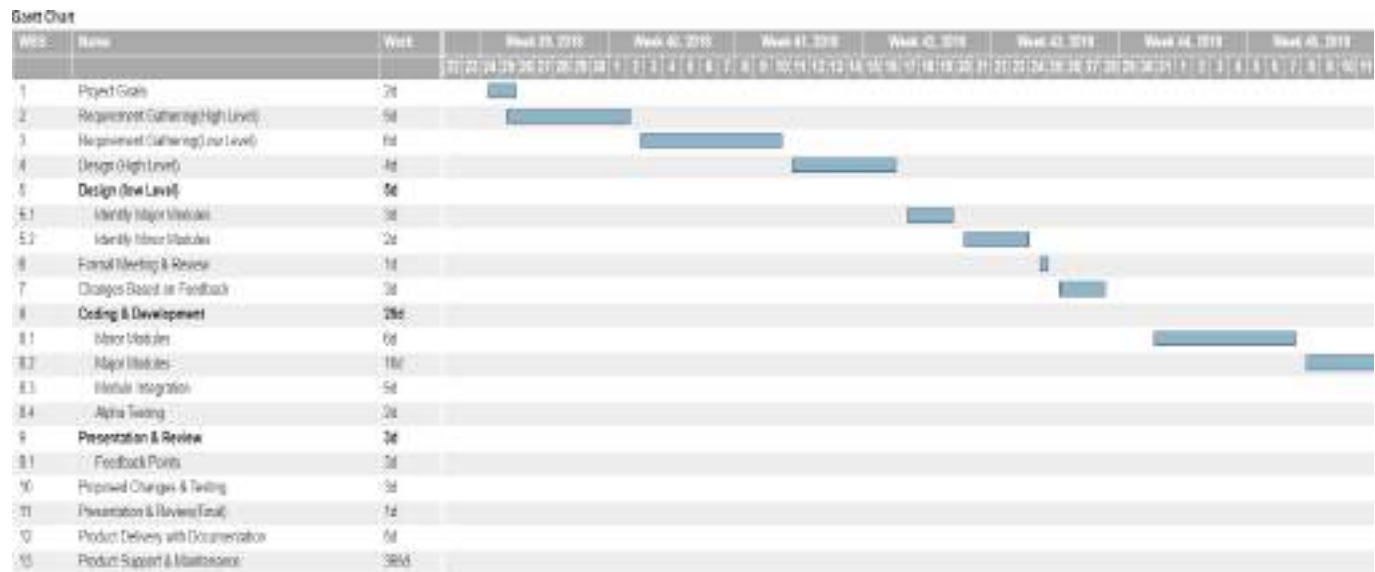


Figure 5.1: Gantt Chart

Tasks

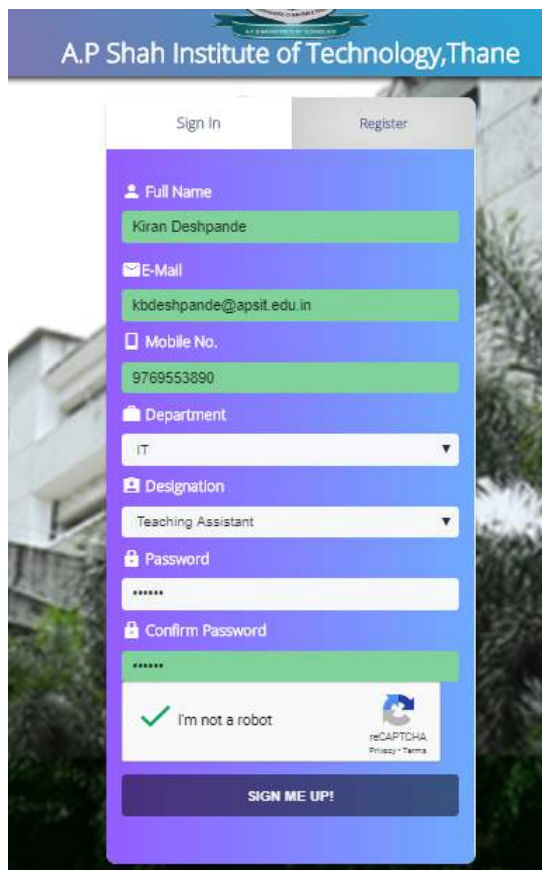
WBS	Name	Start	Finish	Work	Complete
1	Project Goals	Sep 24	Sep 25	2d	0%
2	Requirement Gathering(High Level)	Sep 25	Oct 1	5d	0%
3	Requirement Gathering(Low Level)	Oct 2	Oct 9	6d	0%
4	Design (High Level)	Oct 10	Oct 15	4d	0%
5	Design (Low Level)	Oct 16	Oct 22	6d	
5.1	Identify Major Modules	Oct 16	Oct 18	3d	0%
5.2	Identify Minor Modules	Oct 19	Oct 22	2d	0%
6	Formal Meeting & Review	Oct 23	Oct 23	1d	0%
7	Changes Based on Feedback	Oct 24	Oct 26	3d	0%
8	Coding & Development	Oct 29	Dec 6	28d	
8.1	Minor Modules	Oct 29	Nov 5	6d	0%
8.2	Major Modules	Nov 6	Nov 27	16d	0%
8.3	Module Integration	Nov 28	Dec 4	5d	0%
8.4	Alpha Testing	Dec 5	Dec 6	2d	0%
9	Presentation & Review	Dec 6	Dec 10	3d	
9.1	Feedback Points	Dec 6	Dec 10	3d	0%
10	Proposed Changes & Testing	Dec 11	Dec 13	3d	0%
11	Presentation & Review(Final)	Dec 14	Dec 14	1d	0%
12	Product Delivery with Documentation	Dec 17	Dec 21	5d	0%
13	Product Support & Maintenance	Dec 24	May 15	365d	0%

