INSTITUTE FOR
ADVANCED
COMPUTING AND
SOFTWARE
DEVELOPMENT
AKURDI, PUNE

Documentation On

**"EMPLOYEE PAYROLL
MANAGEMENT"**
E-DAC MAY 2021

Submitted By:
Group No: 74
TANVI MADHALE 1102
MADHAVI BANSODE 1103

Prashant Karhale   **Mr. Akshay Parab**
Centre Coordinator   **Project Guide**

Table of Contents

List of Diagrams

ABSTRACT

"Employee Payroll Management System" is an internet based Java application that automates the working of a company or work center that manage and maintain records of the employees in the different department This application is created as a product and can be customized according the specific needs of the client.

The objective of "Employee Payroll Management System" is designing a scheduling system for a work centre. Scheduling is such a tool with which the process of intimating activities and notifications will be easy and even online in the organization where it is installed. But these task of scheduling the different activities if manually done whether they may be personal or official is time consuming and also may lead to confusion if not properly scheduled.

The supervisor which holds the various activities like sending notifications, mark attendance , and deleting the employees and on the other hand employees view their details, view schedule or the notifications or any message form supervisor and view attendance.

1.1 INTRODUCTION

The objective of "Employee Payroll Management System" is designing a **Scheduling System** for a work centre. **Scheduling** is such a tool with which the process of intimating activities and notifications will be easy and even online in the organization where it is installed. But these task of scheduling the different activities if manually done whether they may be personal or official is time consuming and also may lead to confusion if not properly scheduled.

Scheduling becomes such an easy task such that it reduces much time when compared to previous methods. This enables for the employee to check the task that is assigned to them.

The **Admin** is already having the account on the server and therefore will have login **username and password.** The Admin is allowed to delete employees, set work or task, mark attendance, Add salary and manage leaves etc. The **Employee** will have to register in the database and once registered he/she would be logged in to the system through his/her employee id and password. The employee is allowed to update his/her details, view schedule set by the Admin, view Time Sheet etc**.**

1.1.1 Architecture

"Employee Payroll Management System" will be adopting **3-Tier Architecture.** The front- end will be HTML pages with Java Script for client side validation where as all business logics will be in Java reside at middle layer. And these layers will interact with third layer of database, which will be MYSQL database. The web server will be Apache Tomcat 8.5. To start working on this project environment required is a server having Tomcat as web server, Oracle as database and Java Runtime Environment (JRE) as development environment.

1.1.2 Platform

"Employee Payroll Management System" software shall be designed and developed on Open Platform i.e. J2EE. MYSQL Server shall be used to maintain the database. To achieve a high degree of modularity, scalability and maintainability , it is recommended to adopt n- tier architecture while designing Employee Payroll Management System. Accordingly, it is proposed that the entire application development logic, the database logic and presentation logic shall be segregated.

It is proposed that EMS software package shall be running on Apache Tomcat Web Server. The Web Server will be responsible for rendering the JSP pages and result is shown back to the end-user.

1.2. OBJECTIVE

This report documents the process of designing, building and testing a software system to be used in a company. The piece of software, and therefore the project, is known as a **"Employee Payroll Management System".** This project basically includes two modules i.e. Supervisor and Employee.

The system will do the following:

1)Admin:

- Admin can Add/Edit/delete the employees.
- Admin can Add/Edit/delete the schedule the work of the employees.
- Admin can Add/Edit/delete mark the attendance of employees.
- Admin can Add/Edit/delete Leaves and time sheet of the employee.

2) Employee:

- Registration according to category (developer, tester and designer).
- Employee can view his/her schedule set by Admin.
- Employee can check his/her attendance.
- Employee can update his/her details.
- Employee can re-set password.

1.3. NEED OF THE SYSTEM

The existing system is based on the manual work carried out by the different department, where you have to do all jobs manually and do not allow the automation of system and transparency to all users of system.

Disadvantage of existing system is that everything is on paper, like for fixing the schedule of the particular employee, to do so either that employee should be in contact or should be known by some other source resulting in the wastage of time. So, this whole task of scheduling the activities whether may be personal or official is very time consuming and may lead to confusion if not properly scheduled. The system which was developed now, makes this process of scheduling much easier and computerized. By this system the **manager or top level designated** employee can fix the schedule of any employee working under him. Thus the top level management can easily fix the process of scheduling, and even can change the appointment which is reflected immediately to the related employee avoiding direct contact of the employee resulting in saving a lot of time and work overhead.

1.4 ADVANTAGE OF THE SYSTEM

This project offers employees to enter the data through simple & interactive forms. This is very helpful for the client to enter the desired information through so much simplicity. New Employees when registering themselves register according to category wise (i.e. Developer, Tester and Designer).Here, Employee Id is automatically generated once the user got registered. So, user logged in to the system and access to its profile and rights using that Employee Id and password.

The front view of the system consists of two modules i.e. Supervisor and Employee. When clicking on the Employee tab the pop up window would open up and ask for the login, if the employee is not yet register then clicking on **"New User Register Here"**, user can register themselves. After registering they can logged in to the system using password and Employee Id which is automatically generated after registration and access their rights like view schedule, view attendance, update details, view details etc.

Supervisor can logged in to the system using username and password and can set

schedule, mark attendance, check category wise description of the employees etc. Hence, the system has various advantages:

- Transparency to all the user of system.
- Less paper use and removal of redundancy.
- Less prone to errors.
- The whole system is interactive.[9]

ANALYSIS AND FEASIBILITY OF STUDY

2.1 INTRODUCTION

The objective of **"Employee Payroll Management System"** is designing a **Scheduling System** for a work centre. **Scheduling** is such a tool with which the process of intimating activities and notifications will be easy and even online in the organization where it is installed. But these task of scheduling the different activities if manually done whether they may be personal or official is time consuming and also may lead to confusion if not properly scheduled.

Scheduling becomes such an easy task such that it reduces much time when compared to previous methods. This enables for the employee to check the task that is assigned to them.

System Analysis is the first and foremost phase for the large scale development project, and for this to achieve we start from the basic need of the system mentioned above. In this phase, we normally jotted down all the exact requirements of the system. This task may be performed by the customer, the developer, a marketing organization, or any combination of the three. System Analysis process generally involves these two phases: **problem understanding or analysis and requirement specification**.

In System Analysis, the analyst has to understand the problem and its whole descriptions and definations .And on the other hand, In requirements analysis the analyst has to analyse the wholesome requirements of the system which is to be developed.

2.2 IDENTIFICATION OF NEED

This is the most indispensable phase of the system which is to be developed, In this firstly we have mentioned our need which we want to develop. Here, the need and specification phase of system analysis is done to exactly find out the need and the requirements by the customers, and hence all the requirements is collected by the customers.

2.3 PRELIMINARY INVESTIGATION

To evaluate and to define the problem in hand quickly, the preliminary investigation is carried out, to see if it is worthy of the following study and also it suggests some courses of actions if possible.

Following steps are involved in the preliminary investigation:

- The Problem Understanding
- Determining the project boundaries and constraints
- Feasibility study
- Estimation of the time and cost.
- Documentation of Preliminary Report.

2.4 FEASIBILITY OF STUDY

Feasibility study generally determines the need and solutions considered to accomplish the requirements are practically implementable in the software or not, information such as availability of the resource, estimation of cost for the development of the project and the cost which would be incurred on maintenance of the project is carried out in feasibility study.

There are different types of feasibility:

- Technical Feasibility
- Operational Feasibility
- Economic Feasibility

2.4.1 Operational Feasibility

- This site is operational feasible because in this all users can easily operate access the facilities and module meant for according to the type of user
- The well-planned architecture would ensure the optimal utilization of the resources and will be secure for threats.
- Thus provides easy access to all the users with their registered mail Id and password.

2.4.2 Technical Feasibility

Project is technical feasible due to following reasons:

- This site is technical feasible because in this site, technology which is used to develop the site is efficient and is easily upgraded time to time and separated module makes it easy to implement and maintenance.
- Technical guarantees of accuracy, reliability, ease of access and the data security.
- The database's purpose is to create, establish and maintain a workflow among various entities in order to facilitate all concerned users in their various capacities or roles.

2.4.3 Economical Feasibility

Project is technical feasible due to following reasons:

- The system is economically feasible and based on all freely licensed software. It does not require any additional hardware or software. There is nominal expenditure and economical feasibility for certain.
- This can be added to the official website of the college/institution as a module and does not require any separate space.[8]

2.5 PROJECT PLANNING

It is a process which includes the activities required for the successful completion of the project. Project planning generally prevents obstacles that arise in the project such as non-availability of the resources and it also determines project constraints.

Planning is generally done by the project and senior management team.

Senior Management is responsible for employing team members whereas the project management is responsible for making decisions and planning.

In this system also planning is executed for developing the whole project and meeting the requirements of the user.

DESIGN AND ARCHITECTURE

3.1 INTRODUCTION

The purpose of System Design is to create a technical solution that satisfies the functional requirements for the system. At this point in the project lifecycle there should be a Functional Specification, written primarily in business terminology, containing a complete description of the operational needs of the various organizational entities that will use the new system.

Design is the first step in the development phase for any engineered product or system. The designer's goal is to produce a model or representation of an entity that will later be built. At Beginning, once system requirement have been specified and analyzed, system design is the first of the three technical activities -design, code and test that is required to build and verify software.

The importance can be stated with a single word "Quality". Design is the place where quality is fostered in software development. Design provides us with representations of software that can assess for quality. Design is the only way that we can accurately translate a customer's view into a finished software product or system.

During design, progressive refinement of data structure, program structure, and procedural details are developed reviewed and documented. System design can be viewed from either technical or project management perspective.

