**EXAMWIZ**

**A project report submitted in partial fulfillment of the requirements for the award of the degree of**

**BACHELOR OF COMPUTER SCIENCE**

**By**

# DRUPAD M D (Reg no DVAVSCS027)

**GOUTHAM SUNDAR S S (Reg no DVAVSCS029)**

# M P NAVANATH (Reg noDVAVSCS014)

**Under the Guidance of**

**Mrs. DEEPA RAMACHANDRAN**



**DEPARTMENT OF COMPUTER SCIENCE**

**ST. JOSEPH’S COLLEGE(AUTONOMOUS), DEVAGIRI, CALICUT**

**MARCH 2024**



**ST. JOSEPH’S COLLEGE (AUTONOMOUS)**

**DEVAGIRI, CALICUT**

**EXAMWIZ**

Report of the project submitted to St. Joseph’s College (Autonomous) Devagiri, Calicut

Affiliated to the University of Calicut

**DRUPAD M D (REG NO. DVAVSCS027)**

**GOUTHAM SUNDAR S S (REG NO. DVAVSCS029)**

**M P NAVANATH (REG NO. DVAVSCS014)**

Programme: Bachelor of Computer Science

Semester: VI

DEPARTMENT OF COMPUTER SCIENCE

March 2024

Principal Head of the Department Supervisor

College Seal

**PROJECT REPORT**

**ON EXAMWIZ**



**CERTIFICATE**

This Is To Certify That The Project Report Titled

# EXAMWIZ

Submitted By

# DRUPAD M D (Reg no DVAVSCS027)

**GOUTHAM SUNDAR S S (Reg no DVAVSCS029)**

# M P NAVANATH (Reg noDVAVSCS014)

St. Joseph’s College(Autonomous) Devagiri, Calicut in partial fulfillment of the requirements for the award of degree Bachelor of Computer Science during the Year 2021-2024.

Internal Examiner External Examiner

**DECLARATION**

We hereby declare that the project entitled “**EXAMWIZ**” has been undertaken by us for the award of the degree of Bachelor of Computer Science. We have completed this project under the guidance of Mrs. Deepa Ramachandran Department of Computer Science, St. Joseph’s College(Autonomous),Devagiri, Calicut.

Place : Calicut

Date:

# DRUPAD M D (Reg no DVAVSCS027)

**GOUTHAM SUNDAR S S (Reg no DVAVSCS029)**

# M P NAVANATH (Reg no DVAVSCS014)



**CERTIFICATE**

This is to certify that the project report titled **EXAMWIZ**a project work done by them during the academic year 2023-2024 under my guidance and supervision in partial fulfillment of the requirement of Bachelor of ComputerScience.

Place:Devagiri Mrs.Deepa Ramachandran

Date:

# ACKNOWLEDGEMENT

We wish to express our sincere gratitude to **Dr. Boby Jose** , Principal of St. Joseph’sCollege (Autonomous), Devagiri, Calicut for providing necessary facilities for developing your project.

We are also grateful to Ms.Asha Unnikrishnan , Head of the Department in charge,as well as Mrs.Deepa Ramachandran,our internal guide for the irimmense support and valuable suggestions.

We are also grateful to faculty members of,RissTechnologies,Calicut For providing assistance to do this work in their organization.

# DRUPAD M D (Reg no DVAVSCS027)

**GOUTHAM SUNDAR S S (Reg no DVAVSCS029)**

# M P NAVANATH (Reg no DVAVSCS014)

# SYNOPSIS

This is a simple yet effective academic tool for students of all phases. The basic operation of this is to accept Syllabi, a question paper (for identifying pattern) and Study materials(optional) to generate multiple Model Question papers with answer key for practice and evaluation for students and others for the proficiency in their area of study.

The features of this software include (but are not confined to:Generating Question papers,Including reading and oral questions,Finding enough study materials based on the syllabus provided,Generating Answer keys for self-evaluation,Mock test facility,Multiple exam question generator.

The software may include the following increments: ,Predicting the academic level of the student,Voice detection for Speaking input.

# TABLEOFCONTENTS

**CERTIFICATE I Ⅱ**

**[DECLARATION III](#_30j0zll)**

**CERTIFICATE II. IV**

**CERTIFICATE III V**

**[ACKNOWLEDGEMENT V](#_1fob9te)I**

**[SYNOPSIS VI](#_39kk8xu)I**

**CHAPTERI**

1. **INTRODUCTION** 
   1. AN OVERVIEW 1
   2. INTRODUCTION MISSION OF THE PROJECT 1
   3. [BACKGROUND STUDY](#_26in1rg)
      1. THE ORGANIZATION PROFILE 1
      2. STUDY ON EXISTING SYSTEM 2
      3. DISADVANTAGES OF EXISTING SYSTEM 2

**CHAPTER II** 3

1. **SYSTEM ANALYSIS**  4
   1. [STUDY PROPOSED SYSTEM 4](#_3j2qqm3)
      1. [ADVANTAGES OF PROPOSED SYSTEM 5](#_1y810tw)
   2. USER REQUIREMENT SPECIFICATION 5
   3. SOFTWARE REQUIREMENT SPECIFICATION 7
   4. FEASIBILITY STUDY 9
   5. [SYSTEM SPECIFICATION 10](#_49x2ik5)
      1. REQUIREMENTS FOR DEVELOPING APP 11
      2. REQUIREMENTS FOR USING APP 11

2.6 [COST ESTIMATION AND SCHEDULING 11](#_ihv636)

**[CHAPTER III](#_vx1227)**

1. **DESIGN AND DEVELOPMENT PROCESS** 12
   1. [FUNDAMENTAL DESIGN CONCEPTS 13](#_3fwokq0)
   2. [DESIGN NOTATIONS 14](#_1v1yuxt)
      1. [DATA FLOW DIAGRAM 14](#_4f1mdlm)
   3. [DESIGN PROCESS 21](#_1opuj5n)
      1. [DATABASE DESIGN 21](#_19c6y18)
      2. [NORMALIZATION 26](#_1mrcu09)
      3. [INPUT DESIGN 27](#_46r0co2)
      4. [OUTPUT DESIGN 28](#_2lwamvv)

**CHAPTER 1V**

1. **CODING** 29

4.1 FRONT END 30

4.2 BACK END 31

**CHAPTER V**

**5.TESTING AND IMPLEMENTATION** 33

* 1. [TESTING 34](#_4k668n3)
     1. TESTING METHODOLOGIES 34
     2. [DIFFERENT TESTING 35](#_3ygebqi)
  2. [QUALITY ASSURANCE POLICIES](#_4bvk7pj) 42
     1. [GENERIC RISKS 42](#_1664s55)
  3. [SYSTEM IMPLEMENTATION 43](#_25b2l0r)
     1. IMPLEMENTATION PROCEDURES 45
  4. [SYSTEM MAINTENANCE 45](#_1jlao46)

**CHAPTER VI**

1. **CONCLUSION** 47
   1. CONCLUSION48
   2. [SCOPE FOR FURTHER ENHANCEMENTS 48](#_2iq8gzs)
   3. BIBLIOGRAPHY 48

**[ANNEXURE-A](#_3hv69ve)** [49](#_3hv69ve)

1. [INPUT AND OUTPUT DESIGN 50](#_1x0gk37)
2. [SAMPLESOURCE 55](#_2w5ecyt)

**[ANNEXURE-B](#_3vac5uf)** [58](#_3vac5uf)

[ABBREVIATION 59](#_pkwqa1)

# LISTOFTABLES

**SNo Title Page**

|  |  |  |
| --- | --- | --- |
| **3.3.1.1** | **Login Table** | **21** |
| **3.3.1.2** | **User Table** | **21** |
| **3.3.1.3** | **Test Table** | **22** |
| **3.3.1.4** | **Reviews Table** | **22** |
| **3.3.1.5** | **Staff Table** | **23** |
| **3.3.1.6** | **Test Questions Table** | **23** |
| **3.3.1.7** | **Question Table** | **23** |
| **3.3.1.8** | **Result Table** | **24** |
| **3.3.1.9** | **Complaint Table** | **24** |
|  |  |  |

# CHAPTER - I

# INTRODUCTION

## EXAMWIZ : AN OVERVIEW

In response to the increasing demand for efficient examination processes in our digital age, ExamWiz emerges as a comprehensive web application. From preparation to exam management, it caters to the needs of students, educators, and institutions alike. For students, ExamWiz accepts various study materials, generating multiple model question papers for diverse practice. Educators benefit from streamlined assessment creation, with intuitive tools ensuring relevance and adherence to standards. The platform introduces grading automation, saving time and ensuring accuracy. Exam management is centralized, providing a user-friendly interface for scheduling, monitoring, and conducting exams. Results generation is quick and insightful, offering valuable data for performance analysis. With its user-friendly design, ExamWiz redefines academic proficiency in the modern, digital landscape.

## INTRODUCTION MISSION OF PROJECT

Welcome to ExamWizz, a pioneering project revolutionizing education with an innovative examination platform.ExamWizz aims to simplify and optimize the examination process, leveraging technology to empower students, support educators, and enhance institutional efficiency. Our commitment is to redefine academic proficiency through user-friendly interfaces, diverse study material acceptance, and automated grading, fostering a dynamic learning environment for success.

## BACKGROUND STUDY

## THE ORGANIZATION PROFILE

REGIONAL TECHNOLOGIES is a rapidly growing company that provides professional IT services. We are one of the largest and Best software development companies in Kerala with focus on .Net, PHP, Java, Software testing, SEO and Web Design. As a leader in providing Offshore Software Development and related services, REGIONAL TECHNOLOGIES functions from an offshore set-up, based in Kerala, India. Our reputation enables us to serve in terms of Outsourced Software Development, Web Development, designing of websites and their corresponding development. Particularly high end developments using Microsoft .NET, Java J2EE platforms, ASP, ASP.NET, PHP development as well as VB.6.0 Development

adjoins us to a particularly niche sector of the global servicing. We have worked for and have

provided services to the 6 of the top 10 Fortune 500 companies of the world. We aim to provide our clients with comprehensive, end-to-end technology solutions that give them an advantage over the competition. From building applications that increase your productivity, to internet-enabling your business for maximum profits.

## MISSION

The mission of the Exam Wiz project is to provide a comprehensive and user-friendly platform for exam preparation. It aims to enhance the learning experience of students by offering a range of study materials, practice questions, and mock exams tailored to their specific needs. The project also aims to utilize advanced technologies, such as AI and machine learning, to personalize study plans and provide feedback to help students improve their performance. Ultimately, the goal is to empower students to succeed in their exams and achieve their academic goals.

## STUDY EXISTING SYSTEM

Existing system:

There are several methods and technologies for generating questions and answers from study materials, including natural language processing including Google ,ChatGPT,BERT, etc.

Moreover students depend on their ingenuity and intuition to prepare such potential questions.

There are also many systems and formats to conduct and evaluate tests and exams conducted online. But teachers prefer offline mode due to the heftiness related with such systems.

This shows that they are not always reliable and are only confined to their part of specification. They are not flexible systems that can be used for streamlining the entire process in such a uniform way.

## DISADVANTAGES OF EXISTING SYSTEM

* + - * They were not designed for the preparation and conduction of exams.
      * Not reliable.
      * No systems are entirely exam oriented.
      * We cannot generate summary and questions on same platform reliably.
      * Does not provide the facility to conduct and manage examinations.
      * Does not provide methods for student grading and profiling.

# CHAPTER-II SYSTEM ANALYSIS

# 2. SYSTEM ANALYSIS

A system study is a process of studying a procedure to identify the objectives and purposes of a system as well as to analyze the existing system’s problems and drawbacks. The system analysis will create systems and procedures that will achieve these objectives as well as solutions to the problems in an efficient way. It is the process of gathering and interpreting facts, diagnosing problems and using the information to recommend improvements to the system.Training,experienceandcommonsensearerequiredforthecollectionoftheinformationneeded to do theanalysis.

## STUDY PROPOSED SYSTEM

The proposed system of ExamWiz aims to revolutionize the way students prepare for exams by providing a comprehensive and personalized learning experience. Here's an overview of the key features and components of the proposed system:

1. User Registration and Profile Management: Students can create accounts on ExamWiz, where they can manage their profiles, track their progress, and access personalized study recommendations.

2. Study Material Repository: ExamWiz will provide a repository of study materials, including notes, videos, and reference materials, curated by subject matter experts.

3. Practice Questions and Mock Exams: Students can access a vast database of practice questions and full-length mock exams, which are designed to simulate the actual exam environment.

4. Personalized Study Plans: Based on the student's performance in practice tests and mock exams, ExamWiz will generate personalized study plans, suggesting areas for improvement and recommending study materials.

5. AI-driven Performance Analytics: ExamWiz will utilize AI and machine learning algorithms to analyze student performance, identify strengths and weaknesses, and provide actionable insights for improvement.

6. Interactive Learning Tools: The platform will offer interactive learning tools, such as flashcards, quizzes, and games, to make studying more engaging and effective.

7. Community and Collaboration: ExamWiz will facilitate peer-to-peer learning and collaboration through discussion forums, study groups, and live chat support.

8. Mobile Compatibility: The platform will be mobile-friendly, allowing students to access study materials and practice questions on-the-go.

Overall, the proposed system of ExamWiz aims to leverage technology to provide a holistic and personalized approach to exam preparation, empowering students to achieve their academic goals.

## ADVANTAGES OF PROPOSED SYSTEM

* + - * Purely exam oriented software for students and teachers.
      * Reliable and can include pdfs, docs and audio files.
      * Generates question and answers based on the notes directly.
      * Generates summary for easier understanding of topic
      * Generates keywords to efficiently grasp the topic.
      * Also helps to conduct, attend and grade tests easily within the same platform.
      * Natural Language Processing along with Machine Learning will provide human-centric results.

## USER REQUIREMENT SPECIFICATION

After thorough analysis,Our system has been presented with the following modules:

1. Admin
2. User
3. Staff

## ADMIN

In this module the admin has the privilege to control the system.Admin is the main as he can view users and staff.

* + The main function of the Administrator is to control the access and other various managing functions of the software.
  + Login:Admin can login to the website using their username and password.
  + Add and manage Staff: admin can add and manage Staff.
  + View Users: Admin can view users
  + View Complaints and reply: admin can view complaints and reply to users.
  + View reviews ; admin can view reviews.

## USER

* + Login:The user must signup before logging into the application.
  + View and edit profile : Users can view and edit profile.
  + Generate question : Users can use the software to upload their documents to generate question.
  + Generate key-points and summary : Users can use the software to upload their documents to generate key-points and summary.
  + Exam centre : they will be able to attend the examinations conducted by staff and also see the results.
  + View Complaints and reply: admin can view complaints and reply to users.
  + View reviews ; admin can view reviews.

## STAFF

* + Login :The staff must signup before logging into the application.
  + Staff can only enroll into the site with the proof of their professional status. They can upload their files and generate questions and answers.
  + Generate question and answers : They can upload their files and generate questions and answers.
  + They can further use this generated questions if they need to manage and conduct tests in the platform.
  + Test : They can manage and conduct multiple tests and structure the questions in their needs.
  + View Complaints and reply: admin can view complaints and reply to users.
  + View reviews ; admin can view reviews.

## SOFTWARE REQUIREMENTS SPECIFICATION

Once the system requirements are found then we have to determine whether a particular software package fits for those system requirements.