Figure 5.2: Task

Chapter 6

User Manual

6.1 Registration Module



A.P Shah Institute of Technology, Thane

Sign In Register

Full Name
Kiran Deshpande

E-Mail
kdbeshpande@apsit.edu.in

Mobile No.
9769553890

Department
IT

Designation
Teaching Assistant

Password

Confirm Password

☒ I'm not a robot

reCAPTCHA
Privacy Terms

SIGN ME UP!

Figure 6.1: Registration Module

The above figure shows the registration of the employees for various departments in the institute like Mechanical,Civil,Electronics and Telecommunications,Information Technology,Computer Engineering.Along with the designations as teaching assistant, assistant professor,associate professor,HOD ,Principal.



Figure 6.2: OTP

While registering you will be given an OTP on your registered e-mail which you need to input here.

6.2 Dashboard



Figure 6.3: Dashboard

Dashboard is the homepage of this system. It will show the statistics of the leaves according to various categories as well as navigation to various pages.

6.3 Statistics

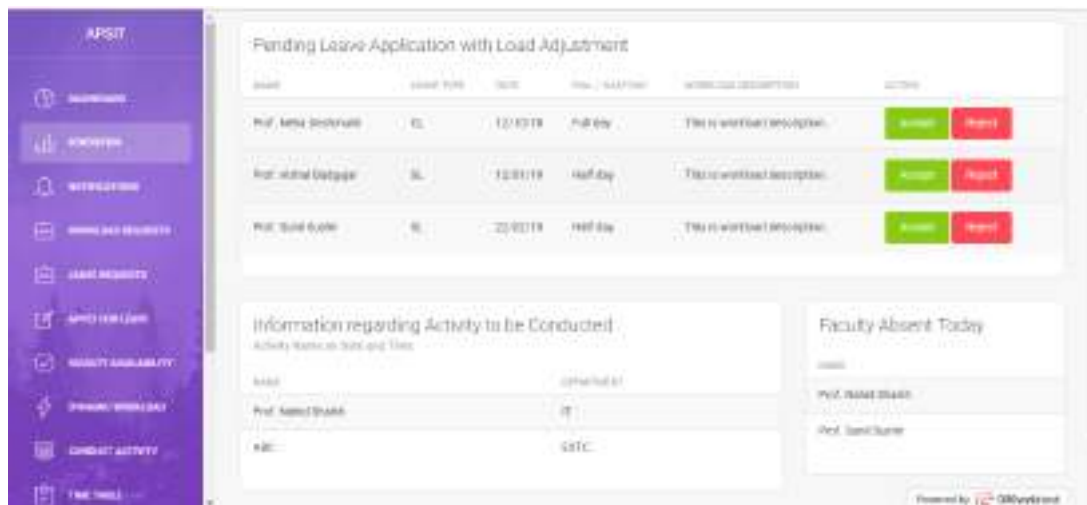


Figure 6.4: Statistics

Statistics will display pending load adjustments, extra activity and list of absent faculties.