Systems design is the process or art of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements.[7]

3.2 MODULARIZATION DETAIL

It is a technique of dividing the whole system into smaller independent modules.

There are two modules in our project:

1. **Supervisor Module:-** Supervisor is responsible to:

- Delete registered employees.
- View category wise details.
- Set schedule of the employees.
- Mark the attendance of particular employee.

2. **Employee Module:-** Employee can:

*   Register
*   Update his/her details.
*   View Details.
*   View schedule set by the supervisor.
*   View his/her attendance.
*   Re-set password.

3.3 DATA FLOW DIAGRAM

In an Information system, the flow of the data around the system is graphically represented by the data flow diagram.

A graphical tool used to describe and analyze the moment of data through a system manual or automated including the process, stores of the data and delays in the system. Data flow diagram the central tool and the basis from which other components are developed.

DFDs are the model of the proposed system. They clearly show the requirements on which the new system should be built. Later during the design activity this is taken as the basis for drawing the system's Structure charts.

The various components of DFDs are:

Dataflow: Data movement form the source to destination is shown by the arrows.

Process: The various activities and the actions performed on the data is represented through circle.
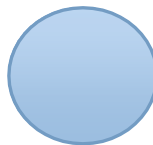
Entities: External sources or information of the data is represented by rectangle.

3.3.1 Zero Level Data Flow Diagram:

F**ig 3.1: Zero-level DFD**

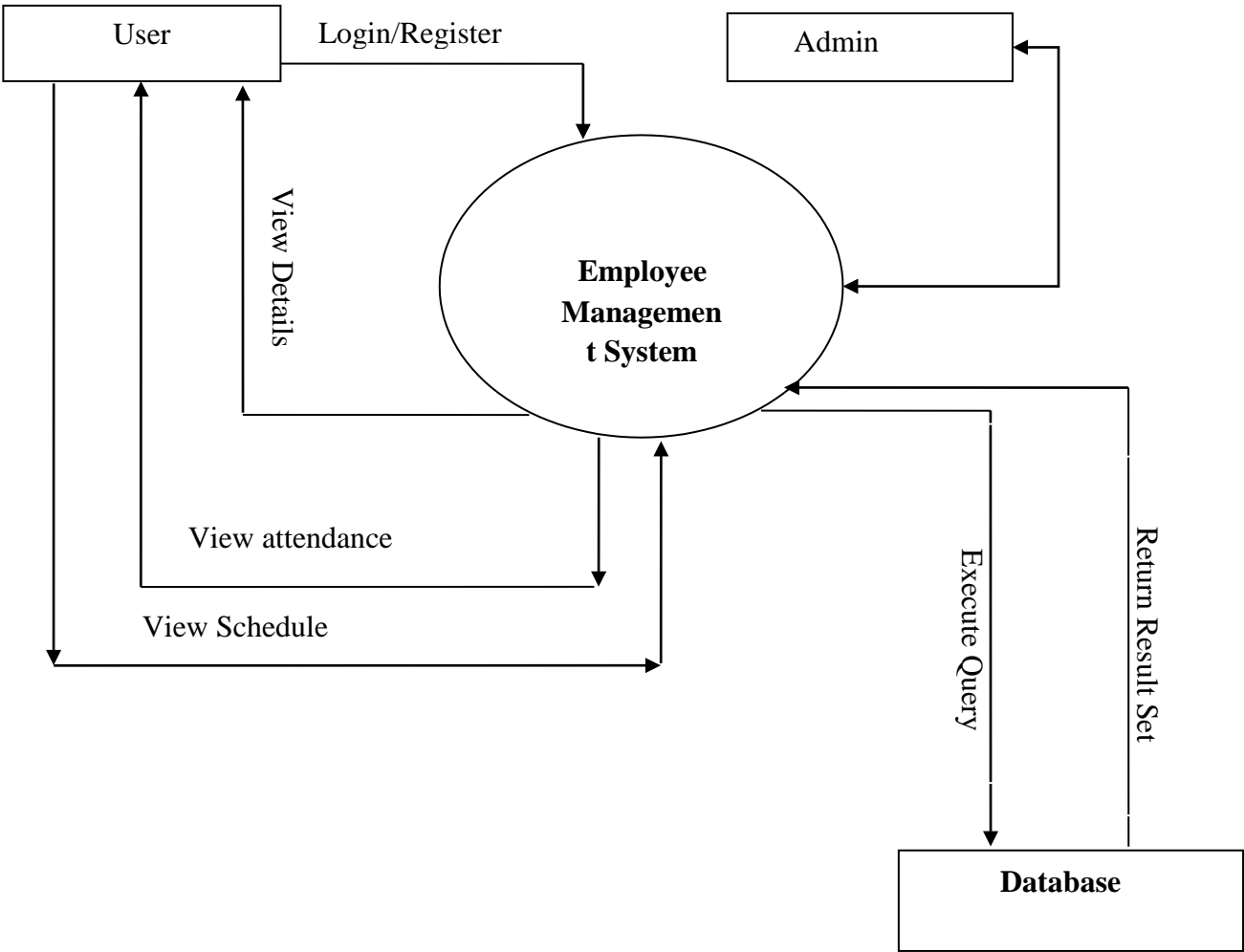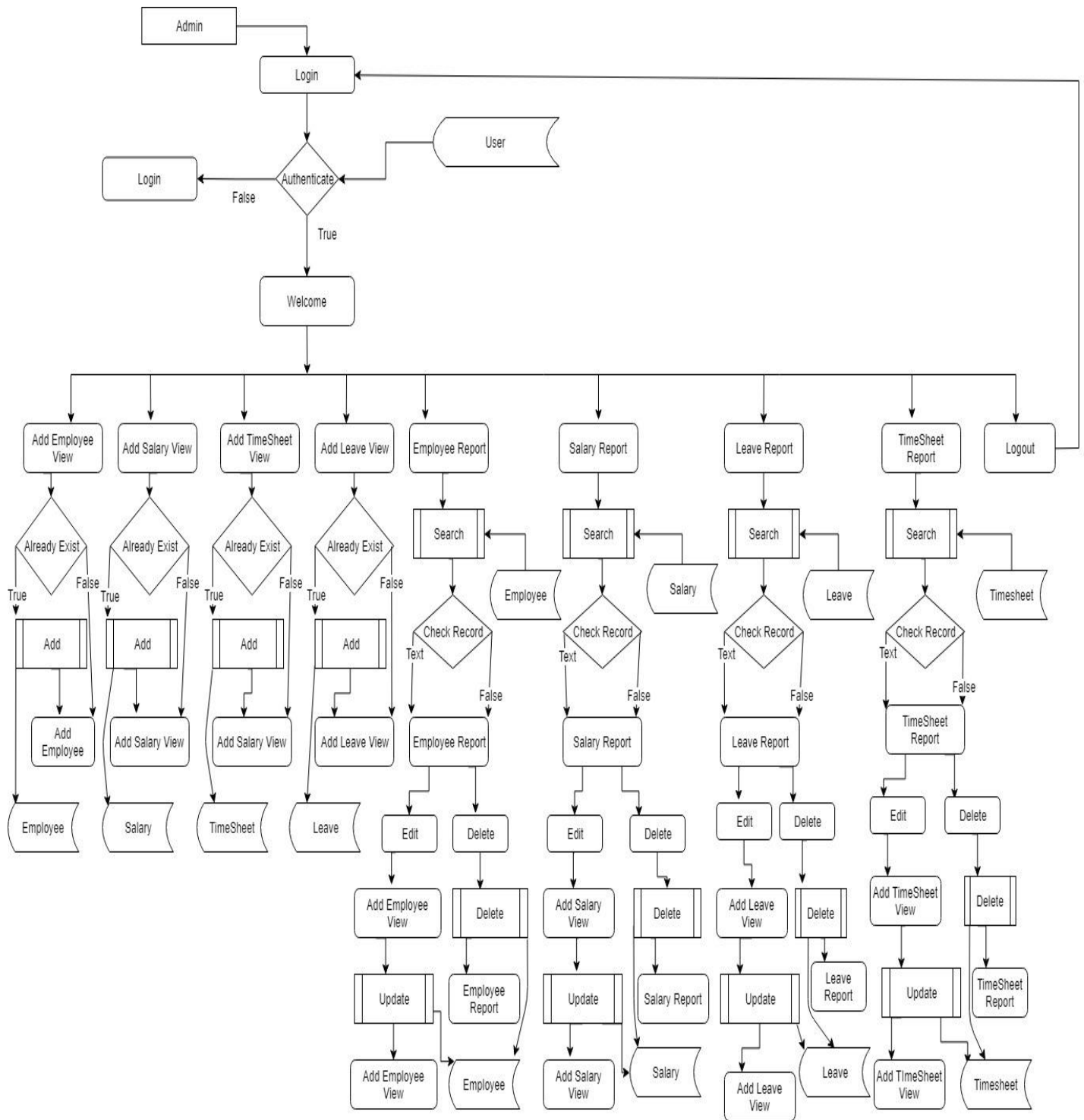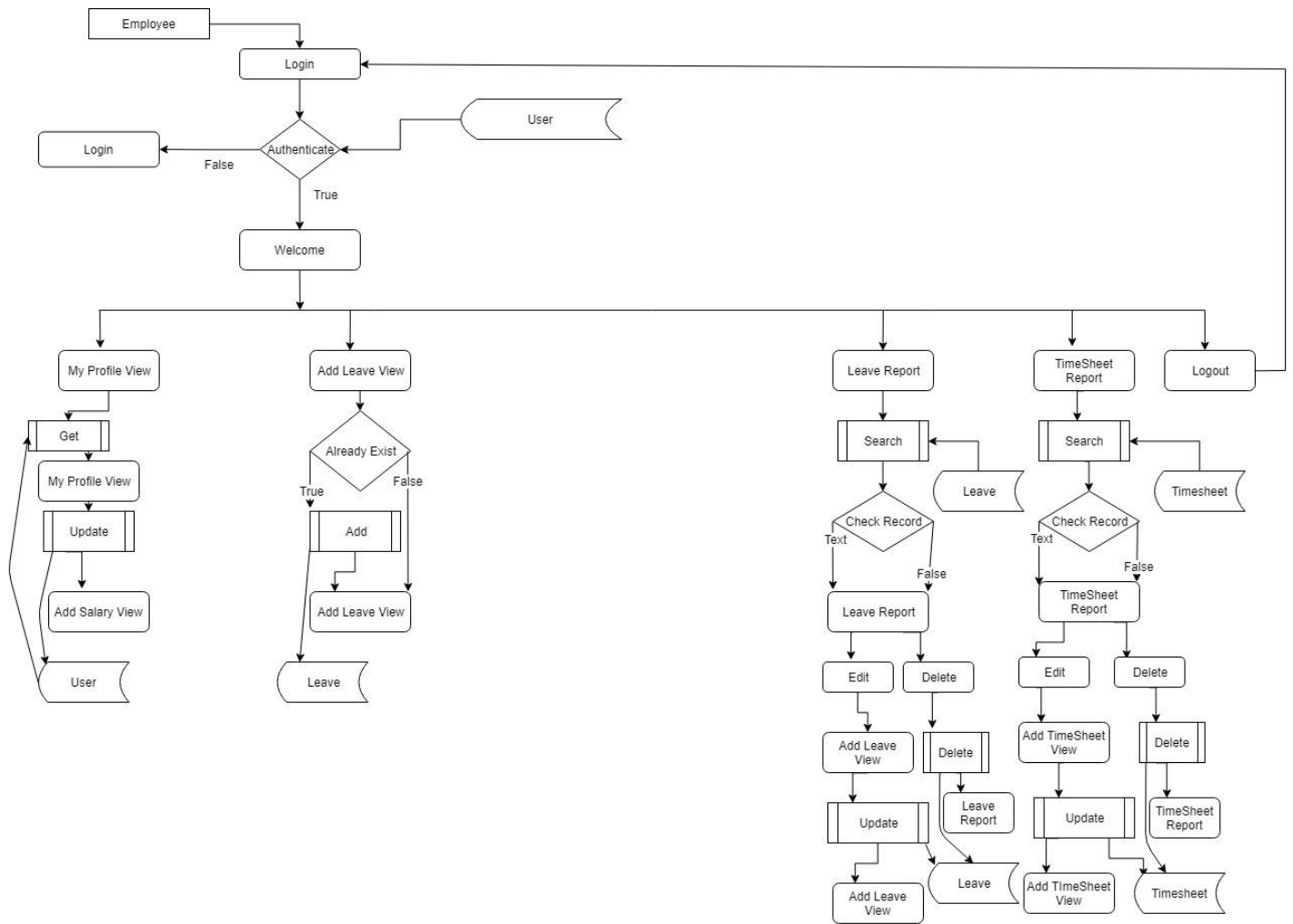**3.3.2 First Level Data Flow Diagram**