## PyCharmIDE2017.1.2

PyCharm is a dedicated Python Integrated Development Environment (IDE) providing a wide range of essential tools for Python developers, tightly integrated to create a convenient environment for productive Python, Web, and data science development. It is developed by the Czech company Jet Brains.It provides code analysis, a graphical debugger, an integrated unit tester, integration with version control systems (VCSes), and supports web development with Django as well as data science with Anaconda.PyCharm cross-platform,withWindows, macOS and Linux versions. PyCharm provides API so that developers can write their own plugins to extend PyCharm features. Several plugins from other JetBrains IDEa also work with PyCharm. There are more than 1000 plugins which are compatible with PyCharm.JetBrains as developed PyCharm as across-platformIDE for Python.In Addition to supporting versions 2.x and 3.x of Python, PyCharm is also compatible with Windows,Linux,andmacOS.At the sametime,the tools and features provide by PyCharm help programmers to write a variety of software applications in Python quickly and efficiently. The developers can even customize the PyCharm UI according to their specific needs and preferences. Also, they can extend the IDE by choosing from over 50 plug-ins to meet complex project requirements. PyCharm makes it easier for developers to implement both local and global changes quickly and efficiently. The developers can even take advantage of the refactoring options provided by the IDE while writing plain Python Code and working with Python frameworks. They can rename and move refactoring forfiles ,classes, functions ,methods,properties,parameters, and local/global variables.Likewise,they can improve code quality by extracting variables,fields,constants,and parameters. Also, PyCharm allows programmers to break up longer classes and methods through extract methods. PyCharm makes it easier for programmers to write various webapplications in Python supporting widely used web technologies like HTML,CSS,JavaScript,Typescript and CoffeeScript.The web developers can use the live editing preview option provided by the IDE to view a single web page simultaneously in the editor and browser.At the same time,the live edit feature provided by the IDE enables programmers to see the changes made to the code instantaneously on a web browser.

PyCharm further allows developers to avail a JavaScript debugger as well as CoffeeScriptand Typescript editors. It even simplifies isomorphic web application development by supporting both Angularjs JSand NodeJS.

## SQLyog

SQLyog is a powerful database management tool designed to simplify database administration tasks for MySQL databases. Developed by Webyog, SQLyog provides a comprehensive set of features for database developers, administrators, and analysts to efficiently manage, monitor, and manipulate MySQL databases.

One of SQLyog's key features is its intuitive user interface, which allows users to easily navigate and interact with their databases. The graphical user interface (GUI) provides a visual representation of the database schema, making it easy to view and modify tables, indexes, and relationships. Users can also write and execute SQL queries directly within the tool, with syntax highlighting and auto-completion features to aid in query construction.

SQLyog offers a range of tools for database management, including backup and restore functionality, data synchronization, and schema comparison. The tool's backup feature allows users to create full or partial backups of their databases, while the restore feature enables them to quickly restore data in the event of a failure. Data synchronization allows users to keep multiple databases in sync, ensuring that data is consistent across different environments. Schema comparison helps users identify and reconcile differences between database schemas, making it easier to manage database changes and updates.

In addition to its management tools, SQLyog also provides a variety of monitoring and optimization features. Users can monitor database performance in real-time, with tools for analyzing query performance, identifying bottlenecks, and optimizing database configuration. SQLyog also includes tools for database maintenance, such as optimizing tables, checking for database corruption, and managing user privileges.

8

## 2.4 FEASIBILITY STUDY

A feasibility study is an assessment of the practicality of a proposed system or project. A feasibility study aims to objectively and rationally uncover weakness and strengths of an existing business or proposed venture,opportunities and threats present in the natural environment, the resources required to carry through, and ultimately the prospects of success.In its simplest terms,the two criteria to judge feasibility are cost required and value to be attained. A feasibility study evaluates the project’s potential for success; therefore, perceived objectivity is an important factor in the credibility of the study for potential investors and lending institutions. It must therefore be conducted with an objective, unbiased approach to provide information upon which decisions can be based. The feasibility study is done in thesephases.

* Technical Feasibility
* Economic Feasibility
* Operational Feasibility

## TECHNICAL FEASIBILITY

It investigates the technical feasibility of each implementation alternative. It analyzes and determines whether the solution can be supported by existing technology or not. The analyst determines whether current technical resources be upgraded or added that fulfill the new requirements. The analyst must find out whether current technical resources can be upgraded or added in a manner that fulfills the request under considerations. Our software is more user friendly. We are using python language for coding. So, it is easy to understand and it is more readable.

## ECONOMICAL FEASIBILITY

The purpose of an economic feasibility study is to demonstrate the net benefit of a proposed project for accepting or disbursing electronic funds/benefits and costs to the agency,other state agencies and the general public as a whole. It is evaluating the effectiveness of candidate system by using cost/benefit analysis method.It demonstrates the net benefits from the candidate system in terms of benefits and costs to the organization. Software is said to be economically feasible if it focuses on issues like cost incurred on software development to produce long term gains for an organization,cost required to conduct full software

investigation, cost of hardware, software, development team, and training.Our application can be economically feasible since it can be used by all sections of the society.

## OPERATIONAL FEASIBILITY

It determines whether the system is operating effectively once it is developed and implemented. It ensures that the management should support the proposed system and its working feasible in the current organizational environment. It analyzes whether the users will be affected and they accept the modified or new business methods that affect the possible system benefits. It also ensures that the computer resources and network architecture of candidate system are workable.

The proposed system is easily understandable.the maintenance and working of proposed system need less human effort. The application can be easily operated via smartphone thus this project passes these entire tests for feasibility and thus found feasible.

## SYSTEM SPECIFICATION

* + 1. **REQUIREMENTS DEVELOPING APP**

HARDWARE:

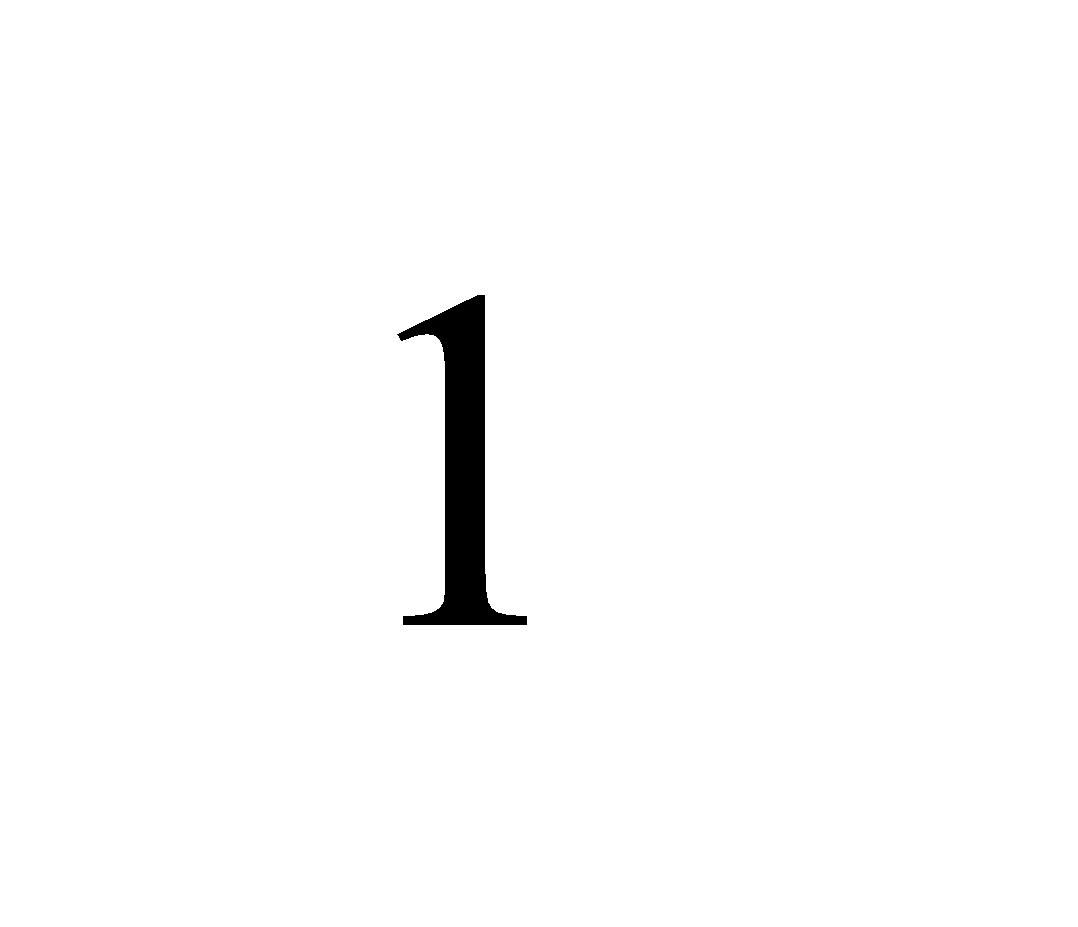
* PROCESSOR:Intel COREi3 (4TH GEN) and above or AMD Dual Core or equivalent.
* RAM:Minimum 4GB and Above
* RAM:Minimum 2GB,4GB Recommended
* DisplayResolution:1280x800px

SOFTWARE**:**

OS:Microsoft Windows 7 And Above(64Bit)

Apps:

* + MySQL
  + PyCharmIDE



## REQUIREMNTSFORUSINGAPP

HARDWARE:

* PROCESSOR:Qualcomm Snapdragon 400 Series and above or Equivalent Exynos or MediaTek Processor
* RAM:Minimum 2GB and Above
* ROM:Minimum 100MB Storage
* INTERNET:Required

SOFTWARE:

OperatingSystem:AndroidVersion5.0andabove

## COST ESTIMATION & SCHEDULING

* H/W:40000
* S/W:5000
* PERSONNEL:3000
* OTHER CHARGES:2000

## 

## SCHEDULING

* Requirements Collection-2 weeks
* Analysis And Design-2 weeks
* Coding-4 weeks
* Testing-1 week

# CHAPTER-III

**DESIGN AND DEVELOPMENT PROCESS**

12

## FUNDAMENTAL DESIGN CONCEPTS

Over the course of software engineering history, a collection of essential design principles has emerged. These principles serve as a cornerstone for software designers, providing them with a solid foundation upon which they can build more advanced design methodologies. These concepts are integral to both traditional and object-oriented software development approaches.:

## Abstraction

Abstraction simply means to hide the details to reduce complexity and increases efficiency or quality. Different levels of abstraction are necessary and must be applied at each stage of the design process so that any error that is present can be removed to increase the efficiency of the software solution and to refine the software solution.

## Architecture

Architecture in software design refers to the methodology of structuring something. In the context of software, it involves the arrangement of different elements and the handling of data within the structure. These elements interact with each other and utilize the structure's data as part of the architectural process..

## Separation Of Concerns

The separation of concerns is a design principle that proposes breaking down a complex problem into smaller, more manageable pieces. By doing so, each piece can be addressed or optimized independently. This approach reduces the effort and time required to solve the problem, making it more manageable overall.

## Modularity

Modularity involves breaking down a system or project into smaller components to simplify its complexity. Similarly, in design, modularity entails subdividing a system into smaller parts, allowing for independent creation. These parts can then be utilized across different systems to perform various functions..

## InformationHiding

Information hiding involves concealing data to prevent access by unauthorized parties. In software design, this is accomplished by structuring modules so that the information they contain is inaccessible to other modules..

## Refinement

Refinement is the process of improving something to eliminate impurities and enhance its quality. In software design, refinement involves developing or presenting the software or system in a detailed manner, elaborating on its structure and functionality. It's crucial for identifying and reducing errors..

## Object-OrientedDesignConcepts

The object-oriented (OO) paradigm is extensively utilized in contemporary software engineering. OO design incorporates various concepts like classes and objects, inheritance, messages, and polymorphism, among others, as its distinguishing features..

## DESIGN NOTATIONS

## DATA FLOW DIAGRAM

A graphical representation is employed to depict and analyze the flow of data within a system, whether manual or automated, encompassing processes, data storage, and system delays. Data flow diagrams serve as a central tool, forming the foundation for the development of other components. The transition of data from input to output via processes can be logically described independently of the system's physical elements. These logical data flow diagrams illustrate the actual implementation and movement of data among individuals, departments, and workstations. DFDs represent one of the most crucial modeling tools used in physical design, illustrating the data flow through various processes within the system.

# Notations used in DFD

In DFD, there are four symbols, they as follows:

* + - * Process



* + - * Data Store



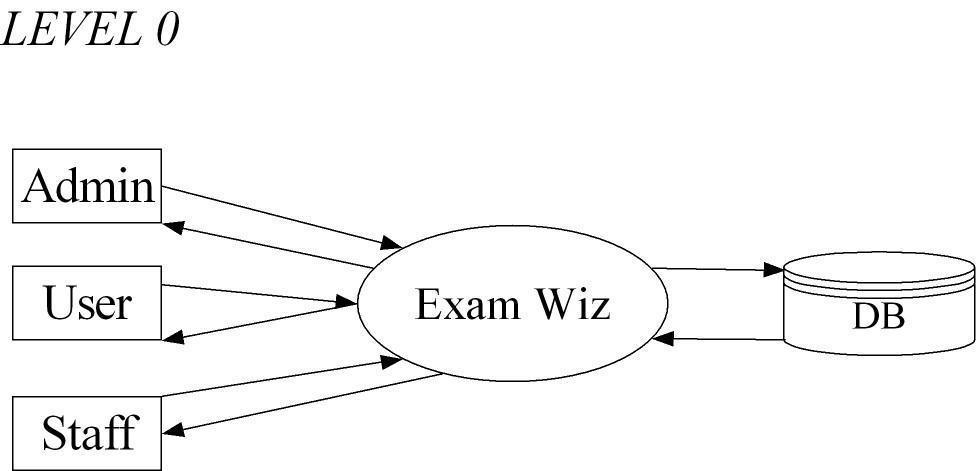
* + - * External Entity



* + - * Direction Of Flow

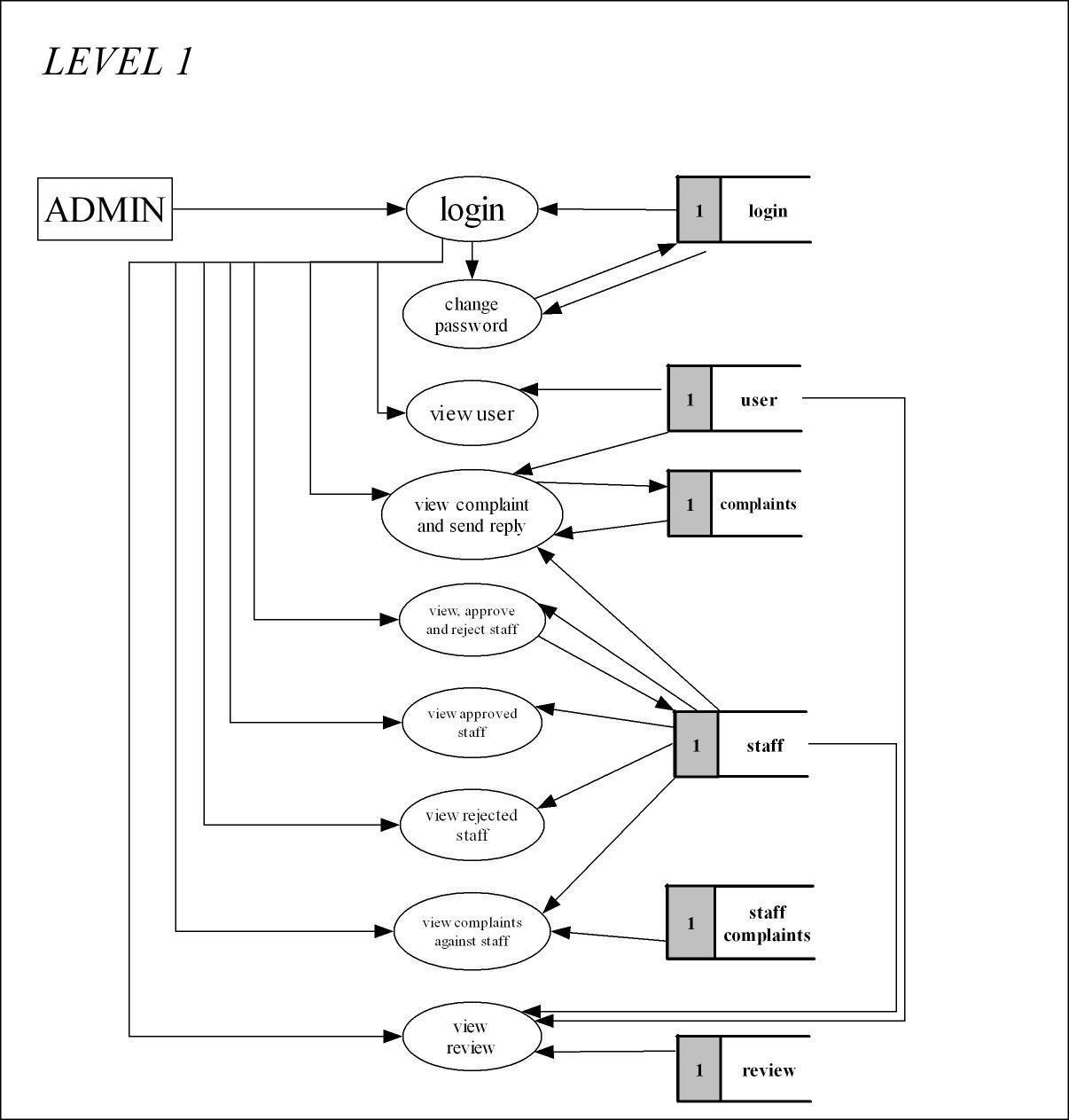


## LEVEL 0 DFD

****

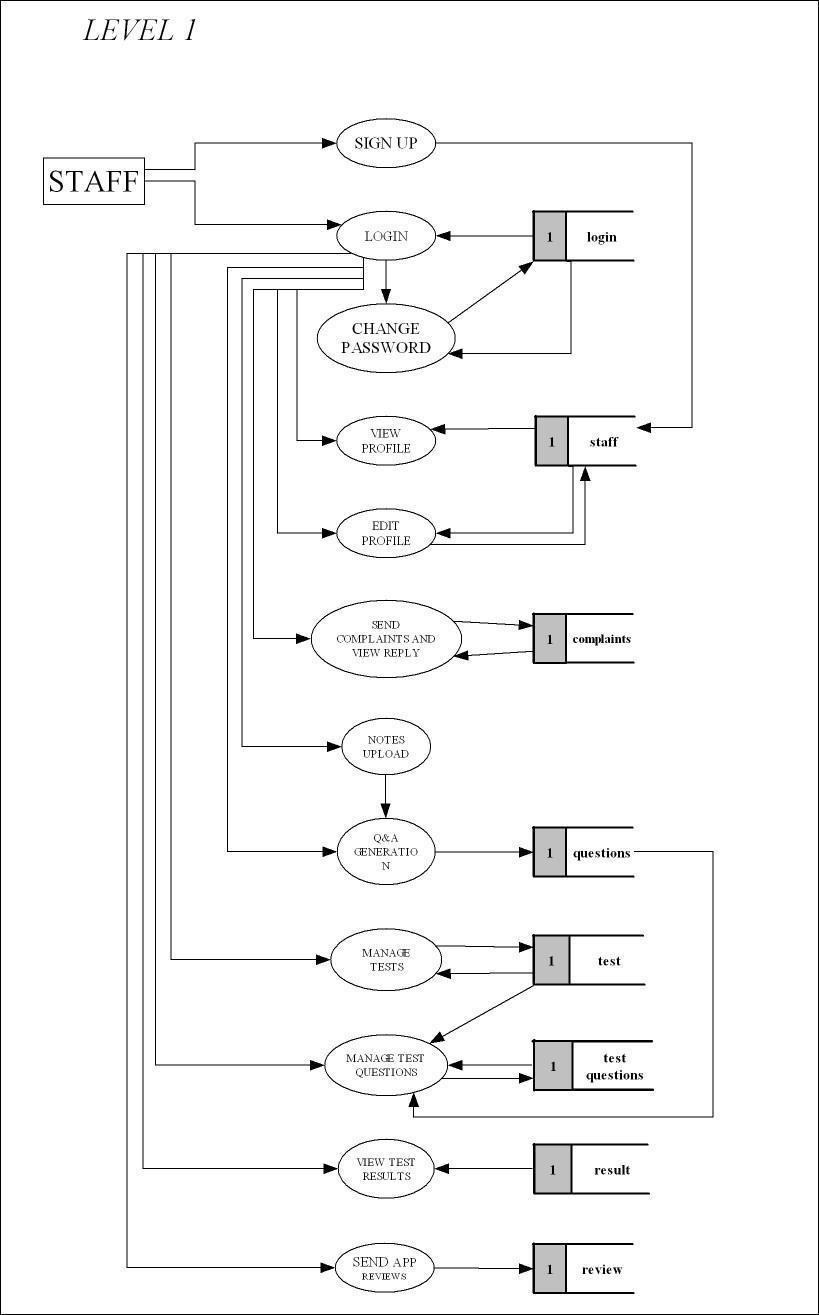
**Figure 3.2.1**

## LEVEL 1.1DFD(Admin)

****

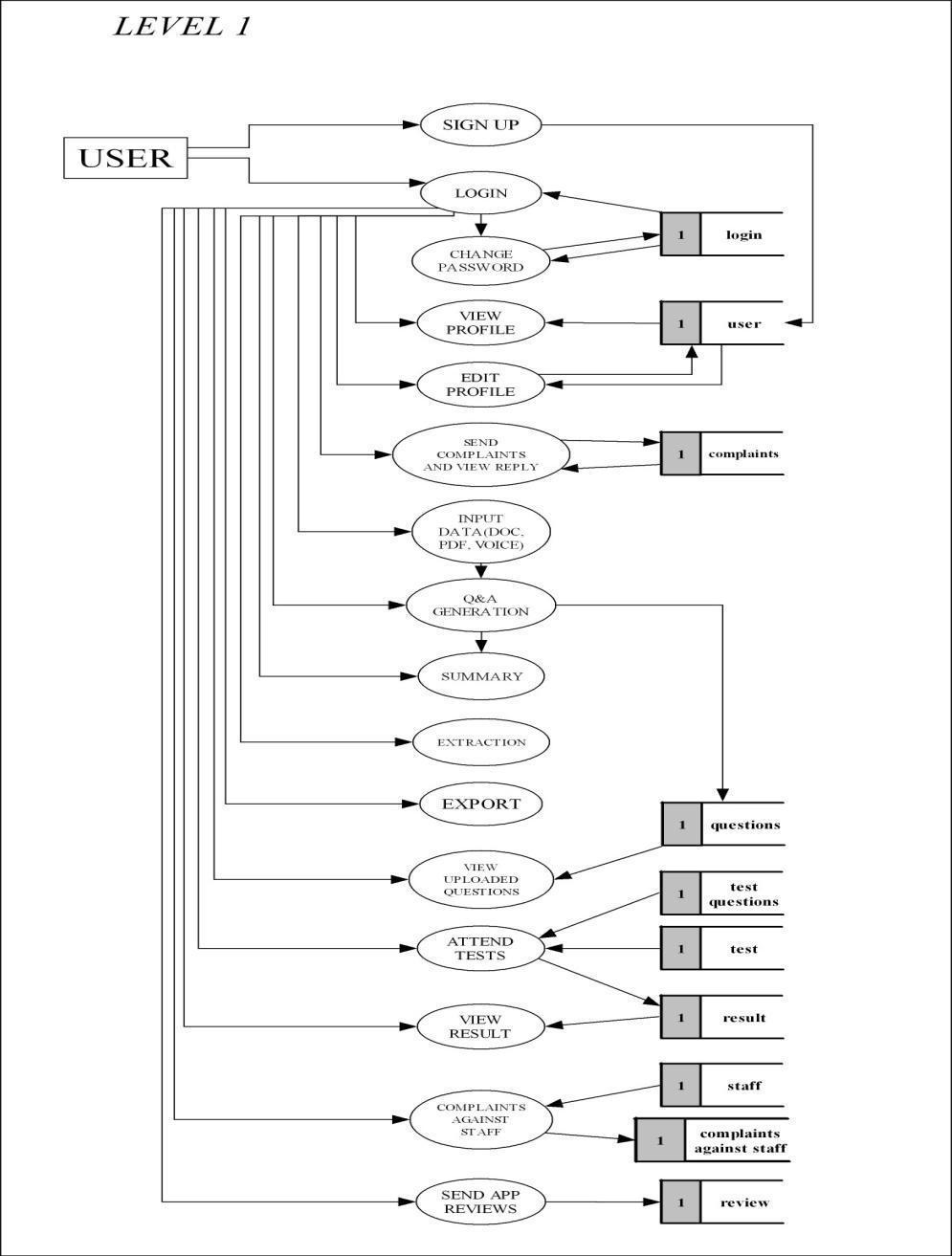
**Figure 3.2.1.1**

## LEVEL 1.2DFD(STAFF)

****

**Figure 3.2.1.2**

## LEVEL 1.3DFD(User)



**Figure 3.2.1.3**

18

## DESIGN PROCESS

## DATABASE DESIGN

A table is a structured data format that arranges information into rows and columns, facilitating both storage and presentation of data. Databases typically consist of multiple tables, each tailored for a specific purpose and containing its own set of fields based on the required data. In database tables, fields are represented as columns, and individual entries, or records, as rows. Accessing specific values from a table involves requesting data from individual columns and rows. Primary key fields are utilized to uniquely identify records within a table.

**Table3.3.1.1:LOGIN TABLE**

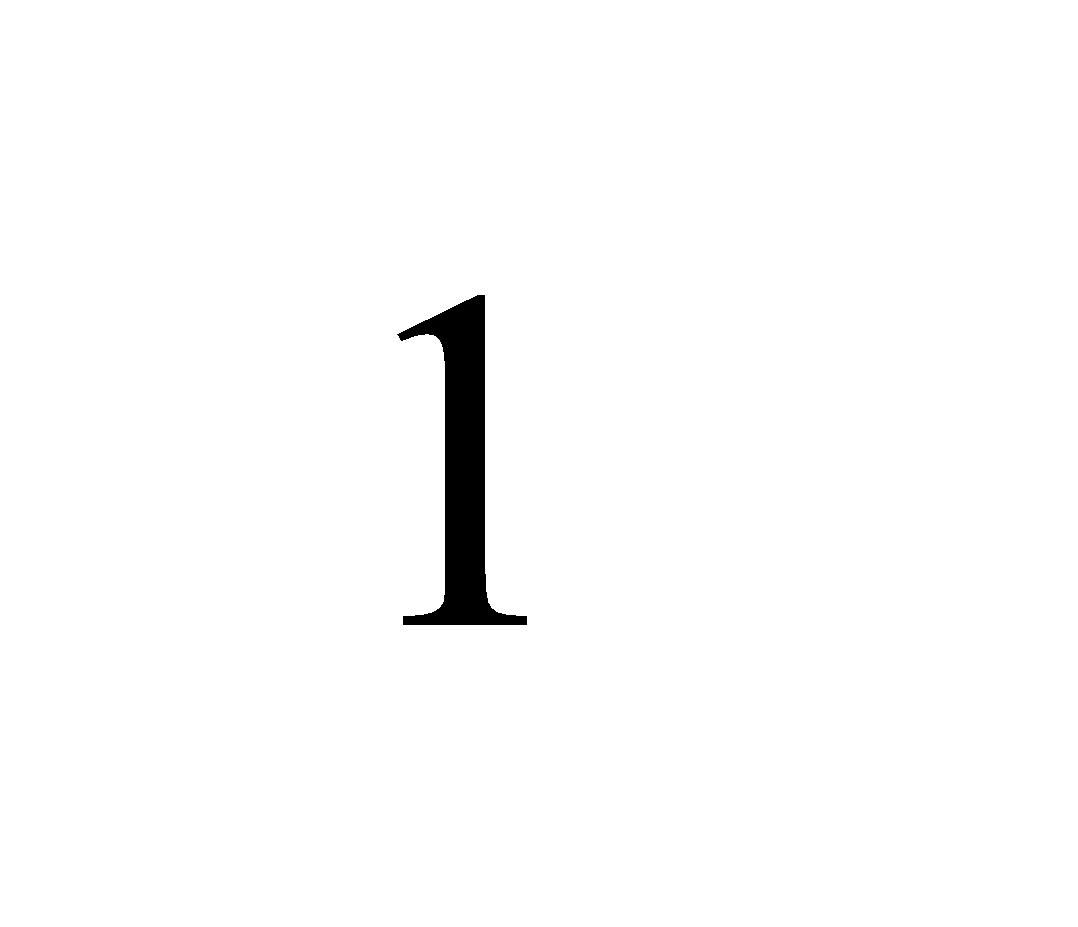
|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | big | 20 | Primary Key |
| username | VARCHAR | 100 | Not Null |
| password | VARCHAR | 100 | Not Null |
| Type | VARCHAR | 30 | Not Null |

**Table 3.3.1.2: USER TABLE**

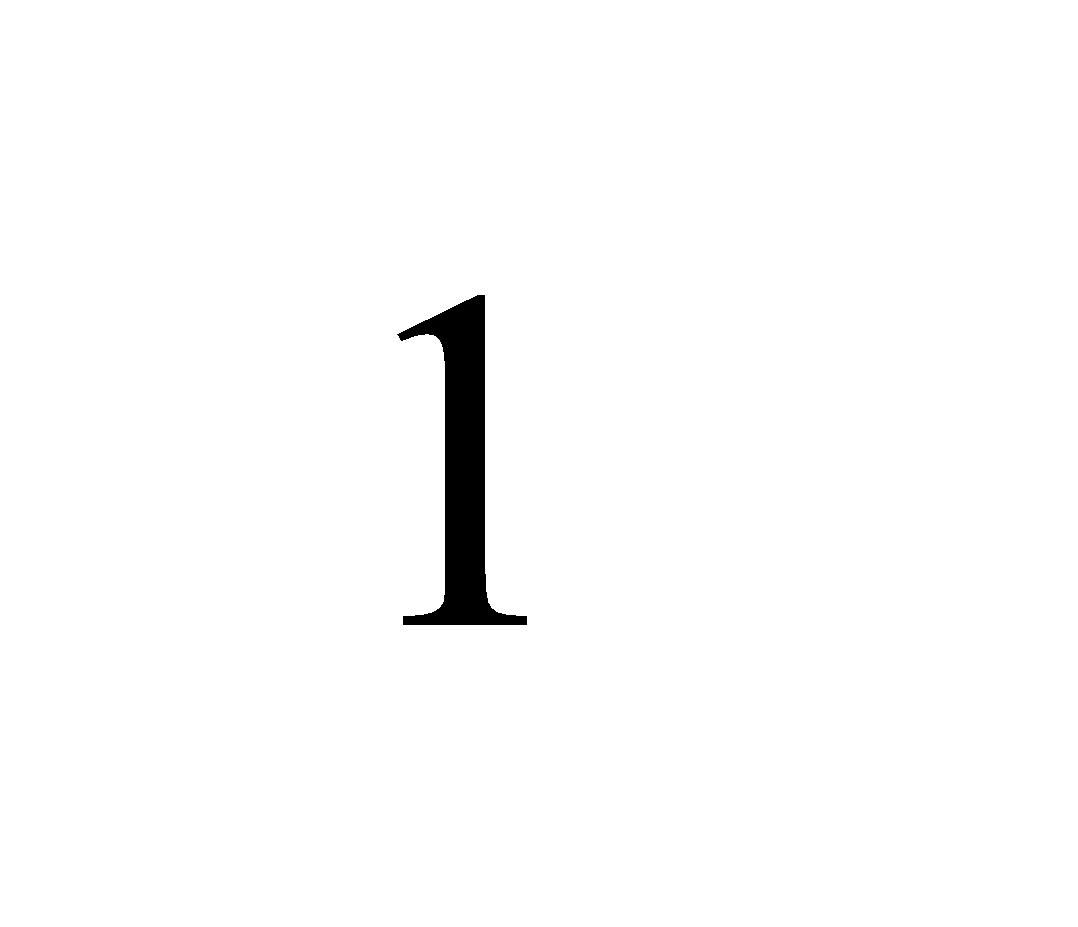
|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | BIGINT | 20 | Primary Key |
| name | VARCHAR | 100 | Not Null |
| gender | VARCHAR | 100 | Not Null |
| dob | DATE |  | Not Null |
| photo | VARCHAR | 300 | Not Null |
| mail | VARCHAR | 100 | Not Null |
| phone | BIGINT | 20 | Not Null |
| place | VARCHAR | 100 | Not Null |
| current\_education\_status | VARCHAR | 100 | Not Null |
| LOGIN\_id | BIGINT | 20 | Foreign Key |