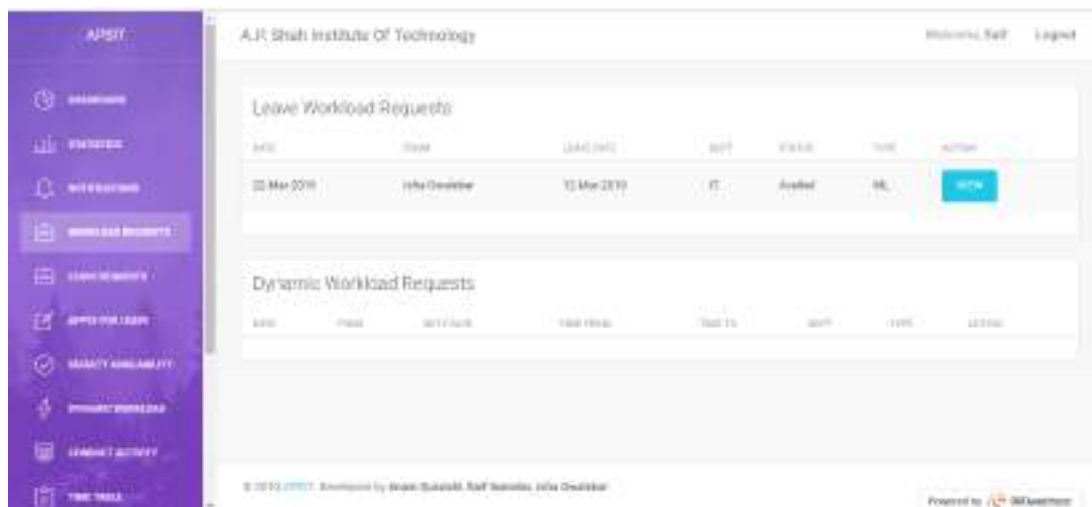
6.4 Notifications



Figure 6.5: Notifications

This will display the notifications of workload sharing, leave requests, extra activity scheduled.

6.5 Workload Requests



The screenshot displays the 'APJIT' system interface for 'A.P. Shah Institute Of Technology'. The sidebar on the left contains navigation links: Dashboard, Statistics, Notifications, Workload Requests (highlighted), Leave Requests, Approvals, Vacancy Availability, Employee Workload, Conduct Activity, and Time Table. The main content area features two tables. The first table, 'Leave Workload Requests', has columns for ID, Name, Leave Date, Shift, Status, Type, and Action. It contains one row for '12 Mar 2019' by 'John Doe' with status 'Approved' and a 'View' button. The second table, 'Dynamic Workload Requests', has columns for ID, Name, Shift Date, Total Hours, Total TS, Shift, Type, and Action, but it is currently empty. The footer indicates the system is powered by 'APJIT' and 'APJIT'.

ID	Name	Leave Date	Shift	Status	Type	Action
12 Mar 2019	John Doe	12 Mar 2019	PT	Approved	HL	<button>View</button>

ID	Name	Shift Date	Total Hours	Total TS	Shift	Type	Action
----	------	------------	-------------	----------	-------	------	--------

Figure 6.6: Workload Requests

This will display the dynamic requests of the employees for workload.

6.6 Leave Requests



The screenshot displays the 'APJIT' system interface for 'A.P. Shah Institute Of Technology'. The sidebar on the left contains navigation links: Dashboard, Statistics, Notifications, Workload Requests, Leave Requests (highlighted), Approvals, Vacancy Availability, Employee Workload, Conduct Activity, and Time Table. The main content area features a table titled 'Leave Requests' with columns for ID, Name, Leave Date, Shift, Status, Type, and Action. The table is currently empty. The footer indicates the system is powered by 'APJIT' and 'APJIT'.

ID	Name	Leave Date	Shift	Status	Type	Action
----	------	------------	-------	--------	------	--------

Figure 6.7: Leave Requests

This will display the leave requests of the employees.

6.7 Faculty Availability

The screenshot shows the 'List of Currently Available Faculty' section. It contains a table with the following data:

Name	Department	Time	Availability
Prof. Subh Rathi	Information Technology	10:00 - 11:00	Class Discussion
Dr. C.	IT	10:00 - 11:00	Lab Assistant
Dr. C.	Information	10:00 - 11:00	Professor

Below the table is a 'Check Free Slot of Faculty' section with the following form fields:

- Select Department:
- From (Date):
- To (Date):
- Time:
- Submit

At the bottom right, it says 'Powered by APSITech.com'.

Figure 6.8: Faculty Availability

It will display list of employee available at the moment.

