



CS 353

DATABASE MANAGEMENT SYSTEMS PROJECT

Project Design Report

National Judiciary Informatics System

05.04.2018

Group 22

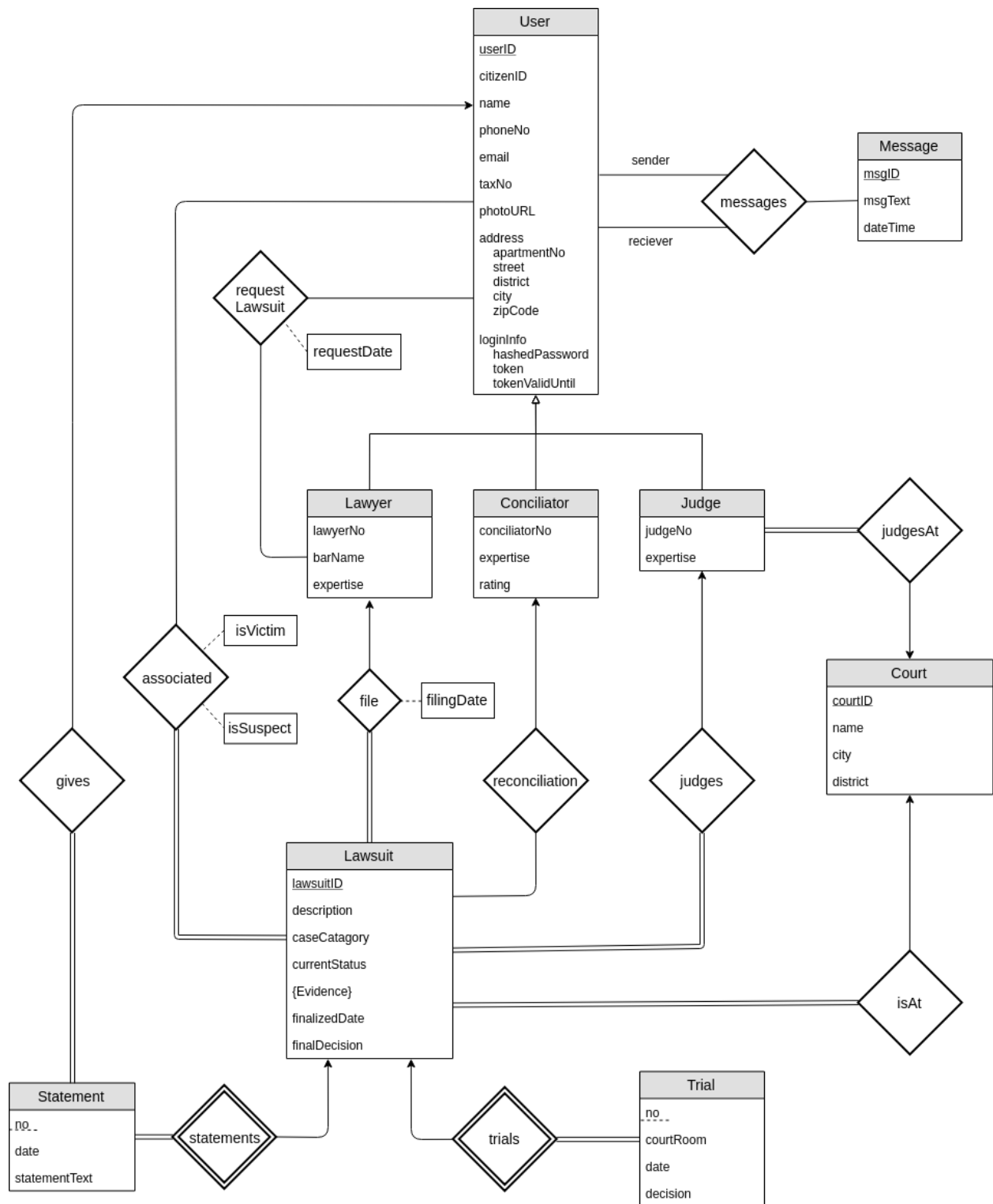
Ali Yümsel - 21601841
Sina Şahan - 21602609
Ömer Fatih Çelik - 21601387
Faruk Ege Hatırnaz - 21601441

Table of Contents

1. REVISED E/R DIAGRAM	3
2. RELATION TABLE SCHEMAS	4
2.1 USER	4
2.2 LAWYER	4
2.3 CONCILIATOR	4
2.4 JUDGE	4
2.5 COURT	4
2.6 LAWSUIT	4
2.7 LAWSUIT EVIDENCE	5
2.8 MESSAGE	5
2.9 TRIAL	5
2.10 STATEMENT	5
2.11 RECONCILIATION	5
2.12 ASSOCIATED	6
2.13 REQUEST LAWSUIT	6
2.14 MESSAGES	6
3. FUNCTIONAL COMPONENTS	7
3.1 USE CASES / SCENARIOS	7
3.1.1 Administrator	7
3.1.2 Citizen	7
3.1.3 Lawyer	7
3.1.4 Judge	8
3.1.5 Conciliator	8
3.2 ALGORITHMS	10
3.2.1 Filing a lawsuit (by a citizen or lawyer)	10
3.2.2. Deciding on a lawsuit (by a judge)	10
3.2.3 Reconciliation (by a conciliator)	10
3.3 DATA STRUCTURES	10
4. UI DESIGN AND SQL STATEMENTS	11
4.1 LOGIN PAGE	12
4.2 SIGN UP PAGE	13
4.3 JUDGE'S WELCOME PAGE	14
4.4 MANAGING ACCOUNT PAGE	15
4.5 SIGN OUT PAGE	16
4.6 MESSAGE PAGE	17
4.7 NOTIFICATION PAGE	18
4.8 ASSIGNED CONCILIATORS	19

4.9 ASSIGN A CONCILIATOR	20
4.10 SORTING IN CASES	21
4.11 TRIALS (EXAMPLE OF JUDGES)	22
4.12 LAWYER'S WELCOME PAGE	23
4.13 LAWYER'S POSSIBLE CASES	24
4.14 FORMING A LAWSUIT	25
4.15 USER'S WELCOME PAGE	26
4.16 REQUEST A LAWSUIT	27
5. ADVANCED DATABASE COMPONENTS	29
5.1 VIEWS	29
5.1.1 RELATED LAWSUITS VIEW	29
5.1.2 JUDGE'S ONGOING LAWSUITS VIEW	29
5.1.3 TOP RATED CONCILIATORS	29
5.2 REPORTS	29
5.2.1 LAST 30-DAY OPEN CASE REPORT:	29
5.2.2 LAST 30-DAY TRIAL REPORT:	30
5.3 TRIGGERS	30
5.4 CONSTRAINTS	30
6. IMPLEMENTATION PLAN	31
7. WEBSITE	31

1. REVISED E/R DIAGRAM



2. RELATION TABLE SCHEMAS

2.1 USER

Relational Model

User(userID, citizenID, name, phoneNo, email, taxNo, photoURL, apartmentNo, street, district, city, zipCode)

Candidate Keys

{ (userID), (citizenID), (phoneNo), (email), (taxNo) }

Attribute Domains

userID:	INT, NOT NULL, UNIQUE
citizenID:	INT, NOT NULL, UNIQUE
name:	VARCHAR, NOT NULL
phoneNo:	VARCHAR, NOT NULL, UNIQUE
email:	VARCHAR, NOT NULL, UNIQUE
taxNo:	INT, NOT NULL, UNIQUE
photoURL:	VARCHAR
apartmentNo:	VARCHAR
street:	VARCHAR
district:	VARCHAR
city:	VARCHAR
zipCode:	VARCHAR

2.2 LAWYER

Relational Model

Lawyer(userID, lawyerNo, barName, expertise)

FK: userID references User

Candidate Keys

{ (userID), (lawyerNo) }

Attribute Domains

userID: INT, NOT NULL, UNIQUE

lawyerNo: INT, NOT NULL, UNIQUE

barName: VARCHAR, NOT NULL

expertise: VARCHAR

2.3 CONCILIATOR

Relational Model

Conciliator(userID, conciliatorNo, expertise, rating)

FK: userID references User

Candidate Keys

{ (userID), (conciliatorNo) }

Attribute Domains

userID: INT, NOT NULL, UNIQUE

conciliatorNo: INT, NOT NULL, UNIQUE

expertise: VARCHAR

rating: INT

2.4 JUDGE

Relational Model

Judge(userID, judgeNo, expertise, courtID)