**Table 3.3.1.4: TEST TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | BIGINT | 20 | Primary Key |
| STAFF\_id | BIGINT | 20 | Foreign Key |
| testname | VARCHAR | 100 | Not Null |
| date | DATE |  | Not Null |
| time | TIME | 6 | Not Null |



**Table 3.3.1.8: REVIEWS TABLE**



|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | BIGINT | 20 | Primary Key |
| date | DATE |  | Not Null |
| type | VARCHAR | 300 | Not Null |
| review | VARCHAR | 300 | Not Null |
| rating | VARCHAR | 5 | Not Null |
| LOGIN\_id | BIGINT | 20 | Foreign Key |

## Table 3.3.1.3: STAFF TABLE

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | BIGINT | 20 | Primary Key |
| name | VARCHAR | 100 | Not Null |
| gender | VARCHAR | 100 | Not Null |
| dob | DATE |  | Not Null |
| photo | VARCHAR | 300 | Not Null |
| mail | VARCHAR | 100 | Not Null |
| phone | BIGINT | 20 | Not Null |
| housename | VARCHAR | 100 | Not Null |
| place | VARCHAR | 100 | Not Null |
| district | VARCHAR | 100 | Not Null |
| pincode | INT | 11 | Not Null |
| state | VARCHAR | 100 | Not Null |
| qualification | VARCHAR | 100 | Not Null |
| proof | VARCHAR | 300 | Not Null |
| status | VARCHAR | 100 | Not Null |
| current\_institution | VARCHAR | 100 | Not Null |
| LOGIN\_id | BIGINT | 20 | Foreign Key |

**Table 3.3.1.7:TEST QUESTIONS TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | BIGINT | 20 | Primary Key |
| TEST\_id | BIGINT | 20 | Foreign Key |
| QUESTION\_id | BIGINT | 20 | Foreign Key |
| option1 | VARCHAR | 100 | Not Null |
| option2 | VARCHAR | 100 | Not Null |
| option3 | VARCHAR | 100 | Not Null |
| option4 | VARCHAR | 100 | Not Null |
| correct\_answer | VARCHAR | 100 | Not Null |

## Table 3.3.1.5: QUESTION TABLE

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | BIGINT | 20 | Primary Key |
| question | VARCHAR | 3000 | Not Null |
| answer | VARCHAR | 3000 | Not Null |
| LOGIN\_id | BIGINT | 20 | Foreign Key |

## 

## 

**Table 3.3.1.6: RESULT TABLE**

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | BIGINT | 20 | Primary Key |
| TEST\_id | BIGINT | 20 | Foreign Key |
| USER\_id | BIGINT | 20 | Foreign Key |
| mark | BIGINT | 20 | Not Null |

## 

## Table 3.3.1.9: COMPLAINT TABLE

|  |  |  |  |
| --- | --- | --- | --- |
| **COLUMN NAME** | **DATATYPE** | **LENGTH** | **CONSTRAINT** |
| id | BIGINT | 20 | Primary Key |
| LOGIN\_id | BIGINT | 20 | Foreign Key |
| date | DATE |  | Not Null |
| complaint | VARCHAR | 300 | Not Null |

## NORMALIZATION

Database Normalization is a methodical technique for structuring data within a database. It involves decomposing tables to eliminate data redundancy and to address undesirable anomalies such as Insertion, Update, and Deletion Anomalies. This multi-step process transforms data into tabular form, removing duplicated data from relational tables. In our design, tables have been normalized up to the third normalization form, which includes :

* FirstNormalForm(1NF)
* SecondNormal Form(2NF)
* ThirdNormalForm(3NF)

## FirstNormalForm

Our application complies with the requirements of the first normal form (1NF), which stipulates that a relation should not contain composite or multi-valued attributes. In other words, for a relation to be in first normal form, every attribute within it must be single-valued. The first normal form states that:

* Every column in the table must be unique
* Separate tables must be created for each set of related data.
* Each table must be identified with a unique column or concatenated columns known as the primary key.
* No rows maybe duplicated
* No columns may be duplicated
* No row/column intersections contain a null value

## Second Normal Form

A relation achieves second normal form (2NF) only when it meets the conditions of first normal form (1NF) for the primary key and when every non-primary key attribute is fully dependent on the primary key alone. If a non-key attribute does not rely on the primary key, it should be extracted from the relation and placed in a separate one. In other words, all fields in a table that is in 2NF are directly related to the primary key.

**Third Normal Form**

A relation achieves third normal form (3NF) when it satisfies the conditions of being in second normal form (2NF), and additionally, non-key attributes do not depend on other non-key attributes. Storing and retrieving data from the database are essential components of system design. At the analysis stage, data elements and structures to be stored are identified, structured, and integrated to create the data storage and retrieval system..

## INPUT DESIGN

Input design serves as the bridge between the information system and the user, encompassing the development of specifications and procedures for data preparation. These steps are essential for transforming transaction data into a usable format for processing, which can involve inspecting the computer, reading data from written or printed documents, or inputting data directly into the system. The primary focus of input design is to regulate the input quantity, minimize errors, prevent delays, streamline processes, and maintain simplicity. Additionally, input design aims to ensure security, user-friendliness, and privacy retention.

27

## OUTPUT DESIGN

Quality output refers to meeting the end user's requirements and presenting information clearly. In any system, the outcomes of processing are communicated to users and other systems through outputs. Output design entails determining how information is displayed for immediate use and also generating hardcopy output. It serves as the primary and direct source of information for users. Effective and intelligent output design enhances the system's ability to assist user decision-making..

28

# CHAPTER-IV

**CODING**

**4.1 FRONTEND:**

**HTML**

Hypertext Markup Language (HTML) serves as the universal markup language for crafting web pages. Its purpose is to outline the framework of a webpage, employing various elements to instruct browsers on how to render content. HTML is widely embraced by developers for website creation due to its straightforward nature, making it accessible and uncomplicated to grasp..

**CSS**

Cascading Style Sheets (CSS) is a language utilized for defining the visual appearance of a document authored in a markup language like HTML or XML (including XML variations such as SVG, MathML, or XHTML). As a foundational component of the World Wide Web, CSS, alongside HTML and JavaScript, plays a crucial role.

CSS is engineered to facilitate the segregation of content and presentation, encompassing aspects like layout, colors, and typography. This division enhances content accessibility, affords greater flexibility and precision in delineating presentation attributes, allows for the reuse of formatting across multiple web pages by specifying CSS in a separate .css file, thus minimizing redundancy and complexity in structural content. Furthermore, caching the .css file enhances page loading speed between pages that share the same formatting, optimizing overall user experience.

**JAVASCRIPT**

JavaScript, often abbreviated as JS, is a high-level, versatile programming language primarily used for enhancing interactivity and functionality in web pages. As a cornerstone technology of the web alongside HTML and CSS, JavaScript enables dynamic updates, interactive features, and complex behavior within web applications. It runs on the client side, typically within a web browser, but can also be executed on the server side through platforms like Node.js. With its broad adoption and extensive ecosystem of libraries and frameworks, JavaScript has become indispensable in modern web development, empowering developers to create dynamic, responsive, and interactive web experiences..

## 4.2 BACKEND:

**PYTHON**

Python is a versatile programming language that supports multiple paradigms, including object-oriented programming, structured programming, functional programming, and aspect-oriented programming. It features dynamic typing and utilizes reference counting and a cycle-detecting garbage collector for memory management. Python also offers dynamic name resolution, binding method and variable names during program execution. Known for its ease of learning and platform independence, Python boasts extensive libraries and supports both procedural-oriented programming (POP) and object-oriented programming (OOP). Hence, we have chosen Python as the backend language for our web development.

.

**PYCHARM IDE**

PyCharm is a popular integrated development environment (IDE) specifically designed for Python programming. Developed by JetBrains, it offers a wide range of features to enhance productivity and streamline the development process. PyCharm provides smart code completion, syntax highlighting, and code analysis to help developers write clean and efficient code. It also includes powerful debugging tools, version control integration, and support for web development frameworks like Django and Flask. With its user-friendly interface and extensive plugin ecosystem, PyCharm is a preferred choice for Python developers seeking a comprehensive development environment.

**MySQL**

MySQL is an open-source relational database management system (RDBMS) supported by Oracle and built around Structured Query Language (SQL). It is compatible with various platforms such as Linux, UNIX, and Windows. While MySQL finds application in diverse domains, it is particularly renowned for its association with web applications and online publishing. Offering high performance and memory efficiency, MySQL is known for its speed and user-friendliness. Given its simplicity and efficiency, we have opted for MySQL as the backend database solution.

TECHNOLOGIES USED:

Machine Learning (ML) transformers

Machine Learning (ML) transformers are a groundbreaking architecture in natural language processing (NLP) and machine translation. They employ a mechanism known as self-attention to process input data in parallel, allowing for more efficient learning and improved performance on sequential tasks. Transformers have revolutionized the field of NLP by enabling the training of deep neural networks on large-scale datasets, leading to state-of-the-art results in tasks such as language translation, text generation, and sentiment analysis. They have become a cornerstone in many modern NLP applications and have greatly contributed to advancements in the field.

# CHAPTER-V

**TESTING AND IMPLEMENTATION**

**5.1 TESTING**

Software testing involves the systematic examination of a system to uncover errors, discrepancies, or deviations from the specified requirements. It is broadly divided into two categories: functional testing and non-functional testing. Ideally, testing should commence as early as possible in the development process to minimize rework and ensure the delivery of bug-free software to clients. However, the exact starting point for testing within the Software Development Life Cycle (SDLC) can vary. Testing may begin as early as the Requirements Gathering phase and continue throughout the development process until the software is deployed in production. The timing of testing activities can also be influenced by the chosen development model. For instance, in the Waterfall model, testing typically begins after the development phase, whereas in the V-model, testing is conducted concurrently with development activities.

**5.1.1 TESTING METHODOLOGY MANUAL TESTING**

Manual testing is a software testing approach where test cases are executed manually, without the use of automated tools. Testers follow end-user perspectives to ensure that the application aligns with the requirements outlined in the documentation. Test cases are meticulously planned and executed to achieve near-complete coverage of the software application. Additionally, manual test reports are generated without automation.

This method is fundamental as it uncovers both apparent and latent defects in the software. Defects, defined as discrepancies between expected and actual outputs, are identified by testers and reported for resolution by developers.

Prior to automated testing, manual testing is imperative for newly developed software. Despite requiring substantial time and effort, manual testing guarantees software reliability. Unlike automated testing, manual testing relies solely on manual testing techniques and does not require knowledge of automated testing tools.

Test engineers who conduct manual testing gain insights into the application from an end-user perspective, facilitating the creation of accurate test cases and prompt feedback on the application's performance.

## AUTOMATION TESTING

Automation testing involves executing test cases using specialized automation tools. These tools control the test case execution and compare actual results with expected outcomes. Automation testing demands significant resources and financial investment.

Primarily, automation testing focuses on repetitive actions, such as regression tests. The tools employed in automation testing serve various purposes, including automated GUI interaction, data setup generation, defect logging, and product installation. The objective of automation testing is to decrease manual test cases while retaining their importance. Automation tools can record test suites, allowing testers to replay them as needed without human intervention.

## 5.1.2 DIFFERENT TESTING

* + - * UnitTesting
      * IntegrationTesting
      * SystemTesting
      * AcceptanceTesting
      * ValidationTesting

## UNITTESTING

The initial stage of testing is referred to as unit testing, where individual modules undergo examination against the specifications outlined during design. Unit testing is crucial for verifying module functionality and assessing internal logic. It is performed across various modules within the project, enabling immediate identification and rectification of errors to enhance program clarity. Testing is integrated into the programming stage itself, ensuring each module operates satisfactorily as per expectations. For instance, the Login page is subjected to testing under three distinct states: positive input, negative input, and zero input. While testing with positive and negative inputs yields expected outcomes, zero input may lead to unexpected results. This underscores the importance of thoroughly testing different states of the code.

## INTEGRATION TESTING

The next level of testing, known as integration testing, involves systematically examining the construction structure of the system. Simultaneously, tests are performed to detect errors within the interface. It is not guaranteed that software modules, which exhibit satisfactory results when run individually, will also perform flawlessly when integrated. Therefore, individual modules are retested, and the results are validated to assess their integration with one another. This testing activity essentially evaluates the design and emphasizes the interaction between modules..

* TopDownIntegration

This approach is incremental in nature and focuses on constructing the program structure step by step. Modules are integrated by progressing downwards through the control hierarchy, starting with the main program module. Subordinate modules are then incorporated into the structure, following either a depth-first or breadth-first manner.

* BottomUpIntegration

This approach initiates the construction and testing with the modules positioned at the lowest level in the program structure. By integrating modules from the bottom upwards, the necessary processing for subordinate modules at any given level is consistently accessible, thereby eliminating the need for stubs.

## SYSTEM TESTING

Testing is a process aimed at confirming the development of a correct system, with the objective of uncovering any faults within it. This activity is not limited to the post-development phase but is ideally conducted concurrently with all stages of system development, beginning with requirements specification. The results of testing, once collected and analyzed, offer a qualitative assessment of software quality and reliability. They also serve as a foundation for potential design modifications if deemed necessary. A project is considered incomplete without thorough testing.

36

System testing involves verifying if the developed system aligns with the initial objectives and requirements. It entails experimental testing with test data to validate that the system operates as per the specified requirements. Once the system is confirmed to be functional, it is then tested with actual data to assess its performance.

## ACCEPTANCE TESTING

Acceptance testing is a method used to assess whether the software system fulfills the specified requirements. Its primary objective is to evaluate the system's alignment with business requirements and confirm if it meets the necessary criteria for delivery to end users.

## VALIDATION TESTING

Validation testing is conducted for each input form to ensure that only permissible values are entered. This process involves testing with incorrect values to validate whether errors are properly detected and handled. Incorrect values are expected to be rejected, and any identified errors are addressed accordingly. Validation verifies if the system aligns with requirements, fulfills intended functions, and meets the organization's objectives and user needs. This validation occurs throughout various testing phases, including feature testing, integration testing, system testing, load testing, compatibility testing, stress testing, and others.

## Front-EndValidation

All validation processes are implemented on the server to prevent unauthorized access or manipulation of the database through alternative clients. The frontend validation is also essential to enhance efficiency and prevent inappropriate data access on the server.

Frontend validation primarily serves to aid data entry and provide contextual messages. This ensures a smooth data entry experience for users and reduces the need for repeated validation checks. Front-end validation encompasses all input verification in modern applications, offering users prompt feedback on potential issues.

In the user interface (UI), basic input validation such as mandatory field checks and email address validity verification is conducted. Additionally, UI controls are updated or disabled based on the validation outcomes..

## Back-EndValidation

Backend validation is equally crucial, as it must guarantee the validity of incoming data. Moreover, depending on the architecture, the middle-tier business logic is often reused across various components. Therefore, it is essential to ensure that the rules applied remain consistent regardless of the frontend logic enforcement.