6.8 Dynamic Workload

The screenshot shows the 'Dynamic Workload' section. It has three tabs: 'Outdoor Duty', 'Indoor Duty', and 'Exam Duty'. The 'Outdoor Duty' tab is selected. The form fields are:

- Name:
- Time From: To:
- Place & Purpose of Duty:
- Alternative Assignment:
- Submit Request

At the bottom right, it says 'Powered by APSITech.com'.

Figure 6.9: Dynamic Workload

It will display the workload assigned to the faculties at the moment. It comprises outdoor duty, indoor duty, exam duty.

6.9 Conduct Activity

Lab No.	Lab	Conductors
10	Aot-Rand-Saak	IT
10	ABC	JNTU

Figure 6.10: Conduct Activity

It will help to send notifications regarding extra activity scheduled for the employees. The higher authorities can write activity title, description, date and time, list of employees required for the activity.

6.10 Time Table

	8:00-9:00	9:00-10:00	10:00-11:00	11:00-12:00	12:00-1:00	1:00-2:00	2:00-3:00	3:00-4:00	4:00-5:00	5:00-6:00
MONDAY										
TUESDAY										
WEDNESDAY										
THURSDAY										
FRIDAY										

Figure 6.11: Time Table

The employees are supposed to fill the lab nos. and blank spaces shall be considered as free-slots.

6.11 User Requests

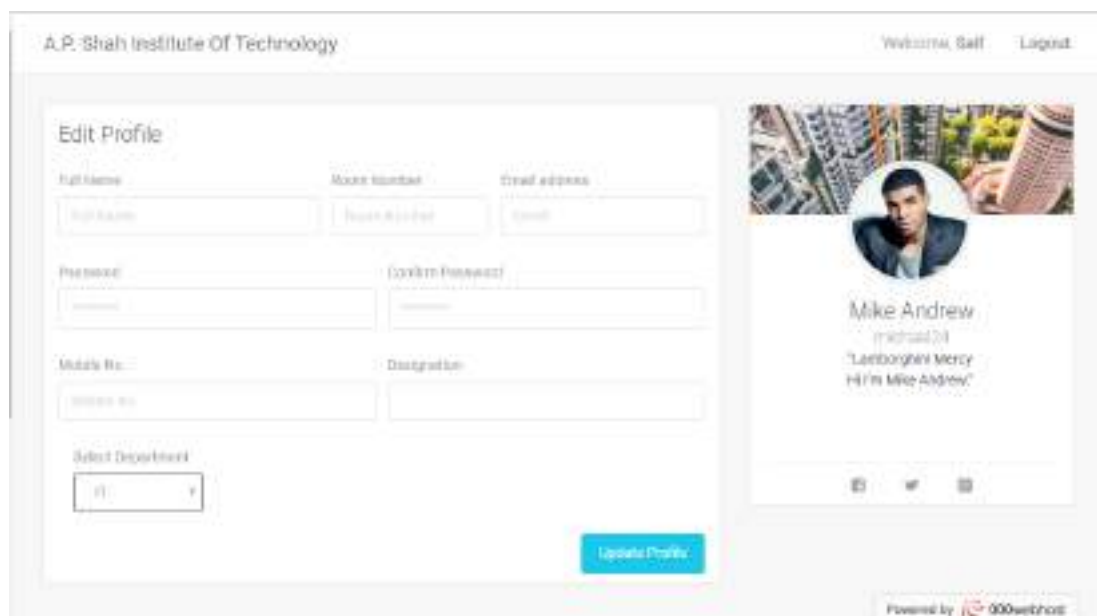


The screenshot shows the 'User Requests' module within the 'A.P. Shah Institute Of Technology' system. At the top right, there are links for 'Welcome Self' and 'Logout'. The main heading is 'User Requests'. Below it, there is a section titled 'New User Requests'. A table is displayed with the following columns: 'NAME', 'ID NO.', 'PHONE NO.', 'SUPPORTER', 'DEPT', and 'ACTION'. The first row of the table is highlighted in grey and contains the text 'No New Requests'.

Figure 6.12: User Requests

This sub-module is provided to the higher authorities which will help them approve the authorized employees only.

6.12 User profile



The screenshot shows the 'User Profile' module within the 'A.P. Shah Institute Of Technology' system. At the top right, there are links for 'Welcome Self' and 'Logout'. The main heading is 'Edit Profile'. The form contains several input fields: 'Full Name', 'Mobile Number', 'Email Address', 'Password', 'Confirm Password', 'Mobile No.', and 'Designation'. There is also a 'Select Department' dropdown menu. A blue 'Update Profile' button is located at the bottom right of the form. On the right side, there is a profile card for 'Mike Andrew' with a profile picture, a bio, and social media icons. The bio text reads: 'Mike Andrew', 'michael24', 'Lamborghini Mercy', 'Hi I'm Mike Andrew'.