User

Login/Register

Admin

View Details

Employee Managemen t System

View attendance

View Schedule

Execute Query

Return Result Set

Database

Fig-3.2: First level DFD

DFD For Admin:

DFD for Employee

Employee → Login

Login → Authenticate

User → Authenticate

Authenticate —False→ Login

Authenticate —True→ Welcome

Welcome branches to:
- My Profile View
- Add Leave View
- Leave Report
- TimeSheet Report
- Logout

My Profile View:
My Profile View → Get → My Profile View → Update → Add Salary View → User

Add Leave View:
Add Leave View → Already Exist
- True → Add → Add Leave View → Leave
- False → Add Leave View

Leave Report:
Leave Report → Search ← Leave
Search → Check Record
- Text → Leave Report
- False → (Edit / Delete)
  - Edit → Add Leave View → Update → Add Leave View / Leave
  - Delete → Delete → Leave Report / Leave

TimeSheet Report:
TimeSheet Report → Search ← Timesheet
Search → Check Record
- Text → TimeSheet Report
- False → (Edit / Delete)
  - Edit → Add TimeSheet View → Update → Add TimeSheet View / Timesheet
  - Delete → Delete → TimeSheet Report / Timesheet

Logout → Login

3.4 E-R DIAGRAM

An E-R model is an abstract way to describe a database. Describing a database usually starts with a relational database, which stores data in tables. Some of the data in these tables point to data in other tables - for instance, your entry in the database could point to several entries for each of the phone numbers that are yours. The ER model would say that you are an entity, and each phone number is an entity, and the relationship between you and the phone numbers is 'has a phone number'. Diagrams created to design these entities and relationships are called entity–relationship diagrams or ER diagrams.
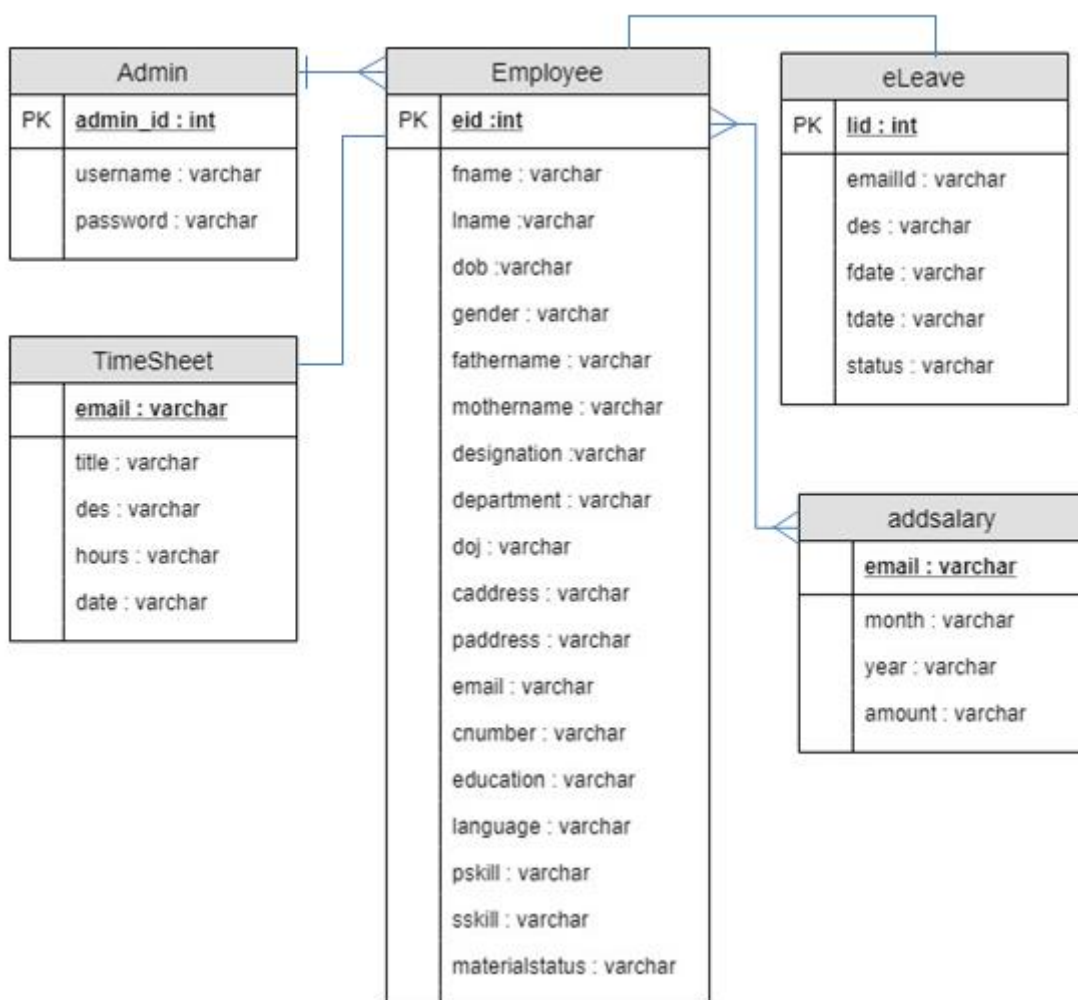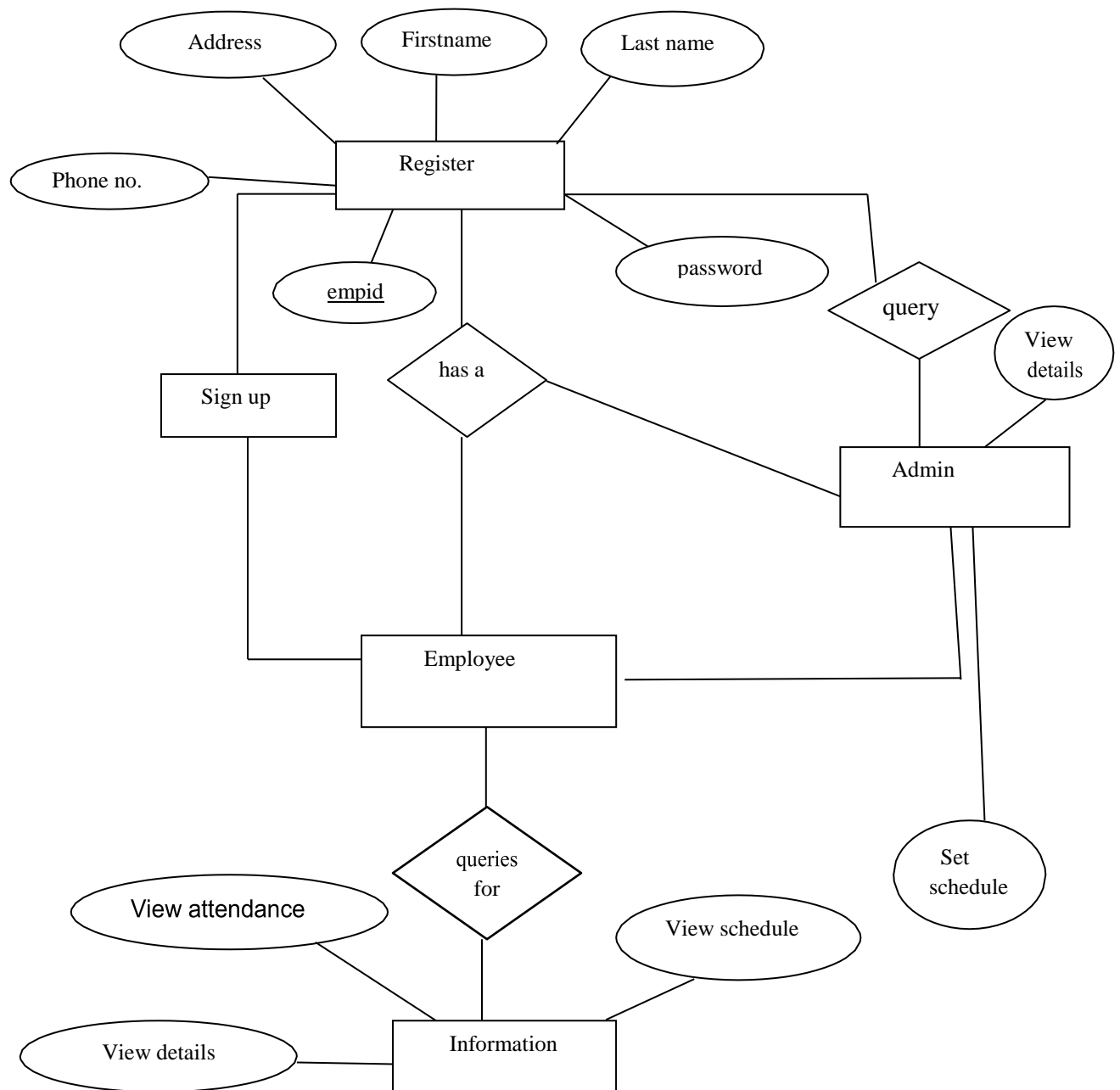
Entity Relationships are three kinds:

1. One-One

2. One-Many

3. Many-Many

1. One-One : One instance of an entity (A) is associated with one other instance of another entity (B). For example, in a database of employees, each employee name (A) is associated with only one social security number (B).

2. One-Many: One instance of an entity (A) is associated with zero, one or many instances of another entity (B), but for one instance of entity B there is only one instance of entity A. For example, for a company with all employees working in one building, the building name (A) is associated with many different employees (B), but those employees all share the same singular association with entity A.

3. Many-Many: One instance of an entity (A) is associated with one, zero or many instances of another entity (B), and one instance of entity B is associated with one, zero or many instances of entity A. For example, for a company in which all of its employees work on multiple projects, each instance of an employee (A) is associated with many instances of a project (B), and at the same time, each instance of a project (B) has multiple employees (A) associated with it.

Admin

PK	admin_id : int
	username : varchar
	password : varchar

TimeSheet

email : varchar
title : varchar
des : varchar
hours : varchar
date : varchar

Employee

PK	eid :int
	fname : varchar
	lname :varchar
	dob :varchar
	gender : varchar
	fathername : varchar
	mothername : varchar
	designation :varchar
	department : varchar
	doj : varchar
	caddress : varchar
	paddress : varchar
	email : varchar
	cnumber : varchar
	education : varchar
	language : varchar
	pskill : varchar
	sskill : varchar
	materialstatus : varchar

eLeave

PK	lid : int
	emailId : varchar
	des : varchar
	fdate : varchar
	tdate : varchar
	status : varchar

addsalary

| |
|---|
| email : varchar |
| month : varchar |
| year : varchar |
| amount : varchar |

Fig 3.3: Entity Relationship Diagram

3.5 USE CASE DIAGRAM

To model a system the most important aspect is to capture the dynamic behaviour. To clarify a bit in details, dynamic behaviour means the behaviour of the system when it is running operating. So only static behaviour is not sufficient to model a system rather dynamic behaviour is more important than static behaviour. In UML there are five diagrams available to model dynamic nature and use case diagram is one of them. Now as we have to discuss that the use case diagram is dynamic in nature there should be some internal or external factors for making the interaction. These internal and external agents are known as actors. So use case diagrams are consists of actors, use cases and their relationships. The diagram is used to model the system/subsystem of an application. A single use case diagram captures a particular functionality of a system. So to model the entire system numbers of use case diagrams are used.

The purpose of use case diagram is to capture the dynamic aspect of a system. But this definition is too generic to describe the purpose. Because other four diagrams activity, sequence. So we will look into some specific purpose which will distinguish it from other four diagrams.

The purposes of use case diagrams can be as follows:

3.5.1 Used to gather requirements of a system.

3.5.2 Used to get an outside view of a system.

3.5.3 Identify external and internal factors influencing the system.

3.5.4 Show the interacting among the requirements are actors.[5]
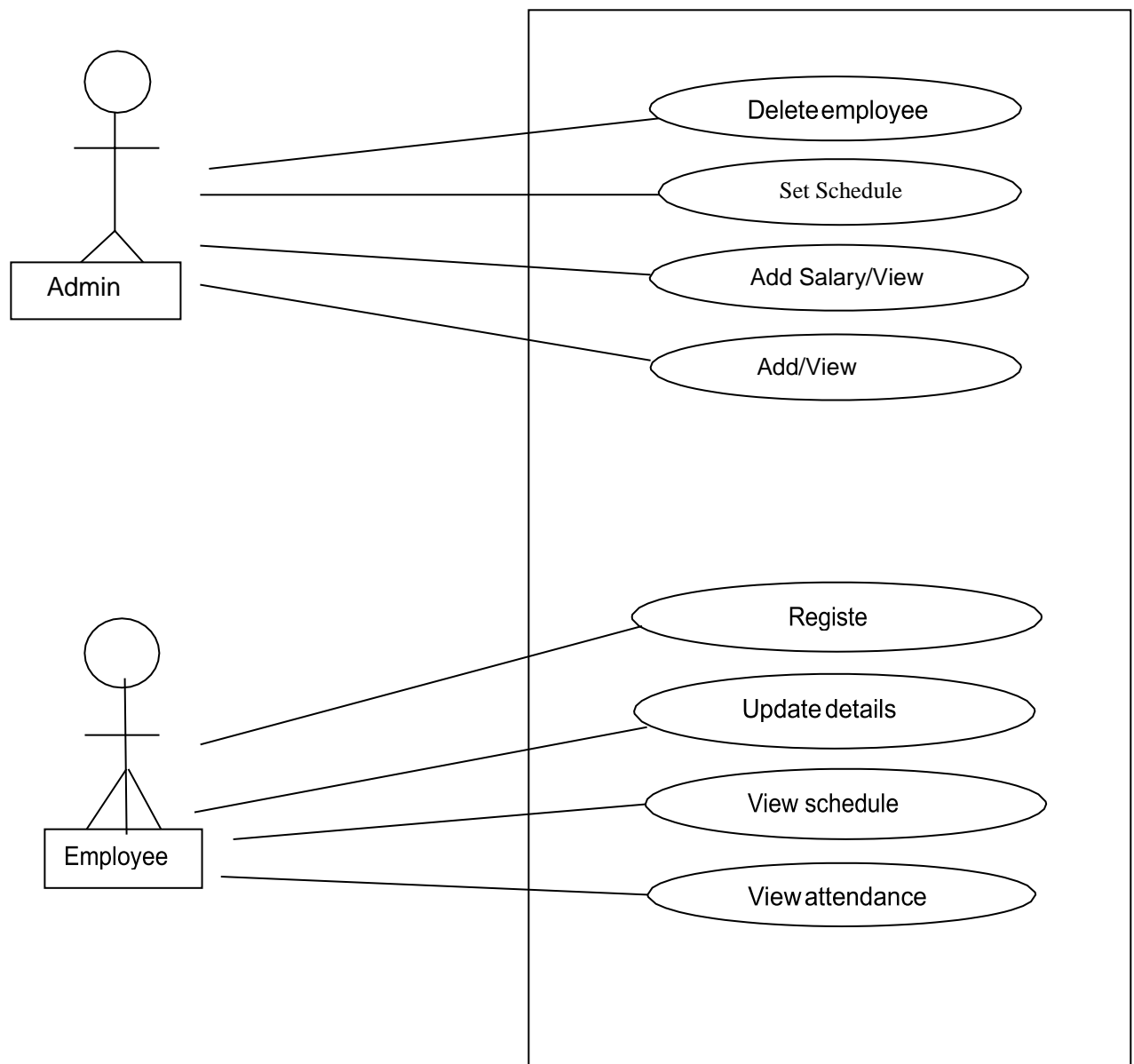
Fig-3.4: Use Case Diagram

3.6 DATA INTEGRITY AND CONSTRAINTS

Data Integrity

It is way through which access to the software or data can be controlled by the unauthorized persons. Through data integrity we can enforce data integrity rules on the data of the database. This also prevents and takes measures to protect the data from the various attacks. Attacks can be made on all three components of software: Programs, data and software. To measure integrity two additional attributes must be defined that is threats and security. The system under consideration does not allow the unauthorized access to the data as well as the document.