FK: userID references User

FK: courtID references Court

Candidate Keys

{ (userID), (judgeNo) }

Attribute Domains

userID: INT, NOT NULL, UNIQUE

judgeNo: INT, NOT NULL, UNIQUE

expertise: VARCHAR

courtID: INT, NOT NULL, UNIQUE

2.5 COURT

Relational Model

Court(courtID, name, city, district)

Candidate Keys

{(courtID)}

Attribute Domains

courtID: INT, NOT NULL, UNIQUE

name: VARCHAR, NOT NULL

city: VARCHAR, NOT NULL

district: VARCHAR, NOT NULL

2.6 LAWSUIT

Relational Model

Lawsuit(lawsuitID, description, filingDate, caseCatagory, currentStatus, finalizedDate, finalDecision, courtID, judgeID, filedLawyerID)

FK: courtID references Court

FK: judgeID references Judge(userID)

FK: filedLawyerID references Lawyer(userID)

Candidate Keys

{ (lawsuitID) }

Attribute Domains

lawsuitID: INT, NOT NULL, UNIQUE

description: VARCHAR

filingDate: DATETIME, NOT NULL

caseCatagory: VARCHAR, NOT NULL

currentStatus: INT, NOT NULL

finalizedDate: DATETIME

finalDecision: VARCHAR

courtID: INT, NOT NULL, UNIQUE

judgeID: INT, NOT NULL, UNIQUE

filedLawyerID: INT, NOT NULL, UNIQUE

2.7 LAWSUIT EVIDENCE

LawsuitEvidence(lawsuitID, evidenceNo, evidenceDescription)

FK: lawsuitID references Lawsuit

Candidate Keys

{ (lawsuitID, evidenceNo) }

Attribute Domains

lawsuitID: INT, NOT NULL, UNIQUE

evidenceNo: INT, NOT NULL

evidenceDescription: VARCHAR

2.8 MESSAGE

Relational Model

Message(msgID, msgText, dateTime)

Candidate Keys

{ (msgID) }

Attribute Domains

msgID: INT, NOT NULL, UNIQUE

msgText: VARCHAR, NOT NULL

dateTime: DATETIME, NOT NULL

2.9 TRIAL

Relational Model

Trial(lawsuitID, no, date, courtRoom, decision)

FK: lawsuitID references Lawsuit

Candidate Keys

{ (lawsuitID, no), (lawsuitID, date) }

Attribute Domains

lawsuitID: INT, NOT NULL, UNIQUE

no: INT, NOT NULL

date: DATETIME, NOT NULL

courtRoom: VARCHAR, NOT NULL

decision: VARCHAR, NOT NULL

2.10 STATEMENT

Relational Model

Statement(lawsuitID, no, date, statementText, userID)

FK: lawsuitID references Lawsuit

FK: userID references User

Candidate Keys

{ (lawsuitID, no) }

Attribute Domains

lawsuitID: INT, NOT NULL, UNIQUE

no: INT, NOT NULL

date: DATETIME, NOT NULL

statementText: VARCHAR, NOT NULL

userID: INT, NOT NULL, UNIQUE

2.11 RECONCILIATION

Relational Model

reconciliation = (lawsuitID, conciliatorID)

FK: lawsuitID references Lawsuit

FK: conciliatorID references Conciliator(userID)

Candidate Keys

{ (lawsuitID, conciliatorID) }

Attribute Domains

lawsuitID: INT, NOT NULL, UNIQUE

conciliatorID: INT, NOT NULL, UNIQUE

2.12 ASSOCIATED

Relational Model

associated = (lawsuitID, userID, isVictim, isSuspect)

FK: lawsuitID references Lawsuit

FK: userID references User

Candidate Keys

{lawsuitID, userID}

Attribute Domains

lawsuitID: INT, NOT NULL, UNIQUE

userID: INT, NOT NULL, UNIQUE

2.13 REQUEST LAWSUIT

Relational Model

requestLawsuit = (userID, lawyerID, requestDate)

FK: userID references User(userID)

FK: lawyerID references Lawyer(userID)

Candidate Keys

{userID, lawyerID, requestDate}

Attribute Domains

userID: INT, NOT NULL, UNIQUE

lawyerID: INT, NOT NULL, UNIQUE

requestDate: DATETIME, NOT NULL, UNIQUE

2.14 MESSAGES

Relational Model

messages = (senderID, recieverID, msgID)

FK: senderID references User(userID)

FK: recieverID references User(userID)

FK: msgID references Message

Candidate Keys

{senderID, recieverID, msgID}

Attribute Domains

senderID: INT, NOT NULL, UNIQUE

receiverID: INT, NOT NULL, UNIQUE

msgID: INT, NOT NULL, UNIQUE

3. FUNCTIONAL COMPONENTS

3.1 USE CASES / SCENARIOS

3.1.1 Administrator

Upgrade citizen accounts: Admins can upgrade citizen accounts to lawyer, judge or conciliator accounts.

Downgrade special accounts: Admins can downgrade lawyer, judge or conciliator accounts to citizen accounts.

View tables: Admins can view all the information in the system

Edit tables: Admins can edit all the information in the system. They can manually add and delete tuples in tables.

3.1.2 Citizen

Create an account: Citizens must create an account to participate in system

Login to system: Citizens must login to participate in system

Edit personal information: Personal information like contact information can be changed by citizens.

Open a lawsuit: Court, suspects, lawsuit type and claim should be given.

View lawsuits related to them: Citizens can see all information about lawsuits related to them, including trials of the lawsuit.

Accept an assigned lawyer: Lawyers assigned to a citizen have to be confirmed to be effective in system.

View a citizen's related lawsuits: The information about these lawsuits will be restricted.

3.1.3 Lawyer

Perform a citizen use case: Lawyers are also a citizen, hence have their use cases included.

Assign to a citizen: Lawyers can add clients.

Open a lawsuit for a client: They can open a lawsuit in name of their client.

View clients' lawsuits: They can see information about their clients' lawsuits, including trials.

3.1.4 Judge

Perform a citizen use case: Judges are also a citizen, hence have their use cases included.

Give a decision to an assigned lawsuit: Judges can finalize a lawsuit.

Setup a trial for an assigned lawsuit: Judges can create a new trial for their assigned lawsuits.

Cancel arranged trial: Judges can cancel a trial they arranged.

Archive trial: Judges will enter the statements of stakeholders in a trial

Assign a conciliator: Judges will be able to assign a conciliator to some type of lawsuits they are assigned to.

