VISVESVARAYA TECHNOLOGICAL UNIVERSITY

BELAGAVI - 590018.

A MINI - PROJECT REPORT ON

"SUPREME COURT DATABASE MANAGEMENT SYSTEM"

submitted in partial fulfillment of requirements for the fifth semester

DATABASE MANAGEMENT SYSTEM LABORATORY

for the course of

BACHELOR OF ENGINEERING

in

COMPUTER SCIENCE

submitted by

RITIKA MISHRA
(1ST19CS728)

NIKHIL RAJ
(1ST19CS721)

under the guidance of

Mrs. Shamala M G
Asst.Prof., Dept. of CSE
&
Mr. Hanumantha Rao K R
Asst.Prof., Dept. of CSE

Department of Computer Science & Engineering

SAMBHRAM INSTITUTE OF TECHNOLOGY

BENGALURU 560097.

SAMBHRAM INSTITUTE OF TECHNOLOGY

M.S. PALYA, JALAHALLI EAST, BENGALURU - 560097.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the mini-project entitled **"SUPREME COURT DATABASE MANAGEMENT SYSTEM"** has been carried out by **RITIKA MISHRA (1ST19CS728)**, **NIKHIL RAJ(1ST19CS721)** bonafide students of **Sambhram Institute of Technology** in partial fulfillment of requirements for the fifth semester **DATABASE Management System Laboratory,** prescribed by the **Visvesvaraya Technological University (VTU)**, Belagavi during the academic year **2021-2022**. It is also hereby, ensured that all corrections/suggestions indicated for internal assessment have been incorporated while submitting this report. This report has been approved as it satisfies the academic requirements with respect to the project work prescribed for the said course.

Mrs. Shamla M G	**Dr. T John Peter**
Asst.Professor, Dept. Of CSE	**Head, Dept. of CSE, SaIT**

External Viva

Names of the examiners **Signatures with date**

1. _____

2. _____

ACKNOWLEDGEMENT

While presenting this mini-project entitled "**Supreme court database management system** " entrusted to us, we feel that it is our duty to acknowledge the guidance and help provided for its success. We take this occasion to thank God, almighty for blessing us with his grace and taking our endeavour to a successful culmination.

We are grateful to our institution, **Sambhram Institute of Technology** for providing us, a congenial atmosphere to carry out this project with all possible amount of ease.

We thank **Dr. H.G Chandrakanth,** Principal, Sambhram Institue of Technology, and **Dr. T John Peter,** HOD, Department of Computer Science & Engineering, for their constant support.

We extend our thanks to our esteemed guides, **Mrs. Shamala M G and Mr. Hanumantha Rao K R** Assistant Professors, Department of Computer Science & Engineering for providing us with the right guidance and advice at the crucial junctures and for showing us the right way.

We are also indebted to our parents and friends for their continued moral and material support throughout the course of this project and helping us in showcasing the project's presentation.

Our thanks to all teaching, non-teaching staffs and others who have contributed bits, bytes and words to accomplish this project.

RITIKA MISHRA

NIKHIL RAJ

ABSTRACT

Supreme court database management system is a project which aims is developing a computerized system to provide e-documentation, e-filling. In Supreme court database management system, it stores records of cases, judgement, caveats, judges bio and advocate bio are maintained and manipulated. There are now some investigations are involved to find out or to correctly done the work. Generally, these works are done and managed manually hence leading to the chances of human errors that may create some problems. Thus, a secured and reliable system is required to handle it. The report generated, thus maintains a log or a track record of the records of the cases and judgements in a simple and an effective format. There by, the options of entering the database also stand enabled. Overall this project of ours is being developed to help the supreme court to maintain them in an optimal way possible and also to minimize effort of human.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION TO DATABASE

A **DATABASE** is a collection of related database. By DATABASE, we mean known facts that can be recorded and that have implicit meaning. For example, consider the names, telephone numbers, and addresses of the people you know. You may have recorded this DATABASE in an indexed address book or you may have stored it on a hard drive, using a personal computer and software such as Microsoft Access or Excel. This collection of related DATABASE with an implicit meaning is a DATABASE.

The preceding definition of database is quite general; for example, we may consider the collection of words that make up this page of text to be related DATABASE and hence to constitute a DATABASE. However, the common use of the term DATABASE is usually more restricted. A DATABASE has the following implicit properties:

- A DATABASE represents some aspect of the real world, sometimes called the **miniworld** or the universe of discourse (**UOD**). Changes to the miniworld are reflected in the DATABASE.
- A DATABASE is a **logically coherent collection of DATABASE** with some inherent meaning. A random assortment of DATABASE cannot correctly be referred to as a DATABASE.A DATABASE is designed, built, and populated with DATABASE for a specific purpose. It has an intended group of users and some preconceived applications in which these users are interested.

A database may be generated and maintained manually or it may be computerized. For example, a library card catalog is a database that may be created and maintained manually. A computerized database may be created and maintained either by a group of application programs written specifically for that task or by a database management system. We are only concerned with computerized database in this report.

Defining a DATABASE involves specifying the DATABASE types, structures, and constraints of the DATABASE to be stored in the DATABASE. The DATABASE definition or descriptive information is also stored by the DBMS in the form of a DATABASE catalog or dictionary; it is called meta-DATABASE.

- **Constructing** the DATABASE is the process of storing the DATABASE on some storage medium that is controlled by the DBMS.
- **Manipulating** a DATABASE includes functions such as querying the DATABASE to retrieve specific DATABASE, updating the DATABASE to reflect changes in the miniworld, and generating reports from the DATABASE.
- **Sharing a** DATABASE allows multiple users and programs to access the DATABASE simultaneously.

An application program accesses the DATABASE by sending queries or requests for DATABASE to the DBMS. A query typically causes some DATABASE to be retrieved; a transaction may cause some DATABASE to be read and some DATABASE to be written into the DATABASE.

Other important functions provided by the DBMS include protecting the DATABASE and maintaining it over a long period of time. Protection includes system protection against hardware or software malfunction (or crashes) and security protection against unauthorized or malicious access. A typical large DATABASE may have a life cycle of many years, so the DBMS must be able to maintain the DATABASE system by allowing the system to evolve as requirements change over time.

It is not absolutely necessary to use general-purpose DBMS software to implement a computerized DATABASE.

We could write our own set of programs to create and maintain the DATABASE, in effect creating our own special-purpose DBMS software. In either case—whether we use a general-purpose DBMS or not—we usually have to deploy a considerable amount of complex software. In fact, most DBMSs are very complex software systems. To complete our initial definitions, we will call the DATABASE and DBMS software together a DATABASE system.

CHAPTER 2

USER REQUIREMENT DEFINITION

2.1 EXISTING SYSTEM

Today, data on cases filed in the Indian judicial system is still gathered and maintained in manual data systems by courts across the country involving manual recording of case and court information in manuals. Each month, considerable time is spent by local courts compiling data from manual registers to submit reports to higher courts.Very little data is available in real time.

2.2 PROPOSED SYSTEM

A comprehensive Court Management Systems is designed for the country that will enhance quality, responsiveness and timeliness of courts. It will also provide a common national platform for recording and maintaining judicial statistics from across the country.A new system of Case Management is to enhance user friendliness of the Judicial System.

2.3 FEASIBILITY STUDY

The feasibility of the project is analysed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the users. For feasibility analysis, some understanding of the major requirements for the system is essential.

2.3.1 Economic Feasibility

This study is carried out to check the economic impact system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus,the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products have to be purchased.

2.3.2 Technical Feasibility

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must have a high demand on the available technical resources. This will lead to high demand being placed on the client. The developed system must have a modest requirement, as only minimal or null changes for the implementing this system.

