LEAVE MANAGEMENT SYSTEM

a project report Submitted by

DHARSHINI M

DHIVIN KUMAR A J

ARUNKUMAR R

for the completion of internship program

under the guidance of

Exson Joseph

Coapps, Tidel Park, "D"North Block, First Floor,

Rajiv Gandhi Salai, Taramani

Chennai - 600113

May 2024

TABLE OF CONTENT

1. INTRODUCTION

The Leave Management System is an Intranet based application that can be accessed throughout the organization or a specified group/Dept. This system can be used to automate the workflow of leave applications and their approvals.

The project has been planned to be having the view of distributed architecture, with centralized storage of the database. The application for the storage of the data has been planned. All the user interfaces have been designed using the Python technologies and using Django framework. The database connectivity is planned using the "SQLite" methodology. The standards of security and data protective mechanism have been given a big choice for proper usage. The application takes care of different modules and their associated reports, which are produced as per the applicable strategies and standards that are put forwarded by the administrative.

2. PURPOSE OF THE PROJECT

This project is aimed at developing a web-based Leave Management Tool, which is of importance to any organization. This is an Intranet based application that can be accessed throughout the organization or a specified group/Dept. This system can be used to automate the workflow of leave applications and their approvals. There are features like cancellation of leave, approval of leave, report generators etc. in this Tool.

3. PROBLEM IN EXISTING SYSTEM

- No use of Web Services and Remoting.
- Risk of mismanagement and of data when the project is under development.

- Less Security.

- No proper coordination between different Applications and Users.

SOLUTION OF THESE PROBLEMS

The development of the new system contains the following activities, which try to automate the entire process keeping in view of the database integration approach.

1. User friendliness is provided in the application with various controls.
2. The system makes the overall project management much easier and flexible.
3. There is no risk of data mismanagement at any level while the project development is under process.
4. It provides high level of security with different level of authentication.

4. SYSTEM ANALYSIS

After analyzing the requirements of the task to be performed, the next step is to analyze the problem and understand its context. The first activity in the phase is studying the existing system and other is to understand the requirements and domain of the new system. Both the activities are equally important, but the first activity serves as a basis of giving the functional specifications and then successful design of the proposed system. Understanding the properties and requirements of a new system is more difficult and requires creative thinking and understanding of existing running system is also difficult, improper understanding of present system can lead diversion from solution.
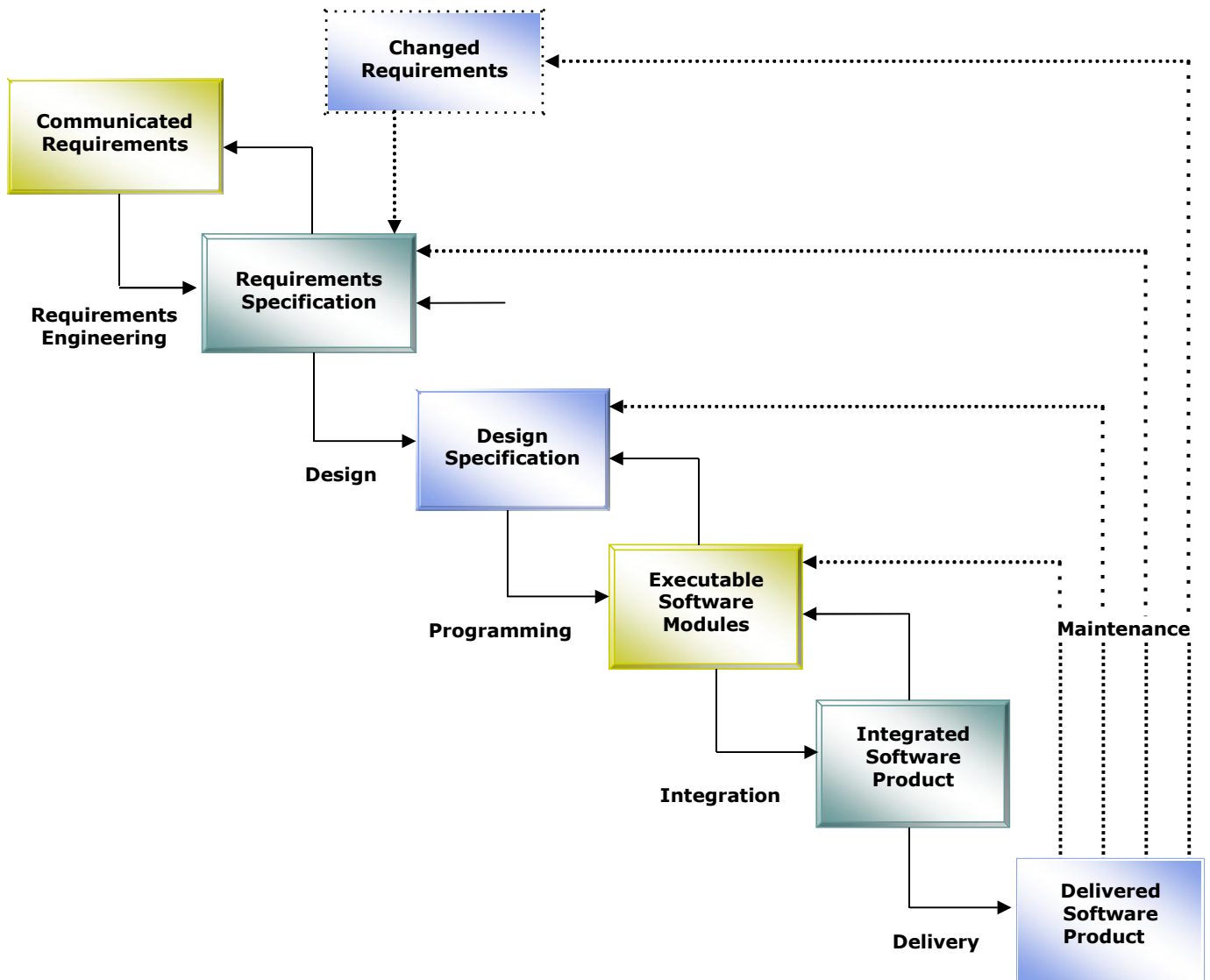
ANALYSIS MODEL

The model that is basically being followed is the WATER FALL MODEL, which states that the phases are organized in a linear order. First of all the feasibility study is done. Once that part is over the requirement analysis and project planning begins. If system exists one and modification and addition of new module is needed, analysis of present system can be used as basic model.

The design starts after the requirement analysis is complete and the coding begins after the design is complete. Once the programming is completed, the testing is done. In this model the sequence of activities performed in a software development project are: -

- Requirement Analysis
- Project Planning
- System design
- Detail design
- Coding
- Unit testing
- System integration & testing

Here the linear ordering of these activities is critical. End of the phase and the output of one phase is the input of other phase. The output of each phase is to be consistent with the overall requirement of the system. Some of the qualities of spiral model are also incorporated like after the people concerned with the project review completion of each of the phase the work done.

WATER FALL MODEL was being chosen because all requirements were known beforehand and the objective of our software development is the computerization/automation of an already existing manual working system.

Changed Requirements

Communicated Requirements

Requirements Engineering

Requirements Specification

Design

Design Specification

Programming

Executable Software Modules

Integration

Integrated Software Product

Maintenance

Delivery

Delivered Software Product

STUDY OF THE SYSTEM

GUI'S

In the flexibility of the uses the interface has been developed a graphics concept in mind, associated through a browses interface. The GUI'S at the top level have been categorized as

1. Administrative user interface

2. The operational or generic user interface

The administrative user interface concentrates on the consistent information that is practically, part of the organizational activities and which needs proper authentication for the data collection. The interfaces help the administrations with all the transactional states like Data insertion, Data deletion and Date updation along with the extensive data search capabilities.

The operational or generic user interface helps the users upon the system in transactions through the existing data and required services. The operational user interface also helps the ordinary users in managing their own information helps the ordinary users in managing their own information in a customized manner as per the assisted flexibilities.

NUMBER OF MODULES

The system after careful analysis has been identified to be presented with the following modules:

The modules involved are:
- Administration
- Staff

ADMIN MODULE DETAILS

- Admin can add a Staff and also update the Staff info.
- Admin can also manage the leave application (approve and not approve).
- Every time when a Staff applies for leave admin will get a notification.
- Admin can change own password after login.
- Admin can also view admin dashboard.

STAFF MODULE

- The Staff can log in with a valid email and password.
- Update his/her profile
- Applied for leave
- View the leave history
- Change his/ her password

PROJECT INSTRUCTIONS:

- Based on the given requirements, conceptualize the Solution Architecture. Choose the domain of your interest otherwise develop the application for ultimatedotnet.com. Depict the various architectural components, show interactions and connectedness and show internal and external elements. Design the web services, web methods and database infrastructure needed both and client and server.
- Provide an environment for upgradation of application for newer versions that are available in the same domain as web service target.