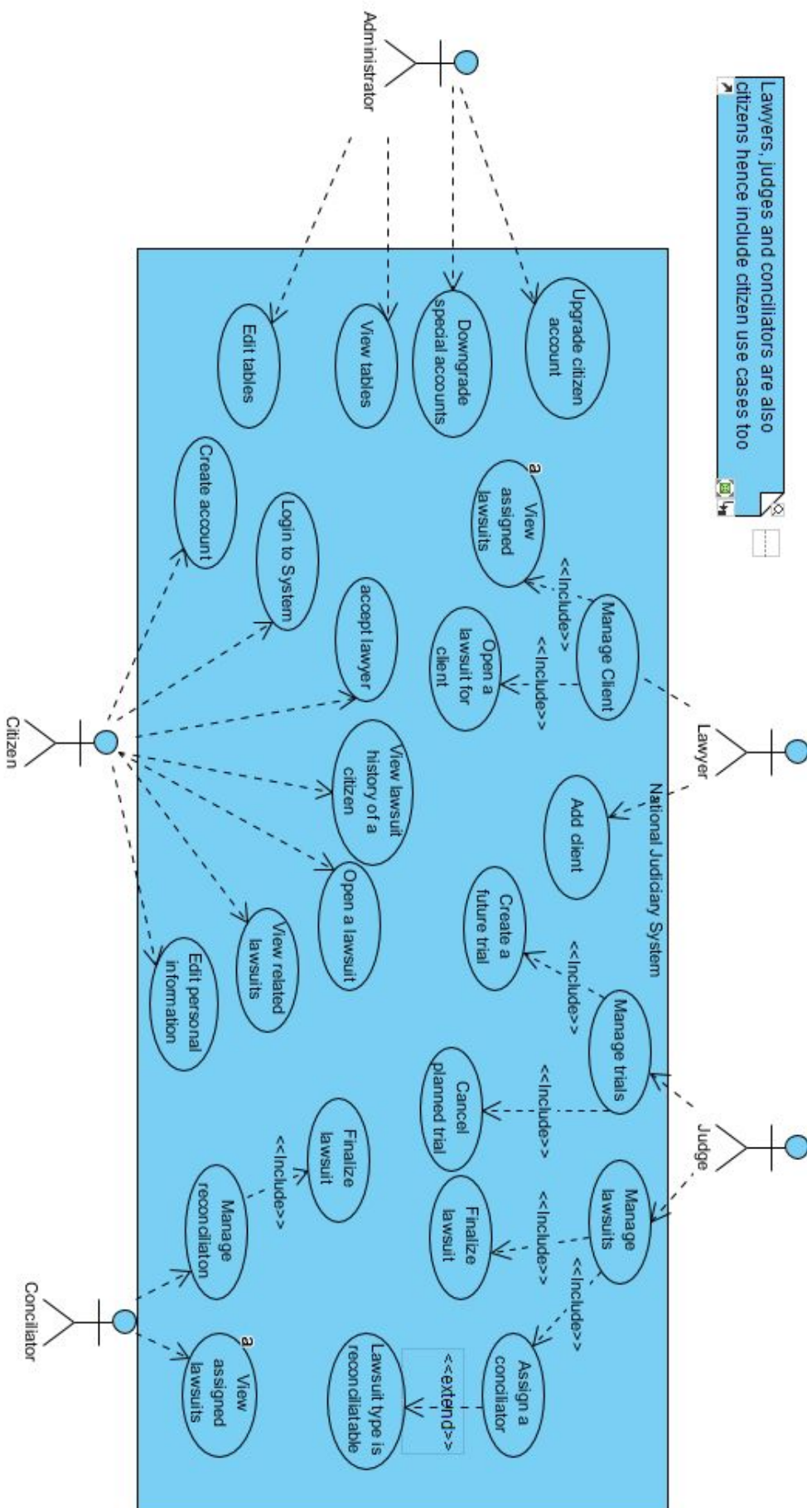
3.1.5 Conciliator

Perform a citizen use case: Conciliators are also a citizen, hence have their use cases included.

View cases assigned: Conciliators can see their assigned cases.

Reconciliate a case: Conciliators can finalize a lawsuit they are assigned by entering agreed terms from both parties.

Lawyers, Judges and conciliators are also citizens hence include citizen use cases too



3.2 ALGORITHMS

3.2.1 Filing a lawsuit (by a citizen or lawyer)

Any citizen can open a lawsuit against any citizen. Additionally, a lawyer of a citizen can open a lawsuit in name of their client. Lawsuits initially include information provided from the victim or their lawyer. This information include court, suspected citizens and type of the lawsuit. For extra information the suing party will also provide a description to the system. After a lawsuit is filed, the system will assign a judge to the lawsuit and begin trialing process, which is also managed by judge. Judges also manage the publicity of the lawsuits, and they can decide for which parts of the lawsuit can be seen by other citizens.

3.2.2. Deciding on a lawsuit (by a judge)

Judges will be assigned to manage lawsuits by the system. Judges will be able to see both their decided and ongoing lawsuits. They also manage trials of the lawsuits, they can arrange and cancel trials. They also fill the information about the trials to the system. Depending on the type of the lawsuit, a judge may decide to assign a conciliator to the lawsuit to grant access to finalize the lawsuit. A lawsuit and its trials are archived after being finalized.

3.2.3 Reconciliation (by a conciliator)

Conciliators manage reconciliation process in the system. They are assigned by judges to finalize lawsuits without unnecessary trials. They can see their assigned lawsuits both archived and ongoing. They can complete a reconciliation case by entering the agreed terms from both parties and finalize the case.

3.3 DATA STRUCTURES

For the numeric types in the project, we are going to use **int**, int variations such as **tinyint**, **shortint**, etc. and **bit**. Small variations of int will be used to decrease the need of storage.

Aside from the numeric types, **int(11)** will be used for dates represented in UNIX timestamps. (We will avoid storing dates in data structures such as **date**, **datetime** or **time**.)

For the string types, **varchar** will be used for shorter strings such as names, telephone numbers, citizen ID, etc. For long texts such as judge verdict details, **text** will be used.

4. UI DESIGN AND SQL STATEMENTS

4.1 LOGIN PAGE

A Web Page

http://nationaljudiciaryinfo.com

Sign Up

Logo

Welcome to National Judiciary Information System !

User ID:

Password:

Login

Forgot my password

Related SQL:

```
"SELECT citizenID FROM User
    WHERE citizenID = '$user_ID'
    AND password = '$pass'";
```

Where \$pass is the hashed password input to be compared with the hashed password stored in the database.

4.2 SIGN UP PAGE

A Web Page

http://nationaljudiciaryinfo.com/sign-up

Sign Up

Surname:*

Name:*

National ID:*

Adress:*

City:*

Zip Code: *

E-mail:*

Phone Number: *

(*) refers to obligatory information to register to the system

Create My Account

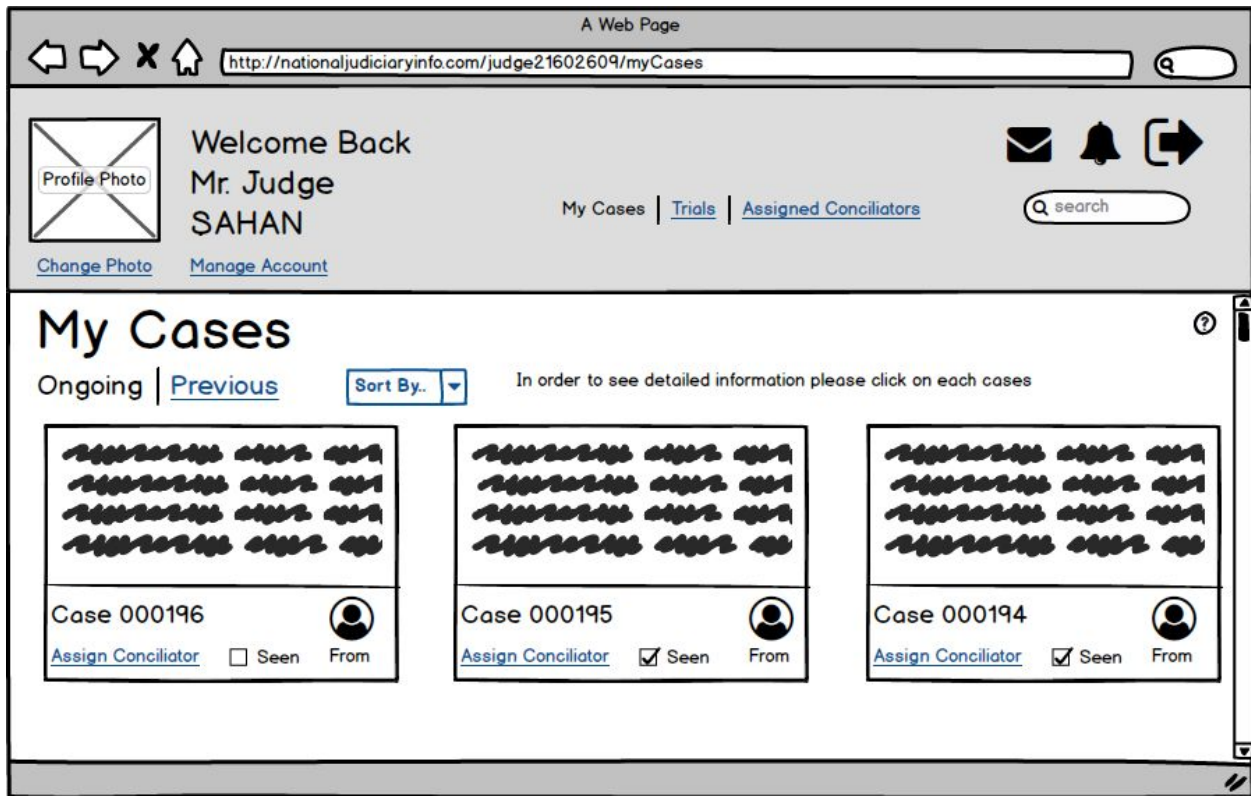
Related SQL:

```
"INSERT INTO User (citizenID, name, phoneNo, email, taxNo, address,
city, zipCode)
VALUES ('$citizen_id', '$name', '$phoneNo', '$email', '$taxNo',
'$address', '$city', '$zipcode');"

```

Where each string variable is the data obtained from manipulated input values.

4.3 JUDGE'S WELCOME PAGE



Related SQL:

```
"SELECT * FROM Lawsuit INNER JOIN LawsuitJudge ON Lawsuit.lawsuitID =  
LawsuitJudge.lawsuitID WHERE LawsuitJudge.judgeNo = '$judgeNo' AND  
Lawsuit.currentStatus = 0";
```

Where currentStatus = 0 indicates that the case is not yet finalized, (i.e. it is an open case) and \$judgeNo is the user's (judge) id.

