MyNoticeBlog_
NoticeBoard Web Application

A Report for the final year project

Submitted by

Priya (17SCSE203078)

in partial fulfilment for the award of the degree

of

MASTERS

IN

COMPUTER APPLICATION

SCHOOL OF COMPUTING SCIENCE AND ENGINEERING

Under the Supervision of

Dr. J.N. Singh,

Professor

APRIL / MAY- 2020

SCHOOL OF COMPUTING AND SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

Certified that this project report "MyNoticeBlog_ NoticeBoard Web Application" is the bonafide work of "Priya (17SCSE203078)" who carried out the project work under my supervision.

SIGNATURE OF HEAD   SIGNATURE OF
                                                                       SUPERVISOR

Dr. MUNISH SHABARWAL   Dr. JN SINGH

PhD (Management), PhD (CS)   Professor

Professor & Dean,   School of Computing Science
School of Computing Science & Engineering   & Engineering

ACKNOWLEDGEMENT

CANDIDATE'S DECLARATION

I hereby certify that the work which is being presented in the MCA Final Report, entitled in Computer Science & Engineering and submitted in the School of Computing Science Engineering of the Galgotias University, Greater Noida is an authentic record of my own work carried out during a period from Jan 2020 to May 2020 under the supervision of, Dr. JN Singh (Professor) School of Computing Science & Engineering, Galgotias University, Greater Noida.

Priya

17SCSE203078

CERTIFICATE

This is to certify that the above statement made by the candidate is correct and to the best of my knowledge.

SUPERVISOR NAME & SIGNATURE

Date: 16-05-2020   **Place: Greater Noida**

Dean-SCSE

ABSTRACT

MyNoticeBlog solves one of the major issues in any institution or enterprise or public like a small or mid size organization. Sending diverse notices day to day is a tedious process. This venture deals with an effective be aware board. It presents a web based notice board incorporating the widely used and reliable web technologies to facilitate the communication of displaying messages on an advanced and smart noticeboard through an online portal. Its operation is based on Python Flask Web Framework [1]. The registered author publishes notices through this portal, viewers can login and view these notices. This project is our experiment on real time information sharing and interactive notice presentation in a blog style to the end user.

TABLE OF CONTENT

INTRODUCTION

- The project 'MyNoticeBlog_' is a web application which provides online notice posting features with facilities like editing and deleting in a blogging style both for registered users and administrators.

- The application allows the users to register and login to access controls for adding the notices with facilities to edit and delete the notices as well

  .

- The notices will be displayed in listed format in the Noticeboard section with a facility to be visible to all the users including notice administrator as well as notice viewers.

- The web application allows a better time management for successful notice posting process.

- It provides features to facilitate easy, error-free processing and retrieval of notice with author name and post date in a very interactive format just like a blog.

PROBLEM STATEMENT

- The problem statement of the project is in consideration of our university and as a student we face problems in getting information and notices from relevant sources.

- There is a lack of a centralized notice management system where students or faculty can get relevant information at one place itself in the form of notices.

- Right now circulation of critical messages and info regarding university topics like examination cancellation or sudden shutdown of university due to unfavourable circumstances are solely dependent on unauthorized sources like students or other entities at unofficial platforms like Whatsapp group or any social group which is not maintained by university or its officials itself.

- The aim of the proposed system is to provide a platform for institutions like universities, colleges or even organizations, to have a centralized space to maintain information in form notices in easy to understand and blog style content which can be accessed by the students or the employees.

EXISTING SYSTEM

Right now if we look at the current domain for Notice blog on the internet. We can find many systems which are not popular and don't have any reorganization among users. Hence this tells us that existing systems are not capable enough to serve the automated action processes. Also these processes are manually which are really lengthy, hectic and exhausting.

The problem of existing system is:

Slow process

Time taking

Difficulty to retrieve information

More human resource involvement

Simultaneous Author and Viewer appearance

More paperwork

Data Redundancy

Data Dependency

PROPOSED SOLUTION

- The aim of the proposed system is to provide a platform for institutions like university, colleges or even organizations, to have a centralized space to maintain information.