HARDWARE SPEDIFICATIONS

HARDWARE REQUIREMENTS:

This software requires following minimum hardware configuration:

- Processor: Pentium-iv and above
- RAM: 256 MB
- Hard Disk: 4.3GB
- Floppy Disk: 1.44 MB
- CD Drive
- Monitor-15" Color Monitor

SOFTWARE REQUIREMENTS:

- OS: Windows, Linux
- Language: Python Runtime Environment
- Database: dbSQLite3
- Browse: Any of Mozilla, Opera, Chrome etc.
- Webserver: Django
- Software Development Kit: Python
- Scripting Language: JavaScript

PYTHON

What is Python?

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

It is used for:

- ✓ web development (server-side),
- ✓ software development,
- ✓ mathematics,
- ✓ System scripting.

What can Python do?

- ✓ Python can be used on a server to create web applications.
- ✓ Python can be used alongside software to create workflows.
- ✓ Python can connect to database systems. It can also read and modify files.

- ✓ Python can be used to handle big data and perform complex mathematics.
- ✓ Python can be used for rapid prototyping, or for production-ready software development.

DJANGO

Django is a Python framework that makes it easier to create web sites using Python.

Django takes care of the difficult stuff so that you can concentrate on building your web applications.

Django emphasizes reusability of components, also referred to as DRY (Don't Repeat Yourself), and comes with ready-to-use features like login system, database connection and CRUD operations (Create Read Update Delete).

Django is especially helpful for database driven websites.

How does Django Work?

Django follows the MVT design pattern (Model View Template).

Model - The data you want to present, usually data from a database.

View - A request handler that returns the relevant template and content - based on the request from the user.

Template - A text file (like an HTML file) containing the layout of the web page, with logic on how to display the data.

SQLite

SQLite is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine. SQLite is the most used database engine in the world. SQLite is built into all mobile phones and most computers and comes bundled inside countless other applications that people use every day.

The SQLite file format is stable, cross-platform, and backwards compatible and the developers pledge to keep it that way through the year 2050. SQLite database files are commonly used as containers to transfer rich content between systems and as a long-term archival format for data.

There are over 1 trillion (1e12) SQLite databases in active uses, SQLite source code is in the public-domain and is free to everyone to use for any purpose.
Latest Release
Version 3.44.2 (2023-11-24).

5. FEASIBILITY REPORT

Preliminary investigation examine project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- Technical Feasibility
- Operation Feasibility
- Economical Feasibility

Technical Feasibility

The technical issue usually raised during the feasibility stage of the investigation includes the following:

- Does the necessary technology exist to do what is suggested?
- Do the proposed equipment have the technical capacity to hold the data required to use the new system?
- Will the proposed system provide adequate response to inquiries, regardless of the number or location of users?
- Can the system be upgraded if developed?
- Are there technical guarantees of accuracy, reliability, ease of access and data security?

Earlier no system existed to cater to the needs of 'Secure Infrastructure Implementation System'. The current system developed is technically feasible. It is a web based user interface. Thus it provides an easy access to the users.

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. Permission to the users would be granted based on the roles specified. Therefore, it provides the technical guarantee of accuracy, reliability and security. The software and hard requirements for the development of this project are not many and are available as free as open source. The work for the project is done with the current equipment and existing software technology.

OPERATIONAL FEASIBILITY

Proposed projects are beneficial only if they can be turned out into information system. That will meet the organization's operating requirements. Operational feasibility aspects of the project are to be taken as an important part of the project implementation. Some of the important issues raised are to test the operational feasibility of a project includes the following:

- Is there sufficient support for the management from the users?
- Will the system be used and work properly if it is being developed and implemented?
- Will there be any resistance from the user that will undermine the possible application benefits?

This system is targeted to be in accordance with the above-mentioned issues. Beforehand, the management issues and user requirements have been taken into consideration. So there is no question of resistance from the users that can undermine the possible application benefits.

The well-planned design would ensure the optimal utilization of the computer resources and would help in the improvement of performance status.

ECONOMIC FEASIBILITY

A system can be developed technically and that will be used if installed must still be a good investment for the organization.

In the economical feasibility, the development cost in creating the system is evaluated against the ultimate benefit derived from the new systems. Financial benefits must equal or exceed the costs.

The system is economically feasible. It does not require any addition hardware or software. Since the interface for this system is developed using the existing resources and technologies. There is nominal expenditure and economical feasibility for certain.
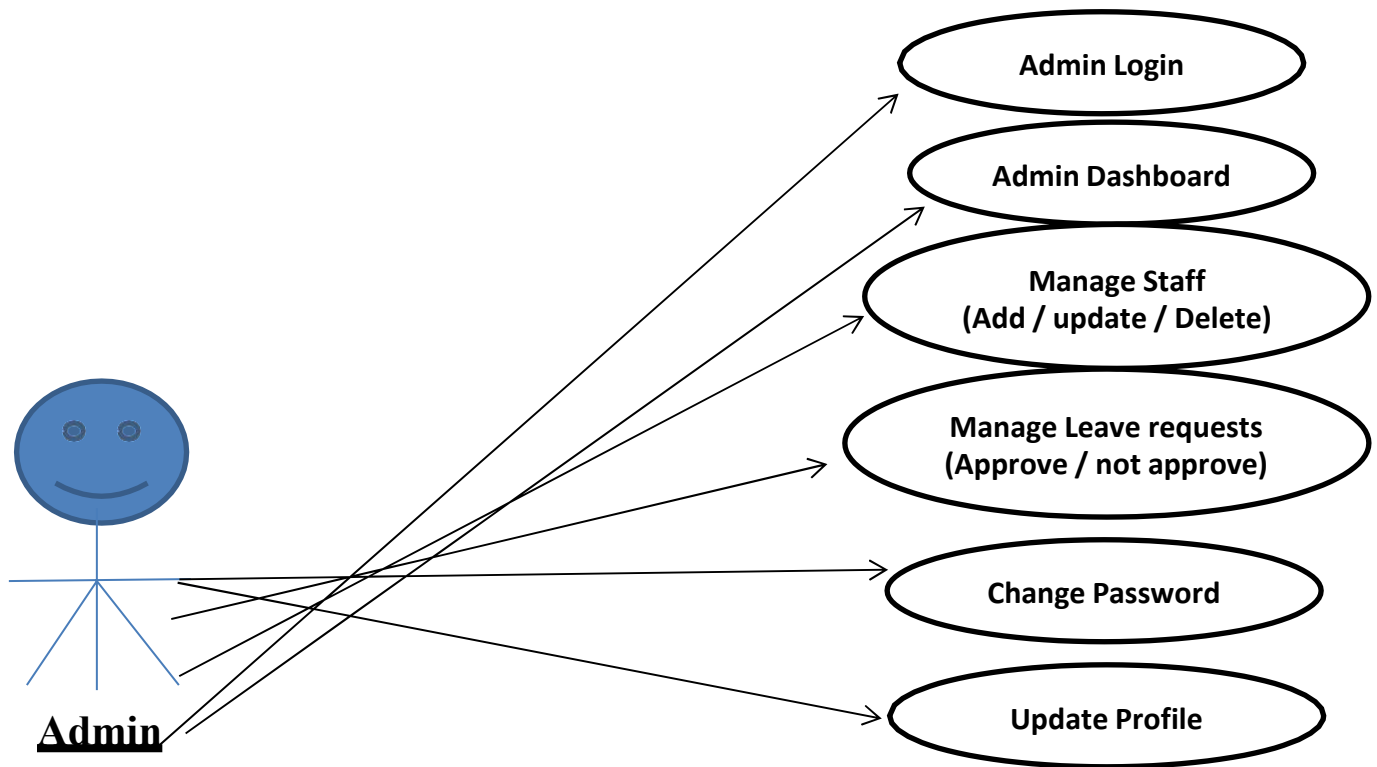
6. SYSTEM DESIGN

INTRODUCTION

Software design sits at the technical kernel of the software engineering process and is applied regardless of the development paradigm and area of application. 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. 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. Software design serves as a foundation for all the software engineering steps that follow. Without a strong design we risk building an unstable system – one that will be difficult to test, one whose quality cannot be assessed until the last stage.