4.4 MANAGING ACCOUNT PAGE

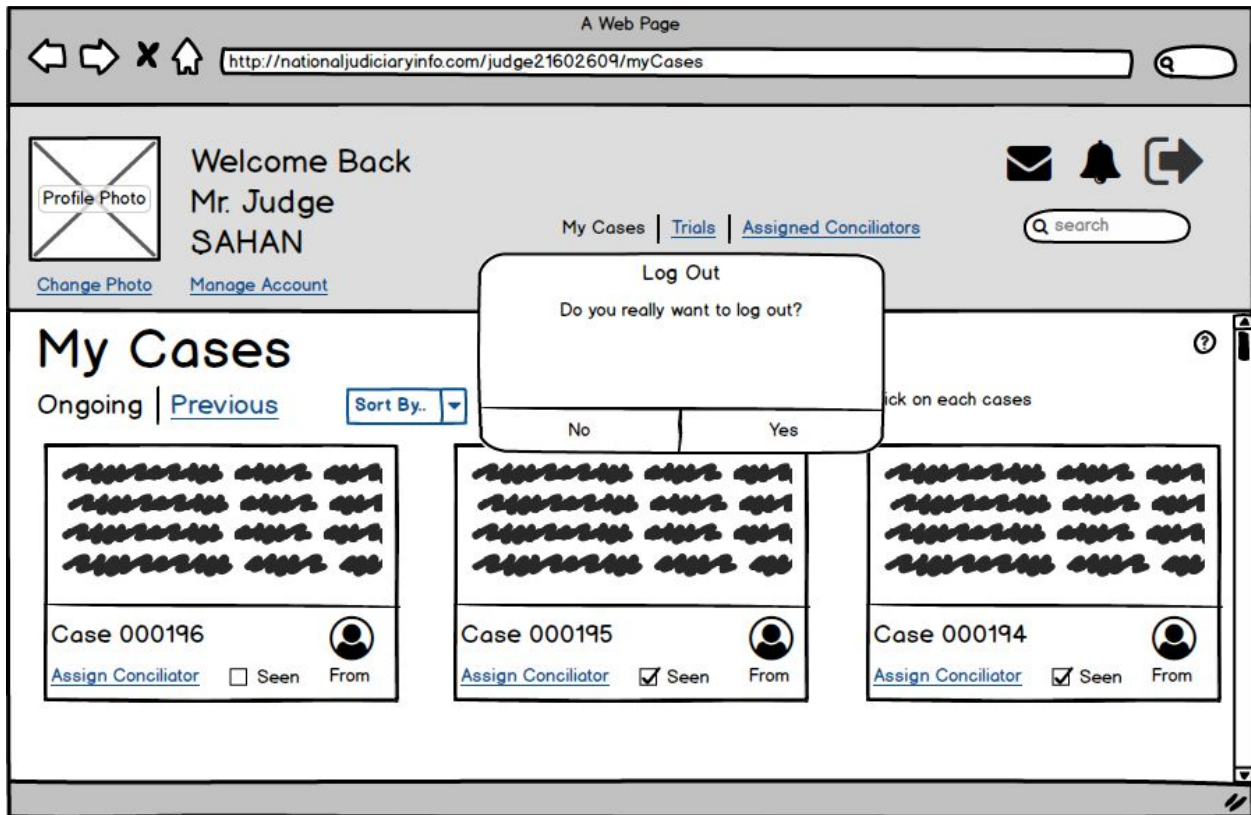
The screenshot shows a web browser window with the address bar displaying `http://nationaljudiciaryinfo.com/judge21602609/myCases`. The page title is "A Web Page". The header section includes a "Profile Photo" placeholder, a welcome message "Welcome Back Mr. Judge SAHAN", and navigation links: "My Cases", "Trials", and "Assigned Conciliators". There are also icons for email, notifications, and a search bar with the text "Q search". Below the header, the "Manage Account" section is visible. It features a large "Profile Photo" placeholder on the left. To the right, there are input fields for "Surname:", "Name:", "National ID:", and "Adress:". The "Adress:" field has a blue "Edit" button next to it. The page also includes a "Change Photo" link and a "Manage Account" link in the header.

Related SQL:

```
"UPDATE User SET photoURL = '$photoURL', address = '$address' WHERE citizenID = '$citizen_id'";
```

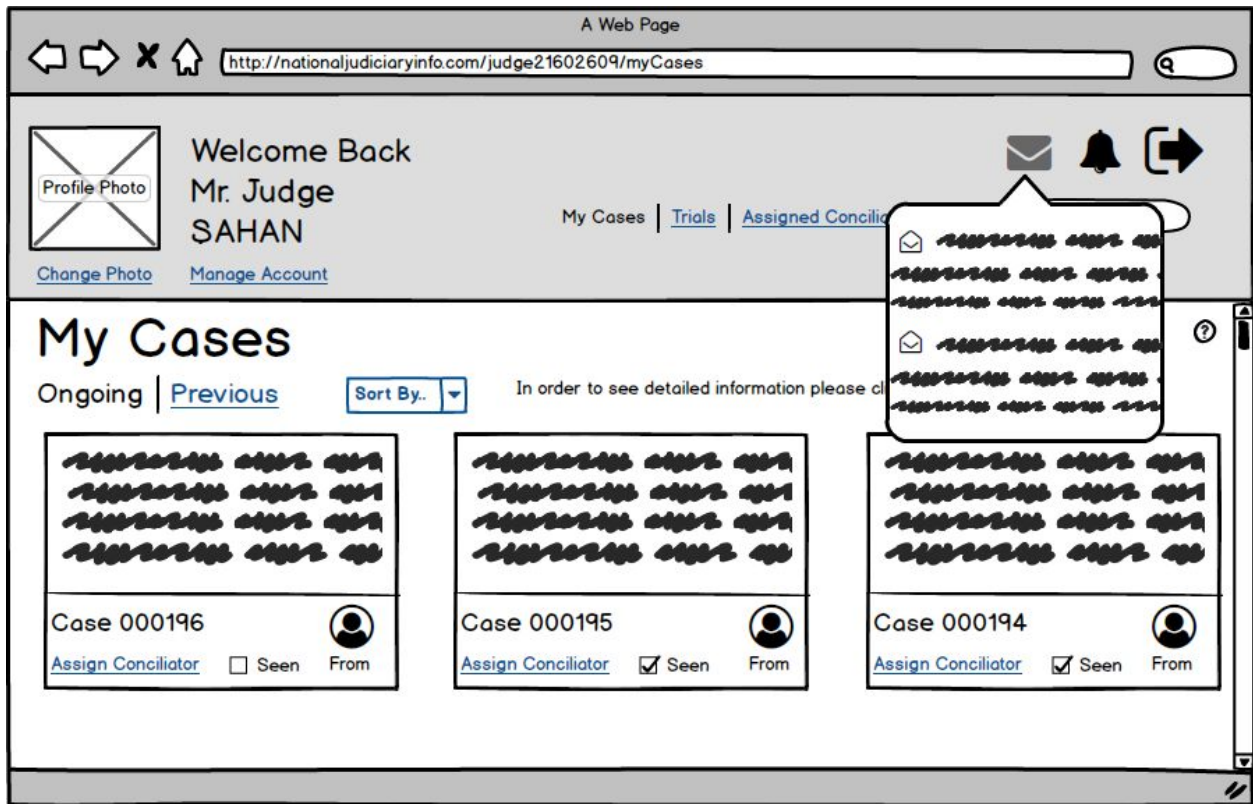
Where \$citizen_id is the variable which indicates the id of the edited account, \$photoURL is the URL which contains the URL of updated picture, \$address is the updated address string.

4.5 SIGN OUT PAGE



This action does not require any SQL statements, only changing the session cookies is necessary for logging out.