- These information will be stored in the form of notices and in an interactive manner.

- Notices will be presented in easy to understand and blog style content which can be accessed by the students or the employees of the organizations like university or school.

PROJECT CATEGORY

The present project is based on the latest, versatile and scalable <u>Web Technologies</u> like Python and its Flask Web Framework for adding business logic and backend processing. HTML, CSS and Bootstrap framework for frontend user interface. These technologies are used as the front end environment for creating interfaces for users to interact with the services of the system.

It also uses **Relational Database Management System (RDBMS)** as its back end environment for the storage and retrieval of data & information.

The present system is a web based online system which relies on **Client-Server Architecture.** Using client-server architecture the system provides an interface to facilitate and connect multiple clients to the server at a same time.

SYSTEM ANALYSIS

6.1 INTRODUCTION

System analysis of the proposed system is conducted for the purpose of studying the system and its parts in order to identify its objectives. To analyze these we will do the feasibility testing to ensure the system is achievable and to ensure that all the components of the system can work efficiently to accomplish their purpose.

6.2 FEASIBILITY

6.2.1 OPERATIONAL FEASIBILITY

Operational requirements of the proposed system are achievable because if we look at the workflow of the whole system there will be clients such as organizations, institutions like university, colleges, schools and any small scale enterprise. These clients will use the system for communicating content to the targeted audience such as employees, students or specific communities. So if we look at the picture the whole scenario with respect to a clients perspective seems successfully operational and also it can be a better and reliable alternative in terms of user's interaction with the system and user experience.

6.2.2 TECHNICAL FEASIBILITY

Now after comparing the operational aspect of the existing and proposed system, we will look whether we have required technology so that we achieve the proposed functionality. As it is a web based solution we will use a web framework based on Python which handles the backend aspects of the project. The frontend requirements are fulfilled by Bootstrap, HTML and CSS. Hence there is enough technical support to build the proposed system.

6.2.3 ECONOMICAL FEASIBILITY

Since this project is for academic purposes and developed with a motive to add developmental knowledge and sharpening technical skill, cost for designing and development can not be considered. All the libraries and tools used in the project will be available for development purpose without any cost. So the proposed system is achievable in terms of development cost.

SOFTWARE REQUIREMENT SPECIFICATION

7.1 REQUIREMENTS

7.1.1 FUNCTIONAL REQUIREMENT

Considering the functional requirements for users (both viewers and admin) the proposed system must be able to:

1. Adding a notice into the system.

2. Adding title, content and metadata like author, publish date related to a notice into the system.

3. Listing all the notice details associated with a particular author and date.

4. Listing all the notices in new to old order.

5. Formatting a particular notice in different styles like bold, italics etc.

6. Updating notice content information.

7. Deleting a particular notice from the system.

7.1.2 NON FUNCTIONAL REQUIREMENT

Non functional requirements for the proposed system must be able to:

1. Performance of the system response time and utilization of resources.

2. Scalability of the system in case the system needs to be enhanced or new features are introduced.

3. Data Integrity – the data must be integral to the users associated with the system.

4. Recoverability if the application gets stuck in adverse conditions.

5. Security – the system must be secure for security attacks which can lead to compromisation of database and business logic.

6. Manageability of the system if the records increase in the system.

7. Interoperability of the system with third party applications or integration with external libraries.

8. Reliability of the system. It must be bug free and must be dependable.

TECHNICAL SPECIFICATION

8.1 FRONTEND SPECIFICATION:

Python Flask Framework – Flask Templating Engine

The front end is specified by the Flask Template Engine based on Python along with HTML5 and Bootstrap CSS. The framework provides a collection of packages along with a built-in dedicated development server.

8.2 BACK END SPECIFICATION:

Flask Web Framework and MySQL Server 5.1 (RDBMS)

Flask Web Framework is used to create APIs to support backend processing. The endpoints are coupled with Flask Template Engine for making requests to the built-in development server. It contains packages that handle form validation functions, URL routing, database connection and manipulation, session storage and retrieval in project.