38

## 5.2 QUALITY ASSURANCE POLICIES

## 5.2.1 GENERIC RISKS

“Risk is future uncertain events with a probability of occurrence and potential for loss” Risk Identification and management are the main concerns in every software project. Effective Analysis of software risks will help to effective planning and assignments of work. Risks are identified,classified and managed before the actual execution of the program.

These Risks are classified into different categories:

1. **Schedule Risk:** Project schedule slips when project tasks and schedule release risks are not addressed properly.Schedulerisksmainlyaffectaprojectandfinallyoncompanyeconomyandmay lead to project failure.

## Schedules often slip due to the following reasons:

* + Wrong Time Estimation
  + Resources are not tracked properly.All resources like staff,systems,skills individuals,etc.
  + Failuretoidentifycomplexfunctionalitiesandtimerequiredtodevelopthosefunctionalities.
  + Unexpected Project Scope Expansions.

## BudgetRisk

* + Wrong Budget Estimation.
  + Cost Overruns
  + Project Scope Expansion

1. **OperationalRisks:** Risksoflossduetoimproperprocessimplementationfailedsystemorsomeexternaleventsrisks. Causes Of OperationalRisks:
   * Failure To Address Priority Conflicts
   * Failure To Resolve The Responsibilities
   * Insufficient Resources
   * No Proper Subject Training
   * No Resource Planning

42

* + No Communication In The Team.

**Technical Risks :c**Technical risks generally lead to failure of functionality and performance. Causes of Technical Risks are:

* + Continuous Changing Requirements
  + Noadvancedtechnologyavailableortheexistingtechnologyisintheinitialstages.
  + The Product Is Complex To Implement.
  + Difficult Project Modules Integration.

1. **Programmatic Risks:** These are the External Risks beyond operational limits. These are all uncertain risks are outside the control of the program. These External Events Can Be:
   * Running Out Of The Fund.
   * Market Development
   * Changing Customer Product Strategy And Priority
   * Government Rule Changes.

## 5.3 SYSTEMIMPLEMENTATION

System implementation is the final phase i.e., putting the utility into action. Implementation is the state in the project where theoretical design turned into working system. Implementation involves the conversion of a basic application to complete replacement with a computer system. It is the process of converting to a new or revised system design into an operation alone.During the design phase,the products structure,its undergoing datastructures,the general algorithms and the interfaces and control/data linkages needed to support communcation among the various sub structures were established. Implementation process issimply a translation of the design abstraction into the physical realization, using the language of the target architecture.

There Are Three Types Of Implementation:

* Implementation of a computer system to replace a manual system.
* Implementation of a new computer system to replace an existing one.
* Implementation of a modified application to replace an existing computer.

43

The Common Approaches For Implementation Are:

## ParallelConversion

In parallel conversion the existing system and new system operate simultaneously until the project team is confident that the new system is working properly. The outputs from the old system continue to be distributed until the new system has proved satisfactorily parallel conversion is a costly method because of the amount duplication involved.

## DirectConversion

Under the direct conversion method the old system is discontinued altogether and the new system becomes operational immediately. A greater risk is associated with direct conversion is no backup in thein the case of system fails.

## PilotConversion

A pilot conversion would involve the changing over of the part of the system either in parallel or directly. Use of the variation of the two main methods is possible when part of the systemcanbetreatedasa separate entity.

## UserTraining

After the system is implemented successfully, training of the user is one of the most important subtasks of the developer. For this purpose, user manuals are prepared and handled over to the user to operate the developed system. Thus, the users are trained to operate the developed system.

Inorder to put new application system into use,the following activities were taken care of:

* Preparation Of User And System Documentation.
* Conducting User Training With Demo And Handson.
* Test run for some period to ensure smooth switching over the system.

44

## 5.3.1 IMPLEMENTATION PROCEDURES

The Major Implementation Procedures Are:

* Test Plans
* Training
* Conversion

## 5.4 SYSTEM MAINTENANCE

The maintenance is an important activity in the life cycle of a software product.Maintenance includes all the activities after the installation of software that is performed to keep the system operational. The maintenance phase of a software life cycle is the time period in which a product performs useful work. Maintenance is classified into four types.

* Corrective Maintenance
* Adaptive Maintenance
* Perfective Maintenance
* Preventive Maintenance

## CORRECTIVE MAINTENANCE

Correctivemaintenancereferstochangesmadetorepairdefectsinthedesign,coding,or implementation of the system. Corrective maintenance is often needed for repairing processing or performance failures or making changes because of previously uncorrected problems or false assumptions.Most corrective maintenance problems surface soon after the installation. When corrective maintenance problems surface, theyare typically urgent and need to be resolved to curtail possible interruptions in normalbusiness activities.

45

## ADAPTIVE MAINTENANCE

Adaptive maintenance involves making changes to an information system to evolve its functionality or to migrate it to a different operating environment. Adaptive maintenance is usually less urgent than corrective maintenance because business and technical changes typically occur some period of time.

## PERFECTIVE MAINTENANCE

Perfective maintenance involves making enhancements to improve processing performance, interface usability, or to add desired, but not necessarily required, system features. Many system professionals feel that perfective maintenance is not really the maintenance but new development.

## PREVENTIVE MAINTENANCE

Preventive maintenance is the only maintenance activity which is carried out without formal maintenance request from the user. When a software company or maintenance agency realizes that the methodologies used in a program have become obsolete, it may decide to develop or modify parts of the program, which do not conform to the current trends. Of these types, more time and money are spent on perfecting than on corrective and adaptive maintenance together .

46

# CHAPTER-VI CONCLUSION

47

## 6.1 CONCLUSION

Legal Advisor is a user friendly app that helps users to find and book the advocates according

to their requirements. Legal Advisor includes the information about the

advocates. This app provides a platform for the users to communicate with the advocates via

online.

## 6.2 SCOPE FOR FURTHER ENHANCEMENTS

Nowadays the role of Legal Consultants is very important because of the popularity of

increasing crimes. So the aim of Legal Advisor is to help the users to find the advocates

according to their requirements. Thus Legal Advisor is useful and very relevant app at

present and future.

## 6.3 BIBLIOGRAPHY

### Textbooks

* Elmasri R.& Navathe S. (2007). Fundamentals of database systems. Boston:Pearson/AddisonWesley.
* Sommerville.(2016).Software Engineering.Harlow:PearsonEducation.
* DarwinI.F.(2017).AndroidCookbook:ProblemsandSolutionsforAndroidDevelopers.Beijing:OReilly Media.

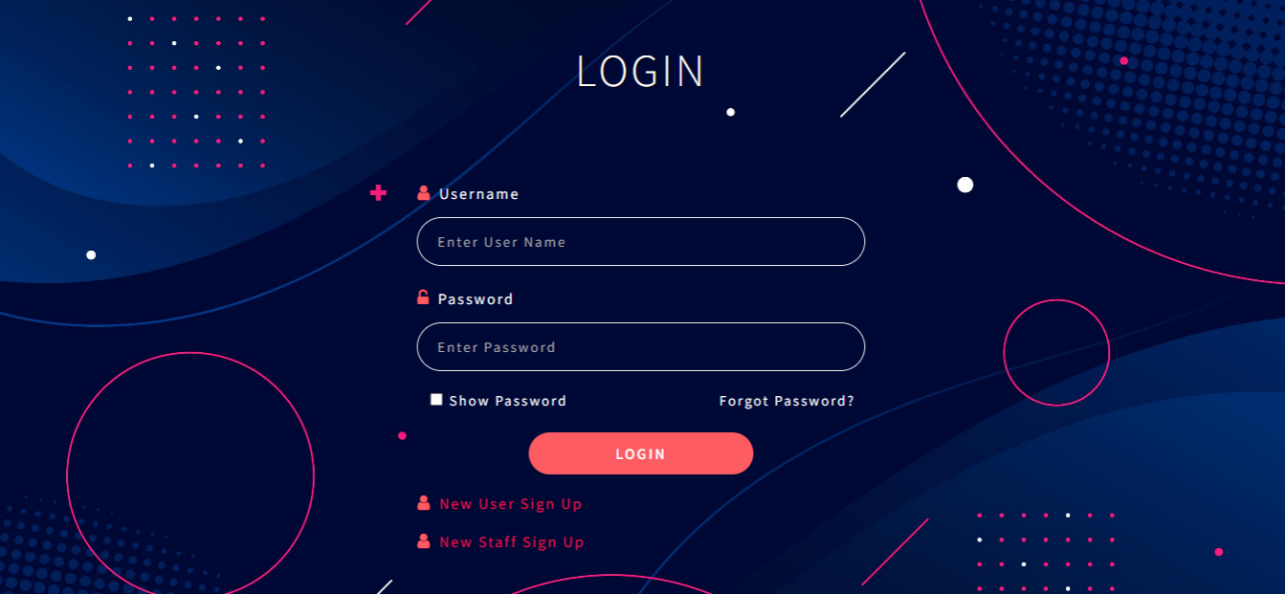
### WebSites

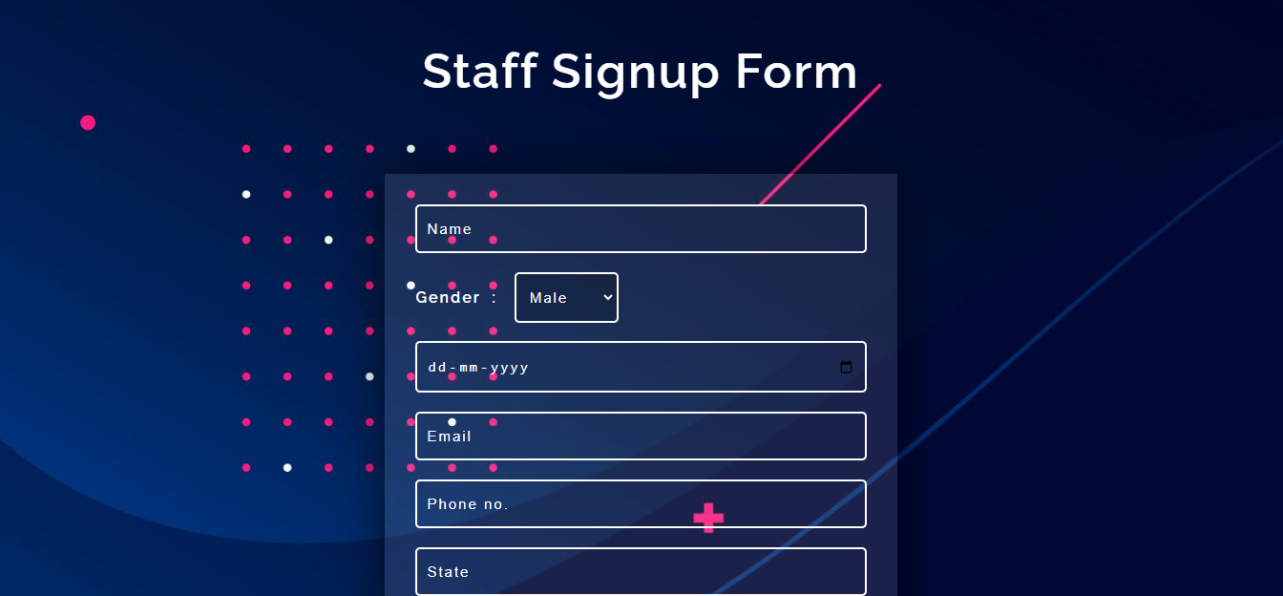
* [www.w3schools.com](http://www.w3schools.com/)
* www.tutorialspoint.com