4.6 MESSAGE PAGE

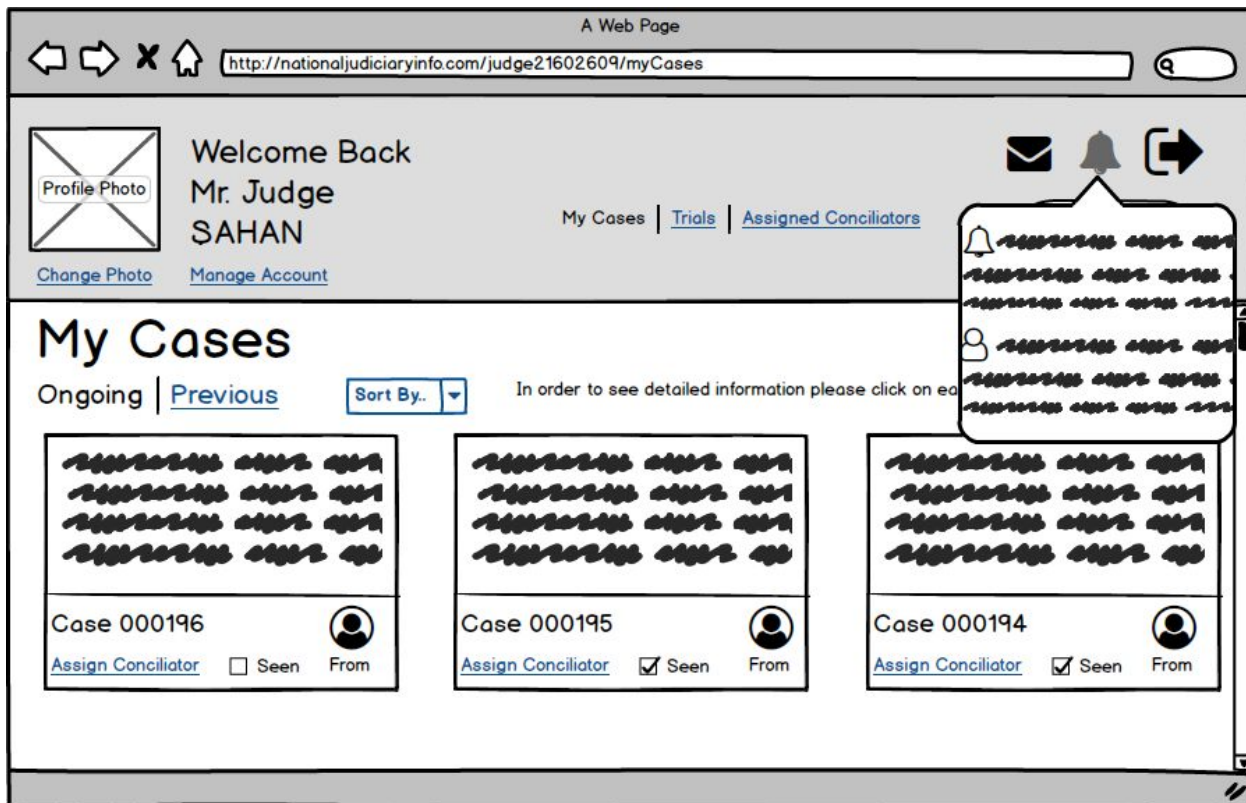


Related SQL:

```
"SELECT sender_id, subject, content FROM UserDMs where recipient_id =  
'$citizen_id'";
```

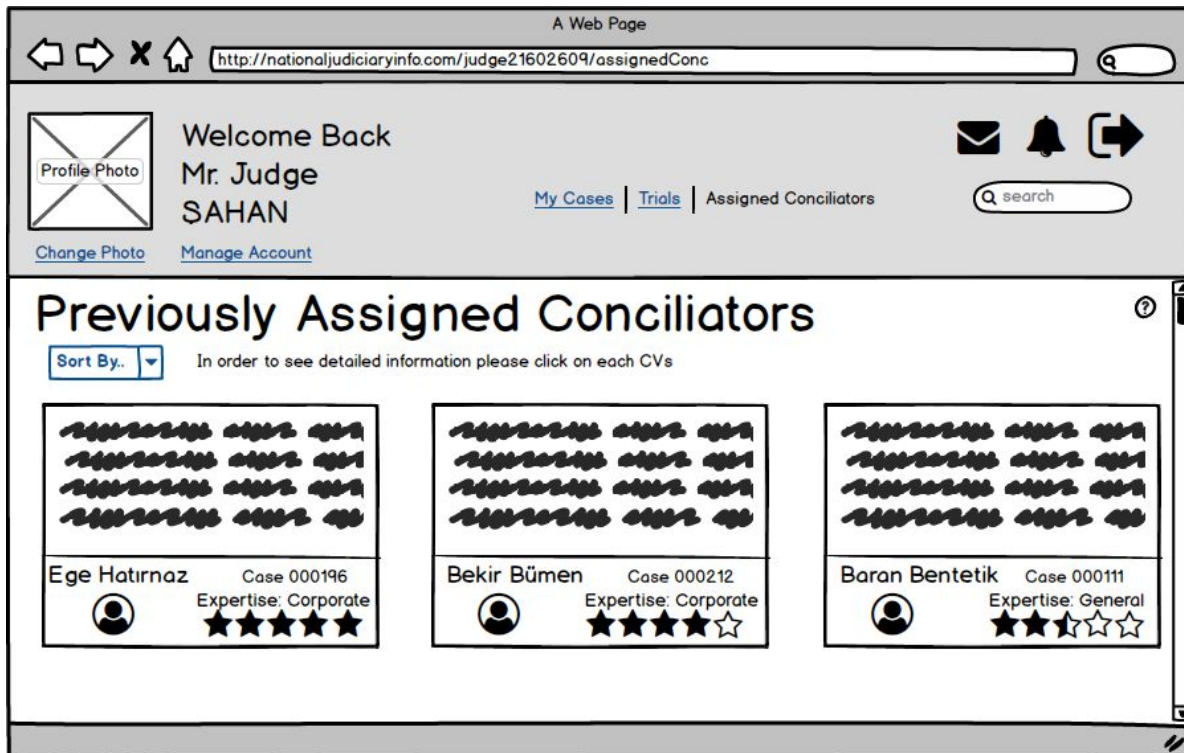
Where \$citizen_id is the user's citizen ID for obtaining the direct messages sent to that particular user.

4.7 NOTIFICATION PAGE



We do not need any SQL query to indicate notification page.

4.8 ASSIGNED CONCILIATORS

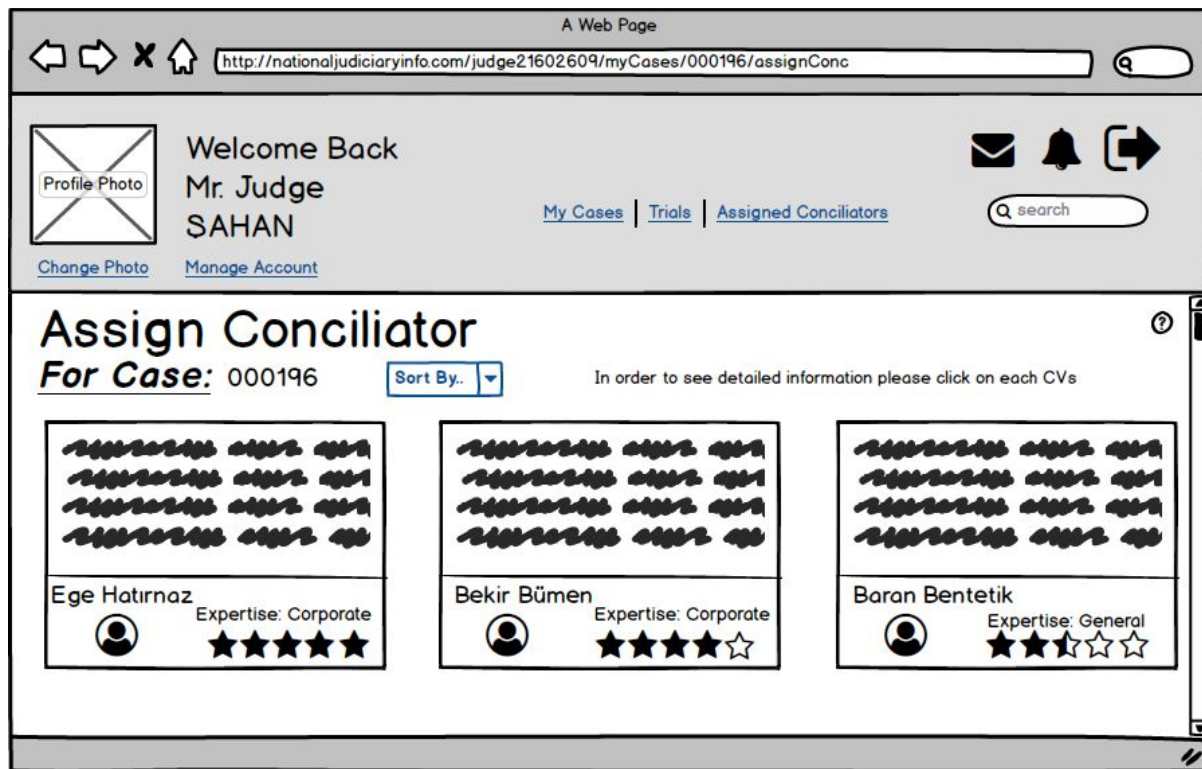


Related SQL:

```
"SELECT * FROM Lawsuit_Conciliator_table ORDER BY rating DESC";
```

Where Lawsuit_Conciliator_table is the table which includes the lawsuitID, conciliatorNo and every other detail regarding to that matchup such as rating, etc. The example is sorted in a descending rating order.