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. From the technical point of view, design is comprised of four activities – architectural design, data structure design, interface design and procedural design.
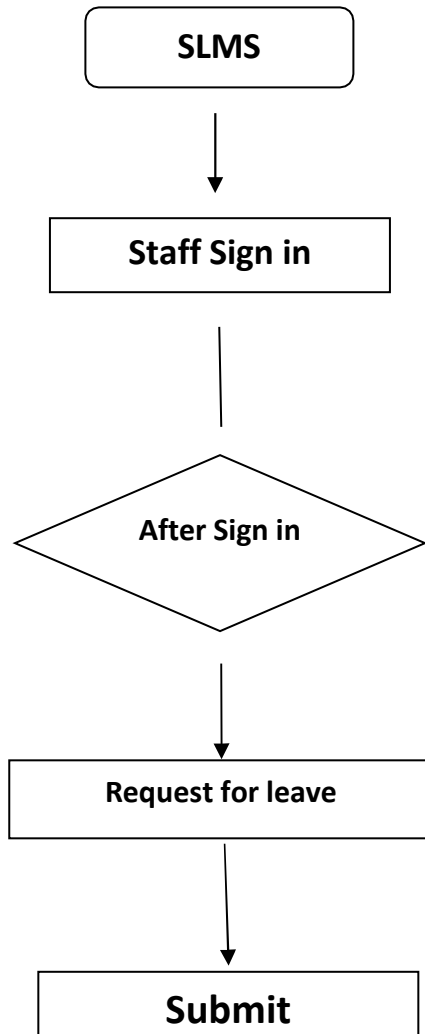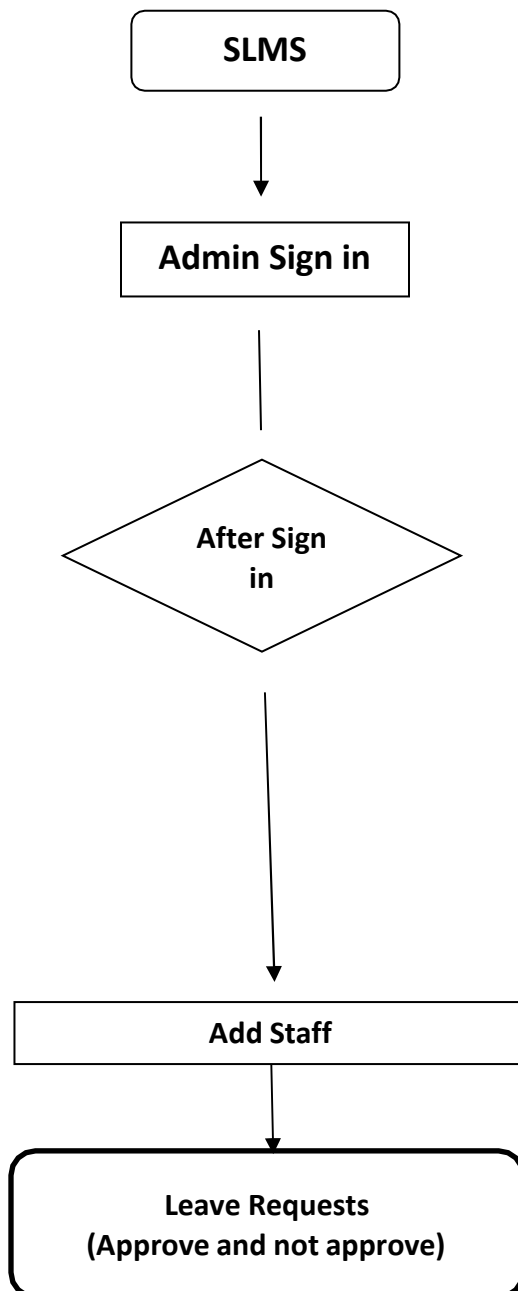
USE CASE DIAGRAM ADMIN

Admin

- Admin Login
- Admin Dashboard
- Manage Staff (Add / update / Delete)
- Manage Leave requests (Approve / not approve)
- Change Password
- Update Profile

USE CASE DIAGRAM STAFF

Staff

- User login
- User Dashboard
- Apply for Leave
- View leave History

15
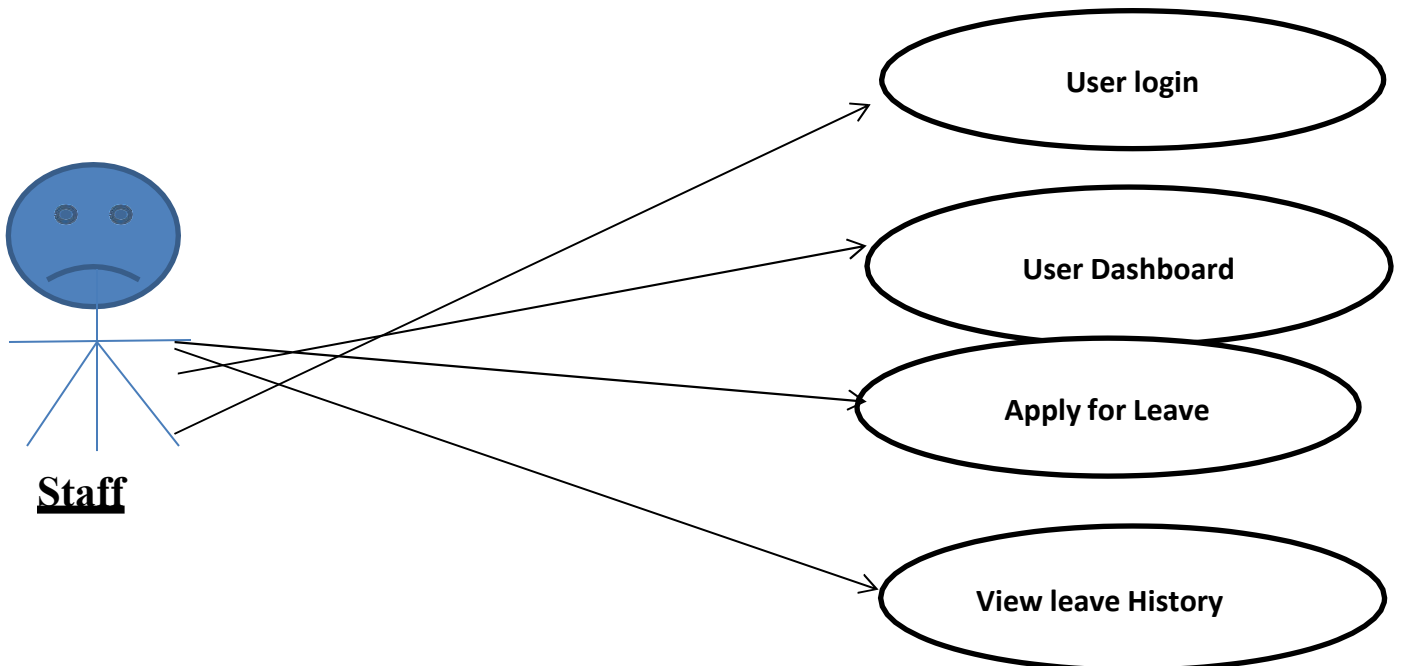
FLOW CHART

USER FLOW CHART

```
        ┌──────────────┐
        │     SLMS     │
        └──────────────┘
                │
                ▼
        ┌──────────────┐
        │ Staff Sign in │
        └──────────────┘
                │
                │
              ╱    ╲
            ╱        ╲
          ╱ After Sign ╲
          ╲    in      ╱
            ╲        ╱
              ╲    ╱
                │
                ▼
        ┌──────────────┐
        │Request for leave│
        └──────────────┘
                │
                ▼
        ┌──────────────┐
        │    Submit    │
        └──────────────┘
```

ADMIN FLOW CHART

```
        ┌─────────────────┐
        │      SLMS       │
        └─────────────────┘
                 │
                 ▼
        ┌─────────────────┐
        │  Admin Sign in  │
        └─────────────────┘
                 │
                 │
              ◇─────◇
            ◇         ◇
          ◇  After Sign  ◇
          ◇     in       ◇
            ◇         ◇
              ◇─────◇
                 │
                 │
                 ▼
   ┌──────────────────────────┐
   │        Add Staff         │
   └──────────────────────────┘
                 │
                 ▼
   ╭──────────────────────────╮
   │      Leave Requests      │
   │  (Approve and not approve)│
   ╰──────────────────────────╯
```

E – R DIAGRAMS

- The relation upon the system is structure through a conceptual ER-Diagram, which not only specifics the existential entities but also the standard relations through which the system exists and the cardinalities that are necessary for the system state to continue.

- The entity Relationship Diagram (ERD) depicts the relationship between the data objects. The ERD is the notation that is used to conduct the date modeling activity the attributes of each data object noted is the ERD can be described resign a data object descriptions.

- The set of primary components that are identified by the ERD are

    ◆ Data object   ◆ Relationships

    ◆ Attributes   ◆ Various types of indicators.