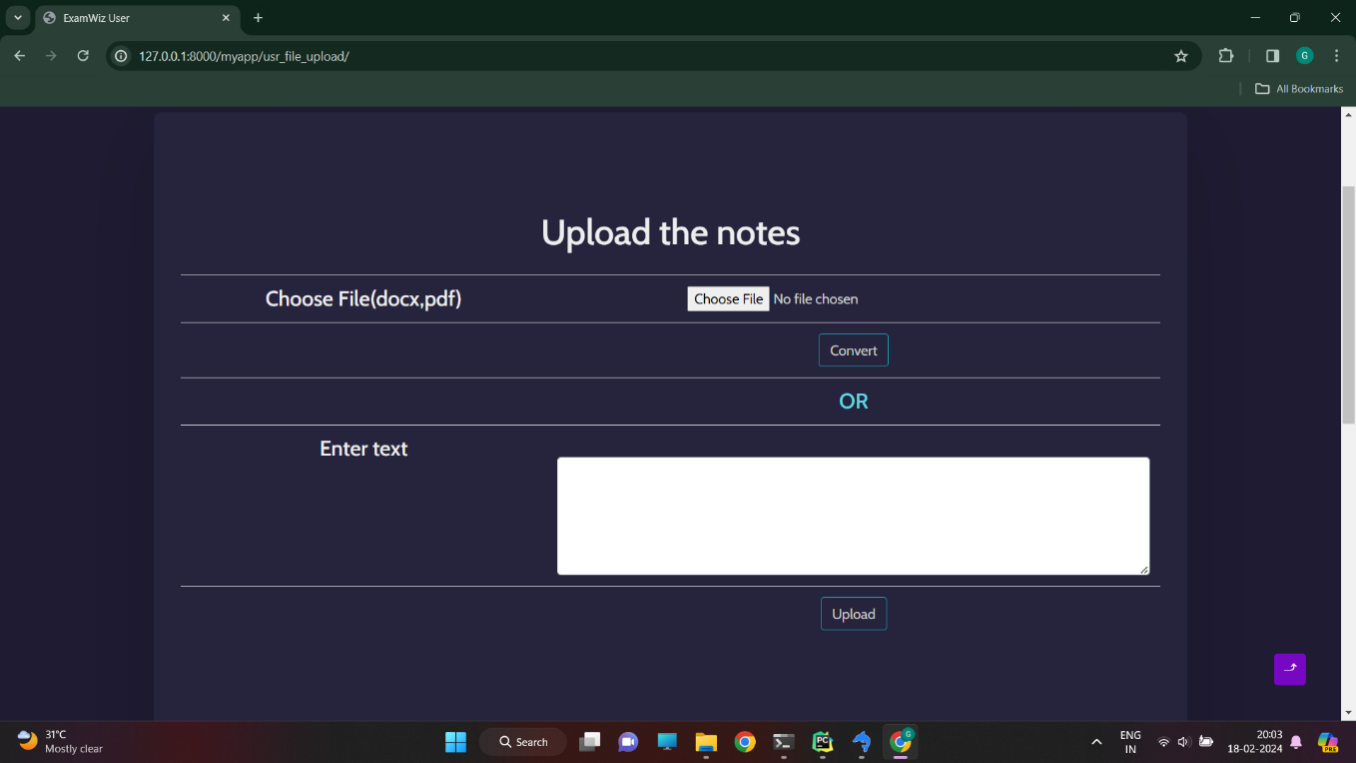
# 

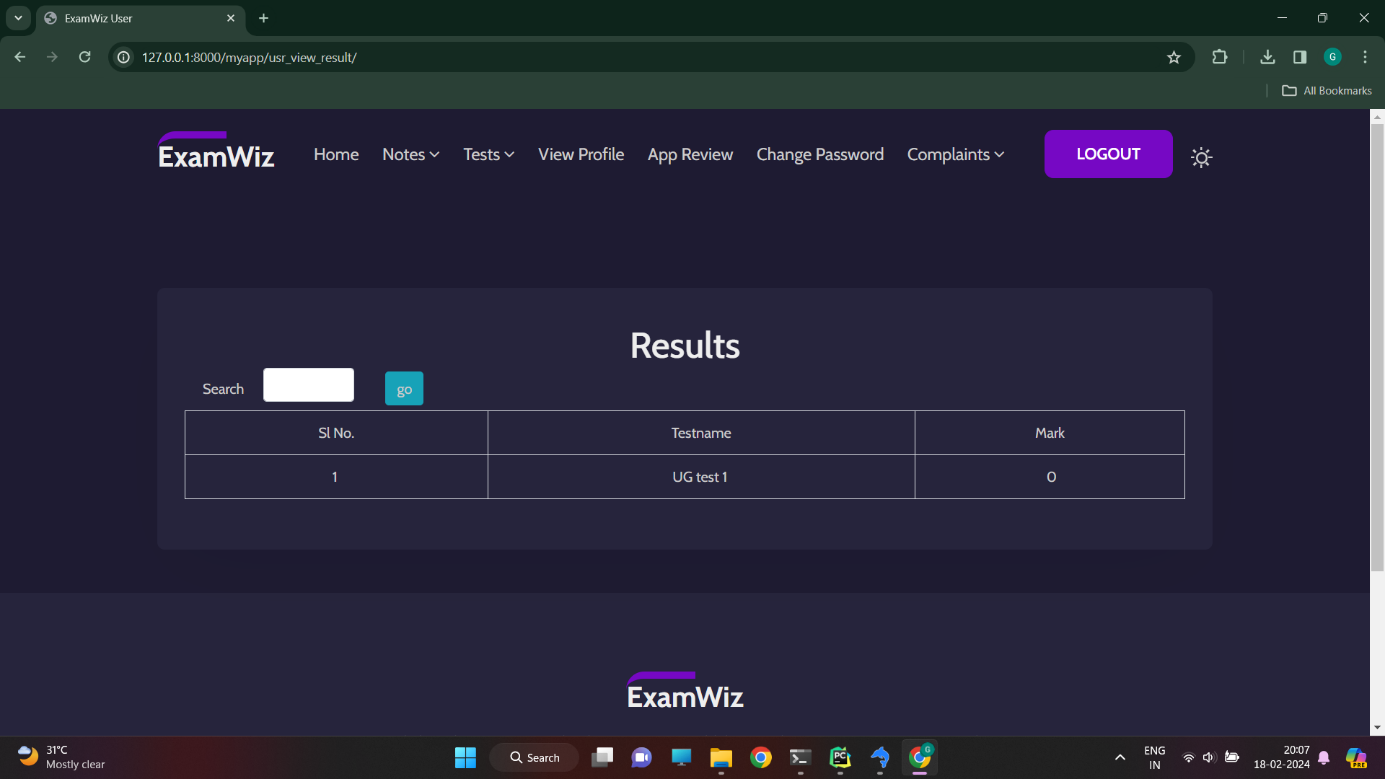
# ANNEXURE-A

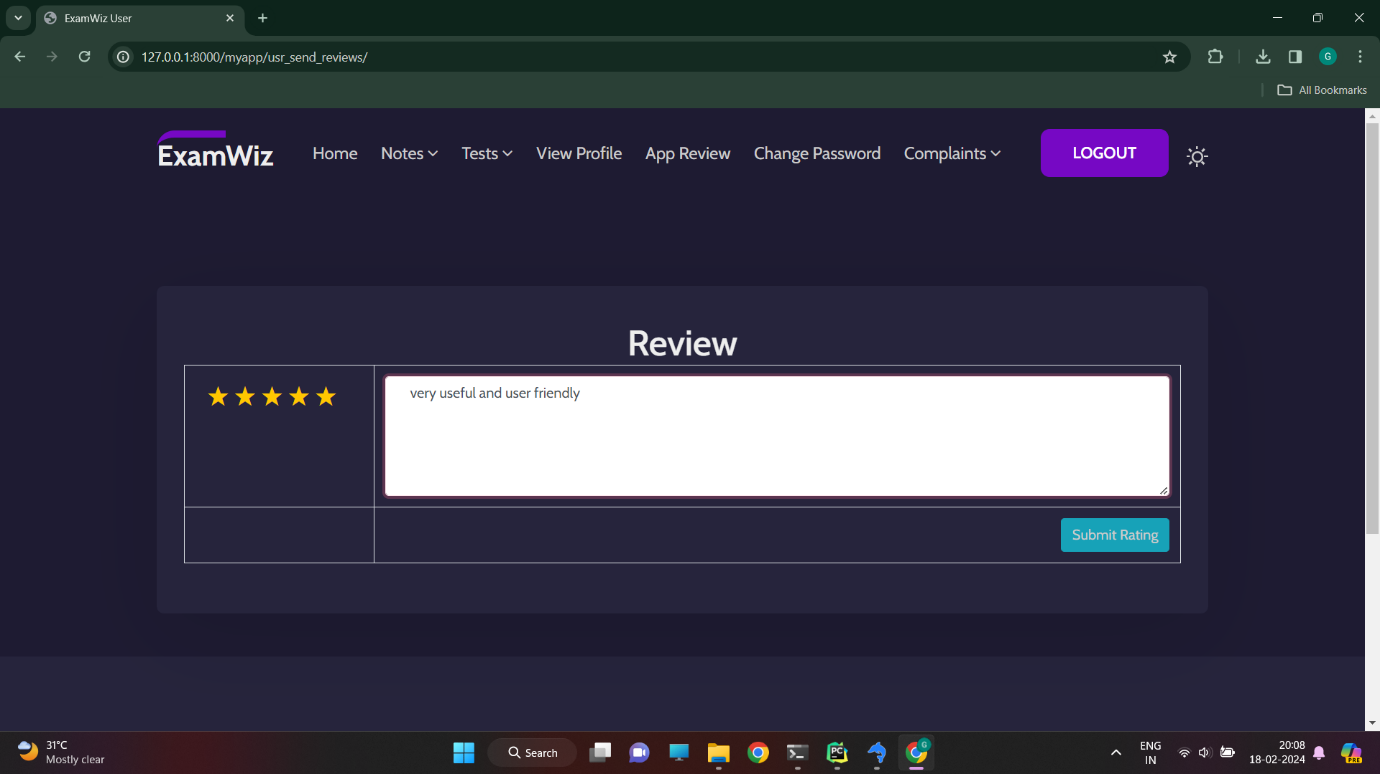
**1.INPUT AND OUTPUT DESIGN**

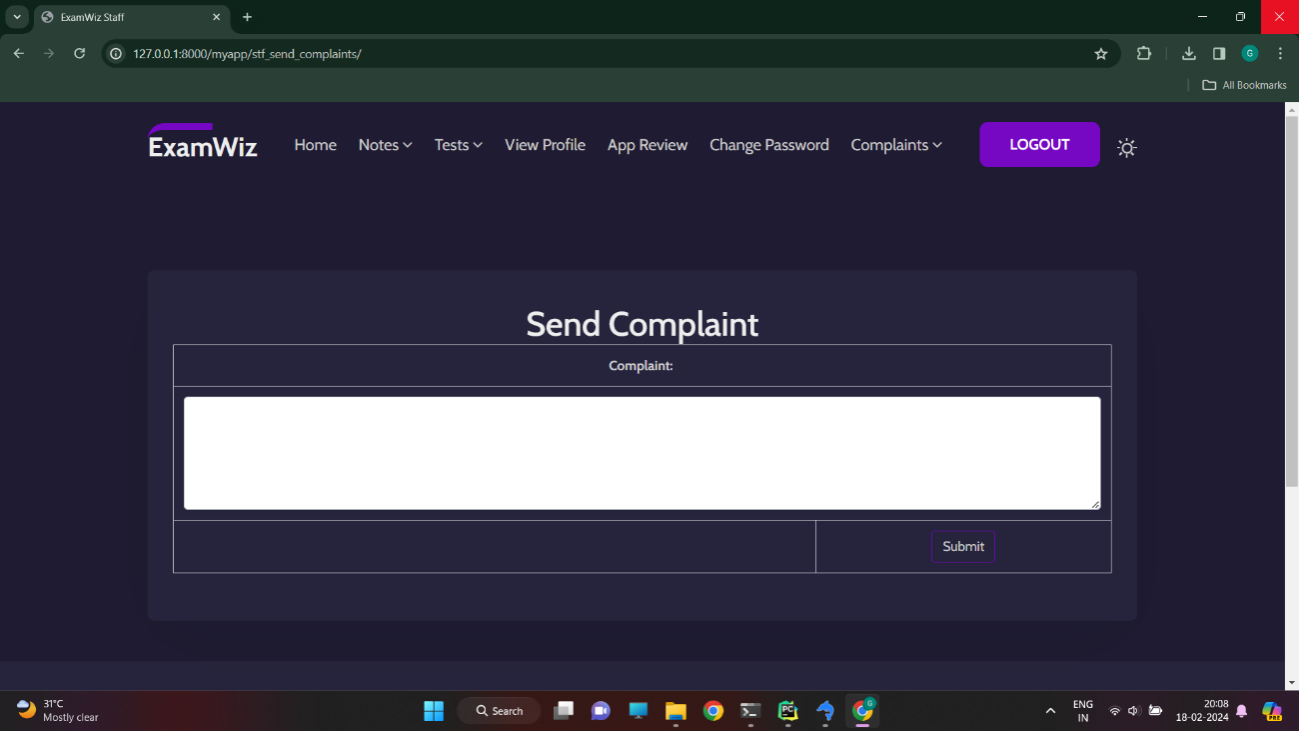
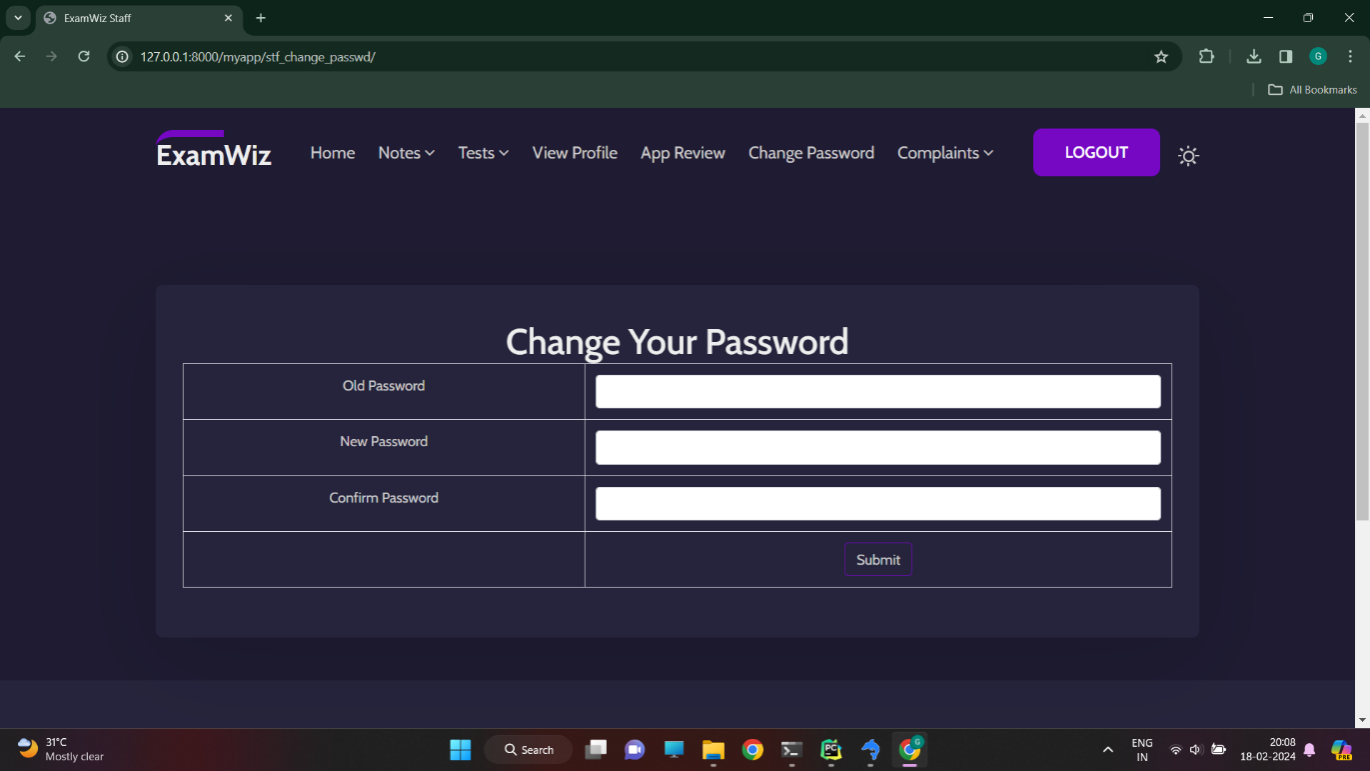
****