2.3.3 Operational Feasibility

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept is as a necessity.

2.4 DESCRIPTION OF THE RELATIONS (TABLES)

(I) Judges : This table is used to store details of the judges appointed in the SC.

Sino	Name	Type	Description
1	id	Varchar (10)	Id of judges
2	name	Varchar (100)	Name of Judges
3	designation	Varchar (100)	Designation
4	dob	Varchar (20)	Date of Birth
5	doa	Varchar (20)	Date of Appointment
6	dor	Varchar (20)	Date of Retirement
7	nh	Varchar (500)	Notable Judgements
8	about	Varchar (900)	Biography

Table 2.4.1: Judges Record

(II) Advocates : This table is used to store the details of the Advocates who are appointed in SC.

Sino	Name	Type	DESCRIPTION
1	Name	Varchar (100)	Name of Advocates
2	Code	Varchar (100)	AoR Code
3	Mobile_no	Int (10)	Contact no. of Advocates
4	Email_id	Varchar (50)	Email_id of Advocates

5	**Address**	**Varchar (100)**	**Address of Avocates**
6	**Charge**	**Varchar (30)**	**Charge Per Appearance**
7	**Acheivments**	**Varchar (100)**	**Case won by Advocates**

Table 2.4.2: Advocates

(III) Case Categories : This table contains the categories of the case which includes the Sub-categories.

Slno	Name	Type	Description
1	ccode	int (11)	Case Code
2	category	varchar (100)	Case category
3	scategory	varchar (100)	Sub-category

Table 2.4.3: Categories of cases

(IV) Court Fees : This table contains the fees of each case charged by SC.

Sl. No.	Name	Type	Description
1	casetype	varchar (100)	Types of Cases
2	category	varchar (100)	Categories of cases
3	scategory	varchar (100)	Sub-categories
4	fees	int (50)	Court Fees

Table 2.4.4: Court Fees

(V) Case Status : This table contains the supplier details with address, id and other Details.

Sino	Name	Type	Description
1	cno	Varchar (10)	Case no.
2	diary_no	Varchar (10)	Cases written in diary no.
3	ctype	Varchar (30)	Case type
4	cyear	Varchar (10)	Year of Case filled
5	pname	Varchar (100)	Petitioner Name
6	rname	Varchar (100)	Respondent Name
7	radvocates	Varchar (100)	Respondent Advocates
8	padvocates	Varchar (100)	Petitioner Advocates
9	judges	Varchar (200)	Judges hearing case
10	nh	Varchar (20)	Next Hearing Date
11	description	Varchar (1000)	Status description

Table 2.4.5: Case Status

(VI) E-File : This table contains the person details with address, id and other details as well as case details.

Sl. No.	Name	Type	Description
1	id	Int(11)	Automatically generated
2	name	Varchar (20)	Person name
3	dob	Date	Date of birth
4	address	Varchar (30)	Person's Address
5	state	Varchar (10)	Name of states person belongs to
6	district	Varchar (10)	District Name
7	pin-code	Varchar (10)	Postal code
8	gender	Varchar (10)	Gender
9	aadharno	Varchar (30)	Aadhar Card No.
10	phone	Varchar (10)	Phone no.
11	email	Varchar (30)	Email-id
12	csubject	Varchar (100)	Case Type
13	csdetail	Varchar (500)	Description

Table 2.4.6: E-File

(VII) Judgement : This table contains the judgement date and judgement as well as verdicts for daily order.

Sl. No.	Name	Type	Description
1	caseno	Int(11)	Case no
2	jd	Varchar (20)	Judgement Date
3	description	date	Judgement Details

Table 2.4.7: Judgement

(VIII)Trigger judgement: This table contains the case & judgement description.

Sino	Name	Type	Description
1	cno	Varchar (10)	Case no.
2	diary_no	Varchar (10)	Cases written in diary no.
3	ctype	Varchar (30)	Case type
4	cyear	Varchar (10)	Year of Case filled
5	pname	Varchar (100)	Petitioner Name
6	rname	Varchar (100)	Respondent Name
7	radvocates	Varchar (100)	Respondent Advocates
8	padvocates	Varchar (100)	Petitioner Advocates
9	judges	Varchar (200)	Judges hearing case

Table 2.4.8: Table for Trigger Operation

CHAPTER 3

HARDWARE AND SOFTWARE REQUIREMENTS

3.1 FUNCTIONAL REQUIREMENTS

Functional requirements are statements of services the system should provide, how the system should react to particular inputs, and how the system should behave in particular situations. In some cases, the functional requirements may also explicitly state what the system should not do.

3.1.1 Valid assumptions and dependencies

- The **assumptions** are:
  - The source code pertaining to each module developed should be error free.
  - The system should be user friendly.
  - The information, modifications and updates to the DATABASE should be available to all the administrators of the library.
  - The system should have feasible storage capacity and should provide faster access to the DATABASE.
  - The system should provide retrieval facility and also facilities for quick and understandable transactions.
  - Users should register themselves in order to procure administrative membership.
  - Valid credentials shall only be keyed-in without which, authentication becomes unsuccessful.
- The **dependencies** are:
  - The specific hardware and software through which the product operates.
  - Thus on the basis of the list of the requirements specification, the application would be developed to a fully functional one
  - The end-users have to possess proper knowledge and understanding in order to work with it.
  - The system should have a general report generator.
  - All updates pertaining to all modules are to be recorded and changes should happen in the referenced modules concurrently.

3.1.2 DATABASE Requirement

The input consists of query to the DATABASE and the output consists of solutions for the query. The output also includes the user receiving the details of their accounts.

In this project, the inputs will be the queries as fired by the users like creating an account, adding information on the basis of various criteria, updating the information pertaining to individual records, deleting individual records, searching and retrieving records, etc.

Now, the solutions for the queries as the respective outputs will be visible when the users request the server through the GUI.

3.1.3 External Interface Requirement

GUI

The application that is developed provides a good graphical user interface for the user or the administrator who operates the system, performs the required tasks such as insert, update, delete and search on the basis of various criteria.

General features of the GUI

- It allows the user to get quick status of case and advocates, judges details.
- Provides search features on basis of case no, diary no.
- All the modules designed are collectively integrated to form the Menu Driven Interface (MDI).
- The design of the interface is simple and all the modules within it, follow a standard template.
- The user interface must be able to interact with the user management module.
- **Login interface or the DBA Staff Authentication Window**

  Log in can be done through the staff authentication window which on successful authentication redirects the staff to the insertion/deletion interface.
- **Menu - oriented views**

  The menu may further show the categories based on which the whole system is supposed to be administered.

- **Control Panel**

  This panel will allow the DBA Staff to update/delete and refresh the contents or records based on a particular criterion.

3.1.4 Valid assumptions and dependencies

The product will operate in Windows environment. The only requirement to use this product would be the installation of supporting software packages like MYSQL DATABASE, APACHE server , XAMPP CONTROL PANEL version 3.2.2.

3.2 NON - FUNCTIONAL REQUIREMENTS

These are constraints on the services or functions offered by the system. They include timing constraints, constraints on the development process, and constraints imposed by standards. Non-functional requirements often apply to the system as a whole, rather than individual system features or services.

3.2.1 Performance requirement

The proposed system that we have developed may be used by court.

Therefore, it is expected that the DATABASE would perform functionally, all the requirements that are specified by the court.

- The performance of the system should be fast and accurate.
- The Supreme Court DATABASE Management System shall handle expected and non-expected errors in ways that prevent loss in information and long downtime period. Thus it should have inbuilt error testing to identify and handle exceptions.
  - The system should be able to handle large amount of DATABASE.