Data Constraints

Constraints means we enforce some rules to enter the data in a correct formatted way which gives the desired output to the user. This attribute measures the system ability to calculate and generate the desired and the correct output for the user.

- Here, this system provides both integrity and constraints to the user.

- In this system, wrong data entry can't be permissible as no fields cannot be left blank, password should be greater than 6 digits etc.

3.7 DATABASE DESIGN AND TABLES

Database is the collection of the data, the database used for this system is **Oracle Server.** Oracle allocates logical database space for all data in a database. The units of database Allocation are data blocks, extents and segments. Oracle manages the space in data files of database in units called **data blocks.** Data block is the smallest unit of data in database.

The snapshots of the structure of tables used in this system are given below:

3.7.1 Description of Tables used in Employee Management System

Table 3.1: Attendance

Description: Table used to store attendance of employee.

S. No.	Field name	Data Types	Constraint/Description
1.	empid	Number	Primary Key, Auto Generated
2.	firstname	Varchar2	Not Null
3.	attend	Varchar2	Not Null
4.	attenddate	Date	Not Null

Table 3.2: Empregister

Description: Table to register the employees.

S. No.	Field name	Data Types	Constraint/Description
1.	empid	Number	Primary Key, Auto Generated
2.	firstname	Varchar2	Not Null
3.	lastname	Varchar2	Not Null
4.	email	Varchar2	Not Null
5.	password	Varchar2	Not Null
6.	dob	Varchar2	Not Null
7.	category	Varchar2	Not Null
8.	phone	Number	Not Null
9.	address	Varchar2	Not Null
10.	identity	Varchar2	Not Null

Table 3.3: Emp_event

Description: Table that set the schedule of employees.

S. No.	Field name	Data Types	Constraint/Description
1.	date	Date	Not Null
2.	Event_description	Varchar2	Not Null
3.	developer	Number	Not Null
4.	designer	Number	Not Null
5.	tester	Number	Not Null

Table 3.4: Suplogin

Description: Table that stores the username and password of supervisor for login.

S. No.	Field name	Data Types	Constraint/Description
1.	supplier	Varchar2	Not Null
2.	password	Varchar2	Not Null

3.8 HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirements

Processor	:	Pentium (or above)
RAM	:	2 GB
Hard Disk	:	40 GB

Software Requirements

Operating System	:	Windows XP or above
Java Runtime Environment	:	JDK/JRE 1.8 or above
Database Server	:	MYSQL
Web Server	:	Apache Tomcat
Browser	:	Java Script Browser

CODING AND IMPLEMENTATION

4.1 TECHNOLOGY OVERVIEW

The most common types of programs written in the Java programming language are applets and applications. If you've accessed the Web, you're probably already familiar with applets. An applet is a program that adheres to certain conventions that allow it to run within a Java-enabled browser.

4.1.1 Java

Java is a programming language and computing platform first released by Sun Microsystems in 1995. There are lots of applications and websites that will not work unless one has installed java and more are created every day. Java is fast, secure and reliable. From laptops to data centers, game consoles to scientific supercomputers, cell phones to the Internet, Java is everywhere!

Java is an object-oriented programming language in the tradition of C and C++. But it is different from other languages in several aspects. One of the main reasons Java is so popular is its platform independence, which means that java programs can be run on different computers. The Java language itself is very simple. However, Java comes with a library of classes that provide commonly used utility functions that most java programs can't do without. This class library, called the java API, is as much a part of Java as the language itself.

However, the Java programming language is not just for writing cute, entertaining applets for the Web. The general-purpose, high-level Java programming language is also a powerful software platform. Using the generous API, you can write many types of programs.

An application is a standalone program that runs directly on the Java platform. A special kind of application known as a server serves and supports clients on a network. Examples of servers are Web servers, proxy servers, mail servers, and print servers. Another specialized program is a servlet. A servlet can almost be thought of as an applet that runs on the server side. Java Servlets are a popular choice for building interactive web applications, replacing the use of CGI scripts. Servlets are similar to applets in that they are runtime extensions of applications. Instead of working in browsers, though, servlets run within Java Web servers, configuring or tailoring the server.

Java is a powerful platform that includes a complete set of APIs for distributed applications, allows programs to run anywhere on the network, runs on top of existing platforms ("WORA") and reduces desktop administration costs. These features make java ideal for programming in the networked, heterogeneous world, realizing web/E-Commerce applications, integrating several technologies and programming handheld devices (PDAs, phones etc). Because of its rich set of API's, similar to Macintosh and Windows, and its platform independence, Java can also be thought of as a platform in itself. Java also has standard libraries for doing mathematics.[4]

4.1.2 About J2EE

The multi-tier architecture such as COBRA has got its own advantages in terms of scalability, performance and reliability. In a multi-tier architecture, a client does not interact directly with the server. Instead, it first contacts another layer called Middleware. The middleware instantiates the server applications and messages the server object. It returns results to the clients. The presence of a middleware layer allows programmers to concentrate on business logic of application. The middleware handles low-lever services, such as thread handling, security, and transactions management.

Sun Microsystems introduced the J2EE application server and the enterprise Java Bean (EJB) specifications as a venture into the multi-tier component architecture. J2EE functions as a middle tier server in three tier architectures. It provides certain specifications that can be used to implement enterprise solutions for certain all types of business requirements. J2EE also offers cost effective solution for business solution.

J2EE is used for developing, deploying and executing applications in a distributed environment. The J2EE applications server acts as a platform for implementing various server side technologies Servlets, Java Server Pages (JSP) and Enterprise Java Bean (EJB). J2EE allows you to focus on your business logic program. The business logic is coded in java program, which are reusable component that can be accessed client program EJB runs on J2EE server. In J2EE security is handled almost entirely by platform and its admin. The developer does not have to worry about writing the security logic.

J2EE Architecture

The J2EE SDK architecture consists of the following components:

- The J2EE server
- The EJB Container
- The Web Container

The J2EE server provides the EJB and web containers. The J2EE server enforces authenticating users. The either service provided by the J2EE server are listed here below.

- It allows client to interact with Enterprise Bean.
- It enables a web browser to access servlets and JSP files
- It provides naming and directory services to enable users and various services to locate and search for services and components.

The EJB container manages the execution of Enterprise Bean for J2EE server. EJB is a specification for making server side component that enable and simplifies the task of creating distributed objects. EJB component provide services such as transaction and security management and can be customized during deployment.

The web container manages the executing of JSP and servlets for J2EE applications web components and their container run on the J2EE server. Servlets of the java program that can be deployed on a java enable web server to enhances and extend the functionality of the web server for example you can write a servlets to add a manager service to a website.

Servlet can also be used to add dynamic content to web pages. Java Server Page (JSP) adds server side programming functionality to java. JSP consists of regular Html tags representing the static content and code enclosed within special tags representing the dynamic content. After compilation, a JSP generates a servlets and therefore incorporates all the servlets functionalities.

J2EE Application

J2EE applications access data from a variety of source and cater to a variety of client. To manage these applications the business function conducted in the middle tier. The J2EE platform acts as a middle tier and provides the necessary environment needed by the application. The J2EE platform provides" write once, run anywhere", portability and scalability for multi-tier application. It also minimizes complexity for building multi-tier application. To create a J2EE application we need to create following three components:

- J2EE application client
- Enterprise Bean
- Web component

Each of these components is packaged into a file with a specified file format. A J2EE application client is a Java application that run in a environment that enable it to access to the J2EE services. A J2EE application client is packaged into a .jar (Java archive) file. An Enterprise Bean consists of three files: the EJB class, Home and Remote Interfaces. The Enterprise Beans are bundled into an EJB.jar file. The .jar, .war and EJB.jar are assembled into a J2EE application, which is an .ear file. The .ear file is then deployed to the J2EE server.

```
┌─────────────────────┐              ┌─────────────────────┐
│   Enterprise Bean   │              │     Component       │
│     (.jar file)     │              │     (.jar file)     │
└─────────────────────┘              └─────────────────────┘
                   ＼                  ／
     Assembled       ＼              ／
                      ＼           ／
                  ┌─────────────────────────┐
                  │ J2EE Application Server  │                Deployed
                  └─────────────────────────┘
                             │
                             ▼
                         ┌────────┐
                         │  J2EE  │
                         │ Server │
                         └────────┘
```
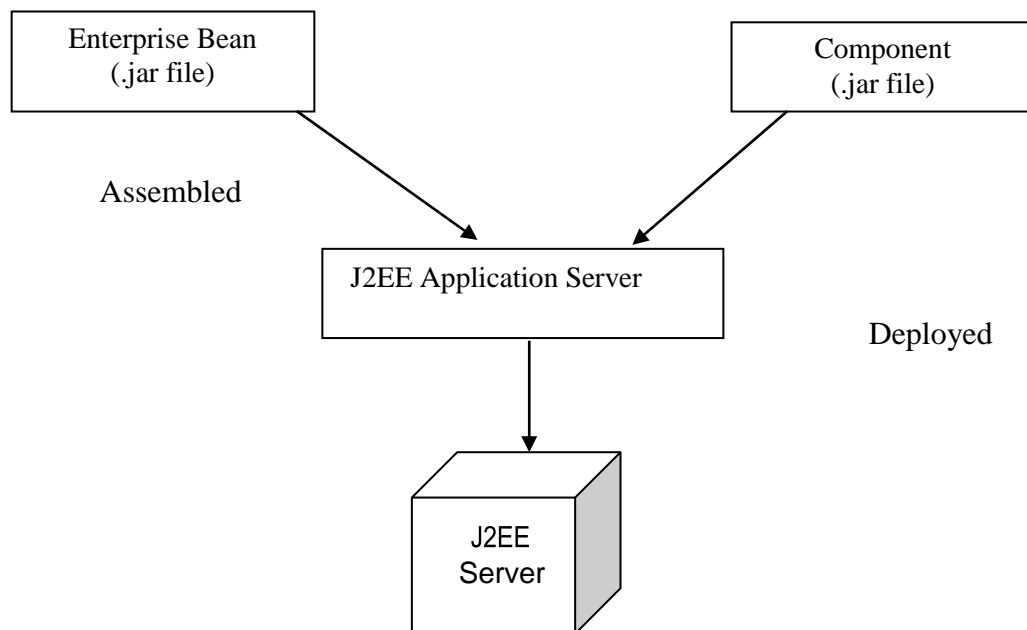
Fig-4.1: Shows Process of creating a J2EE application

