Indraprastha Institute of Information Technology

DATABASE MANAGEMENT SYSTEM

JIM n JAM
(Judicial Improvisation and Judicial Automation)

Project Report

Group Name: pawri ho rahi hai

Group Members:

Avinoor Singh Kohli:2019460

Kartikey Gupta:2019427

Meet Modi:2019435

Palak Goel:2019438
Parag Kumar:2018355

WEEK 1

Contribution: All the team Members Contributed Equally

Meeting with teammates and getting to know each other. Formation of a whatsapp group

WEEK 2

Deciding the Online Application that will be executed for the DBMS project. Pitching of ideas and selecting the best amongst them.

# Contribution: All members Contributed Equally and the work was done Together

Overview

Judicial Automation is the primary need of Courts today which are overloaded with Cases and need a tool to Automate their working.

Over 3 crore cases are pending in the courts. Shocking huh...Imagine the number of cases that have already gone by.Total number of judges as per report are 17000. Just by dividing the figures one can get the idea of the burden of court cases on the judges and on the system. The court proceedings are documented for official records, references and study. But does somebody want to go through millions of records just to find A FILE. I think this is where we come into the picture. How about designing a database which could manage all of the workload. But what's included…

1. A lawyer database and their win/loss ratio,their field of study
2. A judge pending case history
3. Current cases history
4. Police Officer appointment and duty to the case
5. List and link of important records related to case

The database will ensure that the document access is restricted well within the abiding laws and access will be given to the required statutory of the government.

Relevance:

1. It can help automate legal departments of large corporations as well as create strong countrywide e-justice networks.
2. This enables the people living in remote areas to fight for their rights and choose a suitable lawyer
3. One can choose the layer according to win/loss ratio and that the lawyer's degree are justified and legal
4. People can upload and submit the documents when asked from them by the court even in the post covid times
5. Such apps can be used to tackle small magistrate cases from home and itself as the database of the case could be maintained at dispense on a web page and cases could be fought by the lawyers in presence of a judge over video call conference.

NOVELTY

The existing solution comprises of :

Wolter Kluwer

TheLawAssist.com,
Clio
MyCase
Ebizframe Justice

All of these previously mentioned apps are centric around the following features:
Case Management
Document Management
Task Management
Billing Management
Plaintiff Management
Schedule Management

We are intuitively planning to maintain a lawyer database with their details and win/loss ratio with an authentication factor/bar council number.

This will be the first portal which will prove to be a complete interface for all the judicial and customary activities

There are many possibilities of the stakeholders but we narrowed down to the most relevant ones.

WEEK 3

Identifying the Stakeholders and Deciding the role of each stakeholder in the database. Questions that the database will be answering for each stakeholder and the queries.

Contribution: Each Member Contributed Equally and all of the work was done together

Stakeholders:

1. Plaintiff

   Role Of a Plaintiff: The Plaintiff represents an organization or a person who has registered a case. The Role that a Plaintiff plays in this software:

   a. The Plaintiff can use the software to check the status of their case.
   b. #They can use the software to handle their account i.e pending payment or outstanding amount.
   c. Plaintiffs can use this software to handle and maintain documents related to the case

d. It can be used to search lawyers based on their performance and their (win/loss) ratio.
e. They can search authentic lawyers(That are registered to the Bar Association). This protects them from being exploited by the lawyers with fake law degrees.
f. They can manage their schedule i.e the hearing day/time

2. Police Officials (if applicable):
   Role of Police Officials: The police Officials play an important role in the smooth functioning of the judicial system as it is the principal duty of the police to arrest criminals and conduct them until the conclusion of the trial. They can use software for:
   a. For checking the schedule of court hearings so that the witness/accused can be presented in front of court
   b. To access FIR regarding a particular case
   c. To submit FIR connecting the case.
   d. To Check If there was a previous case filed against the accused. This helps in background check for passports, visas etc.

e. To upload various evidences against the case and access the case related documents.

3. Lawyers:
   Role of a Lawyer: The lawyer is the one who represents the accused or plaintiff in the court, and fights the case on their behalf. They can use the software for:
   a. The Lawyer can make an account where they can access their on-going as well as past cases.
   b. They can read about the past cases of other lawyers.
   c. They can find new cases that they can work on.
   d. They can keep a track of the cases they won and lost.
   e. They can add Plaintiff details onto the database whenever a new case is registered with them

4. Judges :

Role of Judge : The Judge gives the final verdict of the case. They can use the software for:

a. Can make an account where one can find about one's past cases.
b. Can keep a track of the on-going cases.
c. Can get the updates of previous and the upcoming hearings.
d. Can read about the past solved cases and the new cases.
e. A judge can access the various Documents Related to

5. #Government Statutory-The govt can use the database for-
    a. Checking all the cases whether pending or closed.
    b. The Government can search a particular case by number.
    c. The Government can see all cases of a Plaintiff.
    d. The Government can add a new case.
    e. The Government Can establish a new Court in the database

WEEK 4-5

Defining the Database Schema and Various constraints like Foreign Key, Primary Key, range Values etc. and Creating the Entity Relationship Diagram

Contribution: Each member contributed Equally and the work was done Together.