3.2.2 Product and usability requirements

- **Availability**: The system is available 100% for the user and can be used on a 24 x 7 basis.
- **Accuracy**: The system shall accurately provide real time information taking into consideration various concurrency issues.

- **Reliability**: The system has to be 100% reliable due to the importance of DATABASE and the damages that can be caused by incorrect or incomplete DATABASE.
- **Maintainability**: Changes (addition of new members, DATABASE changes) must be verified once per day at least.
- **Portability:** The system should also be portable.

3.2.3 Requirements attributes

- There may be multiple administrators creating the project, so, all of them would have rights to create changes to the system.
- The project should be open source.
- The quality of the DATABASE is maintained in such a way so that it can be very friendly to all the users of the DATABASE.
  The users should be able to easily install the software on the system.

3.3 SOFTWARE REQUIREMENTS SPECIFICATION

The application is developed using HTML and PHP as the front end which is supported by Apache server and MySQL DATABASE as the back end for accessing and connecting the front end to the DATABASE.

PHP is a powerful but simple language aimed primarily at developers creating applications by using the Apache server, inherits many of the best features of HTML and php, but few of the inconsistencies and anachronisms, resulting in a cleaner and more logical language.

SQL (Structured Query Language) is used for defining, manipulating, controlling, storing and viewing the information present in a DATABASE.

3.3.1 Specifications of the software used for application development

The following software are required to develop the application:

Front end: PHP

- **Back end: MYSQL**
- **DATABASE used:** MYSQL DATABASE.
- XAMPP CONTROL PANNEL V3.2.2
- **Operating system:** Windows 10 (x64)

3.4 HARDWARE REQUIREMENTS SPECIFICATION

The following hardware requirements are needed to develop the application :

- Computer that has a 1.6 GHz or faster processor (2 GHz recommended).
- 1 GB (32-bit) or 2 GB (64-bit) RAM (add 512 MB if running in a virtual machine).
- 10 GB of available hard disk space.

  5400 RPM hard disk drive.

CHAPTER 4

DESIGN AND IMPLEMENTATION

4.1 ER MODEL:

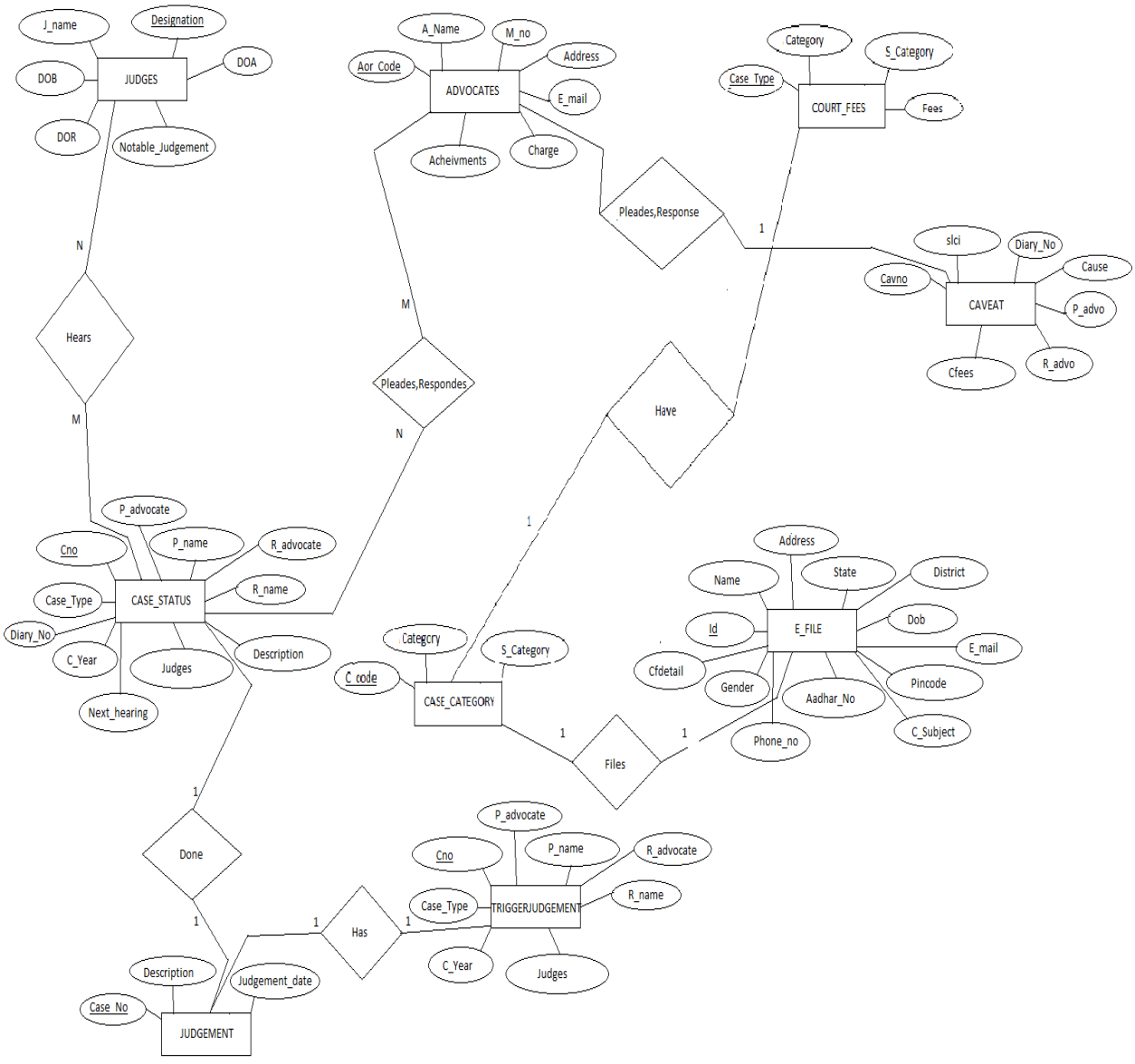


Fig 4.1: ER diagram for a simple SUPREME COURT DATABASE

The ER model describes DATABASE as *entities*, *relationships*, and *attributes*. The basic object that the ER model represents is an **entity**, which is a *thing* in the real world with an independent existence. An entity may be an object with a physical existence or it may be an object with a conceptual existence.

Here each entity has one or more attributes. Fig 4.1 shows the entities with their respective attributes, relationships, with cardinality ratios which further mean something about the quality of participation that may be offered by the entities for their relationships between various other entities