J2EE Technologies

The J2EE includes many technologies such as:

- Enterprise Java Beans (EJB)

- Remote Method Invocation (RMI)

- Java Naming and Directory Interface (JNDI)

- Java Database Connectivity (JDBC)

- Java Transaction API (JTA)

- Java Messaging Services (JMS)

- Java Servlet & Java Server Pages (JSP)

- Extensible Markup Language(XML)

The J2EE Security

The architecture of the J2EE is such that it enforces security in the application. In order to access the J2EE services, a user need to prove his/her identity. Such users are called J2EE users and process is called authentication. The J2EE authentication services are different from security of the operating system. The users of the operating system and the users of the J2EE belong to a different realm. A realm is a group of users that have the same authentication policy. The users of J2EE belong to a two different realms that are respectively authentication by certificates and defaults. J2EE certificate to authenticate a web browser client. In most cases, the J2EE services use the default realm to authenticate a user. J2EE users may also belong to a group. A group is a collection of users who have common feature for example, the user belonging to a group may all belonging to a group coding same module. Similarly project managers might belong to a different group.

When client execute J2EE application it request that you enter login id and password. If the combination of both username and password correct the J2EE allow you to access the services.[3]

4.1.5 Oracle 10g

The race for market share in the database industry has increased with the advent of client-server platforms. Oracle 10g is one of the most successful companies that has released a number of development tools including SQL *PLUS, PL/SQL that enables faster and easier application development and its management. Oracle 10g is the robust Database System, it support very large database. Moreover Oracle 10g is widely used as back end for client / server applications. Administrative tools of Oracle 10g help in securing the Data / Information.

Some of the merits of using Oracle 10g (RDBMS) are as under:

- Centralization of database.
- Client Server Technology.
- Security.
- Normalization of Data Base.
- Relationship.

Hence because of these features we are using Oracle 10g as a back-end technology. Whether you are working on LAN projects or Distributed projects, there are two sides of it:-

- Front End

- Back End

Front End remains on client side. Front end is made for end user who uses our application. Basically in front end, our input-output forms reside which takes the input from the client and gives output back to client. Backend remains on server side and has two components viz.

4.2 REQUIREMENT AND ANALYSIS

Requirements analysis consist of following:

- Discuss with the Guide and other team members about the functionality and implementation of project: Employee Management System
- Analyzed detailed descriptions for each functionality.
- Documented all the required analysis and findings.

4.2.1 Functional requirements

Functional Requirements defines a function of a software system and how the system must behave when presented with specific inputs or conditions.

- Login in website.
- Registration of employee
- Various requests and their handling.
- Showing result to users.
- Recovering password.
- Set Schedule.
- Mark Attendance.

4.2.2 Non-functional requirements

Non-Functional, are those requirements that are not directly concerned with the specific functions delivered by the system. They may relate to emergent system properties such as reliability response time and store occupancy. Alternatively, they may define constraints on the system such as the capability of the Input Output devices and the data representations used in the system interfaces. The key, non-functional requirements are:

- Website should have pleasant look and feel.
- Easy way navigation through web pages.
- Attractive user interface.

TESTING

5.1 INTRODUCTION

Testing is vital to the success of any system. Testing is done at different stages within the development phase. System testing makes a logical assumption that all parts of the system is correct and the goals will be achieved successfully.

Inadequate testing or no testing leads to errors that may come up after a long time when correction would be extremely difficult.

Another objective of testing is its utility as a user-oriented vehicle before implementation.

The testing of the system was done on both test and user data. The following tests are performed.

Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. In fact, testing is the one step in the software engineering process that could be viewed as destructive rather than constructive.

A strategy for software testing integrates software test case design methods into a well-planned series of steps that result in the successful construction of software. Testing is the set of activities that can be planned in advance and conducted systematically. The underlying motivation of program testing is to affirm software quality with methods that can economically and effectively apply to both strategic to both large and small-scale systems.

5.2 STRATEGIC APPROACH TO SOFTWARE TESTING

The software engineering process can be viewed as a spiral. Initially system engineering defines the role of software and leads to software requirement analysis where the information domain, functions, behaviour, performance, constraints and validation criteria for software are established. Moving inward along the spiral, we come to design and finally to coding. To develop computer software we spiral in along streamlines that decrease the level of abstraction on each turn.

A strategy for software testing may also be viewed in the context of the spiral. Unit testing begins at the vertex of the spiral and concentrates on each unit of the software as implemented in source code. Testing progress by moving outward along the spiral to

integration testing, where the focus is on the design and the construction of the software architecture. Talking another turn on outward on the spiral we encounter validation testing where requirements established as part of software requirements analysis are validated against the software that has been constructed. Finally we arrive at system testing, where the software and other system elements are tested as a whole.[2]
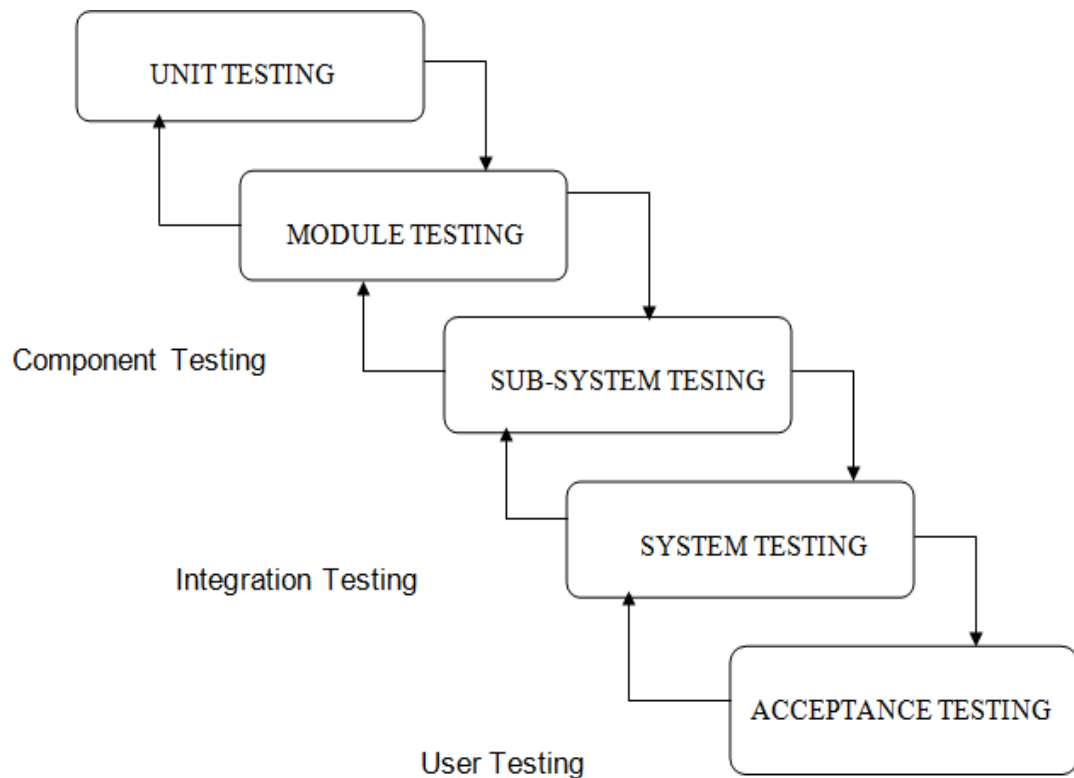
Fig. 5.1: Diagram Types of Testing

5.3 UNIT TESTING

In unit testing the focuses is on the verification of the smallest unit of the project that is a module or a function. In unit testing we work according to white box testing that is providing the input set and checking the output is in accordance with the expected output or not.

1. White Box Testing

This type of testing ensures that

- All the independent modules and function should be executed at least once in the testing phase.
- All the inputs of must include the boundary values & middle values.
- All the logical decisions must be have output as true or false.

To follow the concept of white box testing we have tested each of the above mentioned forms. Tests are done to ensure correct flow of data in the system. All conditions of the system are exercised to check their accuracy

2. Conditional Testing

In Conditional testing, each condition is tested to both true and false aspects. And all the resulting paths of true and false output are tested. So that each path that may be generate on particular condition is traced to uncover any possible errors in the system.

All the conditions like selecting the option button for pdf and word document, not selecting the option button, providing the file path, not providing the file path etc.

3. Data Flow Testing

Data Flow Testing selects the path of the program according to the location of definition and use of variables. This kind of testing is used only when some local variable were declared and their scope in the program is to be tested. The *definition-use chain* method is used in this type of testing. It is used in situations like selecting the word document option button and then checking file is converted to .pdf format or not.[1]

5.4 TEST INTRODUCTION

The test gives a brief idea of the correct expected output and the incorrect output. This test confirms proper behaviour of the GUI during system initialization and startup. The unit under test should initialize without any errors.