MySQL is used as the backend tool for storage, retrieval and maintaining the relevant information using web applications. It has got all the database tools with a rich library and numerous useful functions. It is enabled with features that provide security and maintain the integrity of the system.

SYSTEM DESIGN

9.1 DESIGN OVERVIEW:

The design of the system is done by considering the process flow of the system.

From the analysis performed after examining the functional and nonfunctional

requirements we get inputs to design the system mapped into pictorial or structured

representations. It makes the process clear in mind how things like modules and

interfaces are linked with each other to deliver the working model of the proposed

system made by integrated modules. It provides a better understanding of how well

the system is hooked up with processes running in different layers consisting of

each and every module.

9.2 DESIGN DIAGRAM

The systems representation is designed and represented using two models which

are described in detail in the upcoming sections. This shows the data entry sources,

key inputs, processes involved key output consisting each and every module. Also

it shows the interaction of users with different use cases The design diagram used
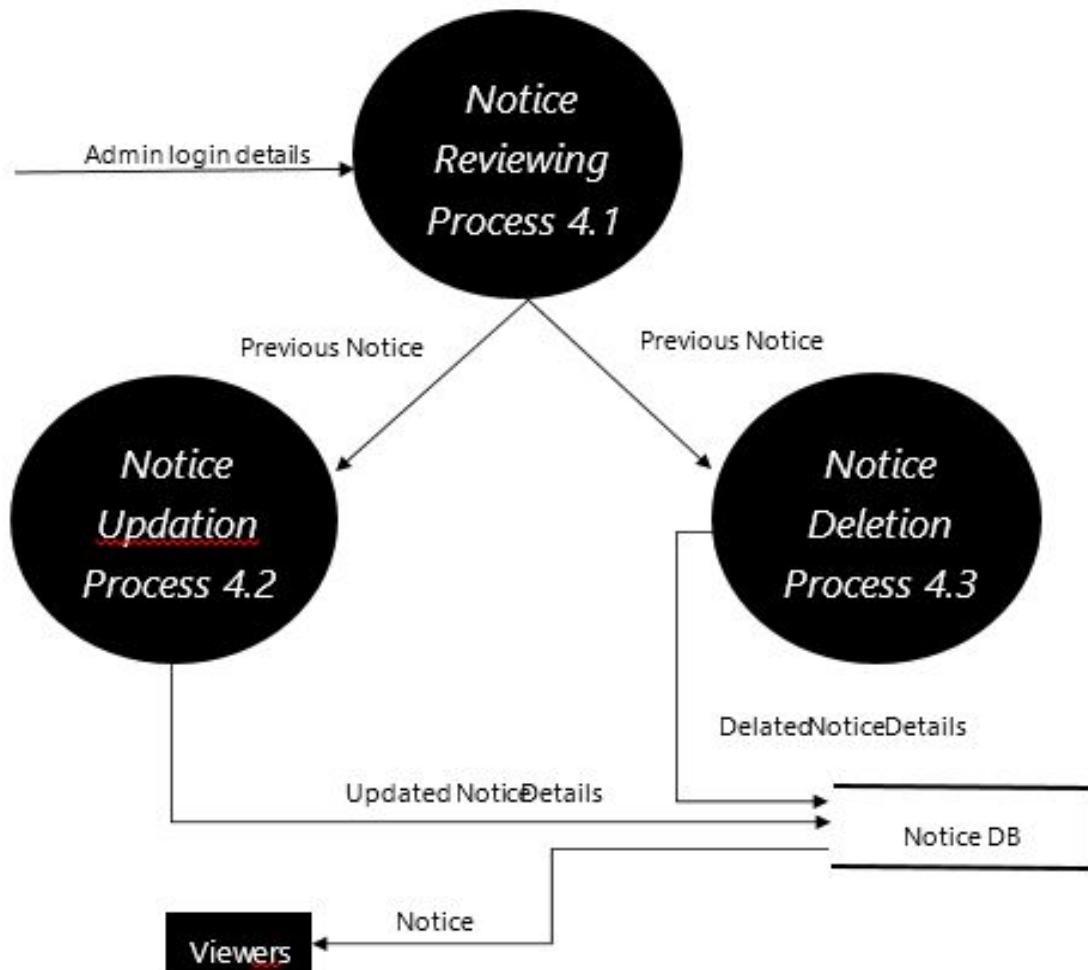
for the illustrations are:

i.)   Data Flow Diagram

DATA FLOW DIAGRAM
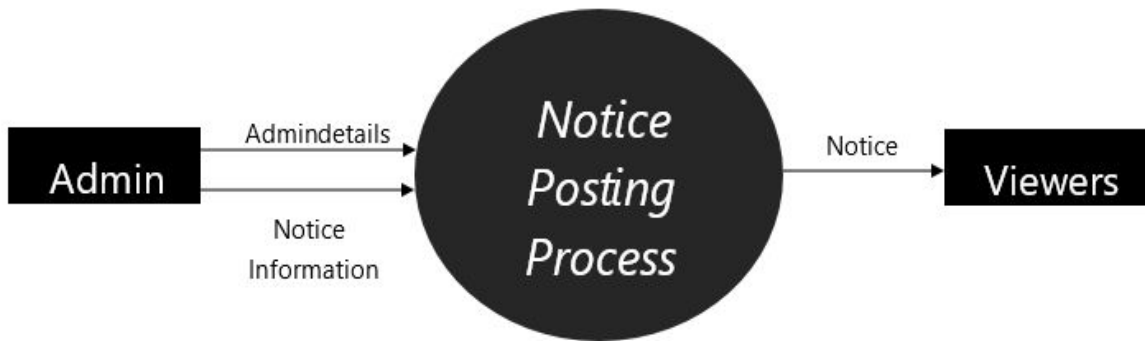
Context Level DFD
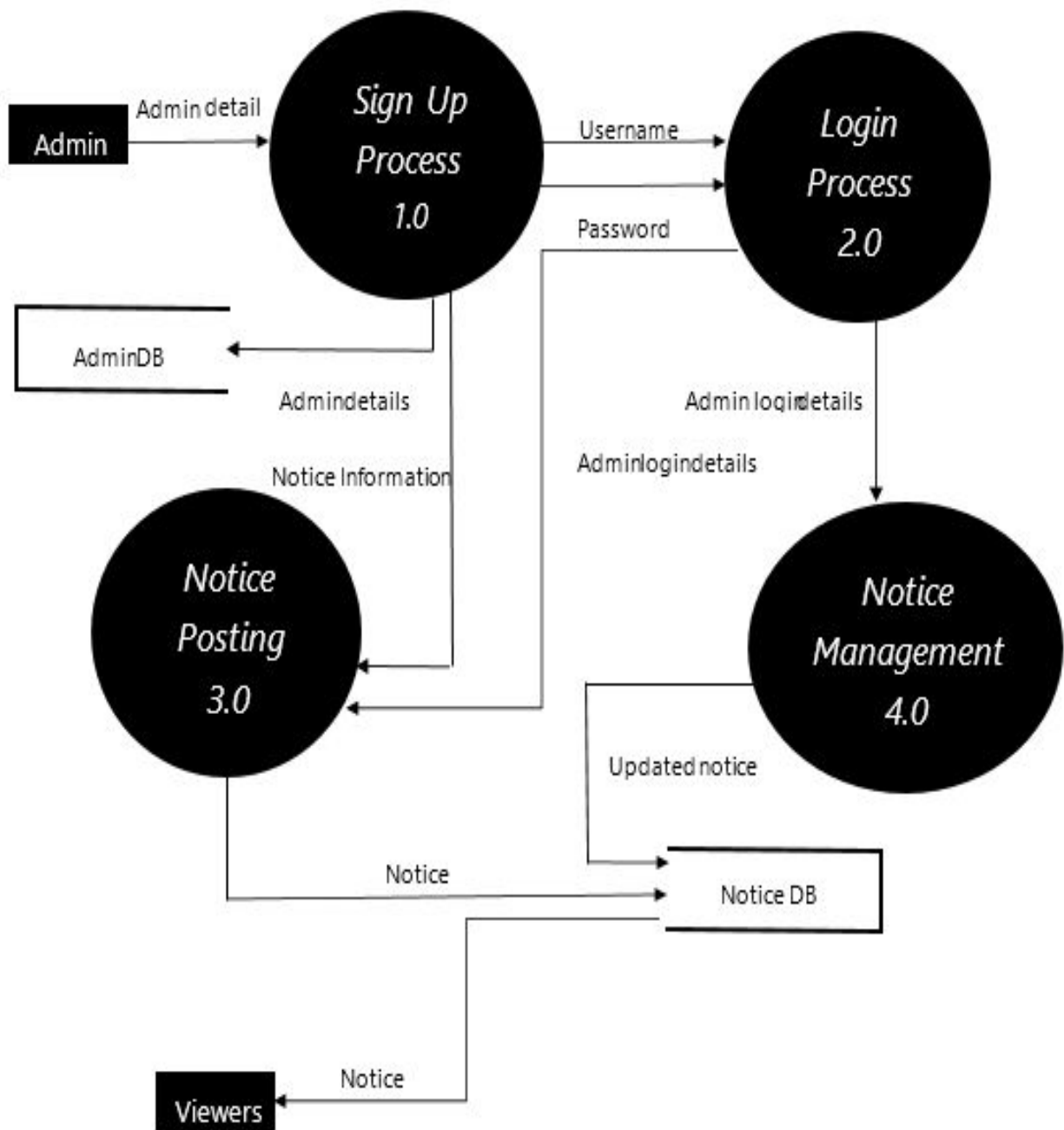
1st Level DFD