Figure 6.13: User profile

The employees can update their personal data using this sub-module.

6.13 Reset

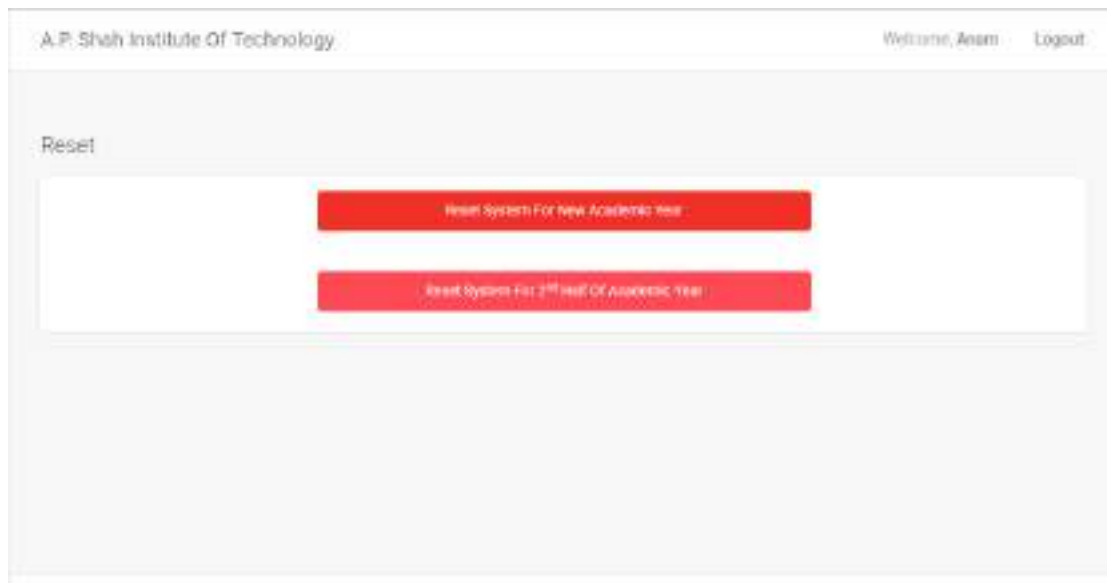


Figure 6.14: Reset

This privilege is given only to the principal who will reset the timetable after every semester.

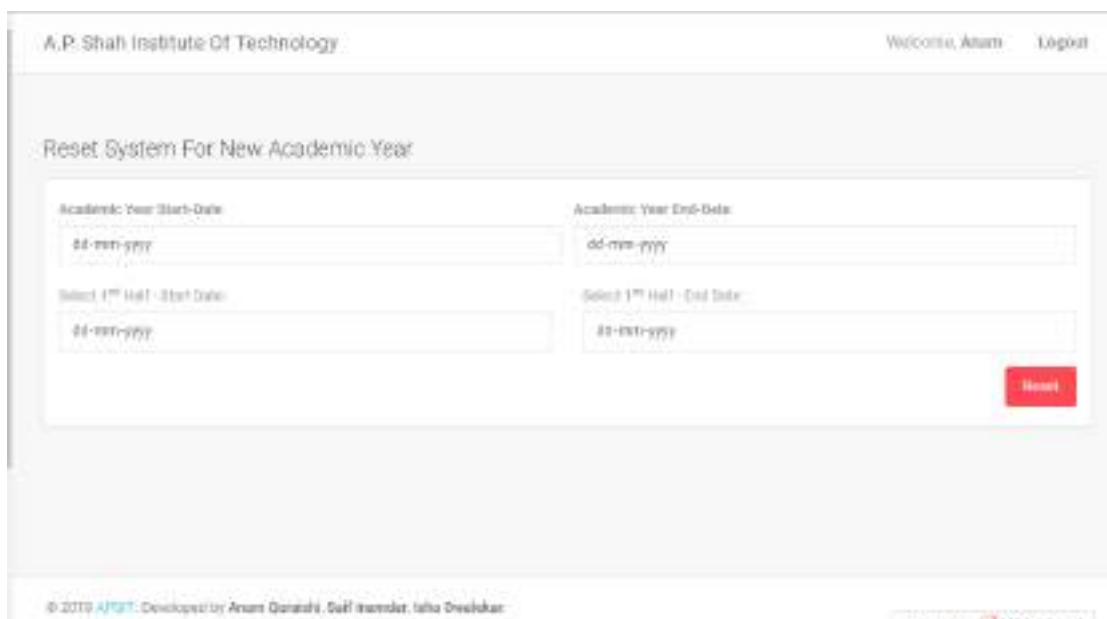


Figure 6.15: Reset for new academic year

The above interface is for 1st half academic year.