4.2 SCHEMA DIAGRAM

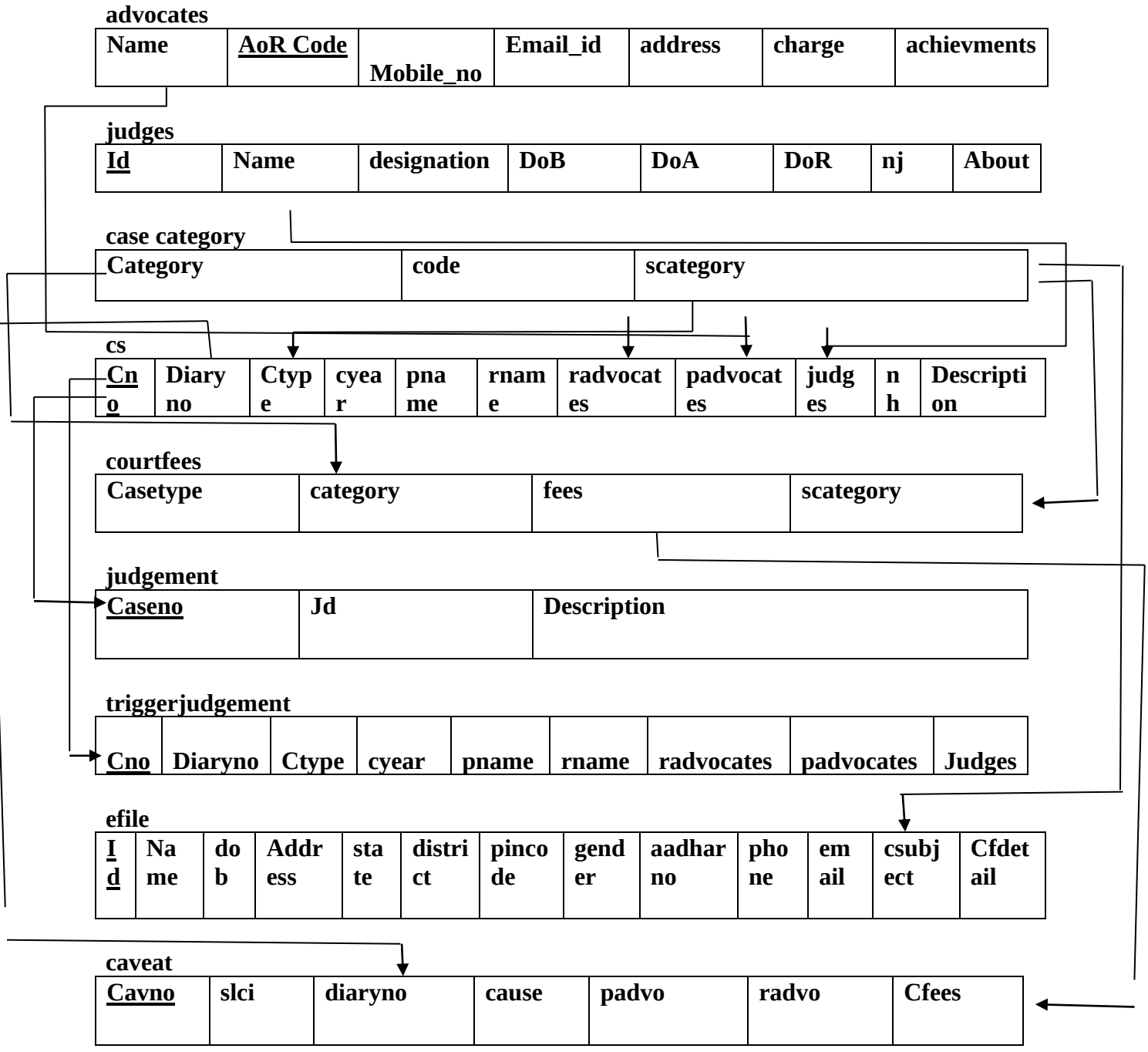


Fig 4.2: Logical schema diagram displaying the referential integrity constraints pertaining to a simple Shop management system.

4.3 CODES FOR FRONTEND

```
<!doctype html>
<html lang="en">

<head>
    <!-- Required meta tags -->
    <meta charset="utf-8">
    <meta name="viewport" content="width=device-width, initial-scale=1">

    <!-- Bootstrap CSS -->
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-1BmE4kWBq78iYhFldvKuhfTAU6auU8tT94WrHftjDbrCEXSU1oBoqyl2QvZ6jIW3" crossorigin="anonymous">

    <!-- Font Awsome Icons -->
    <link rel="stylesheet" href="./css/all.css">

    <!-- External Css -->
    <link rel="stylesheet" href="Style.css">
    <title>Home | Supreme Court Of India</title>
</head>

<body>
    <!-- header -->

    <div class="header">
        <a href="./Home.html" class="logo">
            <img src="./img/peakpx.jpeg" class="header-img" alt="emblumn of india">
        </a>
        <div class="header-left">
            <div>
                <h1>Supreme Court of India</h1>
                <h3 class="headerhead-2">
                    || यतो धर्मस्ततो जय: ||
                </h3>
            </div>
        </div>
    </div>

    <!--Supreme Court Carousel-->

    <div id="SCcarousel" class="carousel slide" data-bs-ride="carousel">
        <div class="carousel-indicators">
            <button type="button" data-bs-target="#SCcarousel" data-bs-slide-to="0" class="active" aria-
current="true" aria-label="Slide 1"></button>
            <button type="button" data-bs-target="#SCcarousel" data-bs-slide-to="1" aria-label="Slide
2"></button>
            <button type="button" data-bs-target="#SCcarousel" data-bs-slide-to="2" aria-label="Slide
3"></button>
            <button type="button" data-bs-target="#SCcarousel" data-bs-slide-to="3" aria-label="Slide
4"></button>
            <button type="button" data-bs-target="#SCcarousel" data-bs-slide-to="4" aria-label="Slide
5"></button>
        </div>
        <div class="carousel-inner">
            <div class="carousel-item active">
                <img src="img/SCimg1.jpeg" class="d-block w-100" alt="Pic1">
            </div>
            <div class="carousel-item">
                <img src="./img/Scimg2.jpeg" class="d-block w-100" alt="Pic2">
            </div>
            <div class="carousel-item">
                <img src="./img/Scimg3.jpeg" class="d-block w-100" alt="Pic3">
            </div>
            <div class="carousel-item">
                <img src="./img/Scimg4.jpeg" class="d-block w-100" alt="Pic4">
            </div>
            <div class="carousel-item">
                <img src="./img/Scimg5.jpeg" class="d-block w-100" alt="Pic5">
```

```
            </div>
        </div>
        <button class="carousel-control-prev" type="button" data-bs-target="#SCcarousel" data-bs-
slide="prev">
                    <span class="carousel-control-prev-icon" aria-hidden="true">

                    </span>
                    <span class="visually-hidden">
                        Previous
                    </span>
            </button>
        <button class="carousel-control-next" type="button" data-bs-target="#SCcarousel" data-bs-
slide="next">
                    <span class="carousel-control-next-icon" aria-hidden="true">

                    </span>
                    <span class="visually-hidden">
                        Next
                    </span>
            </button>
    </div>



    <!-- Nav bar -->

    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
        <div class="container-fluid">
            <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-
label="Toggle navigation">
             Menu<span class="navbar-toggler-icon"></span>
            </button>
            <div class="collapse navbar-collapse" id="navbarSupportedContent">
                <ul class="navbar-nav me-auto mb-2 mb-lg-0">
                    <li class="nav-item">
                        <a class="nav-link active" aria-current="page" href="./Home.html"><i class="fas fa-
home">Home</i></a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="./nc.html">Notice & Circular</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="do.php" tabindex="-1" aria-disabled="true">Daily Order</a>
                    </li>
                    <li class="nav-item">
                        <a class="nav-link" href="./ef.html" tabindex="-1" aria-disabled="true">E-Filing</a>
                    </li>
                </ul>
                <li class="navbar-nav nav-item dropdown">
                    <a class="nav-link dropdown-toggle active" href="#" id="navbarDropdown" role="button"
data-bs-toggle="dropdown" aria-expanded="false">
                        About Court
                    </a>
                    <ul class="dropdown-menu" aria-labelledby="navbarDropdown">
                        <li><a class="dropdown-item" href="./History.html">History</a></li>
                        <li><a class="dropdown-item" href="./Contact.html">Contact</a></li>
                        <li><a class="dropdown-item" href="./StaffLogin.html">Log in</a></li>
                    </ul>
                </li>
            </div>
        </div>
    </nav>

    <!-- Body Content -->

    <div class="row">
        <div class="card col-sm-6">
            <div class=" card-title rounded text-center ">
                Quick Links
```

```
            </div>
            <div class="card-body ">
                <div class="list-group ">
                    <li class="list-group-item list-group-item-action text-center ">
                        <a href="./judge.php" class="card-body-text "><strong>●Judges</strong></a>
                    </li>
                    <li class="list-group-item text-center ">
                        <a href="advocates.php" class="card-body-text "><strong>●Advocates</strong></a>
                    </li>
                    <li class="list-group-item text-center ">
                        <a href="casestatus.php" class="card-body-text "><strong>●Case Status</strong></a>
                    </li>
                    <li class="list-group-item text-center ">
                        <a href="jv.php" class="card-body-text "><strong>●Judgements &amp;
Verdicts</strong></a>
                    </li>
                    <li class="list-group-item text-center ">
                        <a href="cavet.php" class="card-body-text "><strong>●Caveat</strong></a>
                    </li>
                    <li class="list-group-item text-center ">
                        <a href="casecategory.php" class="card-body-text "><strong>●Case
Category</strong></a>
                    </li>
                    <li class="list-group-item text-center ">
                        <a href="cfshow.php" class="card-body-text "><strong>●Court Fees</strong></a>
                    </li>
                </div>
            </div>
        </div>
        <div class="container col-sm-6 offset-sm-3 body-content ">
            <p>The Supreme Court of India is the highest judicial forum and final court of appeal under the
Constitution of India, the highest constitutional court, with the power of judicial review. Consisting of the
Chief Justice of India and a maximum
                of 30 other judges, it has extensive powers in the form of original, appellate and advisory
jurisdictions.</p>
            <br>
            <p>As the final court of appeal of the country, it takes up appeals primarily against verdicts of
the high courts of various states of the Union and other courts and tribunals. It safeguards fundamental
rights of citizens and settles disputes
                between various governments in the country. As an advisory court, it hears matters which may
specifically be referred to it under the constitution by President of India. It also may take cognisance of
matters on its own (or suo moto),
                without anyone drawing its attention to them. The law declared by the supreme court becomes
binding on all courts within India and also by the union and state governments.Per Article 142, it is the
duty of the president to enforce the
                decrees of the supreme court. Government Of India ● 2022</p>
        </div>
    </div>

    <!-- Footer -->
    <footer class="footer ">
        <div class="container ">
            <div class="row ">
                <div class="col-6 col-sm-3 offset-sm-1 align-self-center ">
                    <h5>Links</h5>
                    <ul class="list-unstyled ">
                        <li>
                            <a href="./Home.html"><i class="fa fa-home"></i>Home</a>
                        </li>
                        <li><a href="./Contact.html "><i class="fas fa-id-card"></i>Contact</a></li>
                        <li><a href="./History.html"><i class="fas fa-history"></i>History</a></li>
                    </ul>
                </div>
                <div class="col-6 col-sm-7 offset-sm-1 1align-self-center ">
                    <h5 id="footer-links ">Address</h5>
                    <address class="address ">
                        <strong><h3>Supreme Court of India</h3></strong>
                        Tilak Marg,<br>
                         New Delhi-110001 (INDIA)<br>
                        <i class="fa fa-phone fa-lg "></i>: 011-23385903, 23384874<br>
                        <i class="fa fa-fax fa-lg "></i>: 011-23385551<br>
```

```
                        <i class="fa fa-envelope fa-lg ">:<a href="mailto:scbaec@gmail.com
">scbaec@gmail.com</a></i>

                    </address>
                </div>
            </div>
            <div class="row justify-content-center ">
                <div class="col-auto ">
                    <p>Copyright © 2022 Developed by<br><strong>Nikhil Raj And Ritika Mishra</strong></p>
                </div>
            </div>
        </div>
    </footer>
    <!-- Option 1: Bootstrap Bundle with Popper -->
    <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js "
integrity="sha384-ka7Sk0Gln4gmtz2MlQnikT1wXgYsOg+OMhuP+IlRH9sENBO0LRn5q+8nbTov4+1p " crossorigin="anonymous
">
    </script>
</body>

</html>
```