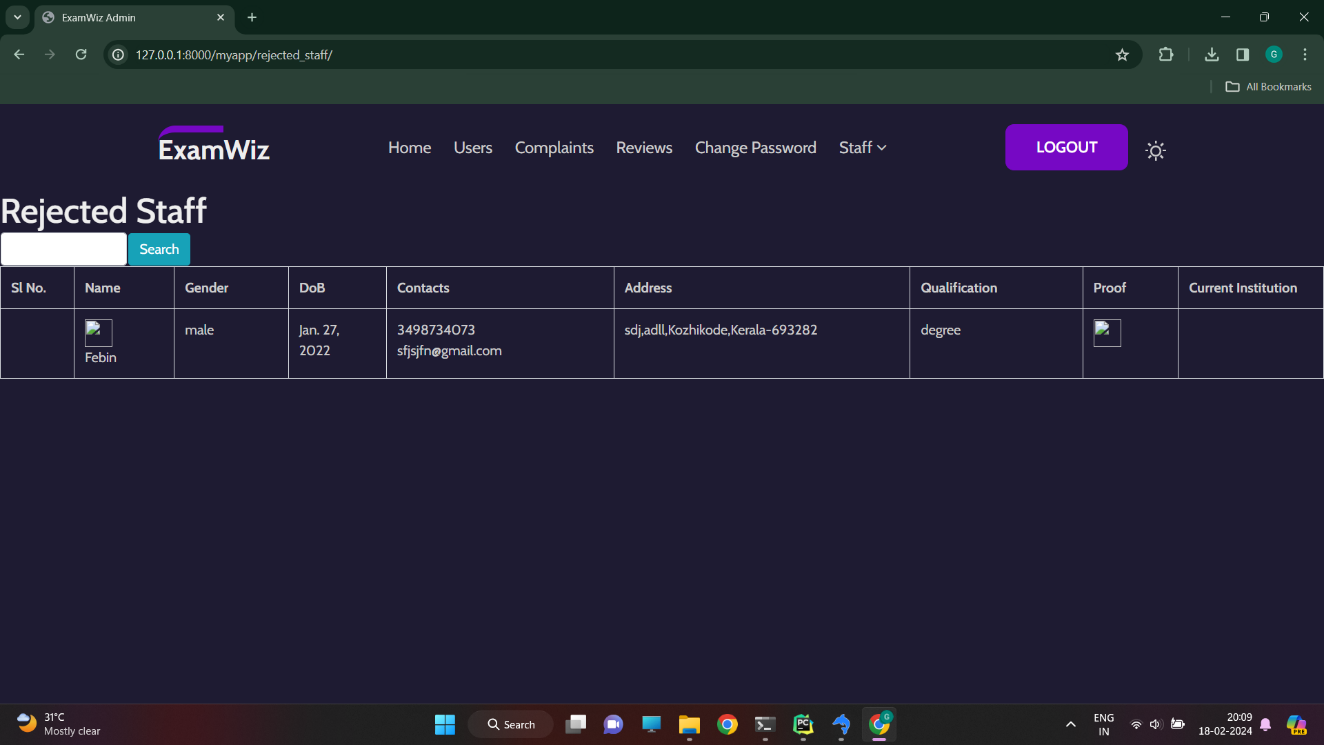
****

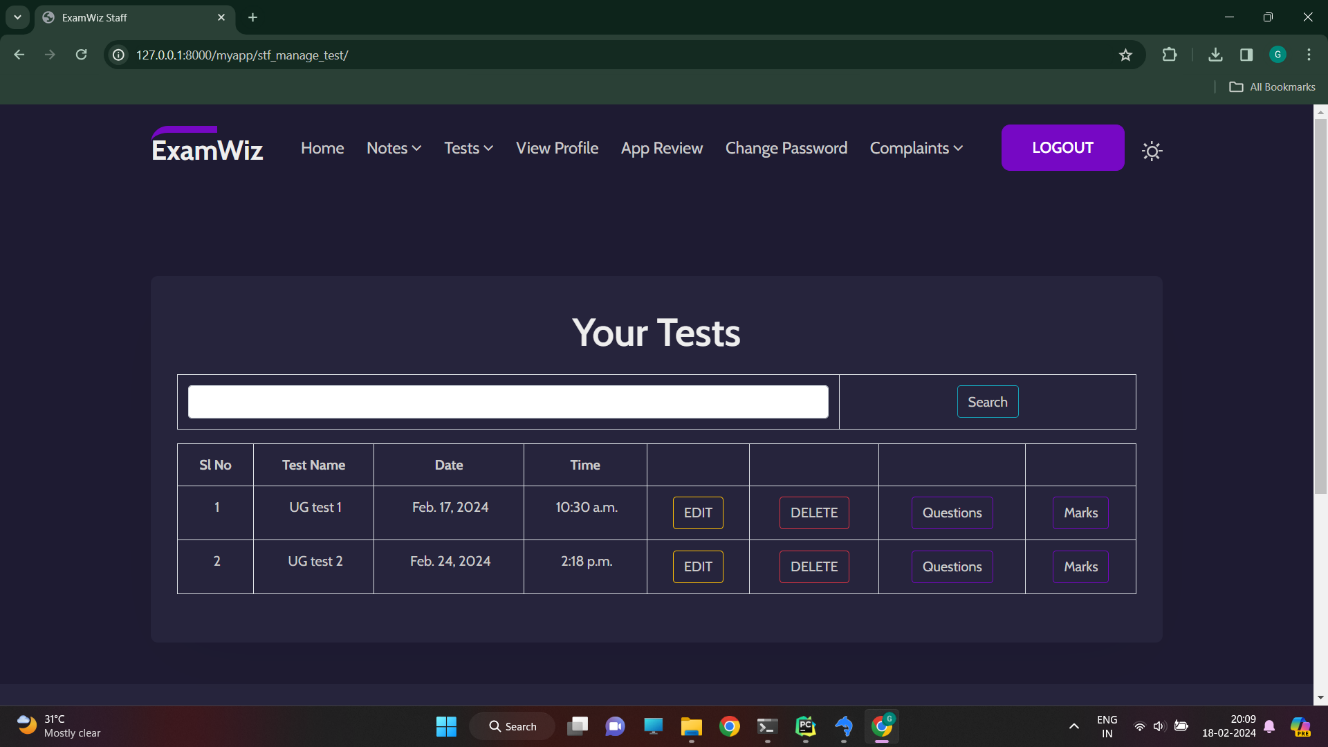
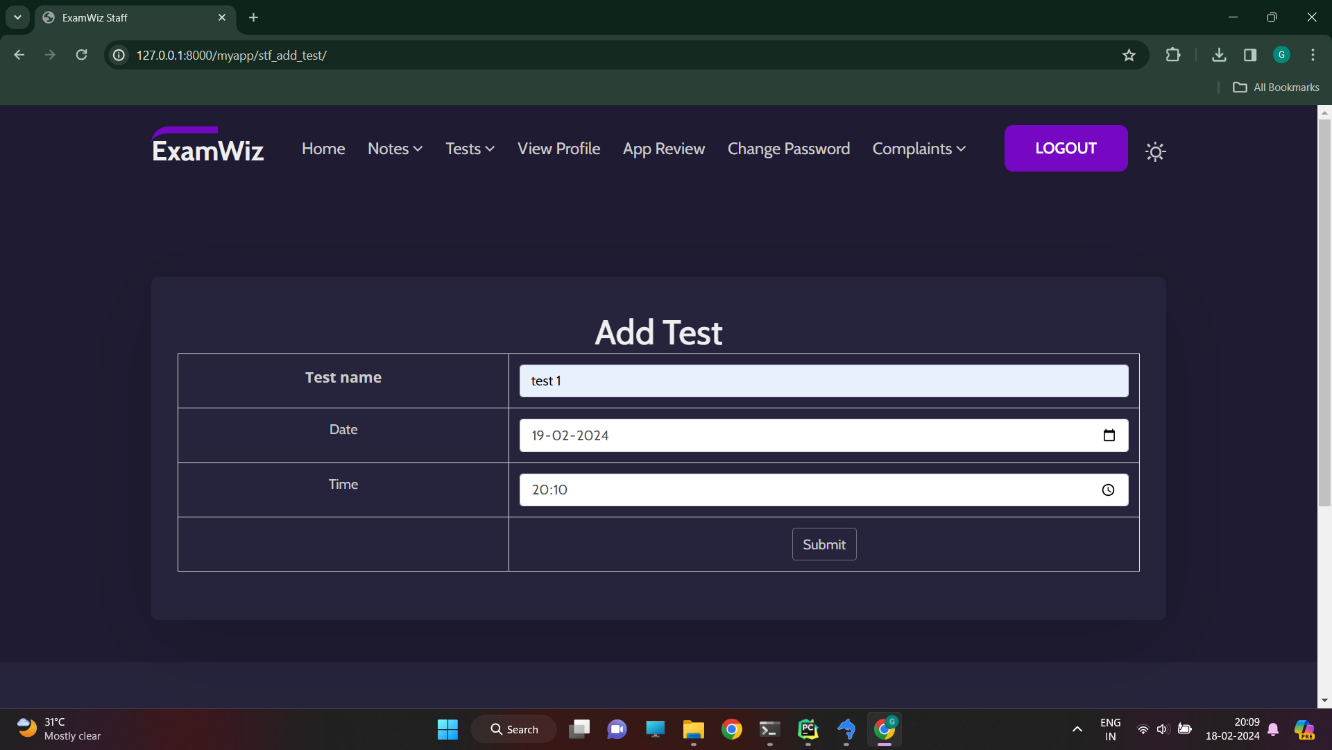
****