- The primary purpose of the ERD is to represent data objects and their relationships.
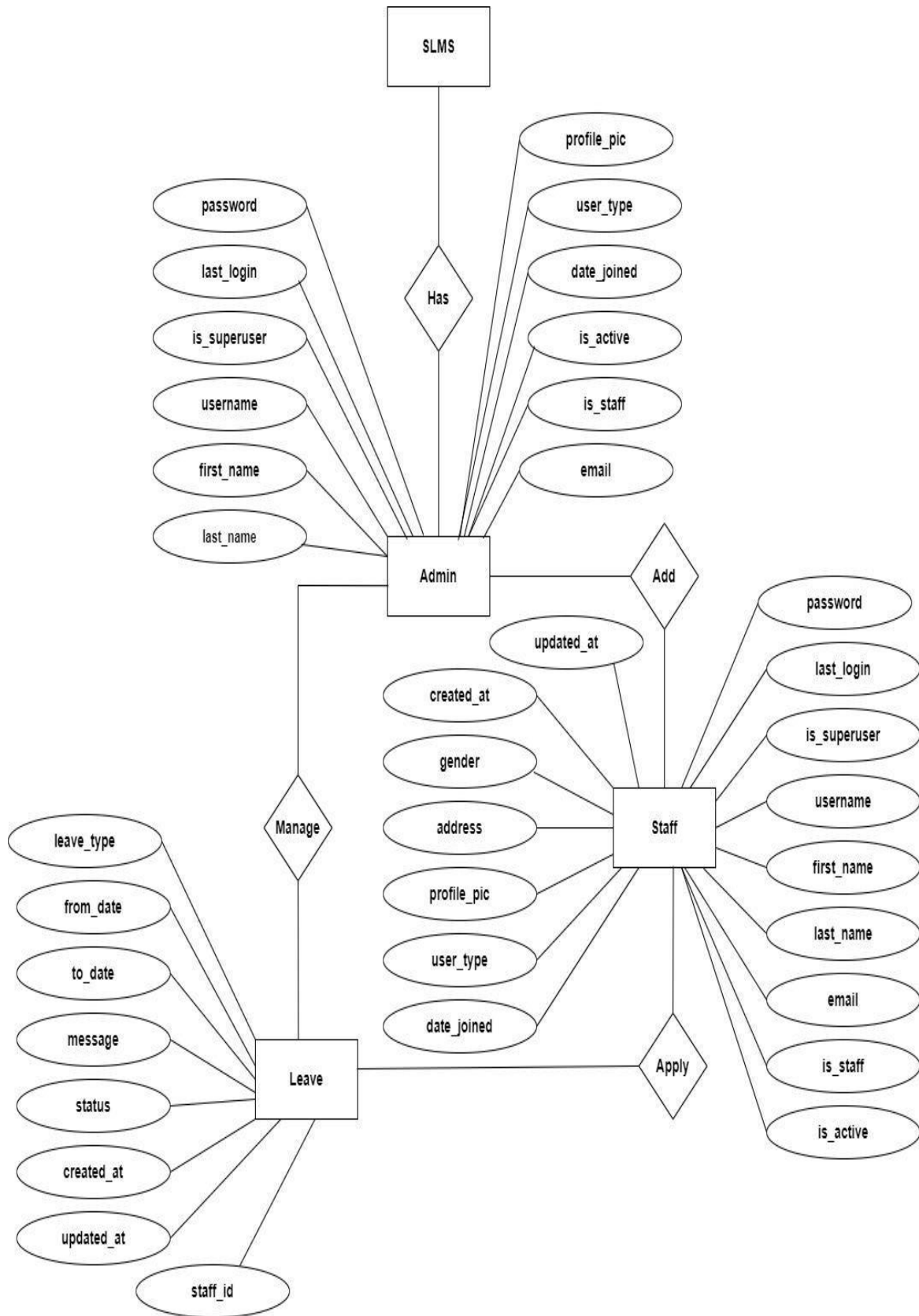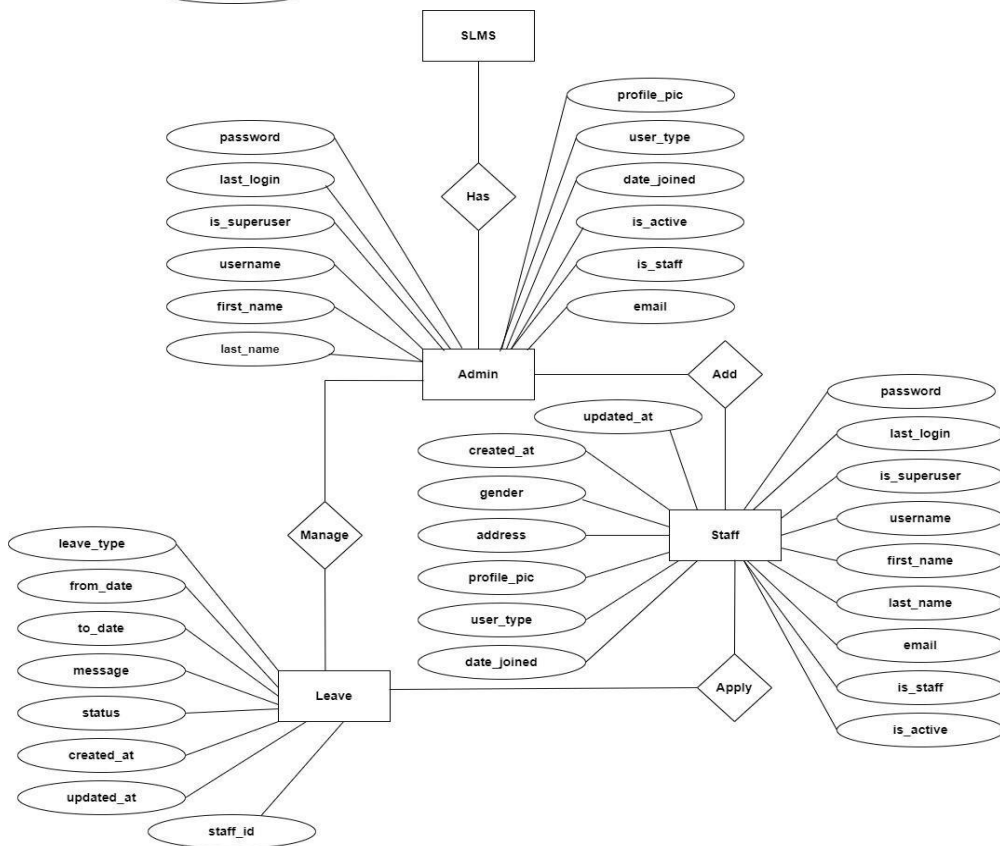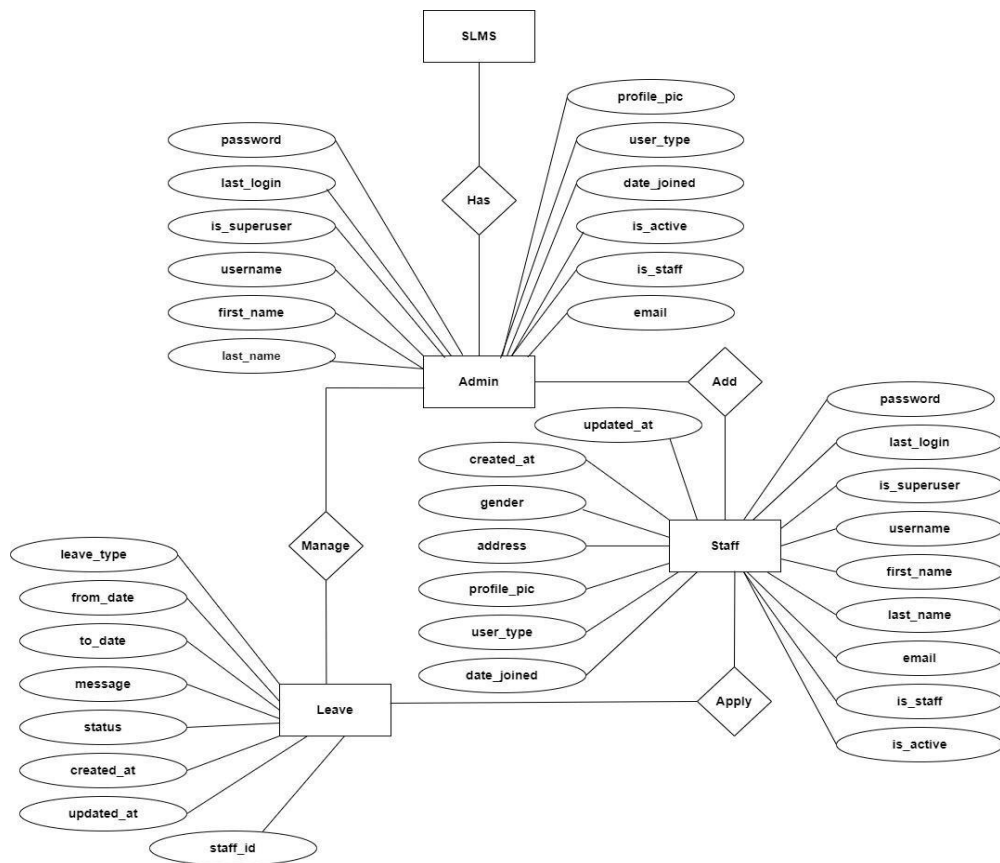
DATA FLOW DIAGRAMS:

The DFD takes an input-process-output view of a system i.e. data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software.