4.9 ASSIGN A CONCILIATOR

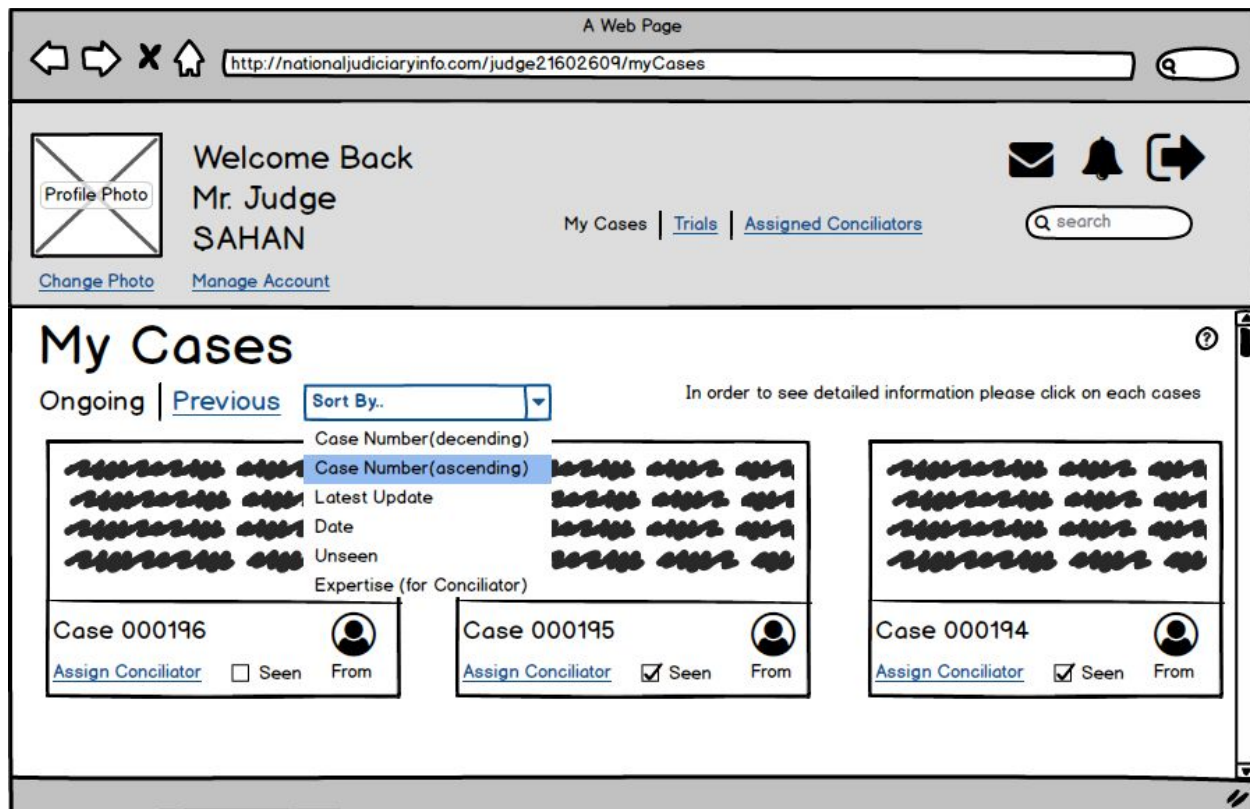


Related SQL:

```
"SELECT * FROM Conciliator INNER JOIN Lawsuit ON  
Conciliator.expertise = Lawsuit.caseCategory WHERE Lawsuit.lawsuitNo  
= '$caseNo' ORDER BY rating DESC";
```

Where \$caseNo is the case which conciliator will be assigned on. This will give only the related conciliators for the case at hand. The example is sorted in a descending rating order.

4.10 SORTING IN CASES

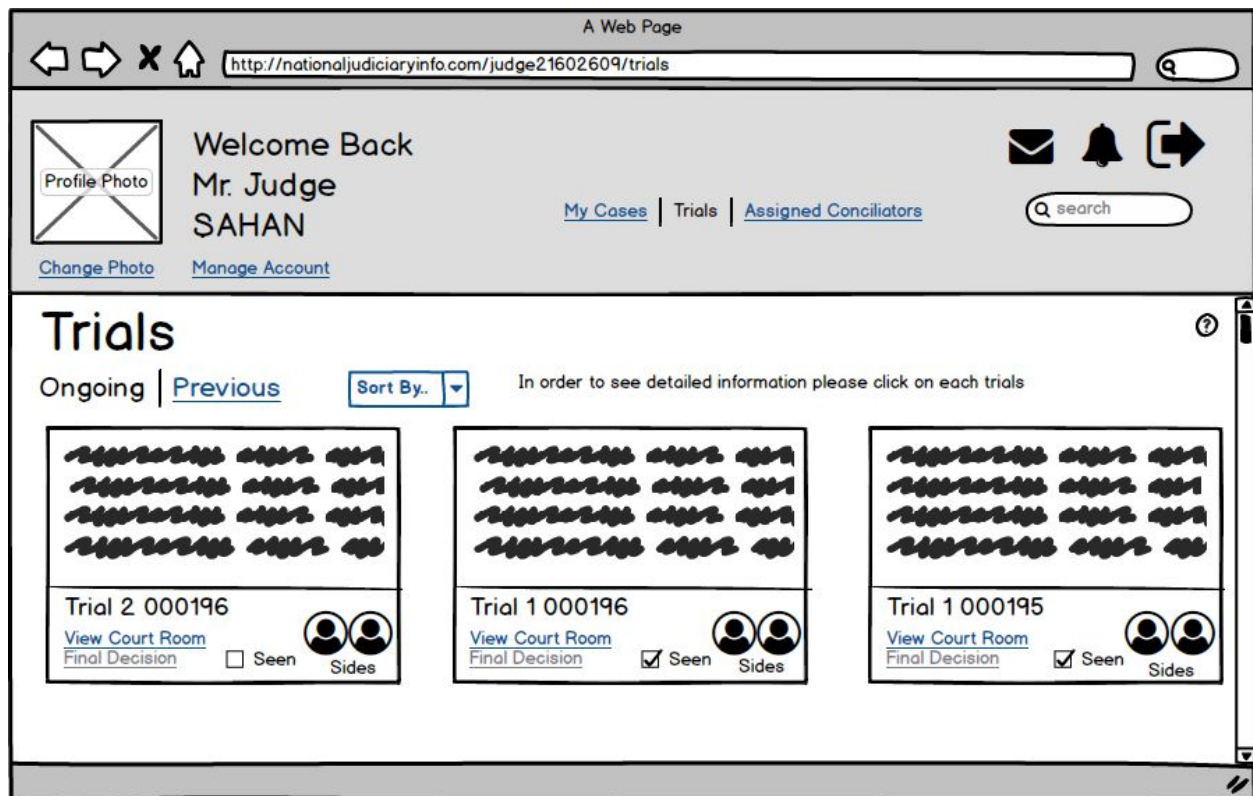


Related SQL:

```
"SELECT * FROM Lawsuit INNER JOIN LawsuitJudge ON Lawsuit.lawsuitID =  
LawsuitJudge.lawsuitID WHERE LawsuitJudge.judgeNo = '$judgeNo' AND  
Lawsuit.currentStatus = 0 ORDER BY '$sortingType'";
```

Where `currentStatus = 0` indicates that the case is not yet finalized, (i.e. it is an open case), `$judgeNo` is the user's (judge) ID and `$sortingType` is the string which would sort the data. `$sortingType` might be "caseNo DESC" which would indicate the cases will be sorted in a descending case number.