DataBase Schema

Person:

Person_ID	**Bigint(12) Not Null**
First_Name	varchar(30) Not Null
Last_Name	varchar(30)
Phone_Number	Bigint(11)
Gender	varchar(2)
Email	varchar(50)
City	varchar(50)
State	varchar(50)
Password	varchar(50)Not Null

Primary Key: Person_ID
Foreign Key: None

Login_Stats (Weak Entity):

Person_ID	**Bigint(12) Not Null**
Login_Date	**Date (Not Null)**
Login_Time	**DateTime (Not Null)**

Primary Key:N/A
Discriminant: Person_ID, Login_Date, Login_Time
Foreign Key: **Person_ID**

Lawyer:

Bar_Council_Number	**varchar(10) Not Null**
Person_ID	Bigint(12)(Not Null)
Category	varchar(20)(Not Null)

Primary Key: Bar_Council_Number
Foreign Key: Person_ID

Lawyer_Stats (Weak Entity):

Bar_Council_Number	**varchar(10) Not Null**
Wins	int(3)(Not Null)
Loss	int(3)(Not Null)
ActiveCases	int(2)(Not Null)

Primary Key: Bar_Council_Number
Foreign Key: Bar_Council_Number

FIGHTS:

Bar_Council_Number	varchar(10) Not Null
Case_ID	int(8) Not Null
Lawyer_Category	varchar(20) Not Null

Primary Key: Bar_Council_Number + Case_ID
Foreign Key: Case_ID, Bar_Council_Number

Plaintiff:

Plaintiff_ID	**varchar(12)** **Not null**
Person_ID	Bigint(12)

Primary Key: Plaintiff_ID
Foreign Key: Person_ID

FILES_Case:

Plaintiff_ID	varchar(12) Not Null
Case_ID	int(8) Not Null
Filing_Date	Date Not Null

Primary Key: Plaintiff_ID + Case_ID
Foreign Key: Plaintiff_ID, Case_ID

_Case:

Case ID	**int(8)** **Not Null**
_Status	varchar(20) (Not Null) (Active, On Hold, Complete)
Judge_Bar_Council_Number	varchar(10)Not Null

Primary Key: Case_ID

Foreign Key: Judge_Bar_Council_Number

Documents:

Document_ID	**varchar(5)** **Not Null**
Document_Category	varchar(20)(Not Null)
Document_Generation_Date	Date (Not Null)
Case_ID	int(8) Not Null

Primary Key: Document_ID

Foreign Key: Case_ID

Hearing Schedule(Weak Entity):

Case_ID	int(8) Not Null
_Date	DateTime
Start_Time	DateTime
End_Time	DateTime

Primary Key:N/A
Discriminant: Case_ID,_Date
Foreign Key: **Case_ID**

Accused (Weak Entity):

Accused_AADHAR_Number	Bigint(12) (Not Null)
Case_ID	int(8)
Charges	varchar(50)
Bar_Council_Number	varchar(10) Not Null

Primary Key: N/A
Discriminant:Accused_AADHAR_No,Case_ID
Foreign Key: Bar_Council_Number

Judge :

Judge_Bar_Council_ Number	**varchar(10) Not null**
Person_ID	Bigint(12)Not Null
Court_ID	int(5)Not Null

Primary Key: Judge_Bar_Council_Number
Foreign Key: COURT_ID, Person_ID

Court :

Court ID	int(5) (Not Null)
State	varchar(20) (Not Null)
City	varchar(20) (Not Null)
Category	varchar(20)(Not Null) (High Court, Supreme Court, District Court)
Address	Text

Primary key: Court ID
Foreign key: None

Police Official:

Badge_Number	**varchar(4)** **Not null**
Person_ID	Bigint(12)Not Null
Category	varchar(20)(Not Null) (Inspector, Constable)
Police_Station_ID	int(7) Not Null

Primary Key: Badge_Number
Foreign Key: Police_Station_ID, Person_ID

Investigates:

Case ID	**int(8)** **Not Null**
Badge_Number	**varchar(4)** **Not null**

Primary Key: Badge_Number + Case_ID
Foreign Key: Case_ID, Badge_Number

Police Station:

Police_Station_ID	**int(7) (Not Null)**
Jurisdiction_Area	varchar(30)(Not Null)
Address	Text
Police_Incharge(Badge Number)	varchar(20)Not Null

Primary Key: Police Station ID
Foreign Key: Police Incharge

FIR:

FIR_ID	**int(6) (Not Null)**
FIR_Date	DateTime
Category	varchar(20)Not Null
Description	Text
Police_Station_ID	int(7) Not Null
Badge_Number	varchar(4)Not Null
Plaintiff_ID	varchar(12)Not Null

Primary Key: FIR_ID , Foreign Key: Police_Station_ID, Badge_Number, Plaintiff_ID

FILES_FIR:

Plaintiff_ID	varchar(12) Not Null
FIR_ID	int(6) (Not Null)

Primary Key: FIR_ID
Foreign Key: Plaintiff_ID, FIR_ID

FIGHTS_FOR:

Plaintiff_ID	varchar(12) Not Null
Bar_Council_Number	varchar(12) (Not Null)