4.4: DATABASE CODE SNIPPETS

```
<?php
$servername = "localhost";
$username = "root";
$password = "";
$dbname = "scdatabase";
$name=$_POST["j_id"];
$code=$_POST["j_name"];
$mobile=$_POST["j_designation"];
$email=$_POST["j_dob"];
$address=$_POST["j_doa"];
$charge=$_POST["j_dor"];
$acheivments=$_POST["j_nj"];
$about=$_POST["j_about"];
// Create connection
$conn = new mysqli($servername, $username, $password, $dbname);
// Check connection
if ($conn->connect_error) {
    die("Connection failed: " . $conn->connect_error);
}

$sql = "INSERT INTO judges (id,name,designation,dob,doa,dor,nj,about)
VALUES ('$name','$code','$mobile','$email','$address','$charge','$acheivments','$about')";

if ($conn->query($sql) === TRUE) {
    echo  "<script>
    alert('Detail Inserted');
    window.location.href='loggedinJudges.php';
    </script>";
} else {
   echo "<script>
    alert('Judge already Exhist');
    window.location.href='loggedinJudges.php';
    </script>";
}

$conn->close();
?>
```

4.5 DATABASE CONNECTIVITY USING PHP

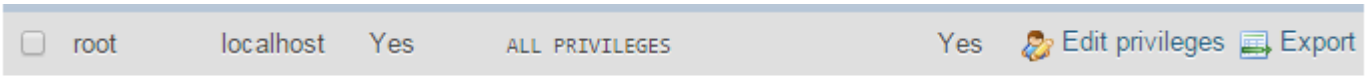
In order to establish connection to the DATABASE, the following directives are needed. They are:

In the first instalment of this MYSQL series, I introduced DATABASE and DATABASE management system. I also presented a brief overview of a popular DBMS.

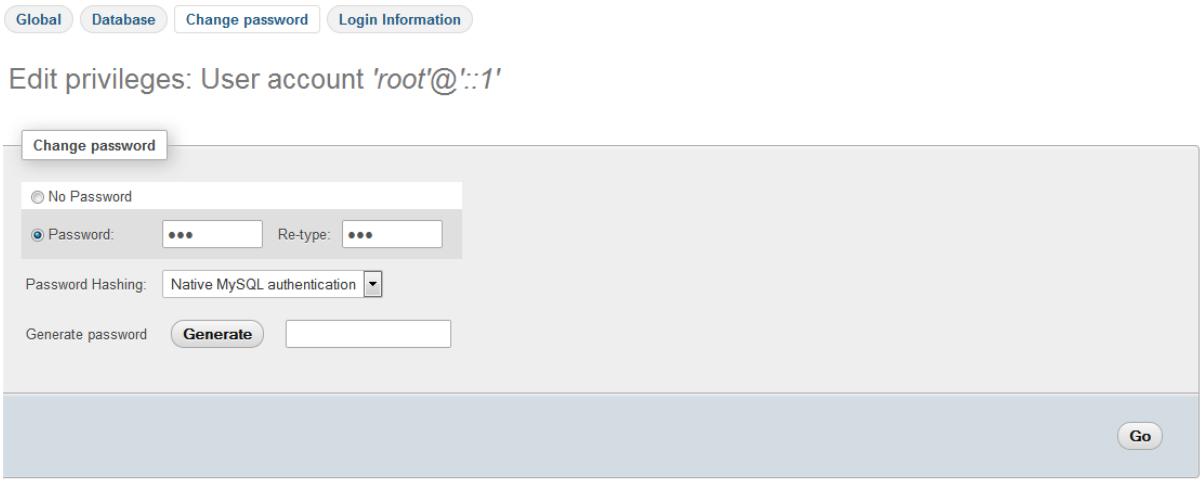
Step1: CHANGING ADMIN PASSWORD

First, let me tell you what PHPMyAdmin is. It is a control panel from where you can manage your DATABASE that you have created. Open your browser and go to localhost/PHPMyAdmin or click “Admin” in XAMPP UI.