Data objects represented by labeled arrows and transformation are represented by circles also called as bubbles. DFD is presented in a hierarchical fashion i.e., the first data flow model represents the system as a whole. Subsequent DFD refine the context diagram (level 0 DFD), providing increasing details with each subsequent level.

The DFD enables the software engineer to develop models of the information domain & functional domain at the same time. As the DFD is refined into greater levels of details, the analyst performs an implicit functional decomposition of the system. At the same time, the DFD refinement results in a corresponding refinement of the data as it moves through the process that embody the applications.

A context-level DFD for the system the primary external entities produce information for use by the system and consume information generated by the system. The labeled arrow represents data objects or object hierarchy.

Zero Level DFD

First Level DFD

Leave Management

Authorization Management

Login Management

SLMS

Changing Password Management

Admin Management

Staff Management

Second Level DFD

Admin

Login to system

Check Roles of access

Add Staff

Update Staff

Delete Staff

Manage Applied Leave

Update Profile

Change Password

Forgot Password

Check Credential

Manage Modules
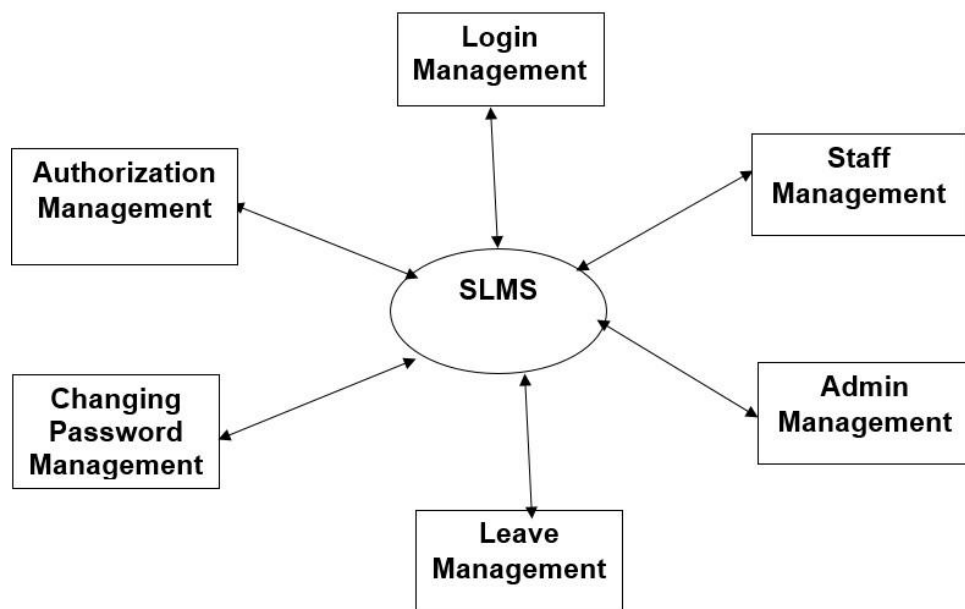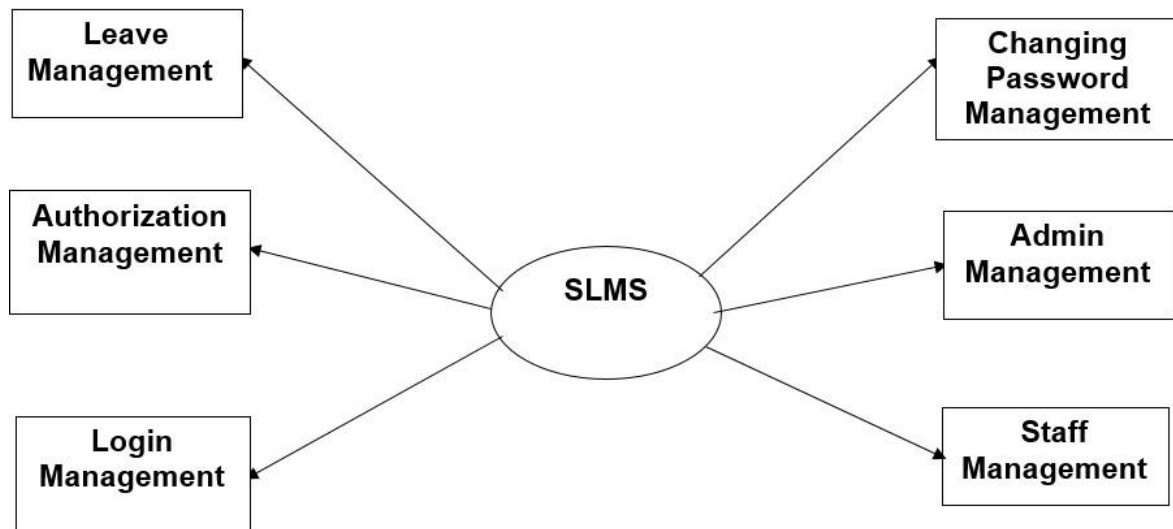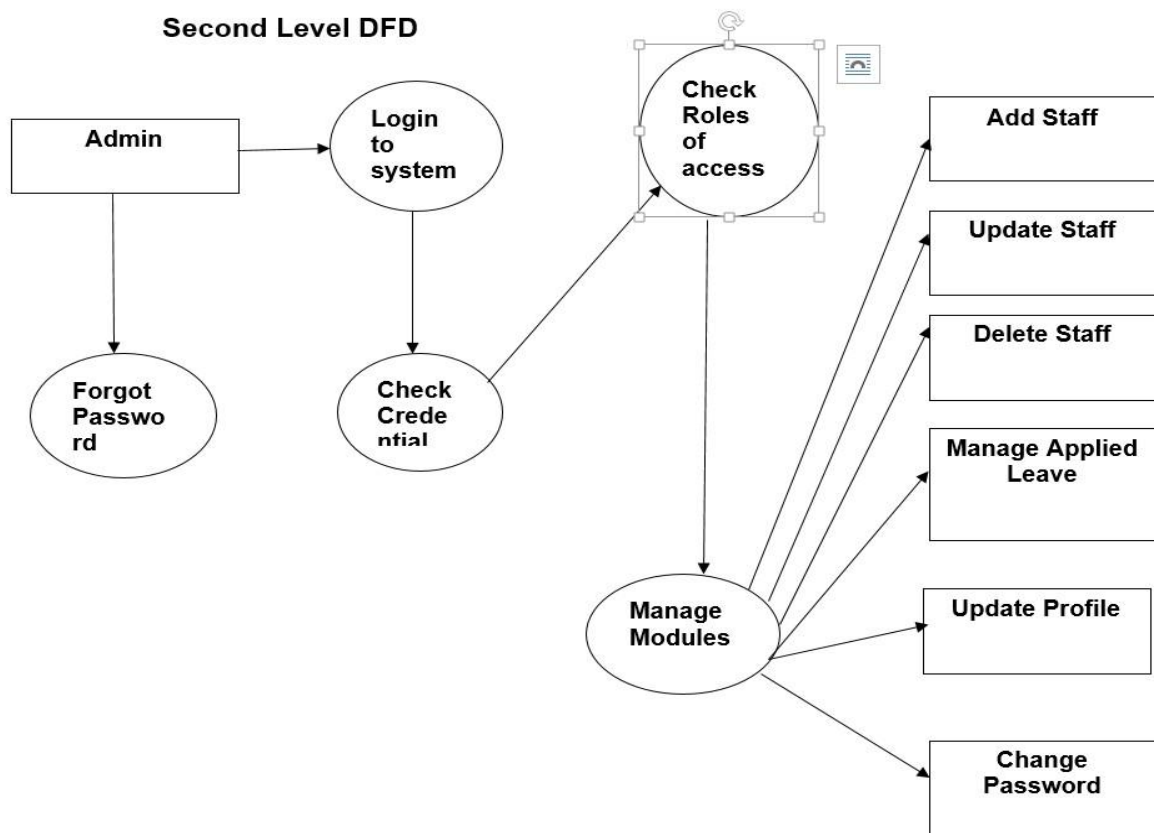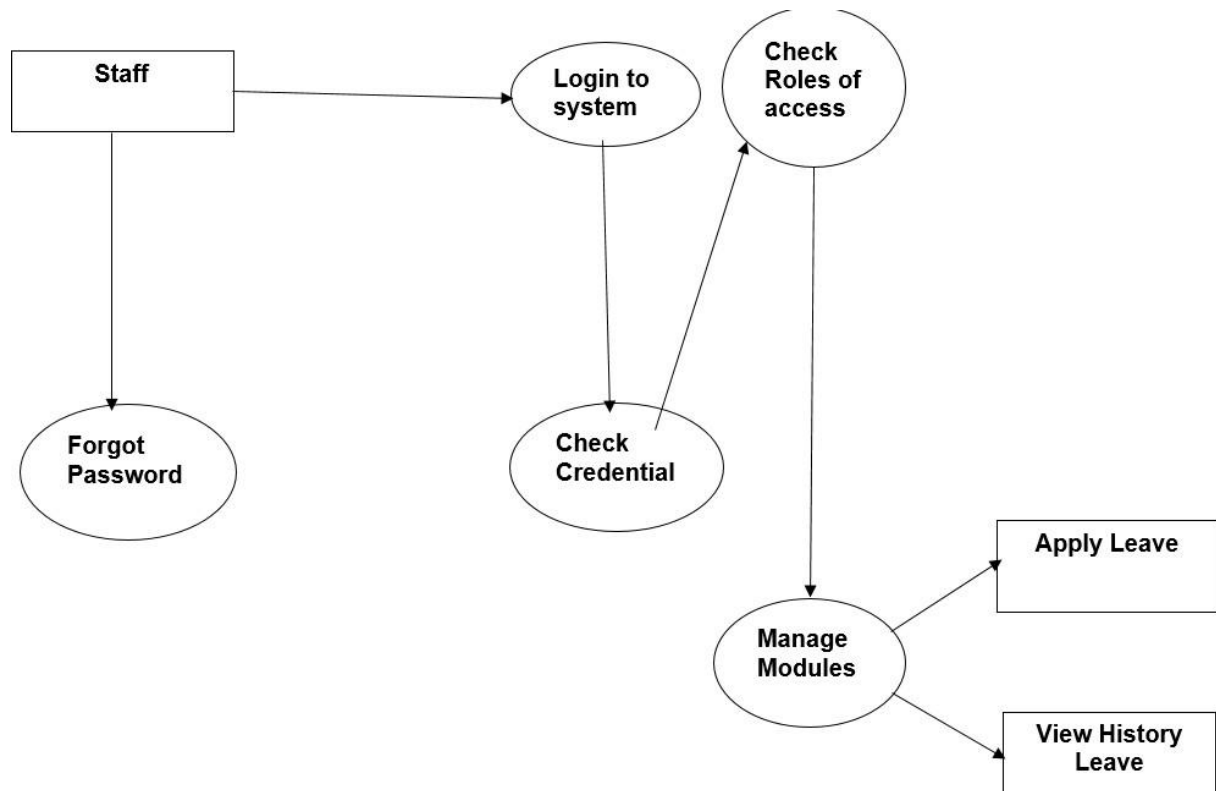
7. DATABASE DESIGN