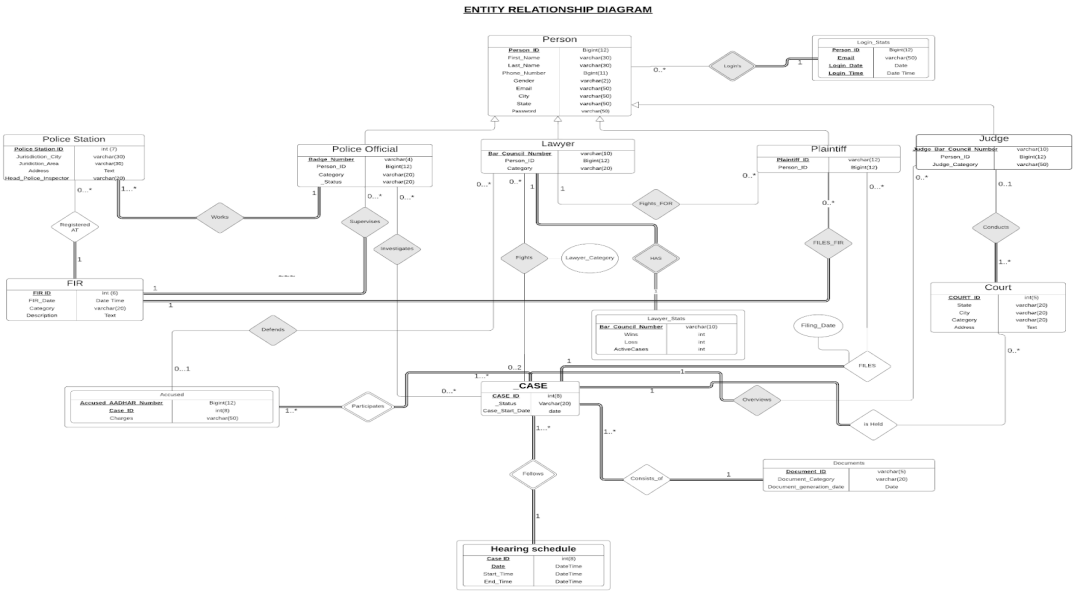
RELATIONSHIPS :

- Person ~~~ Login's ~~~ Login_Stats
  (one to many)
- Lawyer ~~~ Fights ~~~ Case
  (Many - Many)
- Lawyer ~~~ Fights ~~~ For Plaintiff
  (Many - Many)
- Lawyer ~~~ Defends ~~~ Accused
  (One - Many)
- Lawyer ~~~ Has ~~~ Lawyer_Stats
  (One - One)
- Judge ~~~ Overview ~~~ Case
  (One - Many)
- Judge ~~~ Conducts ~~~ Court
  (Many - One)
- Plaintiff ~~~ Sued/Files ~~~ Case
  (One - Many)
- Plaintiff ~~~ Files-FIR ~~~ FIR
  (One - Many)
- PoliceOfficial ~~~ Assists/Investigates ~~~ Case
  (Many - Many)
- PoliceOfficial ~~~ Works ~~~ Police Station
  (Many - one)
- PoliceOfficial ~~~ Supervises ~~~ FIR
  (One - Many)
- Case ~~~ is Held ~~~ Court
  (Many - One)

- Case ~~~ Follows ~~~ Hearing Schedule (One - Many)
- Case ~~~ Consists_of ~~~ Documents (One - Many)
- Accused ~~~ Participates ~~~ Case (Many - Many)
- FIR ~~~ Registered At ~~~ Police Station (Many - One)

Entity Relationship Diagrams:

To view ER Diagram see the link below:
Diagram:



Relationship Schema:

- Person (**Person_ID**, First_Name**,** Last_Name, Phone_Number, Gender, Email, City, State,Password)
- Login_Stats (**Person_ID**, Email, Login_Date, Login_Time)
- Lawyer(**Bar_Council_Number,** Person_ID, Category)
- Lawyer_Stats (**Bar_Council_Number**, Win, Loss, ActiveCases)
- Judge(**Judge_Bar_Council_Number,** Person_ID, Judge_Category,Court_ID)
- Police Official (**Badge_Number,** Person_ID, Category, _Status,Police_Station_ID)
- _Case (**Case_ID,** Case Description, _Status (Active/Inactive), Case_Start_Date,Judge_Bar_Council_Number,Court_ID)
- Documents (**Document_ID**, Document_Category, Document_generation_date,Case_ID)

- Plaintiff(**Plaintiff_ID,** Person_ID,Bar_Council_Number)
- Files (**Plaintiff_ID, Case_ID**, Filing_Date)
- Court (**Court_ID**, State, City, Category, Address)
- Police Station(**Police_Station_ID,** Jurisdiction_City, Jurisdiction_Area, Address, Head_Police_Inspector)
- FIR(**FIR_ID**, FIR_Date, Category, Description,Police_Station_ID,Badge_Number,Plaintiff_ID)
- Fights (**BAR_COUNCIL_Number**, **CASE_ID**, Lawyer_Category)
- Accused (**Accused_AADHAR_Number**, **Case_ID**, Charges,Bar_Council_Number)
- Hearing Schedule (**Case_ID**, **_Date**, Start_Time, End_Time)

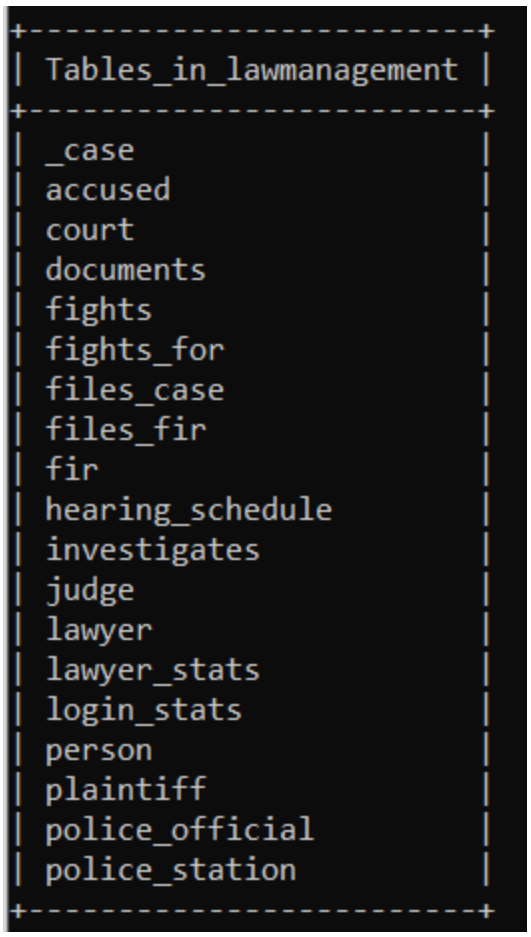
WEEK 5-6

Populating the Database with the Dummy Data from various websites

Work Distribution : DataBase schema Kartikey and Parag

Table Population: Avinoor Meet and Palak

Rest of the work was done Equally



Tables_in_lawmanagement
_case
accused
court
documents
fights
fights_for
files_case
files_fir
fir
hearing_schedule
investigates
judge
lawyer
lawyer_stats
login_stats
person
plaintiff
police_official
police_station