A.P. Shah Institute Of Technology

Welcome, Anant

Logout

Reset System For 2nd Semester:

Select 2nd half start Date:

Select 2nd half end Date:

Caution Record Will Be Carried Forward!

Reset

Figure 6.16: Reset for 2nd half

The above interface is for 2nd half academic year.

6.14 Code for Dynamic Workload

```
<?php
session_start();
include('session_verify.php');
include('dbconnect.php');

$dynamic_id= $_GET['viewform'];
$query="select * from `dynamic_duty` where id=$dynamic_id ";
$queryres=mysqli_query($conn,$query);
if(mysqli_num_rows($queryres)==1)
{
    $queryresult=mysqli_fetch_assoc($queryres);
    if($_SESSION['designation']=='HOD' || $_SESSION['designation']=='Principal' )
    {
        $applier_name=$queryresult['name'];
        $applier_dept=$queryresult['dept'];
        $applier_designation=$queryresult['designation'];
        $type=$queryresult['type'];
        $description=$queryresult['description'];
        $alternative_arrangement=$queryresult['alternative_arrangement'];
        $approval_status=$queryresult['approval_status'];
        $applied_on=new DateTime($queryresult['applied_on']);
        $duty_from=new DateTime($queryresult['duty_from']);
        $duty_to=new DateTime($queryresult['duty_to']);
    }
}
else{
    ?>
</script>
```

Figure 6.17: Dynamic Workload

The above code mentioned is written for dynamic workload interface which will be used only when the employees are given outdoor duty(dte maharashtra work,visiting CSI seminars.),indoor duty,exam duty based on the time-slots in which they would be available.The employees will be selected based on the parameters written in the above code.

6.15 Workload Approval

```
$leave_id= $_GET['viewform'];
$query="select * from leave_applications where id=$leave_id ";
$queryres=mysqli_query($conn,$query);
if(mysqli_num_rows($queryres)==1)
{
    $queryresult=mysqli_fetch_assoc($queryres);
    if($queryresult['selected_faculty']==$_SESSION['full_name'])
    {
        $applier_name=$queryresult['name'];
        if($queryresult['h_from']==' && $queryresult['f_from']!='')
        {
            $ld_from=new DateTime($queryresult['f_from']);
            $ld_to=new DateTime($queryresult['f_to']);
            $ld_fromview=$ld_from->format('d-M-Y');
            $ld_toview=$ld_to->format('d-M-Y');
            $availed_on=new DateTime($queryresult['applied_on']);
            $interval=($availed_on->diff($ld_to))->format('%a');
        }
    }
elseif($queryresult['f_from']==' && $queryresult['h_from']!='')
{
    $ld_from=new DateTime($queryresult['h_from']);
    $ld_to=new DateTime($queryresult['h_to']);
    $ld_fromview=$ld_from->format('d-M-Y H:i:s');
    $ld_toview=$ld_to->format('d-M-Y H:i:s');
    $availed_on=new DateTime($queryresult['applied_on']);
    $interval=($availed_on->diff($ld_to))->format('%a');
}
}
```

Figure 6.18: Workload Approval

The above mentioned code is used for workload approval that is, if any employee is not available for full day/half day/a slot then the code above ensures that the concerned employees will get a pool of employees of their designation to adjust their workload with.

6.16 Leave Requests

```
<?php
session_start();
include('session_verify.php');
$session_name=$_SESSION['full_name'];
$session_dept=$_SESSION['dept'];
include('dbconnect.php');
if($_SESSION['designation']=='HOD' || $_SESSION['designation']=='Principal')
{
    if($_SESSION['designation']=='HOD')
    {
        $query="SELECT * FROM `leave_applications` WHERE dept='$session_dept' AND approval_status='Waiting
for HOD Confirmation' ";
    }
    if($_SESSION['designation']=='Principal')
    {
        $query="SELECT * FROM `leave_applications` WHERE approval_status='Waiting for Principal Confirmation'
OR approval_status='Waiting for HOD Confirmation' ";
    }
    if($queryresult=mysqli_query($conn,$query))
    {
    }
}
else{
    ?>
    <script>
    var successcode = "Database ERROR";
    window.alert(successcode);
    location.href="homepage.php";
    </script>
    <?php
```

Figure 6.19: Leave Requests

The code above generates leave requests of the employees to the higher authorities for their approval.