Test Cases

The test cases should be written to get the basic idea of the input provided to the system and the expected output to be received from the system.

1. In Employee module

When employee logins

- If click on Update Details

  Expected-New data saved.

- If click on View Schedule

  Expected-navigate to view schedule page.

- If click on View Attendance

  Expected- Navigate to related page.

- If click on View Details

  Expected- Navigate to related page.

- If click on Forgot Password

  Expected- Navigate to related page.

2. <u>In Supervisor Module</u>

- When click on Delete Employee

  Expected- Navigate to empdelete1 & empdelete2 to delete employee.

- When clicks on Set Schedule.

  Expected- Navigate to empeventdesc1 to finally setting the schedule.

- When clicks on Mark Attendance.

  Expected- Navigate to the related pages.

- When clicks on View Details.

  Expected- Navigate to details, detailsprocess, detailsprocess2 to view details.

SYSTEM SECURITY

6.1 INTRODUCTION

Security means different things to different people depending upon their perspective. In the context of our product it means only valid users can login into the system and each user can only access the functionality authorized to the user. To prevent unauthenticated access, form based authentication implemented through Front Controller is to be used. To prevent unauthorized access, different roles are to be created by the administrator and access to features is to be controlled through these roles.

The protection of computer based resources that include hardware, software, data, procedures and people against unauthorized use is known as System Security.

System Security can be divided into four related issues:

- Security
- Integrity
- Privacy
- Confidentiality

Certain concepts recur throughout different fields of security:

- **Assurance** - assurance is the level of guarantee that a security system will behave as expected

- **Countermeasure** - a countermeasure is a way to stop a threat from triggering a risk event

- **Defense in depth** - never rely on one single security measure alone.

- **Risk** - a risk is a possible event which could cause a loss

- **Threat** - a threat is a method of triggering a risk event that is dangerous

- **Vulnerability** - a weakness in a target that can potentially be exploited by a security threat

- **Exploit** - a vulnerability that has been triggered by a threat - a risk of 1.0 (100%).

6.2 DATA SECURITY

Data security is the protection of data from loss, disclosure, modification and destruction. Usually a DBMS includes a password system that controls access to sensitive data. By limiting their access to read-only, write-only, or specified records, or

even fields in records, passwords can prevent certain users from retrieving unauthorized data.

Data security refers to protective digital privacy measures that are applied to prevent unauthorized access to computers, databases and websites. Data security also protects data from corruption. Data security is the main priority for organizations of every size and genre. Data security is also known as information security (IS) or computer security. Integrated Data Loss Prevention extends your existing security with single-click deployment of data loss prevention (DLP) capabilities built into Trend Micro endpoint, email, web, and messaging gateway security. It also includes USB device control for endpoints. And with central management for security and data protection, DLP policies can be enforced across multiple layers of security to prevent data loss via email, USB, and the web. Integrated DLP modules are available for:

- Endpoint Security.
- Mail Server Security.
- Security for Microsoft SharePoint.
- Gateway Messaging Security.
- Control Manager.

Techniques of data security are as follows:

1.Disk Encryption: Disk encryption refers to encryption technology that encrypts data on a hard disk drive. Disk encryption typically takes form in either software (see disk encryption software) or hardware (see disk encryption hardware). Disk encryption is often referred to as on-the-fly encryption (OTFE) or transparent encryption.

2.Hardware-based mechanisms for protecting data: Hardware-based or assisted computer security offers an alternative to software-only computer security. Security tokens such as those using PKCS#11 may be more secure due to the physical access required in order to be compromised. Access is enabled only when the token is connected and correct PIN is entered (see two-factor authentication). Newer technologies in hardware-based security solve this problem offering fool proof security for data.

- A hardware device allows a user to log in, log out and set different privilege levels by doing manual actions.

- The device uses biometric technology to prevent malicious users from logging in, logging out, and changing privilege levels. The current state of a user of the device is read by controllers in peripheral devices such as hard disks.

- Hardware-based access control is more secure than protection provided by the operating systems as operating systems are vulnerable to malicious attacks by viruses and hackers.

- The data on hard disks can be corrupted after a malicious access is obtained. With hardware-based protection, software cannot manipulate the user privilege levels. It is impossible for a hacker or a malicious program to gain access to secure data protected by hardware or performs unauthorized privileged operations. This assumption is broken only if the hardware itself is malicious or contains a backdoor.

- The hardware protects the operating system image and file system privileges from being tampered. Therefore, a completely secure system can be created using a combination of hardware-based security and secure system administration policies.

3.Backups: Backups are used to ensure data which is lost can be recovered and nowadays it's very important to keep a backup of any data.

4.Data Masking: Data Masking of structured data is the process of obscuring (masking) specific data within a database table or cell to ensure that data security is maintained and sensitive information is not exposed to unauthorized personnel. This may include masking the data from users (for example so banking customer representatives can only see the last 4 digits of a customer's national identity number), developers (who need real production data to test new software releases but should not be able to see sensitive financial data etc.

5.Data Erasure: Data erasure is a method of software-based overwriting that completely destroys all electronic data residing on a hard drive or other digital media to ensure that no sensitive data is leaked when an asset is retired or reused.
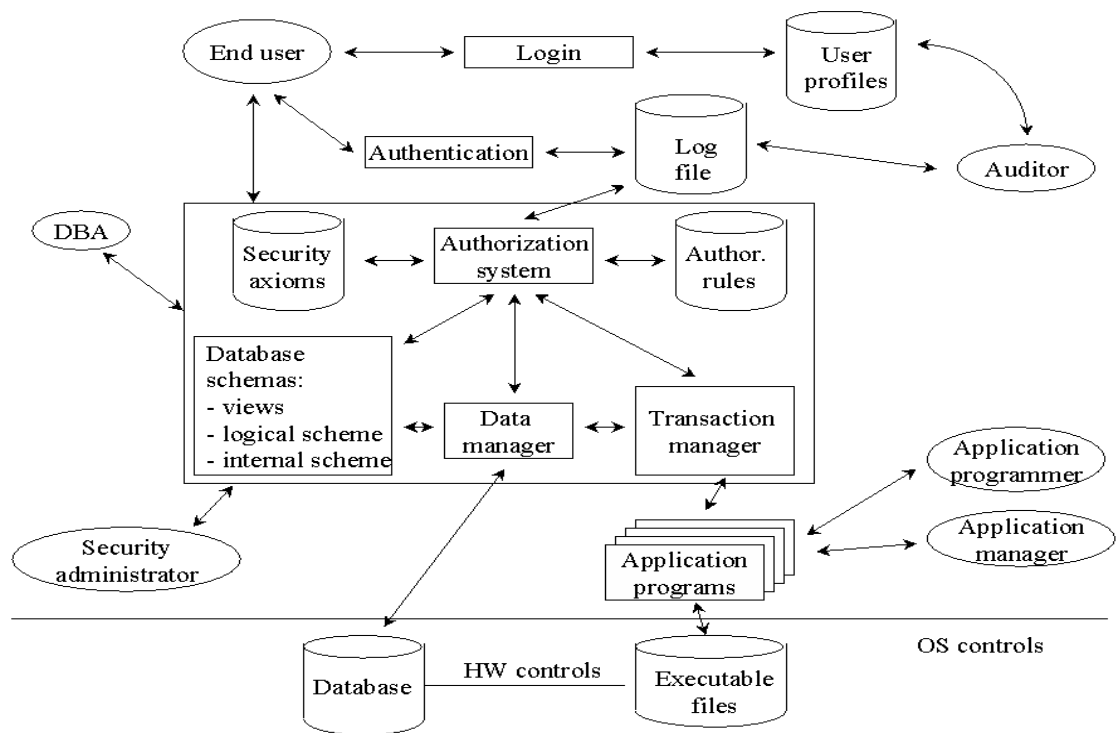
Fig. 6.1: Data Security Diagram

6.3 DATABASE SECURITY

Database security concerns the use of a broad range of information security controls to protect databases (potentially including the data, the database applications or stored functions, the database systems, the database servers and the associated network links) against compromises of their confidentiality, integrity and availability.

- It involves various types or categories of controls, such as technical, procedural/administrative and physical. Database security is a specialist topic within the broader realms of computer security, information security and risk management.

- The existing system has been maintained manually. The system, which has been maintained manually, had been complex and complicated.

- In the existing system file is not transfer encrypt and decrypt.

- The existing system we can use Encrypted format secure text Data will be loss.

- No security Issue.

- In existing system no authentication and authorization.

Security risks to database systems include, for example

- Unauthorized or unintended activity or misuse by authorized database users, database administrators, or network/systems managers, or by unauthorized users or hackers (e.g. inappropriate access to sensitive data, metadata or functions within databases, or inappropriate changes to the database programs, structures or security configurations).

- Malware infections causing incidents such as unauthorized access, leakage or disclosure of personal or proprietary data, deletion of or damage to the data or programs, interruption or denial of authorized access to the database, attacks on other systems and the unanticipated failure of database services.

- Overloads, performance constraints and capacity issues resulting in the inability of authorized users to use databases as intended.

- Physical damage to database servers caused by computer room fires or floods, overheating, lightning, accidental liquid spills, static discharge, electronic breakdowns/equipment failures and obsolescence.

- Design flaws and programming bugs in databases and the associated programs and systems, creating various security vulnerabilities (e.g. unauthorized privilege escalation), data loss/corruption, performance degradation etc.