The data in the system has to be stored and retrieved from database. Designing the database is part of system design. Data elements and data structures to be stored have been identified at analysis stage. They are structured and put together to design the data storage and retrieval system.

A database is a collection of interrelated data stored with minimum redundancy to serve many users quickly and efficiently. The general objective is to make database access easy, quick, inexpensive and flexible for the user. Relationships are established between the data items and unnecessary data items are removed. Normalization is done to get an internal consistency of data and to have minimum redundancy and maximum stability. This ensures minimizing data storage required, minimizing chances of data inconsistencies and optimizing for updates. The SQLite database has been chosen for developing the relevant databases.

Leave Management System contains 3 SQLite tables :
- ➢ admin
- ➢ Staffs
- ➢ Staff leaves

Admin Table : This table stores admin login details

Structure of admin table

Name	Type
password	CharField(max_length=128)
last_login	DateTimeField(blank=True, null=True)
is_superuser	BooleanField()
username	CharField(unique=True, max_length=150)
first_name	CharField(max_length=150)
last_name	CharField(max_length=150)
email	CharField(max_length=150)
is_staff	BooleanField()
is_active	BooleanField()
date_joined	DateTimeField()
user_type	CharField(max_length=50)
profile_pic	CharField(max_length=100)

Staff Table : This table stores staff details.

Structure of Staff table

Name	Type
password	CharField(max_length=128)
last_login	DateTimeField(blank=True, null=True)
is_superuser	BooleanField()
username	CharField(unique=True, max_length=150)
first_name	CharField(max_length=150)
last_name	CharField(max_length=150)
email	CharField(max_length=150)
is_staff	BooleanField()
is_active	BooleanField()
date_joined	DateTimeField()
user_type	CharField(max_length=50)
profile_pic	CharField(max_length=100)
address	TextField()
gender	CharField(max_length=100)
created_at	DateTimeField()
updated_at	DateTimeField()

Staffs Leave Table : This table stores the details of staff leave.
Structure of StaffLeave table

Name	Type
leave_type	CharField(max_length=100)
from_date	CharField(max_length=100)
to_date	CharField(max_length=100)
message	TextField()
status	IntegerField()
created_at	DateTimeField()
updated_at	DateTimeField()
staff_id	ForeignKey(SlmsappStaff, models.DO_NOTHING)

8. OUTPUT SCREENS

HOME PAGE
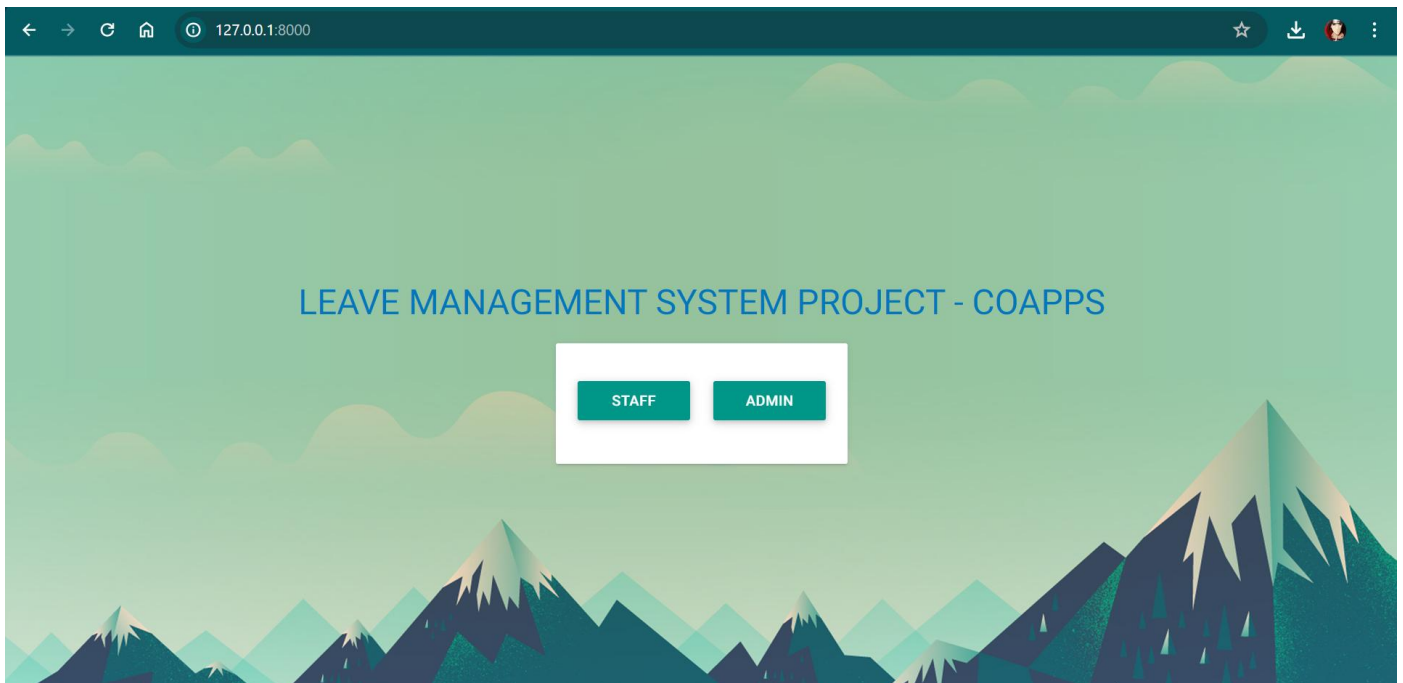
LOGIN PAGE
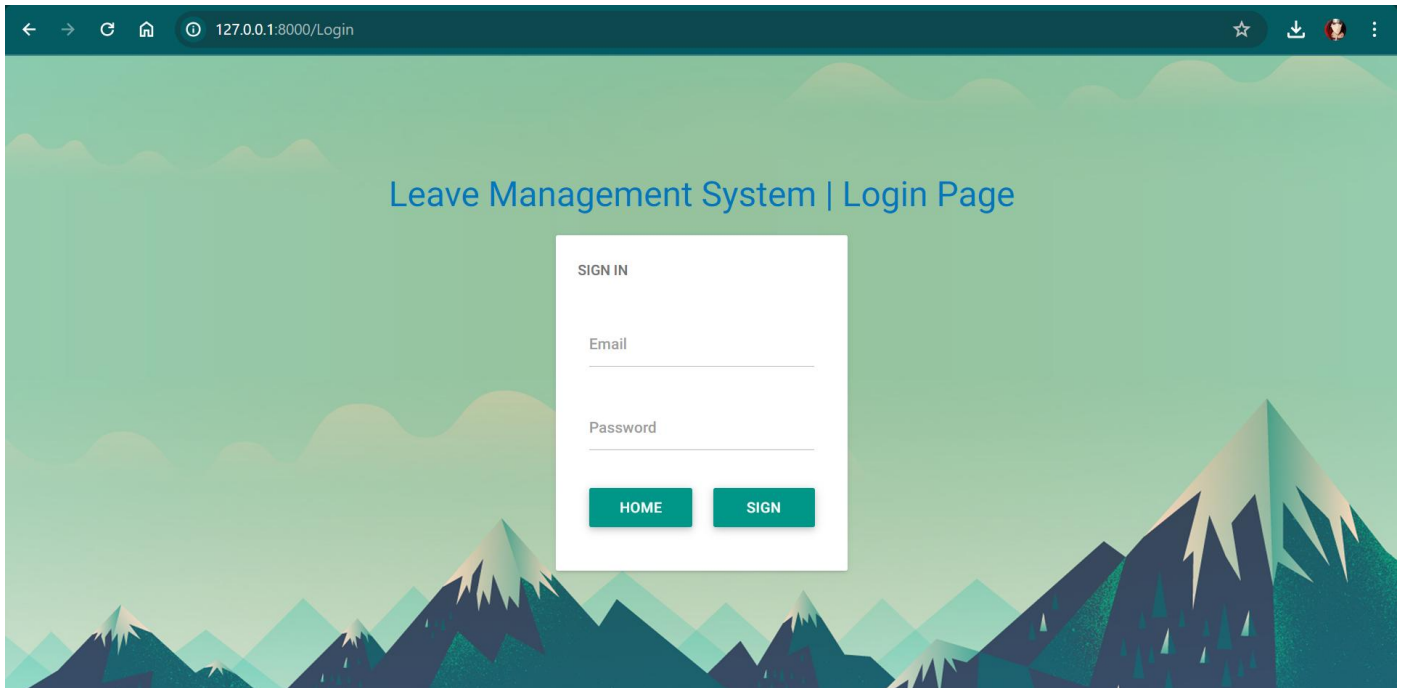
ADMIN PANEL