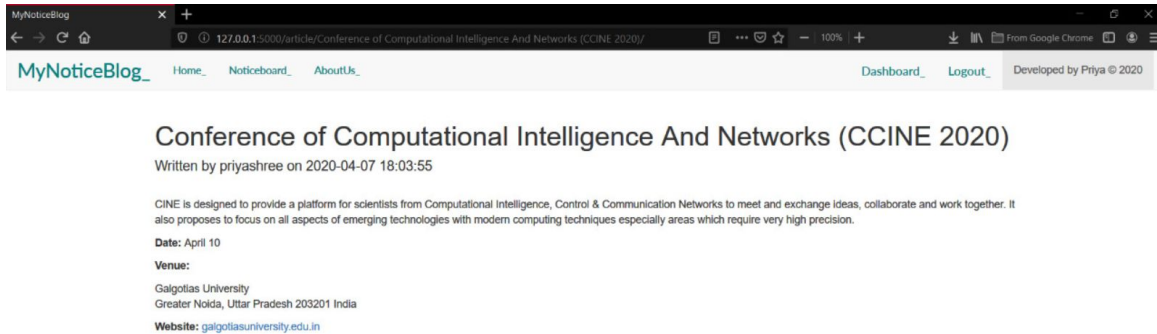# 2nd Level DFD of Notice Management Process 4.0
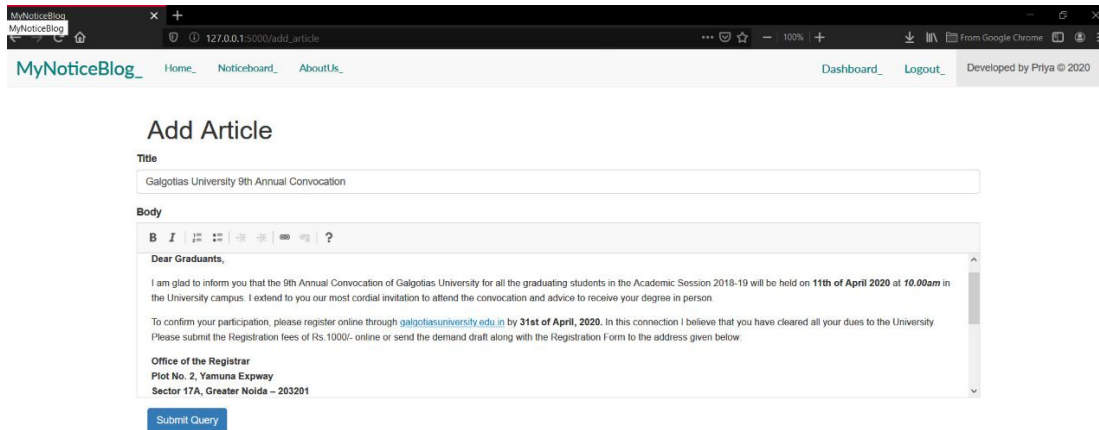
INPUT OUTPUT SCREEN