- Data corruption and/or loss caused by the entry of invalid data or commands, mistakes in database or system administration processes, sabotage/criminal damage etc.
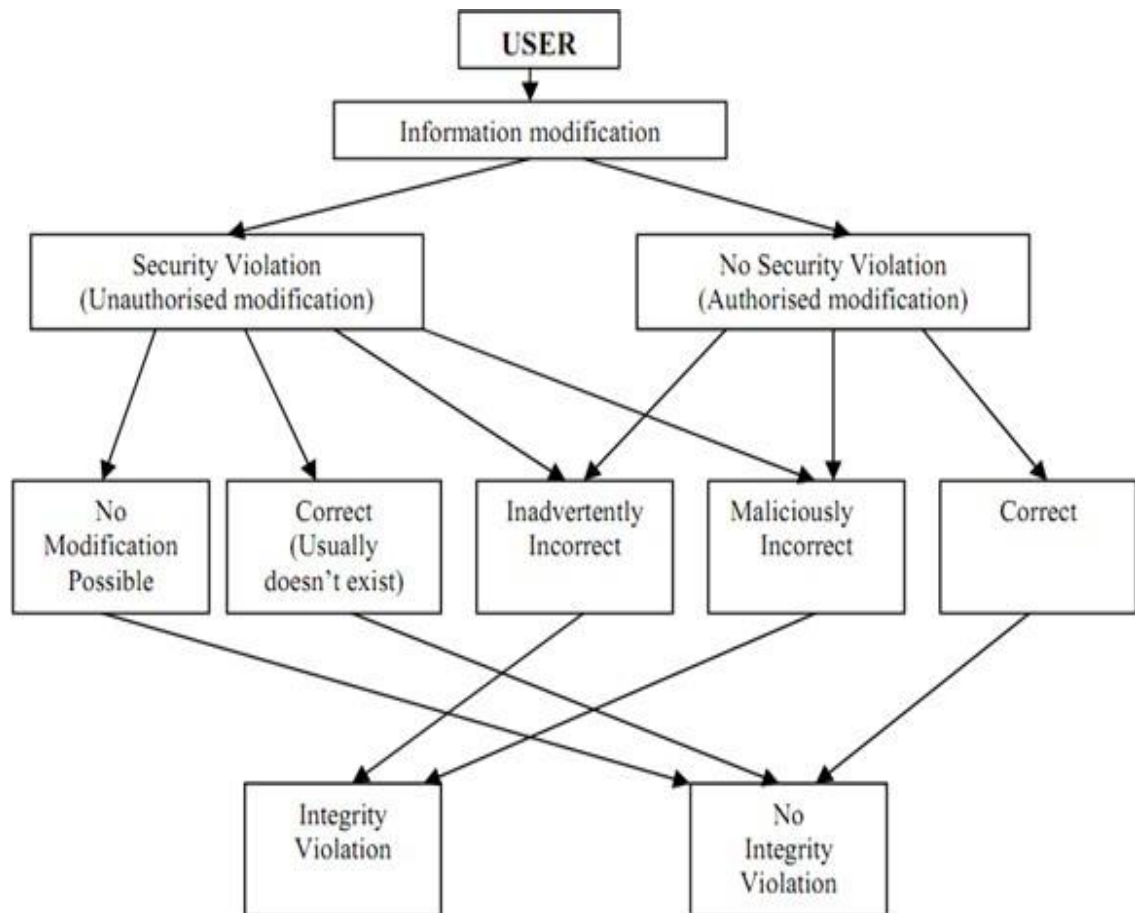
Fig. 6.2: Database Security Diagram

Screen Shots

This chapter displays the results and discusses on the project **"Employee Payroll Management System".**
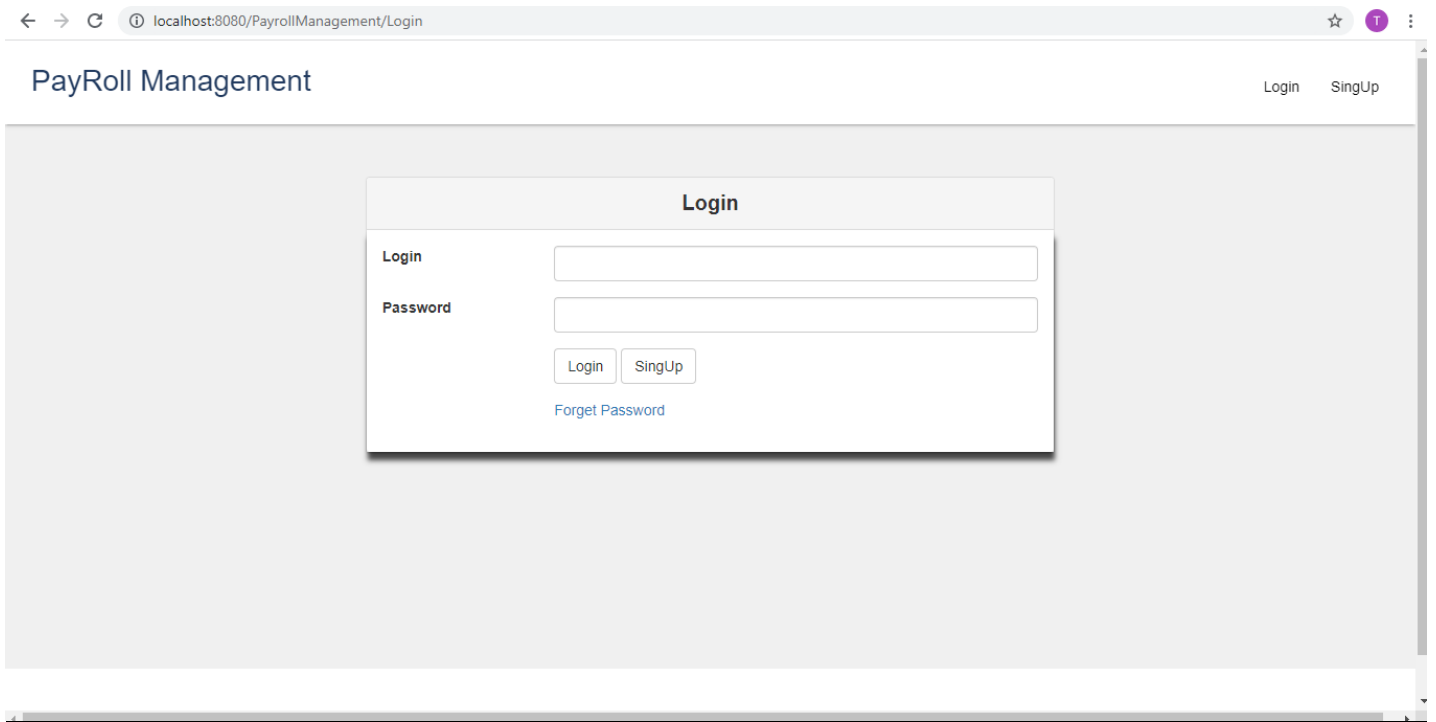
7.1 RESULT & DISCUSSION

The front view of the system consists of different services provided by the system and a login page with which a user is prompted to login in to the system through his user name and password. Upon the student's login, his/her details are updated in to the system. When all data is entered, the student can not only view their details but also view their exams and report details. When the student's session ends, all data is saved. Faculty can also login to the system with their id and password and has access to the system.

1.Welcome Screen:

<u>2. Login:</u>

After admin login:

Add Employee:

Add Salary:

Employee List:

Salary List:

Employee Dashboard:

CONCLUSION

8.1 CONCLUSION

Overall, the system is useful for all the users to maintain information at various levels. It connects supervisor and employee and thus easy to maintain.

Now supervisor can easily set the schedule or any notifications to the respective employees without having a person to send to employees.

This project proved good for us as it provided practical knowledge of not only programming in J2EE and Oracle Server Developer working of web based application, but also about all handling procedure related with Advance and new technology. It also provides knowledge about the latest technology used in developing web enabled application and client server technology that will be great demand in future. This will provide better opportunities and guidance in future in developing projects independently.

8.2 FUTURE SCOPE

The developed system is flexible and changes can be made easily. The system is developed with an insight into the necessary modification that may be required in the future. Hence the system can be maintained successfully without much rework.

Any product despite of its meticulous design and features needs enhancement with time.EMS being no exception needs active enhancement of features and functionality. Following features are proposed to be implemented in future to make the product more feature rich.

Moreover, it is just a beginning; further the website can be enhanced by adding following facilities in the existing software:

- **Email:** In each organization, there is always a need of efficient paperless, secure, and private communication medium that has the retention capabilities. We are proposing to add Intranet messaging facility to fill this requirement as an independent module in future release.
- **Template Based Look & Feel:** Being a product, it is proposed to be implemented at the site of different clients. Each client has different set of preferences for look feel of the application. To minimize the customization

process, look and feel of the application is proposed to be based on templates in future releases.

- **Complete Employee Record Management:** In future we propose to do all record management online or web based automated system which only requires officially needed paper work not more than that.

- **Attendance through email:** A complete email module for sending the attendance to respective employees which is developed separately can be integrated to the system.

REFERENCES

1. **Gray and Reuter**, "*An Introduction to Database Systems*" Fifth Edition, Wesley Publication.

2. **Pressman, Roger S**, "*Software Engineering: A Practitioner's Approach*" 6th Edition, Mass: McGraw-Hill Publication.

3. **George Franciscus and Danilo Gurovich**, *"Struts Recipes",* Manning Publications Company

4. **Emmanuel Bernard, John Griffin**, "*Hibernate Search in Action*", Manning Publications Company.

5. **Simon Singh, "***The Code Book", Wiley Publication*

6. **** www.java.sun.com (Official Java Website).

7. **** www.java.sun.com/developer/onlineTraining/J2EE/Intro2/j2ee.html

8. **** www.java.sun.com/j2se/1.4.2/docs/api/index.html

9. **** www.w3schools.com