Person_ID	First_Name	Last_Name	Phone_Number	Gender	Email	City	State	Password
101927119606	Ezekiel	Priver	1838912219	Male	epriverp@miibeian.gov.cn	Seremban	Mizoram	3f9C68dRA84a
108256118568	Deana	Brandenberg	1875357738	Male	dbrandenberg1b@google.co.jp	Damatulan	Dehi	7t9A06gWW63u
120948927603	Tonie	Pecha	1025259420	Female	tpechaj@issuu.com	Port Blair	Andaman Islands	0b3J59kUI62p
124510386764	Wit	Raisbeck	9683280087	Male	wraisbeck6@mit.edu	Changkeng	Sikkim	3k3C42xSV32a
140085118175	Gwenette	Dodell	3556442116	Female	gdodellt@tinyurl.com	Hyderabad	Telanagana	1h4N69rRQ20j
140196159163	Ivie	Andrewartha	1445099013	Male	iandrewartha2g@163.com	Jayaraga Kaler	Ladakh	5c8Z35vTT98c
142174288221	Ole	Churching	5872813259	Male	ochurchingd@illinois.edu	Tikiw	Mizoram	5u7S72fAD17d
151781891068	Mel	Hardisty	5519875616	Male	mhardisty31@51.la	Leran	Sikkim	9i7C20bQT87h
160255019795	Susana	Physic	5111059775	Female	sphysic3p@themeforest.net	Houjiachuan	Nagaland	0g4H50eTB95i
172593408469	Dalston	Regglar	8026110056	Male	dregglar1q@about.me	Callao	J&K	7q6M55fKK13n
173965447459	Allard	Baskerfield	1526976617	Female	abaskerfield20@yandex.ru	Girihieum	West Bengal	1l8Y38eVT75i
179594872437	Siana	Brickham	2434378294	Male	sbrickham1k@whitehouse.gov	KÃ¸benhavn	Maharashtra	8v9L84oXR30w
183561809511	Gelya	Gaucher	1391730276	Female	ggaucherf@gmpg.org	Fortaleza	Odisha	5h3Z31nWT65s
183566652863	Domenico	Tattershaw	7079212958	Female	dtattershaw2i@privacy.gov.au	Al á¸¨arÃ?k	Goa	0x7Z39zMH89l
202044327958	Constantin	Turfes	9573993628	Female	cturfesk@sitemeter.com	Sukasada	Assam	1v1W84eBL66t
211062988452	Zackariah	Bricham	1571803935	Male	zbricham35@blog.com	Jizhou	Ladakh	4l3Z10lPG49v
226871529134	Arlie	Kellar	4449162570	Female	akellar2l@privacy.gov.au	Buesaco	Haryana	6o1K72xPO12q
229230583962	Stanton	Shears	6198123224	Male	sshears2e@springer.com	Myrzakent	Nagaland	2e7E37lGI09s
238376643934	Antonius	Gerge	2638722663	Female	agergex@psu.edu	Kudamatsu	Meghalaya	7b1A20nWI95r
243119759081	Silvan	Pozzo	5378107037	Male	spozzo23@topsy.com	Yueyang	Pondicherry	6k2Y67wFL10h
246290760409	Natka	Sweett	9798345644	Female	nsweett2a@zimbio.com	Shuiyuan	Maharashtra	0b1X16lUB53v
254396440719	Krista	Baurerich	7095196821	Female	kbaurerich1g@flickr.com	Varkaus	Bihar	0h9I93nNY27j
257327080492	Ardis	Chartman	4072363715	Female	achartman1s@japanpost.jp	Talisayan	Madhya Pradesh	2x6X40tNR25e
264858957560	Ardenia	Astman	9016143201	Female	aastman2m@scribd.com	Carson City	Kerala	4d0G99zQA25z
271395448669	Harley	Snaddon	1357086890	Male	hsnaddon3e@live.com	Shaoyang	Odisha	6g2S48rUY05h
275273313647	Basilio	Arzu	3459768858	Female	barzu1n@smugmug.com	ZeleneÃ?	Uttar Pradesh	3w9I35yXQ36r
284549233800	Leonore	Paz	7401175971	Male	lpaz1i@fema.gov	Higuerote	Mizoram	6r6L66hOJ60t
285922133360	Lucilia	Banville	9949520783	Female	lbanville2x@tinyurl.com	Saint-Bruno-de-Guigues	Mizoram	7r8X92vGY23z
295991678240	Jermayne	Dilloway	8006256345	Male	jdilloways@nsw.gov.au	Raipur	Chattisgarh	7b0X79jAX37l
310354632709	Brandise	Bednall	1235465282	Male	bbednall3t@eventbrite.com	Comrat	Sikkim	1a8C27mBN10k
310420996250	Jehu	Wesley	4677778409	Female	jwesley27@prnewswire.com	Chubek	Daman and Diu	7s3N86zJA01a
310847561489	Igor	Haydney	9203177580	Female	ihaydneyi@google.co.uk	Bhopal	Madhya Pradesh	5q6D34iGS76f
314804325712	Zola	Alaway	5735857077	Female	zalaway1@wunderground.com	Kemang	Telanagana	9m0S50fJD43n