4.11 TRIALS (EXAMPLE OF JUDGES)

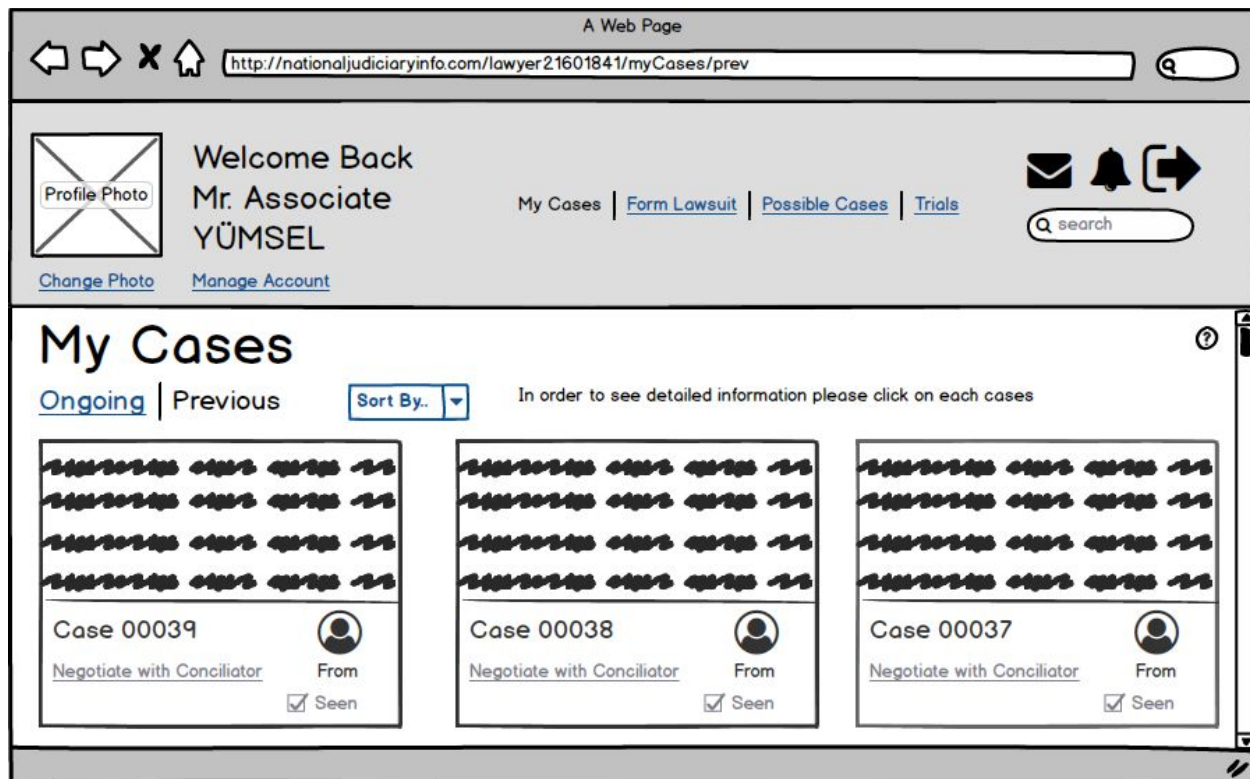


Related SQL:

```
"SELECT * FROM Trial INNER JOIN TrialJudge ON Trial.no =  
TrialJudge.trialNo WHERE TrialJudge.judgeNo = '$judgeNo'";
```

Where \$judgeNo is the user (judge) ID.

4.12 LAWYER'S WELCOME PAGE

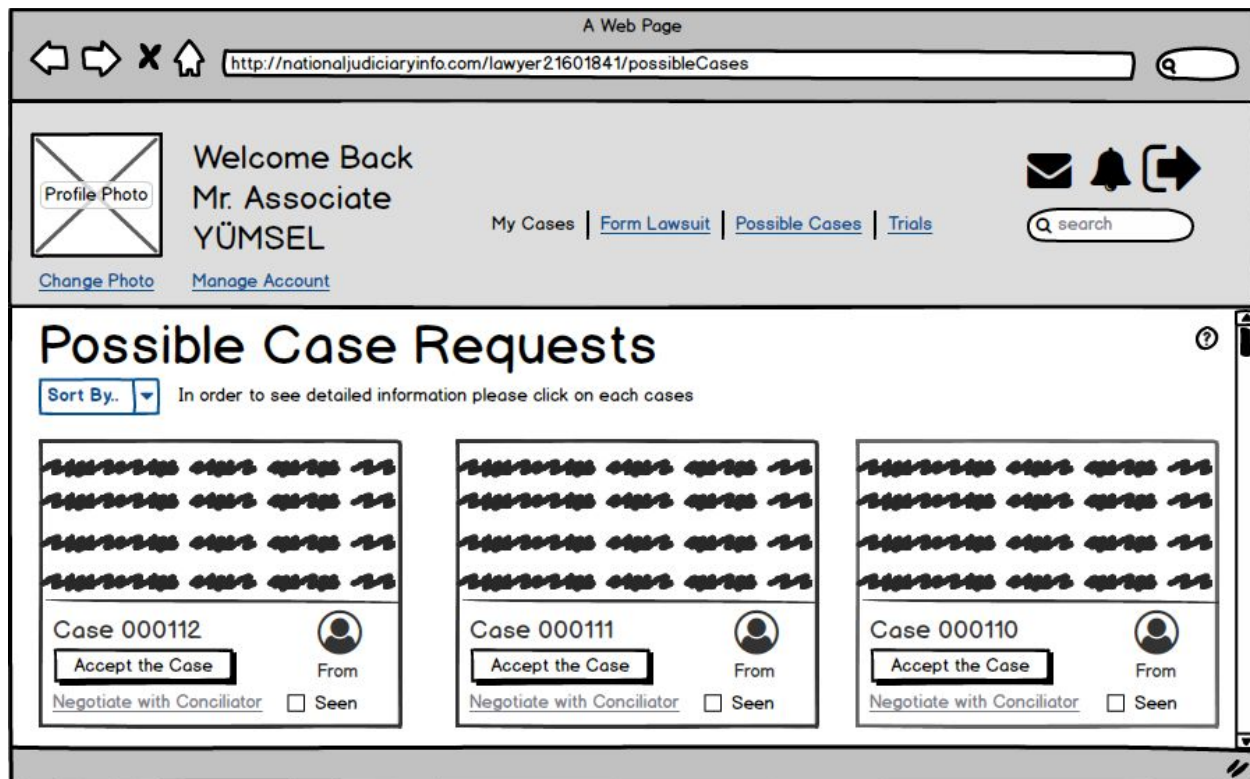


Related SQL:

```
"SELECT * FROM Lawsuit INNER JOIN LawsuitLawyer ON Lawsuit.lawsuitID  
= LawsuitLawyer.lawyerNo WHERE LawsuitLawyer.lawyerNo = '$lawyerNo  
AND Lawsuit.currentStatus = 0";
```

Where currentStatus = 0 indicates that the case is not yet finalized, (i.e. it is an open case) and \$lawyerNo is the user's (lawyer) id. LawsuitLawyer is the table which stores the lawsuit id's with corresponding lawyer id's.

4.13 LAWYER'S POSSIBLE CASES



Related SQL:

```
"SELECT * FROM Lawsuit INNER JOIN LawsuitLawyer ON Lawsuit.lawsuitID
= LawsuitLawyer.lawsuitID WHERE LawsuitLawyer.lawyerNo = '$lawyerNo
AND Lawsuit.currentStatus = 3";
```

Where `currentStatus = 3` indicates that the case is a request, (i.e. not opened yet and it is incomplete) and `$judgeNo` is the user's (judge) id. `LawsuitLawyer` is the table which stores the lawsuit id's with corresponding lawyer id's.

4.14 FORMING A LAWSUIT

The screenshot shows a web browser window with the address bar displaying `http://nationaljudiciaryinfo.com/lawyer21601841/formLawS`. The page header includes a welcome message for 'Mr. Associate YÜMSEL' and navigation links for 'My Cases', 'Form Lawsuit', 'Possible Cases', and 'Trials'. There is also a search bar and icons for email, notifications, and a share button. The main content area is titled 'New Lawsuit' and contains the following form elements:

- Victim:** A text input field.
- Category:** A dropdown menu with 'Categories' selected.
- Description:** A large text area for entering the lawsuit details.
- Suspect:** A text input field.
- Court:** A dropdown menu with 'Courts' selected.
- Form:** A button at the bottom of the form.