When you first installed XAMPP, it only created the username for it to be accessed, you now have to add a password to it by yourself. For this, you have to go to User account where the user is same as the one shown in this picture:



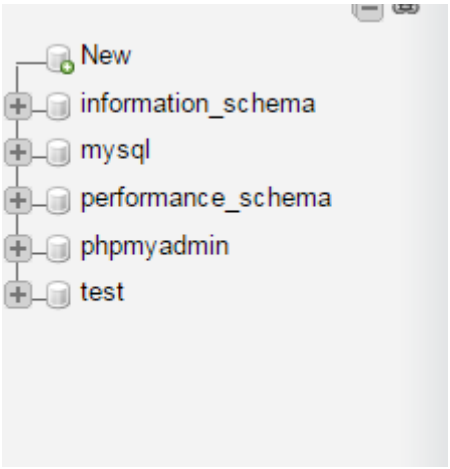
Now click edit privileges and go to change password, type your password there and save it. Remember this password as it will be use to connect to your DATABASE.



Note: It is not necessary to change password to access DATABASE on local host. It is a good practice and that is why we have used a password.

Step 2: CREATE DATABASE

Now return to homepage pf phpmyadmin. Click New button to create a new DATABASE.



In the new window, name your DATABASE as per your need, I am naming it "practice". Now select collation as utf8_general _ci, as we are using it for learning purpose and it will handle all of our queries and DATABASE that will be covered in this tutorial series. Now click on create and your DATABASE will be created.

The newly created DATABASE will be empty now, as there are no tables in it. I will be covering that in the upcoming series where we will learn how to create tables and insert DATABASE in it. In this tutorial, we are going to connect this DATABASE to a localhost using PHP.

STEPS 3: CREATE A FOLDER IN HTdocs

Now, locate the folder where you installed XAMPP and htdocs folder (usually c:/xampp). Create a new folder inside c:/xampp/htdocs/ and name it. we will place web files in this folder. Why we have created folder in htdocs? XAMPP uses folders in htdocs to execute and run your PHP sites.

Note: If your using WAMP, then add your practice folder in c:/wamp/www folder.

STEP 4: CREATE DATABASE CONNECTION FILE IN PHP

Create a new php file and name it some name db_connection.php and save it. Why am I creating a separate DATABASE connection file? Because if you created multiple files in which you want to insert DATABASE or select DATABASE from the DATABASE, you don't need to write the code for DATABASE connection every time. You just have to include it by using PHP custom function **include** on the top of your code and call its function and use it. It also helps when you are moving your project location from one PC to another and you have to change the values on the single file and all the changes will be applied to all other files automatically. Write the following code in your DATABASE file.

STEP 5: CREATE NEW PHP FILE TO CHECK YOUR DATABASE CONNECTION

Create a new php file to connect to your DATABASE. Name it index.php and add this code in this file.

STEP 6: RUN IT!

Now open your browser and go to localhost/practice/index.php and you should see the message '**CONNECTED SUCCESSFULLY' in** t

CHAPTER 5

SNAPSHOTS

5.1 USER HOME PAGE

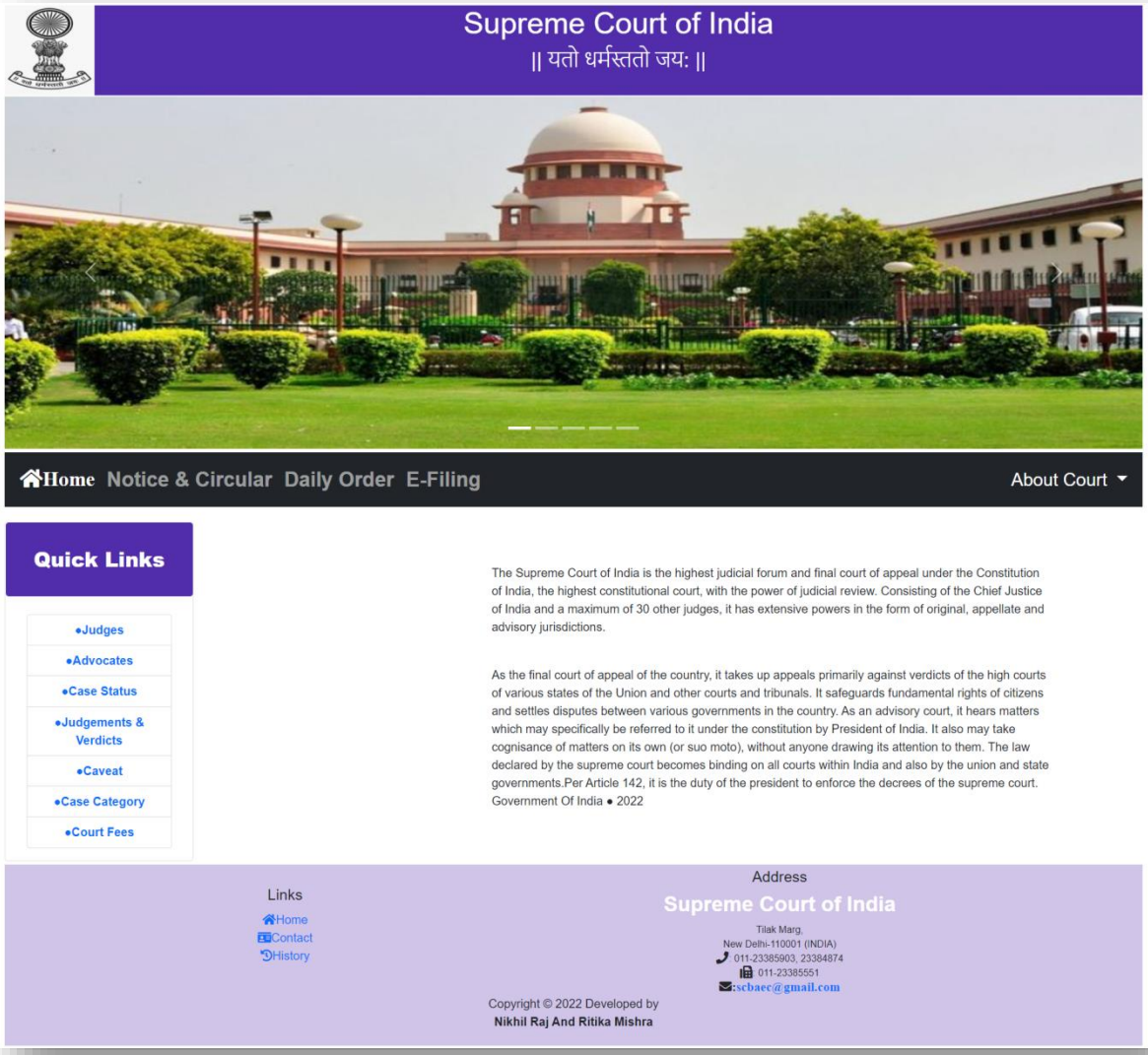


Fig 5.1: The User Front page or the user, which acts as the home page

The **User's Front** page or the **USER** is the first window that appears when the application is initiated. With this the administrators provide valid DATABASE and search option to the users.