Case_ID	_Date	Start_Time	End_Time
10789913	2020-05-09 00:00:00	15:00:00	19:00:00
10789913	2020-05-14 00:00:00	15:00:00	19:00:00
10789913	2020-05-21 00:00:00	15:00:00	19:00:00
11376825	2019-12-08 00:00:00	15:00:00	19:00:00
13213798	2020-03-26 00:00:00	15:00:00	19:00:00
13213798	2020-03-31 00:00:00	15:00:00	19:00:00
13213798	2021-04-10 00:00:00	15:00:00	19:00:00
13531749	2020-04-09 00:00:00	15:00:00	19:00:00
14269164	2019-05-15 00:00:00	15:00:00	19:00:00
14269164	2019-05-22 00:00:00	15:00:00	19:00:00
14269164	2019-05-31 00:00:00	15:00:00	19:00:00
14696518	2020-06-27 00:00:00	15:00:00	19:00:00
14696518	2020-07-08 00:00:00	15:00:00	19:00:00
14696518	2021-07-20 00:00:00	15:00:00	19:00:00
15353682	2020-08-29 00:00:00	15:00:00	19:00:00
15353682	2020-09-09 00:00:00	15:00:00	19:00:00
15353682	2021-09-20 00:00:00	15:00:00	19:00:00
16084001	2019-03-07 00:00:00	15:00:00	19:00:00
16084001	2019-03-09 00:00:00	15:00:00	19:00:00

Data Definition Language:

```
create database LawManagement;
use LawManagement;
create table Person
(       Person_ID bigint(12) NOT NULL,
        First_Name varchar(30) NOT NULL,
        Last_Name varchar(30),
        Phone_Number bigint(11) Not Null,
        Gender enum("Male","Female") Not Null ,
        Email varchar(50) Not Null,
        City varchar(50) Not Null,
        State varchar(50) Not Null,
        Password varchar(50) NOT NULL,
        Primary Key (Person_ID)
);
create table Login_Stats
(       Person_ID bigint(12) NOT NULL,
        Login_Date date NOT NULL,
        Login_Time time NOT NULL,
    Primary Key(Person_ID,Login_Date,Login_Time),
        Foreign Key(Person_ID) references Person(Person_ID)
    On update cascade
    On delete Cascade
);
create table Lawyer
(       Bar_Council_Number varchar(10) NOT NULL,
```

```
        Person_ID bigint(12) NOT NULL,
        Category varchar(20) NOT NULL,
        Primary key (Bar_Council_Number)

);
create table Lawyer
(       Bar_Council_Number varchar(10) NOT NULL,
        Person_ID bigint(12) NOT NULL,
        Category varchar(20) NOT NULL,
        Primary key (Bar_Council_Number),
    Foreign Key(Person_ID) references Person(Person_ID)
    On Update Cascade
    On delete Cascade

);
create table Lawyer_Stats
(       Bar_Council_Number varchar(10) NOT NULL,
        Wins int(3) NOT NULL,
        Loss int(3) NOT NULL,
        ActiveCases int(2) NOT NULL,
        Primary key(Bar_Council_Number),
        Foreign key(Bar_Council_Number) references
Lawyer(Bar_Council_Number)
    On Update Cascade
    On Delete Cascade
);
```

```
 create table Court
(       Court_ID int(5) NOT NULL,
        State varchar(20) Not NUll,
        City varchar(20) NOT NULL,
        Category varchar(20) NOT NULL,
        Address text,
        Primary key(Court_ID)
);
create table Judge
(       Judge_Bar_Council_Number varchar(10) NOT NULL,
        Person_ID bigint(12) NOT NULL,
        Court_ID int(5) NOT NULL,
        Primary key(Judge_Bar_Council_Number),
        Foreign key(Court_ID) references Court(Court_ID)
    on Update Cascade
    On delete Cascade,
        Foreign key(Person_ID) references Person(Person_ID)
    On Update Cascade
    On delete Cascade
);
create table Police_Station
(       Police_Station_ID int(7) NOT NULL,
        Jurisdiction_Area varchar(30) Not Null,
        Address text Not Null,
        Police_Incharge varchar(20) NOT NULL,
        Primary Key(Police_Station_ID)
);
```

```
create table Police_Official
(       Badge_Number varchar(4) NOT NULL,
        Person_ID bigint(12) NOT NULL,
        Category enum
("Constable","SubInspector","Inspector") NOT NULL,
        Police_Station_ID int(7) NOT NULL,
        Primary Key(Badge_Number),
        Foreign Key(Police_Station_ID) references
Police_Station(Police_Station_ID)
    On Update Cascade
    On delete Cascade,
        Foreign key(Person_ID) references Person(Person_ID)
    On Update Cascade
    On delete Cascade
);
Alter table Police_Station Add Constraint Foreign Key
Police_Station(Police_Incharge) references
Police_Official(Badge_Number) on update Cascade On
delete Cascade;
```

```
create table _Case
(	Case_ID int(8) NOT NULL,
	_Status varchar(20) NOT NULL,
	Judge_Bar_Council_Number varchar(10) NOT NULL,
	Primary key(Case_ID),
	Foreign key(Judge_Bar_Council_Number) references
Judge(Judge_Bar_Council_Number)
   On Update Cascade
   On delete Cascade
);

 create table Documents
(	Document_ID varchar(5) NOT NULL,
	Document_Category varchar(20) NOT NULL,
	Document_Generation_Date date NOT NULL,
	Case_ID int(8) NOT NULL,
	Primary key(Document_ID),
	Foreign Key(Case_ID) references _Case(Case_ID)
   on Update Cascade
   On delete Cascade
);
create table Accused
(	Accused_AADHAR_Number bigint(12) NOT NULL,
	Case_ID int(8),
	Charges varchar(50),
	Bar_Council_Number varchar(10) NOT NULL,
```

```
        Foreign key(Bar_Council_Number) references
Lawyer(Bar_Council_Number)
    on Update Cascade
    on delete Cascade,
    Foreign Key(Case_ID) references _Case(Case_ID)
    on Update Cascade
    on delete Cascade
);
create table Fights
(       Bar_Council_Number varchar(10) NOT NULL,
        Case_ID int(8) NOT NULL,
        Lawyer_Category enum("Claimant","Defendant") Not
Null,
        Primary key(Bar_Council_Number,Case_ID),
        Foreign key(Case_ID) references _Case(Case_ID)
    on Update Cascade
    on delete Cascade,
        Foreign key(Bar_Council_Number) references
Lawyer(Bar_Council_Number)
    on update cascade
    on delete cascade
);
create table Plaintiff
(       Plaintiff_ID varchar(12) NOT NULL,
        Person_ID bigint(12),
        Primary key(Plaintiff_ID),
        Foreign key(Person_ID) references Person(Person_ID)
```

```
    on update cascade
    on delete cascade
);
create table Files_Case
(       Plaintiff_ID varchar(12) NOT NULL,
        Case_ID int(8) NOT NULL,
        Filing_Date date NOT NULL,
        Primary key(Plaintiff_ID,Case_ID),
        Foreign key(Plaintiff_ID) references
Plaintiff(Plaintiff_ID)
    on update cascade
    on delete cascade,
        Foreign key(Case_ID) references _Case(Case_ID)
    on delete cascade
    on update cascade
);
create table FIR
(       FIR_ID int(6) NOT NULL,
        FIR_Date datetime,
        Category varchar(20) NOT NULL,
        Description text,
        Police_Station_ID int(7) NOT NULL,
        Badge_Number varchar(4) NOT NULL,
        Primary key(FIR_ID),
        Foreign key(Police_Station_ID) references
Police_Station(Police_Station_ID)
    on update cascade
```

```
    on delete cascade,
        Foreign key(Badge_Number) references
Police_Official(Badge_Number)
    on update cascade
    on delete cascade
);
create table Files_FIR
(       Plaintiff_ID varchar(12) NOT NULL,
        FIR_ID int(6) NOT NULL,
    Foreign Key (FIR_ID) references FIR(FIR_ID)
    on delete cascade
    on update cascade,
    Foreign Key (Plaintiff_ID) references
Plaintiff(Plaintiff_ID)
    on delete cascade
    on update cascade
);
create table Investigates
(       Case_ID int(8) NOT NULL,
        Badge_Number varchar(4)NOT NULL,
        Primary key(Badge_Number,Case_ID),
        Foreign key(Case_ID) references _Case(Case_ID)
    on delete Cascade
    on update cascade,
        Foreign key(Badge_Number) references
Police_Official(Badge_Number)
    on delete cascade
```

```
    on update cascade
);
create table Fights_For
(
        Plaintiff_ID varchar(12) Not Null,
    Bar_Council_Number varchar(10) NOT NULL,
    Foreign Key(Plaintiff_ID) references
Plaintiff(Plaintiff_ID)
    on delete Cascade
    on update Cascade,
    Foreign Key(Bar_Council_Number) references
Lawyer(Bar_Council_Number)
    on delete cascade
    on update cascade
);
create table Hearing_Schedule
(       Case_ID int(8) NOT NULL,
        _Date datetime,
        Start_Time time,
        End_Time time,
    Primary Key(Case_ID,_Date),
        Foreign key(Case_ID) references _Case(Case_ID)
    on Update Cascade
    on delete Cascade
);

#Indexing
```