Fig 1. Detailed Notice View

Fig 2. Notice Posting Process
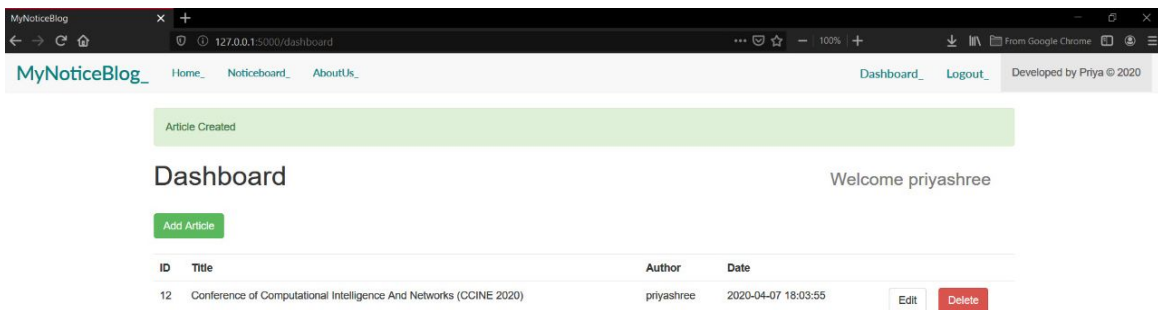

Fig 3. Notice List View

TOOLS & ENVIRONMENT

Minimum Hardware Requirements:

Processor   : Intel Pentium 4

RAM   : 512 MB

Input   : Mouse and Keyboard

Software Requirements:

(For Client)

Operating System   : Windows/Linux/Mac

Web Browser   : IE/Google Chrome/Firefox

(For Server)

Operating System   : Windows/Unix Servers

Web Server   : Flask Web Built-In Server

Front End   : Flask Template Engine,
HTML5, Bootstrap & CSS

Back End   : Flask Web Framework ,MySQL

FUTURE SCOPE

'MyNoticeBlog_'is based on client-server architecture and it means working in a networked environment. This system is versatile and flexible and can be easily modified to suit any further requirements and updates in future. The system has been designed in such a way that hierarchical modification/additional capabilities can be applied at any level.

There is always room for further enhancement of this system in the following areas:

- Addition of notice categories to expand the range of accessibility for the user.

- If required, a distributed database can be implemented. From which many servers hold data and reliability as well as backup will be high.

- Back up of users and notices can also be added to the system to provide real time status to the users.

- Data recovery and zipping mechanism can be implemented to empower the present system.

CONCLUSION

The system is an interactive and minimal blog web application which is capable to serve as an internal notice blogging tool in organisations for a kind of people associated with the same work. It targets small and mid size organisation's requirements. And can be updated to fulfil the extra specific requirements too.

REFERENCES

[1] Patrick Vogel, Thijs Klooster, Vasilios Andrikopoulos, Mircea Lung: A Low-Effort Analytics Platform for Visualizing Evolving Flask-Based Python Web Services. *IEEE Working Conference on Software Visualization (VISSOFT) 2017*.

[2] Erroll Wood, Peter Robinson: NetBoards: Investigating a Collection of Personal Digital Noticeboard in the Workplace, *ITS '14: Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces, 2014*.

[3] Greenberg, S., and Rounding, M. The notification collage: posting information to public and personal displays. In Proc. *SIGCHI conference on Human factors in computing systems, ACM, 2001*.