After Home Page

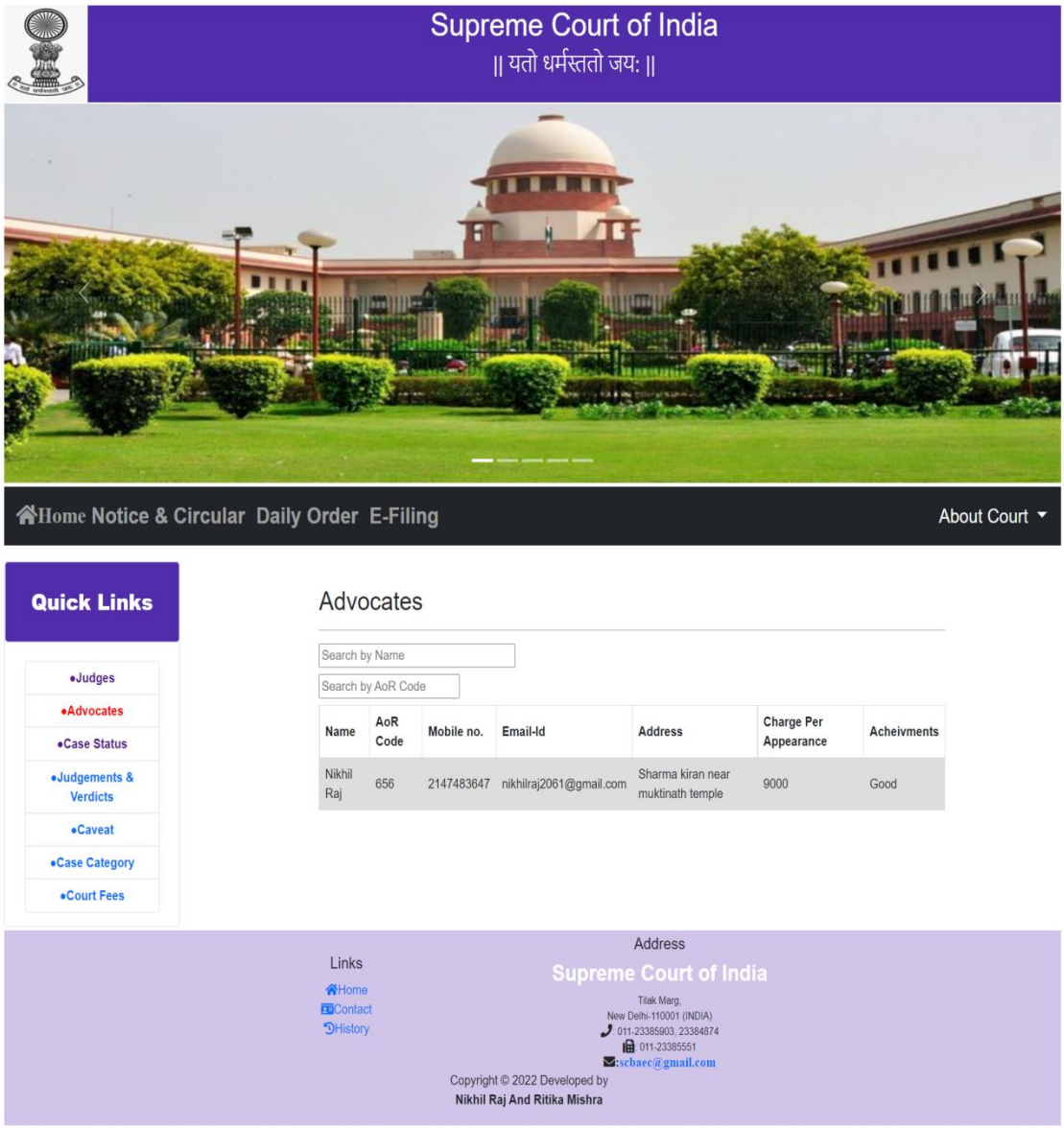


Fig 5.2: Search Page

Here user can search the judges, advocates or any case details or even do e-filing.

The users can also figure out details of judges or advocates based on their AOR code or by the name also.

Log In Page For DBA Staff

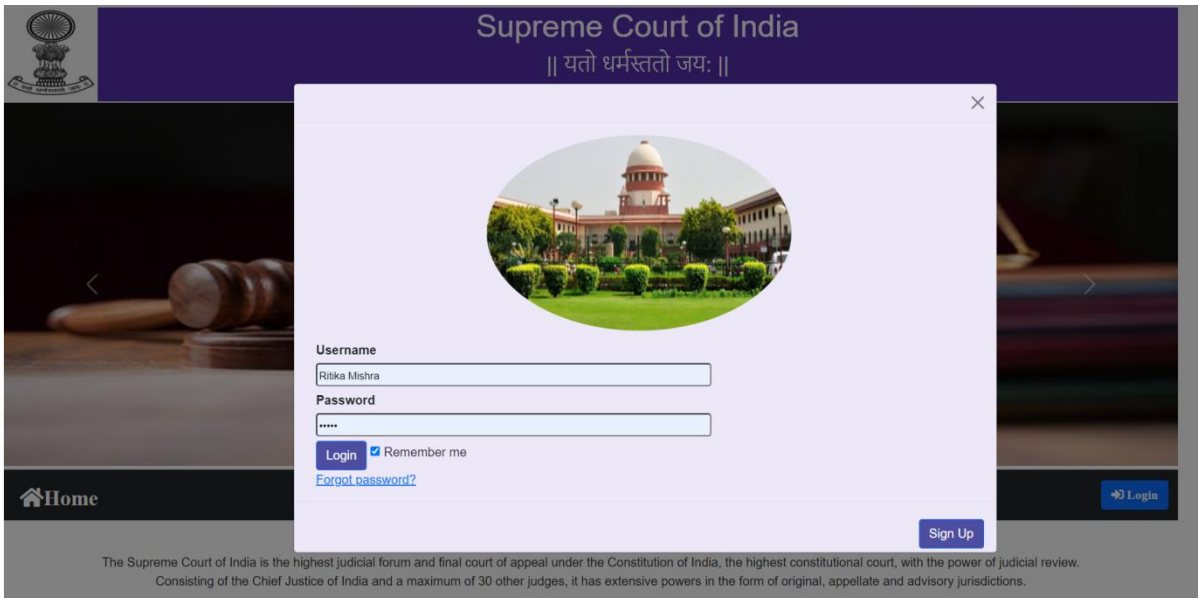


Fig 5.3: Login Modal

Here DBA Staff can Log in to insert and delete DATABASE.

They also have the authority to make updations that will be showed to user’s.