create index idx_case on _case(_status, Case_ID);
create index idx_accused on accused(Case_ID, charges);
create index idx_judge on
judge(Judge_Bar_Council_Number);
create index idx_sched on hearing_schedule(Case_ID);
create index idx_person on person(Person_ID);
create index idx_plaintiff on plaintiff(Plaintiff_ID);
create index idx_filecase on files_case(Plaintiff_ID);
create index idx_lawyer on lawyer(Person_ID, category);

Local Constraints:

1. FIR DATE>=REGISTRATION DATE && FIR DATE<=LAST LOGIN DATE
2. JUDGE REG DATE<=CASE START DATE
3. LAWYER REG DATE<=CASE START DATE
4. Foreign Keys are kept consistent throughout the databases.
5. The primary key of different tables are kept unique among themselves
6. Accused Lawyer and Plaintiff Lawyer cannot be same
7. POLICE OFFICIAL REG DATE<=FIR DATE
8. Police Station incharge can be only be a designated as Inspector
9. The police officers allotted to a particular and uniques police station

10. Aadhar of the plaintiff and defendant are unique amongst themselves.

Assumptions

1. One court has one judge
2. Everybody has their AADHAR Number whether it's the plaintiff or the accused
3. Each Police Station is headed by a single police official
4. There are are two lawyers involved in a case, one from the defendant and other being the prosecutor
5. Accused does not register on portal instead the plaintiff files the details as in against whom the case is being registered
6. FIR and Cases are independent
7. Each case is investigated by a single police officer and each FIR is also monitored by a single police official
8. FIR could be registered for certain categories only
9. Case Statement is user input and data filled is dummy data.

Note:The relationships,tables and the data they contain are dummy and tentative to change during the course of the project.