6.17 Statistics

```
<th>LEAVE TYPE</th>
<th>DATE</th>
<th>FULL / HALF DAY</th>
<th>WORKLOAD DESCRIPTION</th>
<th>ACTION</th>

</thead>
<tbody>
<tr>
<td>Prof. Neha Deshmukh</td>
<td>CL</td>
<td>12/10/18</td>
<td>Full day</td>
<td>This is workload description.</td>
<td>
<a href="approval.php">
<button type="submit"
class="btn btn-success btn-fill">Accept</button>
</a>
<button type="submit" class="btn btn-danger btn-fill">Reject</button>
</td>
</tr>
<tr>
<td>Prof. Vishal Badgujar</td>
<td>SL</td>
<td>12/01/19</td>
<td>Half day</td>
<td>This is workload description.</td>
<td>
<button type="submit" class="btn btn-success btn-fill">Accept</button>
```

Figure 6.20: Statistics

This part of code will display the live statistics of the faculties present in the respective department to their HODs. The principal will have an authority to view live statistics about the employees present in various departments. The real-time data which will be shown would be of the employees who are absent, who have pending leave requests and information regarding the activities to be conducted.

6.18 Time Table

```
`sem2_from` AND
`sem2_to` ORDER BY id DESC LIMIT 1 ";
$queryresult1=mysqli_query($conn,$query1);
$queryresult2=mysqli_query($conn,$query2);
if(mysqli_num_rows($queryresult1)==1)
{
    $current_sem="1<sup>st</sup> Semester";
    $queryres1=mysqli_fetch_assoc($queryresult1);
    $current_sem_from=new DateTime($queryres1['sem1_from']);
    $current_sem_to=new DateTime($queryres1['sem1_to']);
    $display_current_from=$current_sem_from->format('d-F-Y');
    $display_current_to=$current_sem_to->format('d-F-Y');
    $date_interval=$current_sem_from->diff($current_sem_to);
    $interval=$date_interval->format('%a')+1;
}
elseif(mysqli_num_rows($queryresult2)==1)
{
    $current_sem="2<sup>nd</sup> Semester";
    $queryres2=mysqli_fetch_assoc($queryresult2);
    $current_sem_from=new DateTime($queryres2['sem2_from']);
    $current_sem_to=new DateTime($queryres2['sem2_to']);
    $display_current_from=$current_sem_from->format('d-F-Y');
    $display_current_to=$current_sem_to->format('d-F-Y');
    $date_interval= $current_sem_from->diff($current_sem_to);
    $interval=$date_interval->format('%a')+1;
}
else{
    echo "Error".mysqli_error($conn);
    $current_sem="Database Error";
}
```

Figure 6.21: Time Table

The above code generates sessions for academic timetable of the employees. The employees are supposed to fill this timetable at the start of every semester.

Chapter 7

Test Report

Testing : The aim of testing is to prove that the developed system addresses the predefined business requirements and will perform reliably and efficiently when running live. The testing methodology used here are **alpha** and **ad-hoc** testing. Since, alpha testing is conducted at the developers site, in-house virtual user environment was created for this type of testing. Moreover, the objective behind the ad-hoc testing is it is conducted with no reference to the test case and also without any plan or documentation in place.

7.1 Test Scenario 1



Figure 7.1: Invalid Credentials

Test Case : The input provided by User.

Expected : Valid Credentials.

Actual : Invalid Credentials.

Status : Registration Failed

7.2 Test Scenario 2



Figure 7.2: Valid Credentials

Test Case : The input provided by User.

Expected : Valid Credentials.

Actual : Valid Credentials.

Status : Registration Successful

7.3 Test Scenario 3



Figure 7.3: User not found

Test Case : Login ID and Password provided by User.

Expected : Valid Credentials.

Actual : Invalid Credentials.

Status : Login Failed

7.4 Test Scenario 4



Figure 7.4: Dashboard

Test Case : Login ID and Password provided by User.
Expected : Valid Login ID and Password.
Actual : Valid Login ID and Password.
Status : User is directed to Dashboard.