DASHBOARD

PROFILE

CHANGE PASSWORD

ADD STAFF

MANAGE STAFF

UPDATE STAFF DETAILS

VIEW STAFF LEAVE

STAFF PANEL

LOGIN PAGE
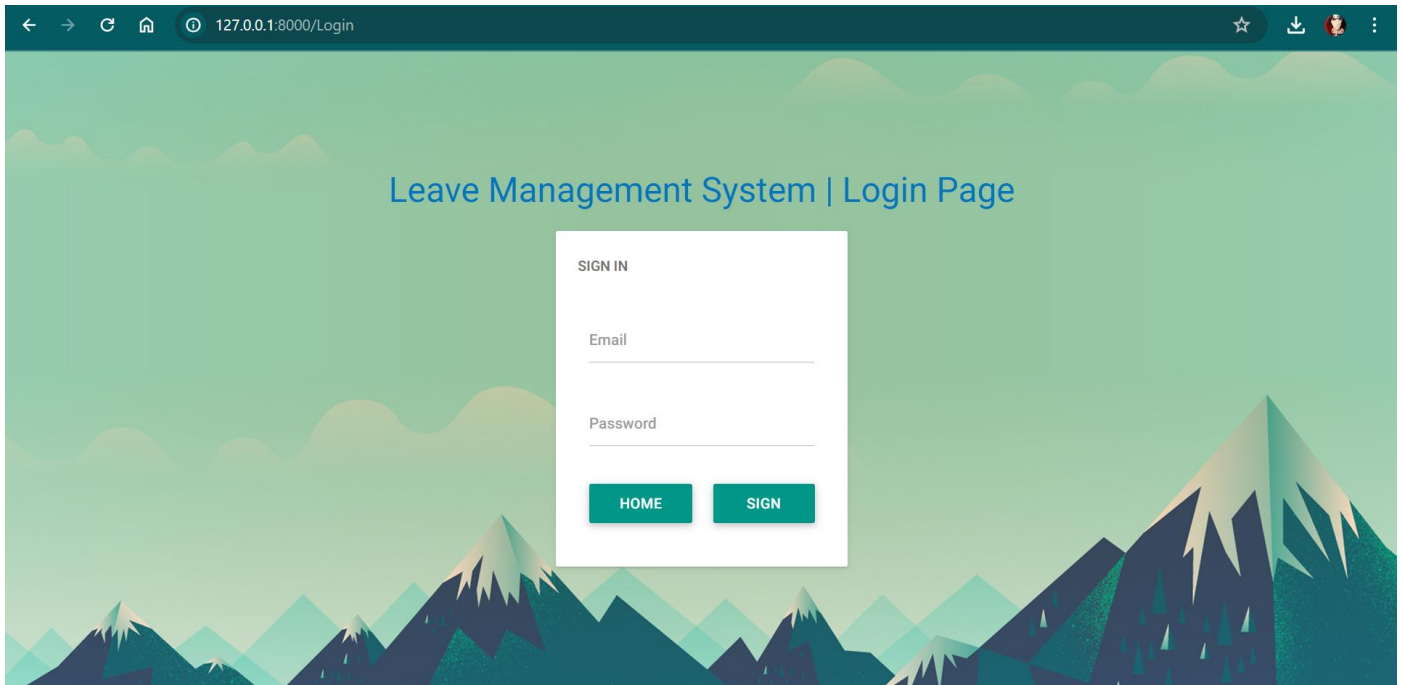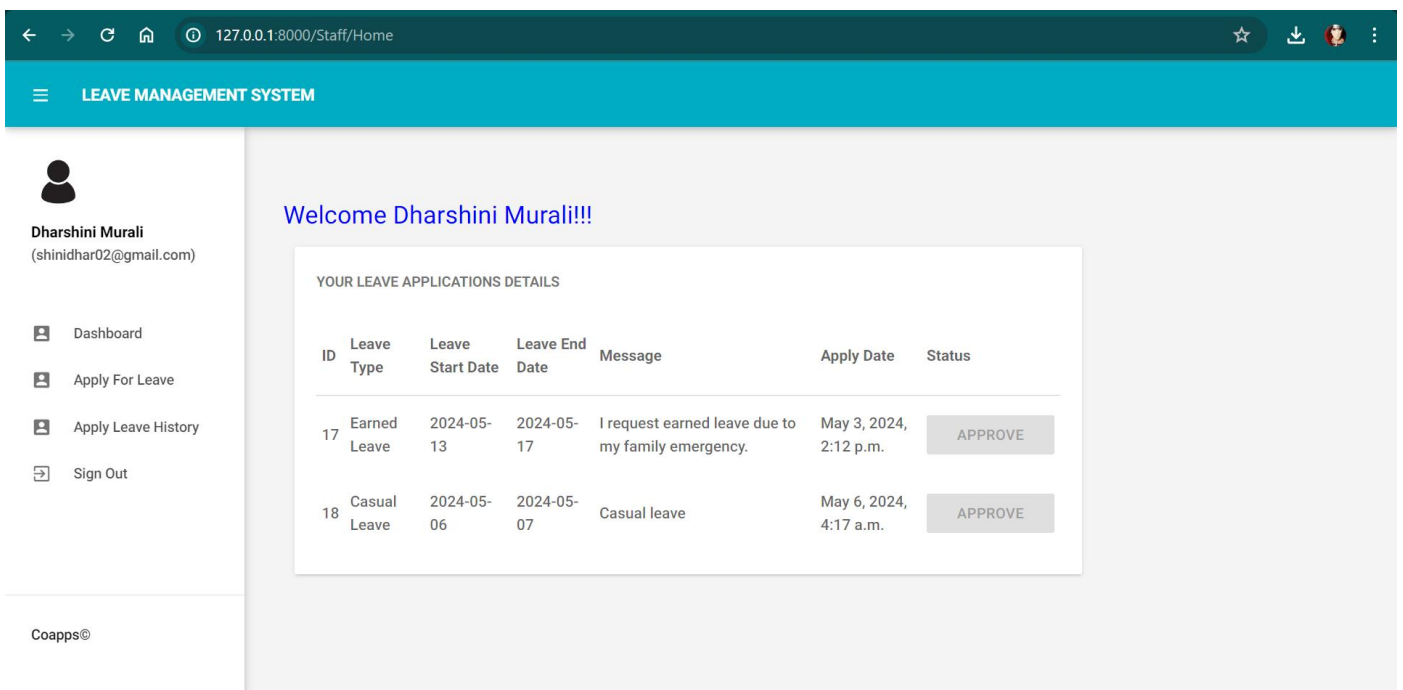
DASHBOARD

APPLY FOR LEAVE

APPLY LEAVE HISTORY

9. SYSTEM TESTING AND IMPLEMENTATION

INTRODUCTION

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.