Queries for Judge

#1) **To get the number of cases that are active and completed in each type of charges**

```
select T1.Charges, count(*) as Frequency from (select *
from _case where _status="Active")
as T2 inner join Accused as T1 on (T2.Case_ID =
T1.Case_ID) group by T1.charges;
```

```
select T1.Charges, count(*) as Frequency from (select *
from _case where _status="Completed")
as T2 inner join Accused as T1 on (T2.Case_ID =
T1.Case_ID) group by T1.charges;
```

```
select T1.charges, count(*) as Frequency from Accused
as T1, _Case as T2 where T1.Case_ID=T2.Case_ID and
T2._Status="Active" group by T1.Charges order by 2
Desc;
```

#2) **Print all the courts that have more than one judge with Respective Judge;**

```
select  distinct T1.Judge_Bar_Council_Number,
T1.Court_ID from judge as T1, judge as T2
where T1.Court_ID=T2.Court_ID and
T1.Judge_Bar_Council_Number!=T2.Judge_Bar_Council_
Number;
```

#3) **To extract the hearing schedule of a Judge within the next 40 days.**

```
select T1.Case_ID,T1.Judge_Bar_Council_Number,
T2._date,T2.Start_Time,T2.End_Time from _case
as T1,hearing_schedule as T2 where
T1.case_id=t2.case_id and T1._status="Active" and
T2._date
between CURDATE() and DATE_ADD(CURDATE(),
INTERVAL 40 DAY);
```

#4) **Details of Name of the Judges with number of active and completed cases**

```
 Select
T4.first_name,T4.last_name,T4.judge_bar_council_numbe
r, T5._status, T5. Frequency from
 (select T2.first_name,T2.last_name,
T3.Judge_Bar_Council_Number from person as T2, Judge
as T3 where T2.person_ID=t3.person_ID) as T4,
 (select T1.Judge_Bar_Council_Number, T1._status,
count(*) as Frequency from _case as T1 group by
T1.Judge_Bar_Council_Number, T1._status ) as T5
where
T4.Judge_Bar_Council_Number=T5.Judge_Bar_Council_
Number;
```

#5) **Tracking the most efficient Judge with max(Completed/active) ratio.**

```
select T3.Judge_Bar_Council_Number ,
(T3._completed/T3._Active) as Ratio from (select
T1.Judge_Bar_Council_Number, T1._Active,
T2._Completed from (select Judge_Bar_Council_Number,
count(*) as _Active from _case where _status="Active"
group by Judge_Bar_Council_Number,_status) as T1
inner join
(select Judge_Bar_Council_Number, count(*) as
_Completed from _case where _status="Completed"
group by Judge_Bar_Council_Number,_status) as T2
on(T1.Judge_Bar_Council_Number=T2.Judge_Bar_Coun
cil_Number)) as T3 order by (T3._Completed/T3._Active)
desc;
```

Plaintiff queries :

```
# Display details of active cases of a plaintiff (Plaintiff ID is
given as an input)
select _case.Case_ID, _case._Status,
accused.Accused_AADHAR_Number, accused.Charges
from _case inner join accused on _case.Case_ID =
```

```
accused.Case_ID where _case.Case_ID in (select
Case_ID from files_case where Plaintiff_ID =
"353418651411") and _case._Status = "Active";

# Display number of cases of a plaintiff based on case
status (Plaintiff ID is given as an input)
select _case._Status, count(Files_Case.Case_ID) from
Files_Case inner join _case on _case.Case_ID =
Files_Case.Case_ID where Files_Case.Plaintiff_ID =
"353418651411" group by _case._status;

# Display details of lawyers with maximum wins in every
category
select lawyer.*, person.First_Name,
person.Phone_Number, person.State from (Lawyer_Stats
inner join lawyer on Lawyer_Stats.Bar_Council_Number =
lawyer.Bar_Council_Number) inner join person on
person.Person_ID = lawyer.Person_ID  where Wins in
(select max(Lawyer_Stats.wins) from Lawyer_Stats inner
join lawyer on lawyer.Bar_Council_Number =
Lawyer_Stats.Bar_Council_Number group by
lawyer.category);

# Details of lawyer based on category and bar council
(Either of the two is input)
select lawyer.Bar_Council_Number, person.First_Name,
person.Phone_Number, person.State from lawyer inner
```

```
join person on person.Person_ID = lawyer.Person_ID
where lawyer.Category = "Divorce";

select lawyer.Category, person.First_Name,
person.Phone_Number, person.State from lawyer inner
join person on person.Person_ID = lawyer.Person_ID
where lawyer.Bar_Council_Number = "6787856943";

# hearing schedule of active cases of a particular plaintiff
(Plaintiff ID is given as an input)
select hearing_schedule.* from hearing_schedule inner
join _case on hearing_schedule.Case_ID =
_case.Case_ID where _case.Case_ID in (select Case_ID
from Files_Case where Plaintiff_ID = "353418651411")
and _case._status = "Active";

# update docs (Inputs reqd)
update documents set Document_Category = " " where
Case_ID = " ";
```

#String Operations(like)

```
Select * from accused where charges like '%Tax Fraud%';
```

#Percentage of Win/loss of a Lawyer(**input reqd)

```
Select * from lawyer_stats having (win/loss)>'some value';
```

#Check Workload of a lawyer cases(** input reqd)

Select * from lawyer_stats having ActiveCases>’2’;

#Display details of a lawyer who represent some plaintiff

select * from lawyer left join fights_for on lawyer.Bar_Council_Number=fights_for.Bar_Council_Number;

#Find out whether the lawyer is claimant/Defendant

select * from fights natural join lawyer where Lawyer_Category='Claimant';