Insertion Page For Advocates to insert DATABASE

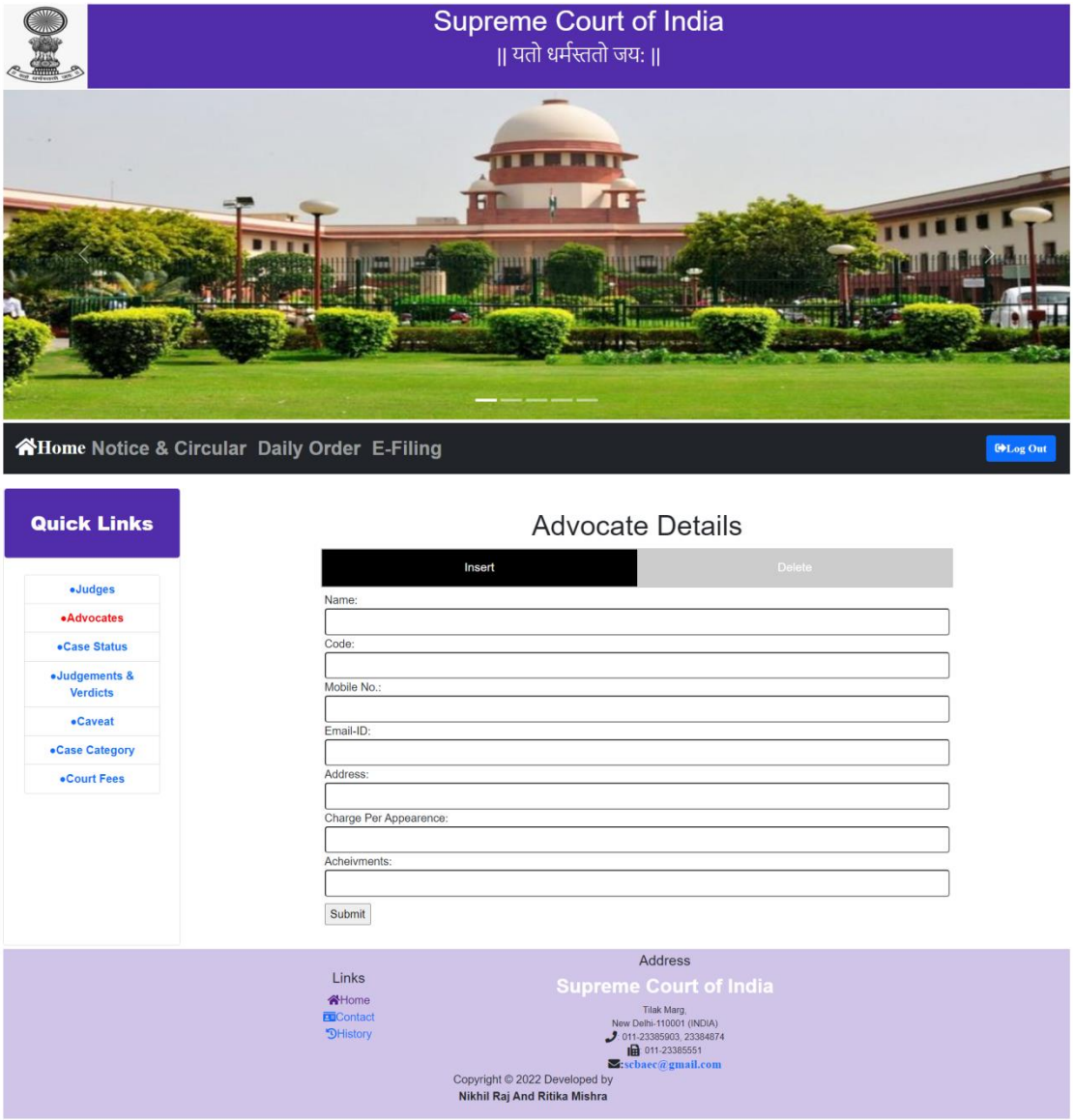


Fig 5.4: Insertion Page

Here DBA Staff can insert DATABASE.

Here we can insert the details of the Judges ,advocates ,judgments,etc;.

Page For deletion of DATABASE by DBA Staff.

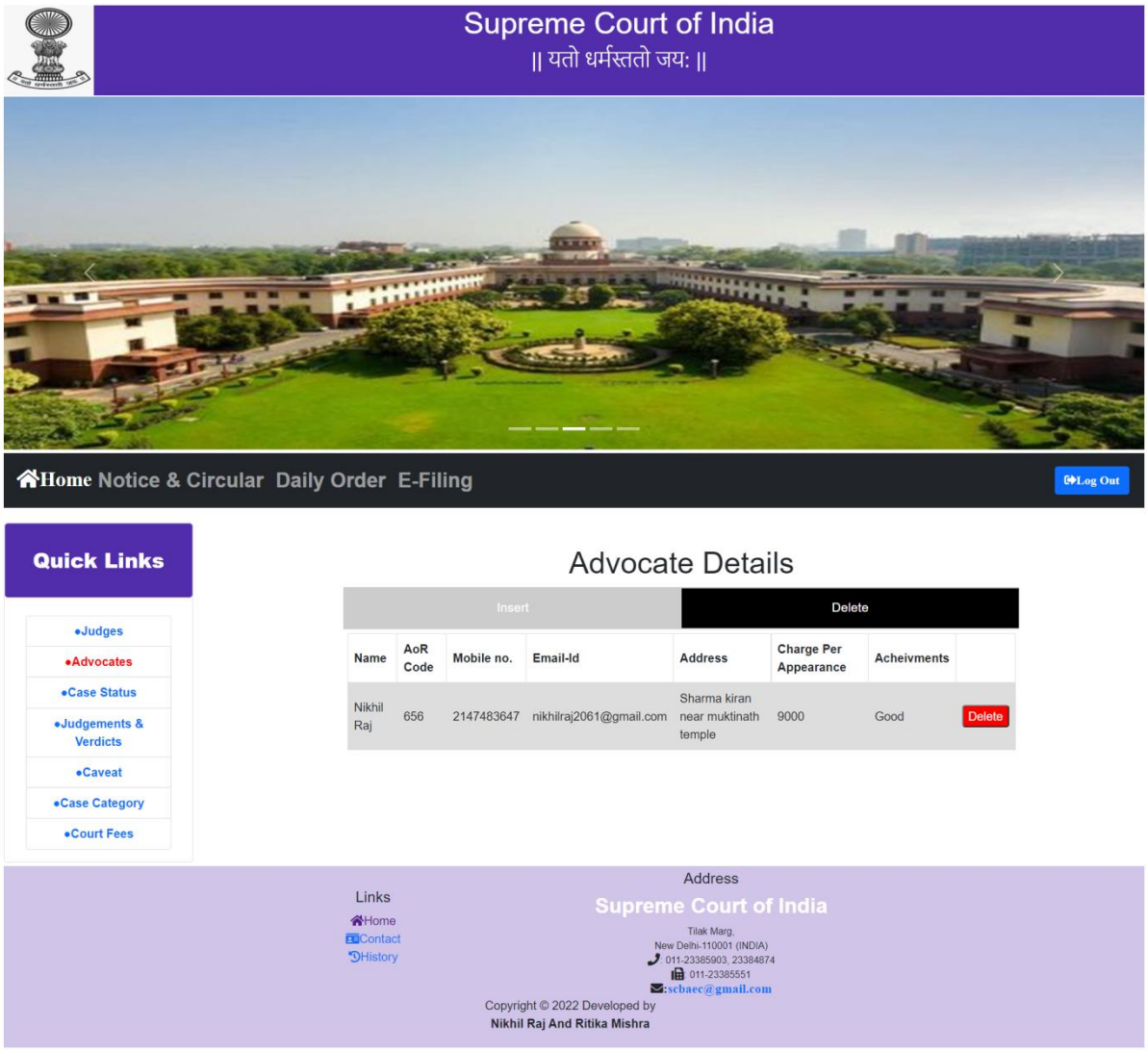


Fig 5.5: Page for deletion

Here we can delete the pages for the judges,advocates,and judgments.

CONCLUSION

Supreme Court could have managed the records in the paper too but, it is not too much reliable to depend. DATABASE written in paper can be lost and even it is tough to search the particular DATABASE. So to provide the information online, it is important to create a DATABASE so that a person related to case or even any other person can know the status of case.We have developed a project “Supreme Court DATABASE Management System”, by implementing that those issues can be solved. Managing DATABASE and retrieval of DATABASE by users can be done easily. As we know DATABASE can handle multiple request at a time so, number of people can use this website at the same time. And it will reduce the paper workload of Supreme Court. This project can also be used by other kinds of court.

BIBLIOGRAPHY

[1] w3schools.com

[2] stackoverflow.com

[3] Fundamentals of DATABASE Systems 7th Edition by Elmasri , Navathe

[4] www.google.co.in

[5] Xamp Applications