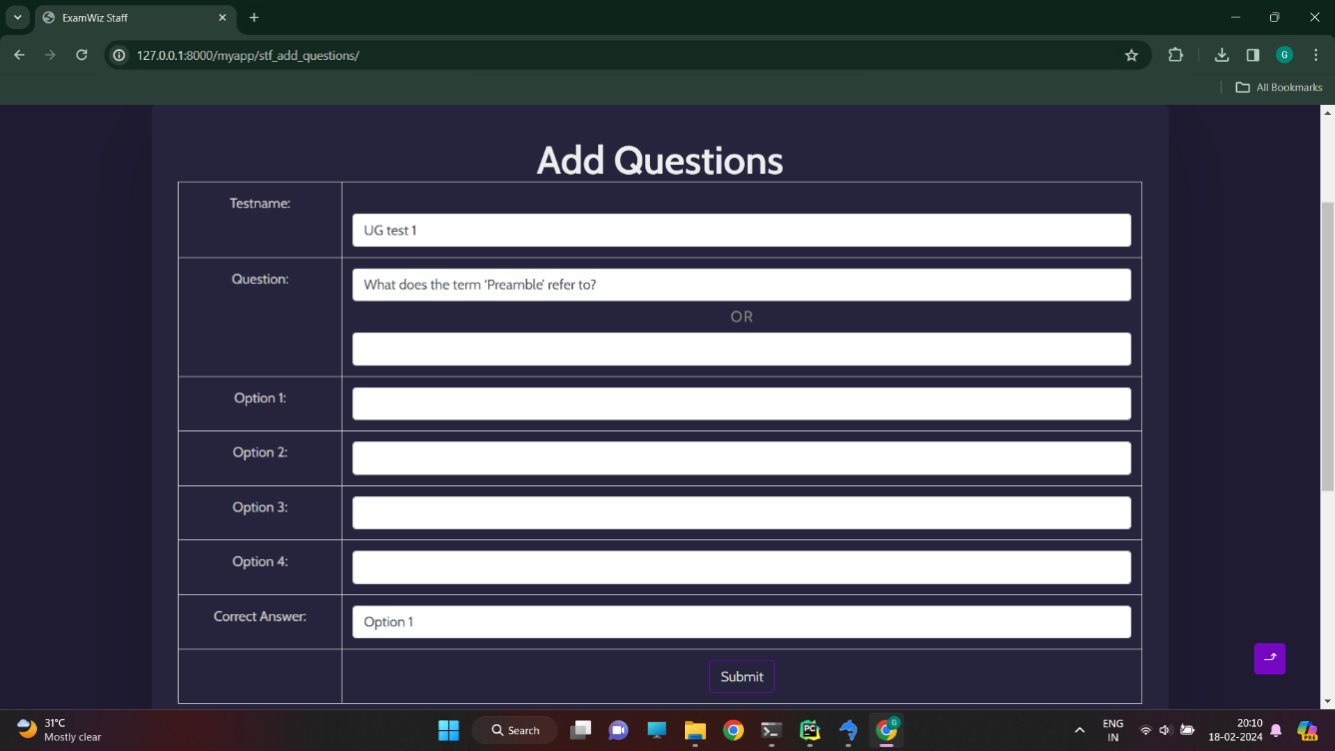


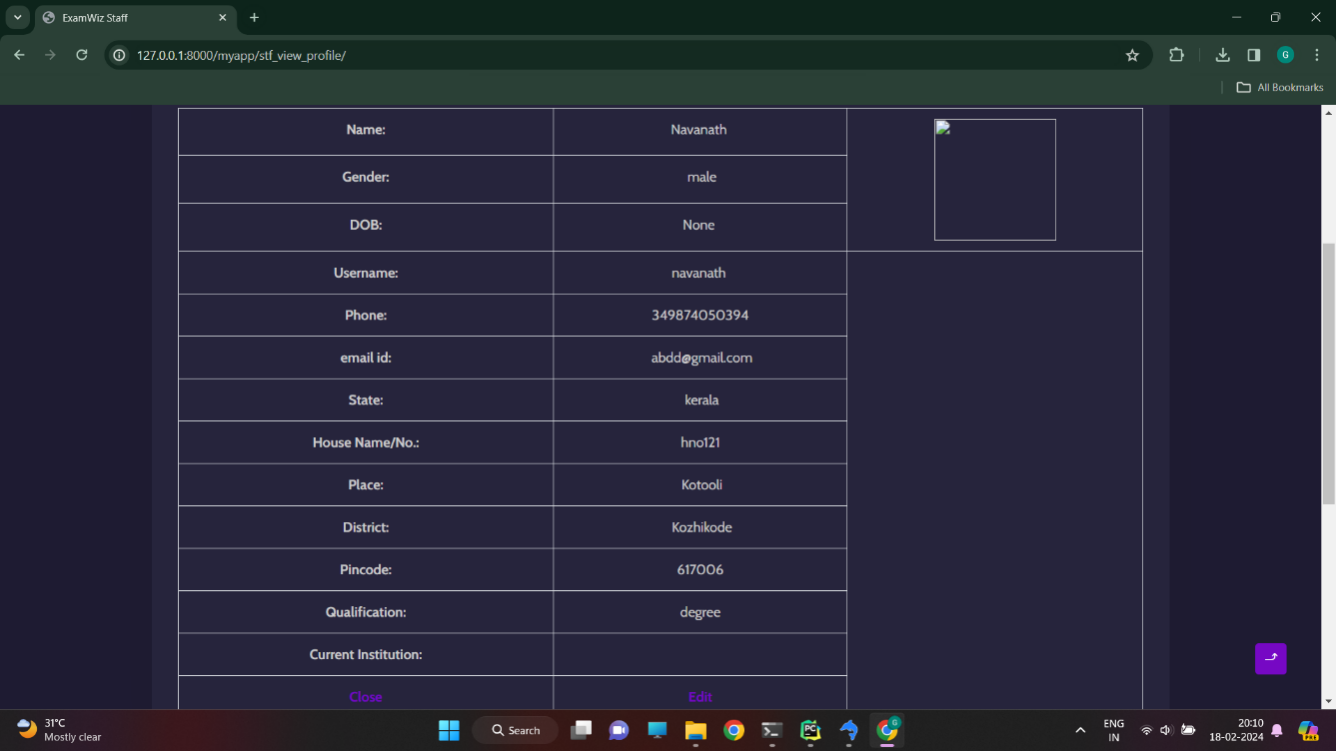


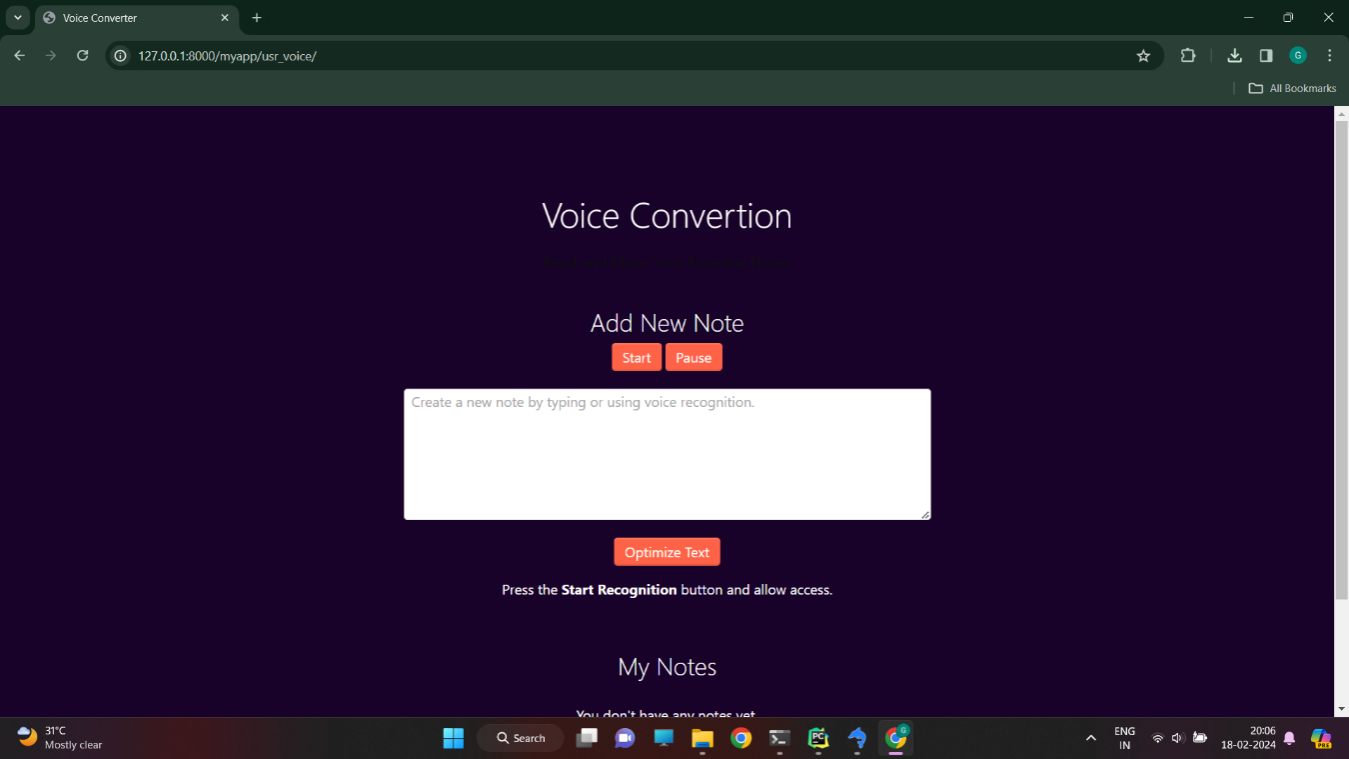


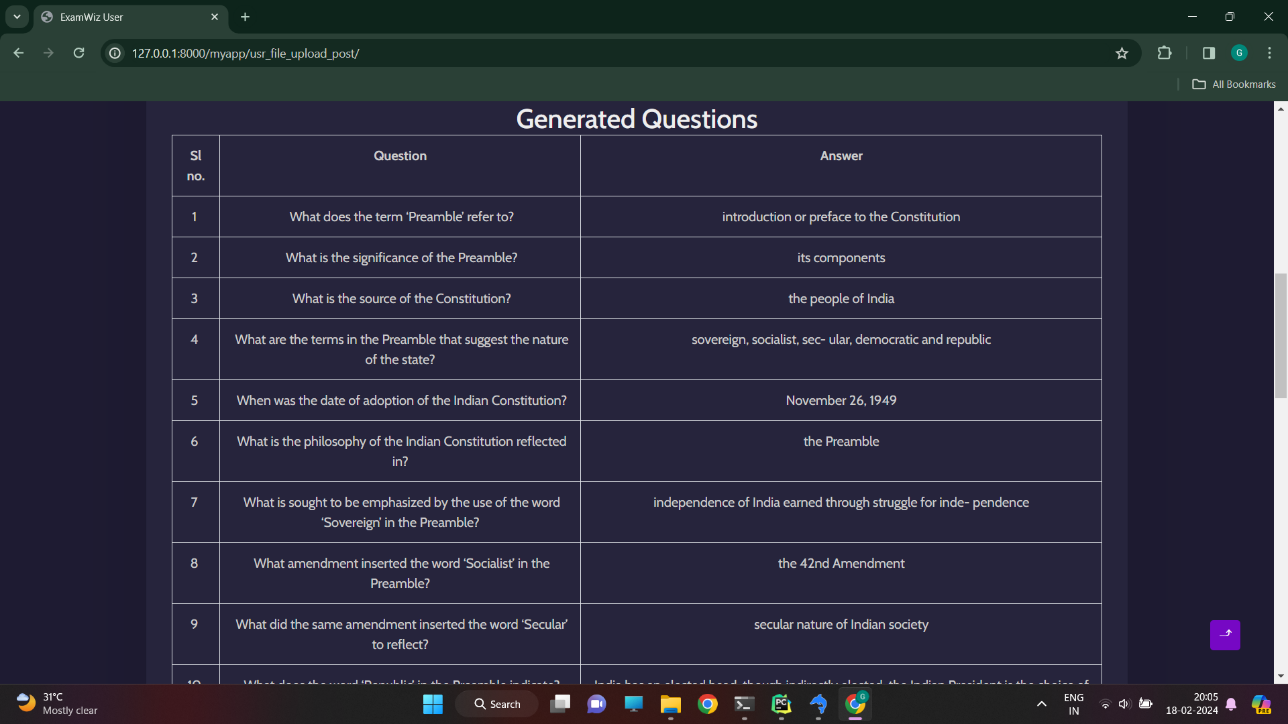


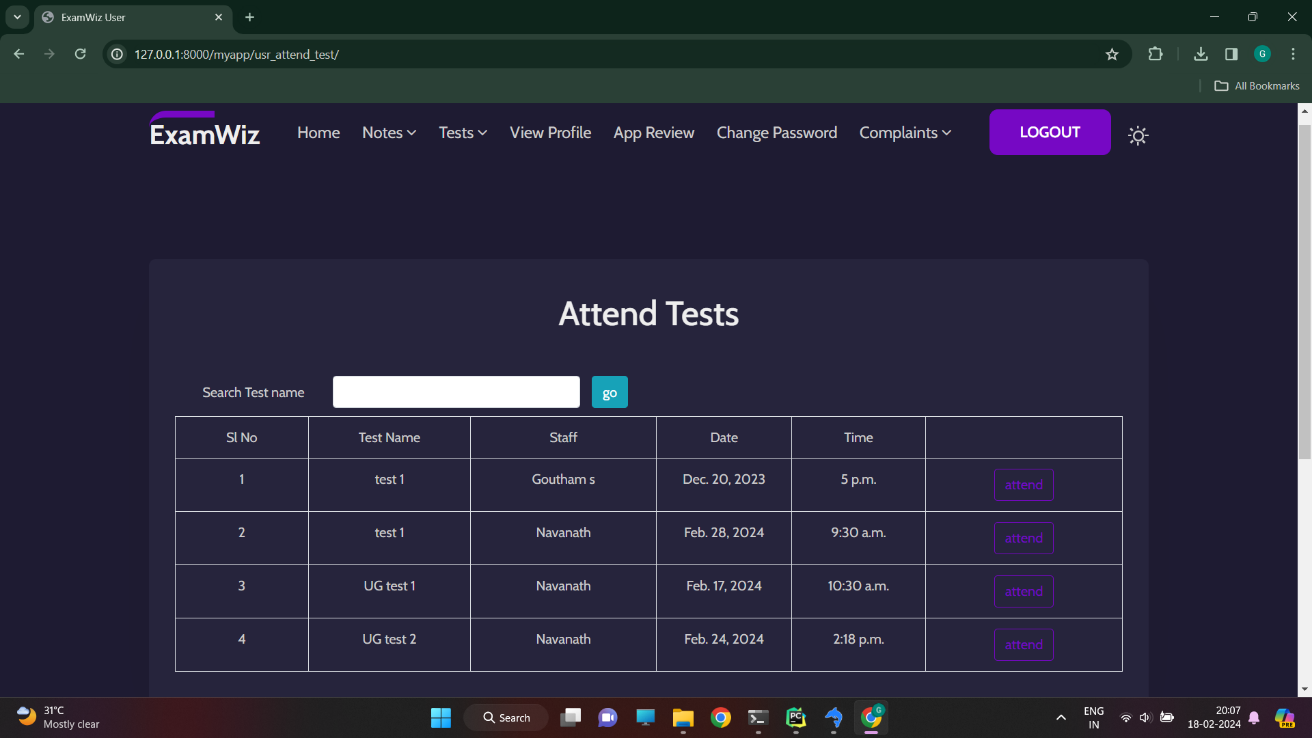
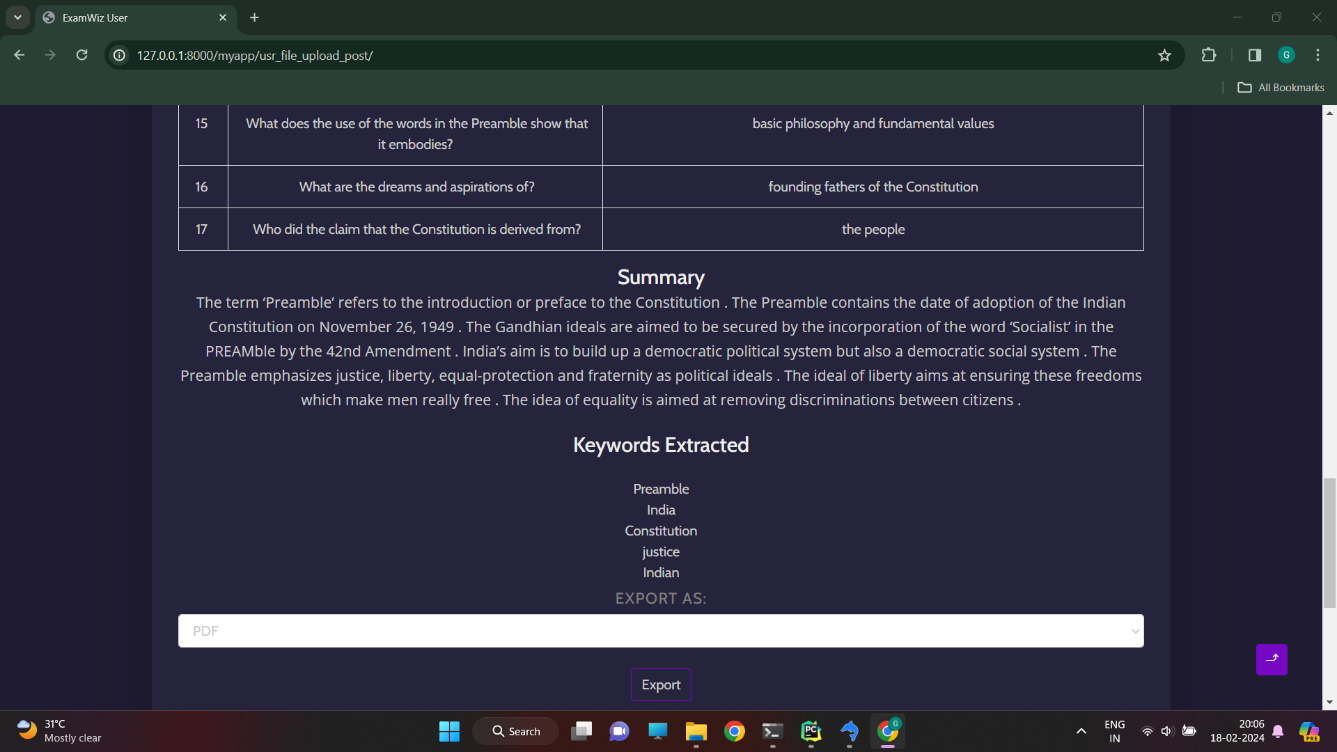




\







**2. SAMPLE SOURCE**

from django.core.files.storage import FileSystemStorage

from django.http import HttpResponse

from django.shortcuts import render

from yake import yake

from .models import \*

# Create your views here.

def login(request):

return render(request,'login\_index.html')

def login\_post(request):

username=request.POST['textfield']

password=request.POST['textfield2']

var=Login.objects.filter(username=username,password=password)

if var.exists():

var = Login.objects.get(username=username, password=password)

request.session["lid"]=var.id

if var.type == "admin":

return HttpResponse('''<script>alert("Logged in Successfully!");window.location="/myapp/adm\_home/"</script>''')

elif var.type == "staff":

return HttpResponse('''<script>alert("Logged in Successfully!");window.location="/myapp/stf\_home/"</script>''')

elif var.type == "user":

return HttpResponse('''<script>alert("Logged in Successfully!");window.location="/myapp/usr\_home/"</script>''')

else :

return HttpResponse('''<script>alert("Log in Unsuccessful!");window.location="/myapp/login/"</script>''')

else :

return HttpResponse('''<script>alert("USER Does not exist");window.location="/myapp/login/"</script>''')

def adm\_logout(request):

request.session["lid"]=""

return HttpResponse('''<script>alert("You have Logged out");window.location="/myapp/login/"</script>''')

def adm\_home(request):

return render(request, 'admin/adm\_home\_index.html')

def adm\_sendreply(request,id):

return render(request,'Admin/send\_reply.html',{'id':id})

def adm\_sendreply\_post(request):

reply=request.POST['reply']

cid=request.POST['cid']

obj=Complaints.objects.get(id=cid)

obj.reply=reply

obj.status='replied'

obj.save()

return HttpResponse('''<script>alert("Reply Sent");window.location="/myapp/view\_complaints/"</script>''')

def adm\_app\_rej\_staff(request):

res = Staff.objects.filter(status='Pending')

return render(request, 'Admin/apprve\_rjct\_staff.html', {'data':res})

def adm\_app\_rej\_staff\_post(request):

search = request.POST['srch']

res=Staff.objects.filter(name\_\_icontains=search,status="Pending")

return render(request, 'Admin/apprve\_rjct\_staff.html', {'data':res})

def approve\_post(request,id):

res=Staff.objects.filter(id=id).update(status="approved")

return HttpResponse('''<script>alert("Approved!");window.location="/myapp/apprvd\_nd\_rjctd\_staff/"</script>''')

def reject\_post(request,id):

res=Staff.objects.filter(id=id).update(status="rejected")

return HttpResponse('''<script>alert("Rejected!");window.location="/myapp/apprvd\_nd\_rjctd\_staff/"</script>''')

def adm\_chngpasswd(request):

return render(request,'Admin/Chng\_psswrd.html')

def adm\_chngpasswd\_post(request):

old\_password=request.POST['textfield']

new\_password=request.POST['textfield2']

confirm\_password = request.POST['textfield3']

obj=Login.objects.filter(id=request.session['lid'],password=old\_password)

if obj.exists():

obj = Login.objects.get(id=request.session['lid'], password=old\_password)

obj.password=confirm\_password

obj.save()

return HttpResponse(

'''<script>alert("Password Changed Successfully.");window.location="/myapp/login/"</script>''')

else :

return HttpResponse('''<script>alert("Incorrect Password");window.location="/myapp/Change\_password/"</script>''')

def adm\_vw\_rejected\_staff(request):

res = Staff.objects.filter(status='rejected')

return render(request, 'Admin/rejected\_staff.html', {'data':res})

def adm\_vw\_rejected\_staff\_post(request):

search=request.POST['srch']

res = Staff.objects.filter(name\_\_icontains=search, status="rejected")

return render(request, 'Admin/rejected\_staff.html', {'data':res})

def adm\_vw\_apprej\_staff(request):

res = Staff.objects.filter(status='approved')

return render(request, 'Admin/adm\_vw\_approved.html', {'data':res})

def adm\_vw\_apprej\_staff\_post(request):

search=request.POST['srch']

res=Staff.objects.filter(name\_\_icontains=search,status="approved")

return render(request, 'Admin/adm\_vw\_approved.html', {'data':res})

def adm\_vw\_complaints(request):

res=Complaints.objects.all()

return render(request,'Admin/view\_complaints.html',{'data':res})

def adm\_vw\_complaints\_post(request):

from\_date=request.POST['from\_date']

to\_date=request.POST['to\_date']

res=Complaints.objects.filter(date\_\_range=[from\_date,to\_date])

return render(request,'Admin/view\_complaints.html',{'data':res})

#=========Staff=============================

def stf\_home(request):

return render(request,'staff/stf\_home.html')

def stf\_adquest(request):

res = Test.objects.filter(STAFF\_\_LOGIN\_id=request.session['lid'])

q = Question.objects.filter(LOGIN\_id=request.session['lid'])

return render(request,'staff/add\_questions.html',{'data':res,'q':q})

def stf\_adquest\_post(request):

qut = request.POST['quest']

qut2 = request.POST['questions']

tq = TestQuestions()

if 'quest' in request.POST:

tq.question = qut

tq.type = 'typed'

else:

tq.question = Question.objects.filter(id=qut2).question

tq.type = 'list'

tq.TEST\_id = request.POST['tname']

tq.option1 = request.POST['opt1']

tq.option2 = request.POST['opt2']

tq.option3 = request.POST['opt3']

tq.option4 = request.POST['opt4']

tq.correct\_answer = request.POST['crctans']

tq.save()

return HttpResponse('''<script>alert("Question added successfully.");window.location="/myapp/stf\_add\_questions/"</script>''')

def stf\_addtest(request):

return render(request,'staff/add\_test.html')

def stf\_addtest\_post(request):

t\_name=request.POST['tname']

date=request.POST['date']

time=request.POST['time']

obj=Test()

obj.testname=t\_name

obj.date=date

obj.time=time

obj.STAFF= Staff.objects.get(LOGIN\_id=request.session['lid'])

obj.save()

return HttpResponse('''<script>alert("Test added successfully.");window.location="/myapp/stf\_add\_test/"</script>''')

# 

# ANNEXURE-B

**ABBREVIATION**

* **IDE**-Integrated Development Environment
* **HTML-**Hypertext Mark-up Language
* **SQL**-Structured Query Language
* **UI**-User Interface
* **PK**-Primary Key
* **FK**-Foreign Key
* **SVGA–**Standard Video Graphics Array

59