#list out how many lawyers are claimant/defendant

select count(Bar_Council_Number) from fights natural join lawyer where Lawyer_Category='Defendant';

#***check how many instances of a crime in FIR***

select count(Category) from fir where Category like '%Drunk%';

#**grant privileges to judge on hearing schedule(***)**

Police Station Queries:

#check the category of Police official

Select Badge_number from police_official where category=’Constable,Inspector’;

#check which person is a Police Official
Select Person_ID from police_official natural join person;

#which case is being investigated by a police officer
select investigates.Case_ID,police_official.Badge_Number from investigates left join police_official on investigates.Badge_Number=police_official.Badge_Numb er;

#Which Police_Official is placed in which station
Select * from police_official natural join police_station

#Display details of a police_official(personal details)
Select * from police_official natural join person;

#**revoke privileges(****)**

#investigates

```
select Case_ID from investigates where
Badge_Number=1014;

#Fir
select FIR_ID from FIR where Badge_Number=1014;

#works
select T1.Case_ID,T1.Badge_Number,
T2._date,T2.Start_Time,T2.End_Time from investigates
as T1,hearing_schedule as T2 where
T1.case_id=t2.case_id and T1._status="Active" and
T2._date
between CURDATE() and DATE_ADD(CURDATE(),
INTERVAL 40 DAY);

select _case.Case_ID from _case join investigates on
_case.case_id=investigates.case_id where
_Status="Active" and badge_number=1001;

#hearing schedule
select * from hearing_schedule where case_id in
(select _case.Case_ID from _case join investigates on
_case.case_id=investigates.case_id where
_Status="Active" and badge_number=1001);

select badge_number,category,police_station_id from
police_official where police_station_id=1000001;
```

```
select badge_number,category,police_station_id from
police_official where police_station_id=1000001 and
Category="Inspector";

select police_station_id,Count(*) as Total_Officials
from police_official group by police_station_id;
#agar kisi ko dekhna ho mere police area mein
category wise fir
select FIR_ID,Description from police_official as a join
fir as b on a.police_station_id=b.police_station_id
where a.badge_number=1001 and b.category="Sexual
Abuse";
#agar active or completed cases

#active cases jinki specific category hai by a special
police man
select * from _Case where case_id in (select
accused.case_id from accused join investigates on
accused.case_id=investigates.case_id and
badge_number=1001 and charges="Embezzlement")
and _status="Active";

#completed cases jinki specific category hai by a
special police man
select * from _Case where case_id in (select
accused.case_id from accused join investigates on
```

```
accused.case_id=investigates.case_id and
badge_number=1001 and charges="Embezzlement")
and _status="Completed";

select _status,count(*) as Total_Cases from _Case
where case_id in (select accused.case_id from
accused join investigates on
accused.case_id=investigates.case_id and
badge_number=1001 and charges="Embezzlement")
group by _status;

#total cases category wise by a policeman
 (select count(*) as Total_Cases,charges from accused
join investigates on
accused.case_id=investigates.case_id and
badge_number=1001 group by charges)

#category wise completed
select _status,count(*) as Total_Cases from _Case
where case_id in (select count(*) as
Total_Cases,charges from accused join investigates
on accused.case_id=investigates.case_id and
badge_number=1001 group by charges) group by
_status;


#jurisdiction areas with fir count >3
```

select Jurisdiction_Area,a.Total_Cases from (select police_station_id,count(*) as Total_Cases from fir group by police_station_id) as a join police_station as b on a.police_station_id=b.police_station_id having total_cases>3 order by Total_cases desc ;

#if lawyer wants to find the frequency of policemen in the completed cases i.e most efficient policemen

#police verification

#police wants to see the accused in case

#plaintiff wants to meet police

#grant

#views
create view case_sched as select 'hearing_schedule.case_ID', 'hearing_schedule._date', 'case._status' from hearing_schedule inner join _case on hearing_schedule.Case_ID = _case.Case_ID;

create view case_accused as select 'accused.Accused_AADHAR_number','accused.case_ID', 'accused.charges', 'case._status' from accused inner join _case on accused.Case_ID = _case.Case_ID;

create view case_investigate as select '_case.case_ID', 'investigates.badge_number', 'case._status' from investigates inner join _case on investigates.Case_ID = _case.Case_ID;

1. **Case aur accused**
2. **Case n hearing sched**
3. **#Accused aur investigates**
4. **Case aur Investigates**
5. **#Lawyer aur Fights**

#indexing

#queries (having,order by,group by,join,in,as,and where,string operations,aliases,set operation [union,intersections],aggr func[min,max,count,sum,group by],nested subqueries,check,optional:group by cube)

#doc theek

#Embedded Sql

1) **Group by maximum engagement in the database and minimum so that the software could be made more accessible to other people.**
2)

#

Lawyer:
Lawyer:view,
Accused:view,
#Documents:view
Hearing schedule:view
Court:view
Judge:view
Police official:view
Police station:view
Plaintiff:view
Case:view
Fights:view
Fights_for:view
Files_case:view
Investigates:view
lawyer_Stats:view

Police official:
Lawyer:view,
Accused:view,
#Documents:edit,view
Hearing schedule:view
Court:view
Judge:view
Police official:view
Police station:view
Plaintiff:view
Case:view
Fights:view
Fights_for:view
Files_case:view
Investigates:view
lawyer_Stats:view
Fir:edit
Files_fir:view

Plaintiff
Lawyer:view,
Accused:view,
Documents:view
Hearing schedule:view
Court:view
Judge:view
Police official:view

Police station:view
Plaintiff:view
Case:view
Fights:view
Fights_for:view
Files_case:view
Investigates:view
lawyer_Stats:view