7.5 Test Scenario 5

Leave Application

Type:

Status of leave:

Duration of leave:

From: To:

Reason For Leave:

Figure 7.5: Apply for Leave

Test Case : Apply for leave.
Expected : Fill all the Details for Leave with workload Description
Actual : Detail such as Type, Status, Duration, Reason, Workloadload sharing description is filled by the Employee who is willing to apply for leave.
Status : Workload description is sent to the other employee for workload acceptance or rejection.

7.6 Test Scenario 6

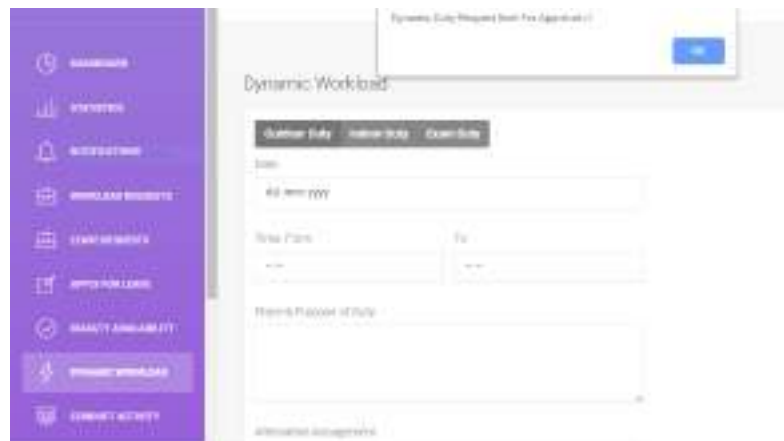


Figure 7.6: Dynamic Workload

Test Case : Apply for Dynamic Workload.

Expected : Type of Duty and Detail regarding the duty must be filled.

Actual : Detail such as Type of Duty must be selected from options - Indoor Duty / Outdoor Duty / Exam Duty. Date, time, place and purpose of duty, Alternative Arrangement must be filled.

Status : Dynamic Duty request will be sent for approval.

7.7 Test Scenario 7



Figure 7.7: Time-Table

Test Case : Enter Timetable.

Expected : Lab number or class room Number must be filled in the given fields and empty slots will be considered as free slot in timetable.

Actual : Time-Table entered as per Employee's schedule.

Status : Time-Table will be stored in the database.

Chapter 8

Conclusion

The work presented in this report is related to employee leave and availability management system. This web system is designed to tackle the problems faced in an organization regarding availability of employees. It will manage all the leave applications, load sharing requests and employee availability in an organization. It will manage the static workload as well as the dynamic workload efficiently. Dynamic workload and activities will be notified to them with the help of notifications through e-mails/SMS. It will also display statistics to employees and higher authorities which will help them in making decisions better. It aims to increase productivity and efficiency of employees in an organization. Besides, the system developed is flexible and modular.

Bibliography

- [1] Attendance Management - <https://www.greythr.com/pricing/> , 1 July 2018.
- [2] Management Solutions - <https://advancesystemsinc.com/absence-management/> , 1 July 2018.
- [3] Leave Management - <https://www.zoho.com/people/leave-management-system.html> , 1 July 2018.
- [4] sumhr Leave tracking - <https://www.sumhr.com/online-leave-policies-leave-application-tracking-2> , 1 July 2018.

Acknowledgement

We have great pleasure in presenting the report on **ERP System For Employee Management**. We take this opportunity to express our sincere thanks towards our guide **Prof. Kiran Deshpande** & Co-Guide **Prof. Neha Deshmukh** Department of IT, APSIT thane for providing the technical guidelines and suggestions regarding line of work. We would like to express our gratitude towards his constant encouragement, support and guidance through the development of project.

We thank **Prof. Kiran B. Deshpande** Head of Department, IT, APSIT for his encouragement during progress meeting and providing guidelines to write this report.

We thank **Prof. Vishal S. Badgujar** BE project co-ordinator, Department of IT, APSIT for being encouraging throughout the course and for guidance.

We also thank the entire staff of APSIT for their invaluable help rendered during the course of this work. We wish to express our deep gratitude towards all our colleagues of APSIT for their encouragement.

Saif Inamdar
16204022

Anam Quraishi
16204017

Isha Owalekar
15104011

Publication

Paper entitled “**Employee Availability at Fingertips**” is presented at “**International Conference on Advances in Science, Technology and Engineering**” (ICASTe) 2019 by “**Saif Inamdar**”, “**Anam Quraishi**”, “**Isha Owalekar**”.