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, behavior, 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.

UNIT TESTING

MODULE TESTING

Component Testing

SUB-SYSTEM TESING

Integration Testing

SYSTEM TESTING

ACCEPTANCE TESTING

User Testing
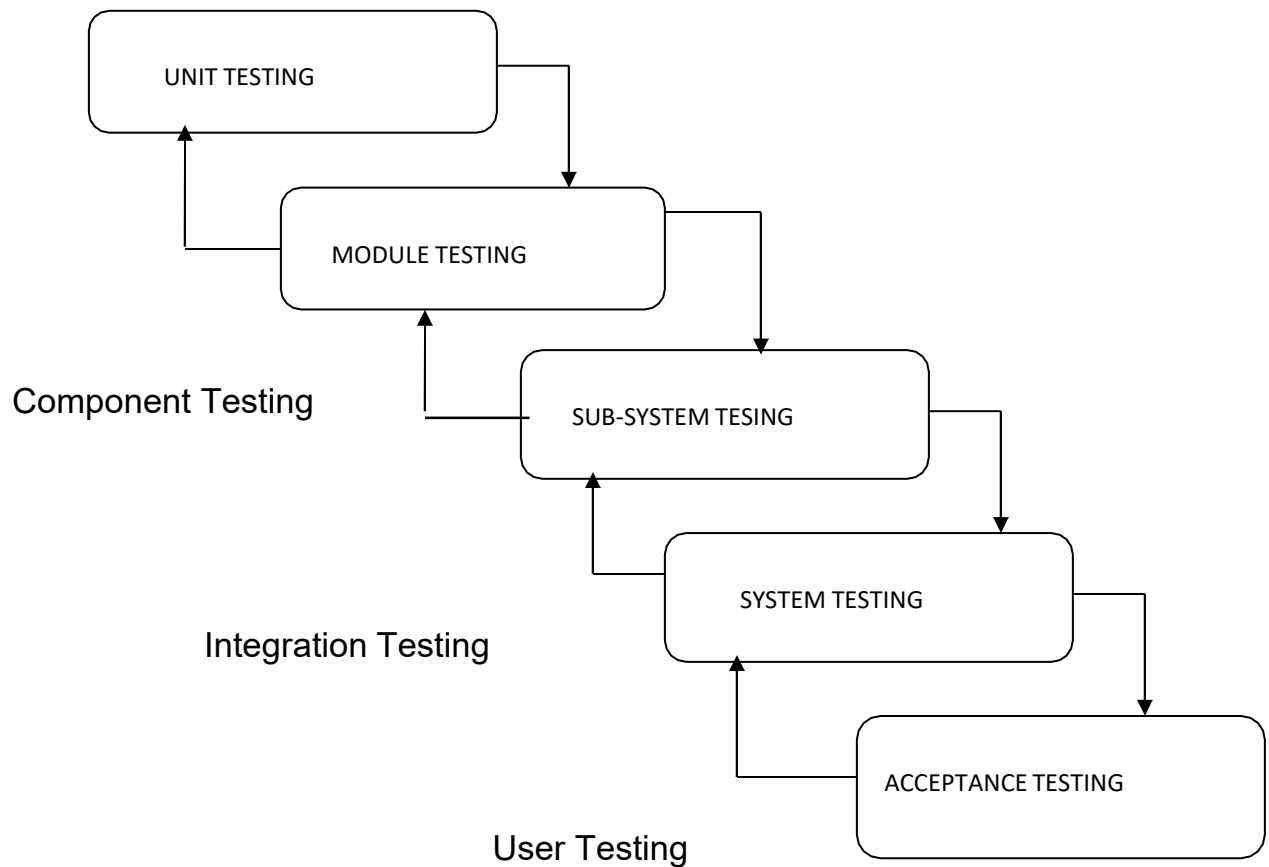
UNIT TESTING

Unit testing focuses verification effort on the smallest unit of software design, the module. The unit testing we have is white box oriented and some modules the steps are conducted in parallel.

1. WHITE BOX TESTING

This type of testing ensures that

- All independent paths have been exercised at least once
- All logical decisions have been exercised on their true and false sides
- All loops are executed at their boundaries and within their operational bounds
- All internal data structures have been exercised to assure their validity.

To follow the concept of white box testing we have tested each form .we have created independently to verify that Data flow is correct, All conditions are exercised to check their validity, All loops are executed on their boundaries.

2. BASIC PATH TESTING

Established technique of flow graph with Cyclomatic complexity was used to derive test cases for all the functions. The main steps in deriving test cases were:

Use the design of the code and draw correspondent flow graph.

Determine the Cyclomatic complexity of resultant flow graph, using formula:

$V(G)=E-N+2$ or

$V(G)=P+1$ or

$V(G)=$ Number Of Regions

Where V(G) is Cyclomatic complexity,

E is the number of edges,

N is the number of flow graph nodes,

P is the number of predicate nodes.
Determine the basis of set of linearly independent paths.

3. CONDITIONAL TESTING

In this part of the testing each of the conditions were tested to both true and false aspects. And all the resulting paths were tested. So that each path that may be generate on particular condition is traced to uncover any possible errors.

4. DATA FLOW TESTING

This type of testing selects the path of the program according to the location of definition and use of variables. This kind of testing was used only when some local variable were declared. The definition-use chain method was used in this type of testing. These were particularly useful in nested statements.

5. LOOP TESTING

In this type of testing all the loops are tested to all the limits possible. The following exercise was adopted for all loops:

- All the loops were tested at their limits, just above them and just below them.
- All the loops were skipped at least once.
- For nested loops test the inner most loop first and then work outwards.

- For concatenated loops the values of dependent loops were set with the help of connected loop.
- Unstructured loops were resolved into nested loops or concatenated loops and tested as above.

Each unit has been separately tested by the development team itself and all the input have been validated.

10. CONCLUSION

It has been a great pleasure for me to work on this exciting and challenging project. This project proved good for me as it provided practical knowledge of not only programming in Python and SQLITE web based application. 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.

BENEFITS:

The project is identified by the merits of the system offered to the user. The merits of this project are as follows: -

- It's a web-enabled project.
- This project offers user to enter the data through simple and interactive forms. This is very helpful for the client to enter the desired information through so much simplicity.
- The user is mainly more concerned about the validity of the data, whatever he is entering. There are checks on every stages of any new creation, data entry or updating so that the user cannot enter the invalid data, which can create problems at later date.
- Sometimes the user finds in the later stages of using project that he needs to update some of the information that he entered earlier. There are options for him by which he can update the records.

- Moreover, there is restriction for his that he cannot change the primary data field. This keeps the validity of the data to longer extent.
- User is provided the option of monitoring the records he entered earlier. He can see the desired records with the variety of options provided by him.
- From every part of the project the user is provided with the links through framing so that he can go from one option of the project to other as per the requirement. This is bound to be simple and very friendly as per the user is concerned. That is, we can say that the project is user friendly which is one of the primary concerns of any good project.
- Data storage and retrieval will become faster and easier to maintain because data is stored in a systematic manner and in a single database.
- Decision making process would be greatly enhanced because of faster processing of information since data collection from information available on computer takes much less time then manual system.
- Allocating of sample results becomes much faster because at a time the user can see the records of last years.
- Easier and faster data transfer through latest technology associated with the computer and communication.
- Through these features it will increase the efficiency, accuracy and transparency,

LIMITATIONS:

- The size of the database increases day-by-day, increasing the load on the database back up and data maintenance activity.
- Training for simple computer operations is necessary for the users working on the system.

REFERENCE

[1] Django homepage. http://www.djangoproject.com/.

[2] Python documentation. http://www.python.org/doc.

[3] Django(web framework). http://en.wikipedia.org/wiki/Django.

[4] Django documentation. http://docs.djangoproject.com.

[5] Python(programming language). http://en.wikipedia.org/wiki/Python.

[6]Books: Web Development with Django by Ben Shaw, Saurabh,

Django 4 By Example by Antonio Mele