Related SQL:

```
"INSERT INTO Lawsuit (description, caseCategory, currentStatus)
VALUES ('$description', '$category', '0');"

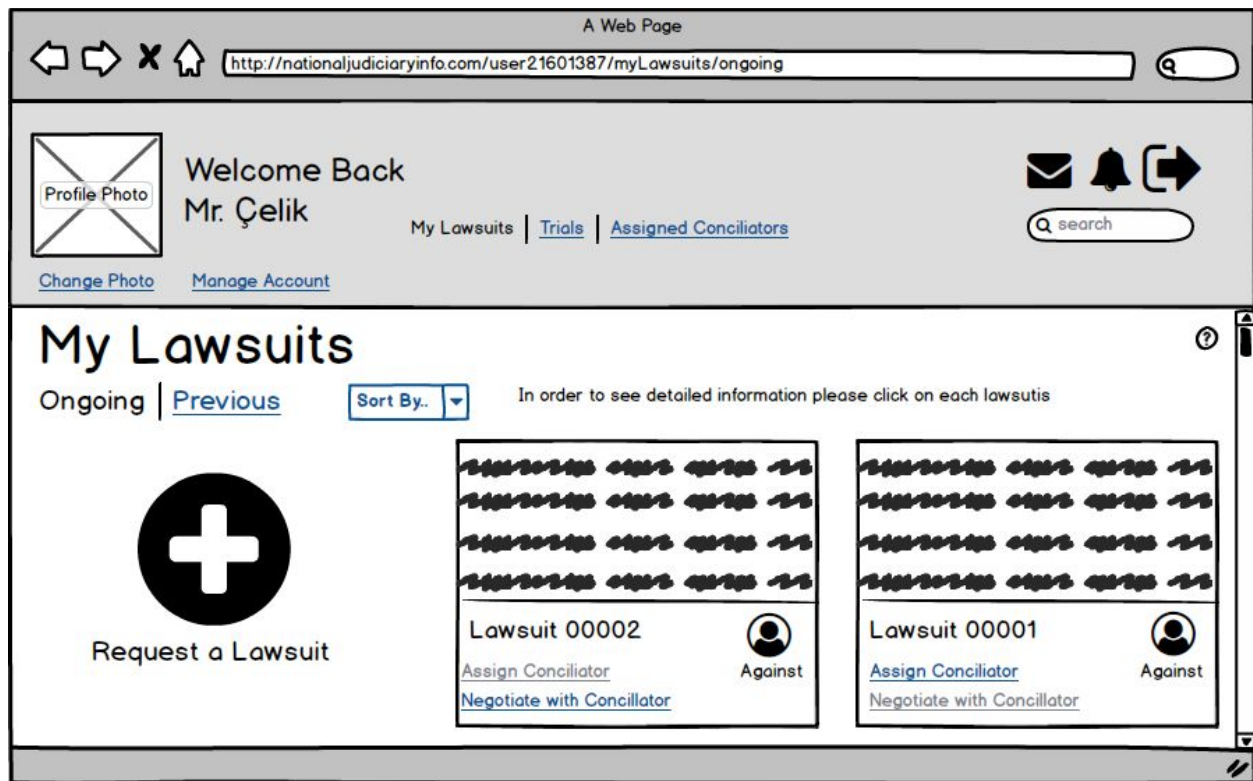
"INSERT INTO LawsuitVictim (lawsuitID, citizenID)
VALUES ('$lawsuitID', '$victimID');"

"INSERT INTO LawsuitSuspect (lawsuitID, citizenID)
VALUES ('$lawsuitID', '$SuspectID');"

"INSERT INTO LawsuitCourt (lawsuitID, courtID)
VALUES ('$lawsuitID', '$courtID');
```

Where lawsuitID is the auto-incremented value of Lawsuit's lawsuitID.

4.15 USER'S WELCOME PAGE



Related SQL:

```
"SELECT * FROM Lawsuit INNER JOIN LawsuitVictim ON Lawsuit.lawsuitID
= LawsuitVictim.lawsuitID WHERE LawsuitVictim.citizenID = '$userID
AND Lawsuit.currentStatus = 0
UNION
SELECT * FROM Lawsuit INNER JOIN LawsuitSuspect ON Lawsuit.lawsuitID
= LawsuitSuspect.lawsuitID WHERE LawsuitSuspect.citizenID = '$userID
AND Lawsuit.currentStatus = 0";
```

Where currentStatus = 0 indicates that the case is not yet finalized, (i.e. it is an open case) and \$userID is the user's (citizen) id.

4.16 REQUEST A LAWSUIT

The screenshot shows a web browser window with the URL `http://nationaljudiciaryinfo.com/user21601387/reqLawS`. The page header includes a user profile for 'Mr. Çelik' with links for 'My Lawsuits', 'Trials', and 'Assigned Conciliators'. The main content area is titled 'Request a Lawsuit' and contains the following form fields:

- Victim**: A text input field.
- Category**: A dropdown menu with 'Categories' selected.
- Description**: A large text area for the lawsuit description.
- Suspect**: A text input field.
- Court**: A dropdown menu with 'Courts' selected.

At the bottom of the form is a button labeled 'Form'.

Related SQL:

```
"INSERT INTO Lawsuit (description, caseCategory, currentStatus)
VALUES ('$description', '$category', '3')";

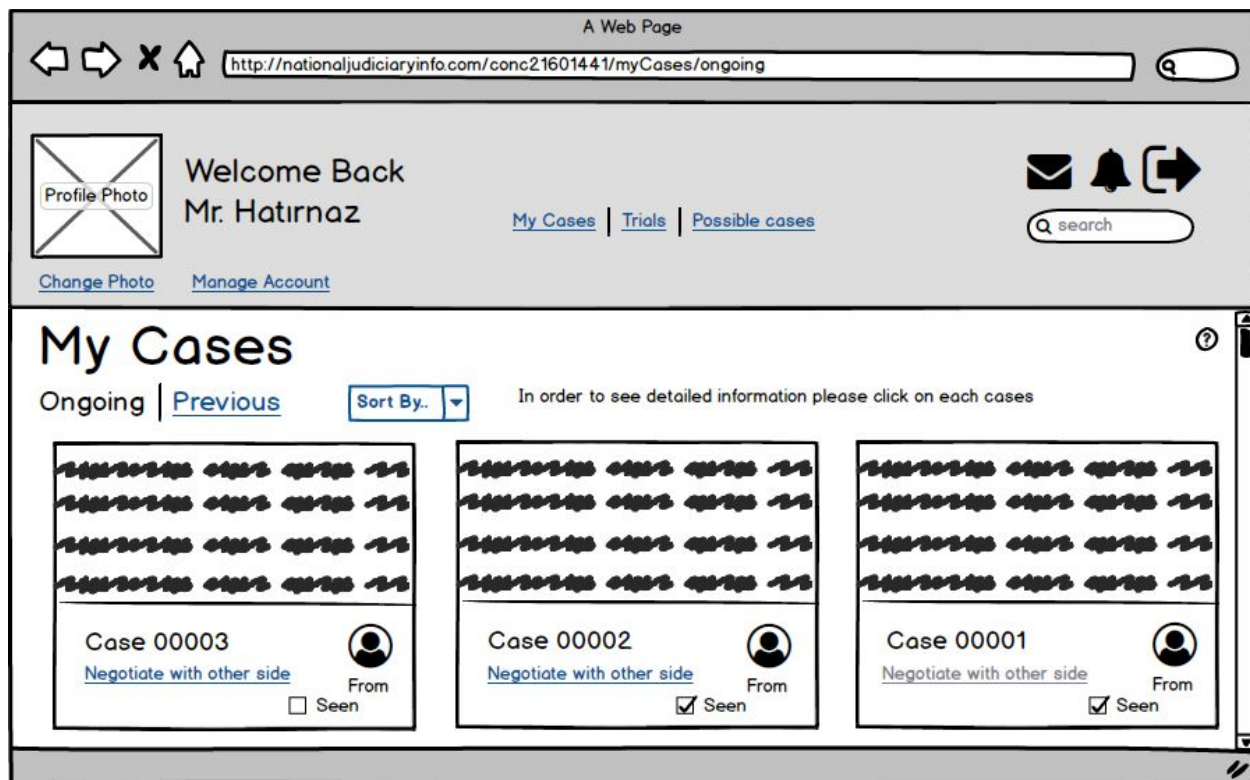
"INSERT INTO LawsuitVictim (lawsuitID, citizenID)
VALUES ('$lawsuitID', '$victimID')";

"INSERT INTO LawsuitSuspect (lawsuitID, citizenID)
VALUES ('$lawsuitID', '$SuspectID')";

"INSERT INTO LawsuitCourt (lawsuitID, courtID)
VALUES ('$lawsuitID', '$courtID')";
```

Where lawsuitID is the auto-incremented value of Lawsuit's lawsuitID. currentStatus = 3 indicates a requested lawsuit.

4.17 CONCILIATOR WELCOME PAGE



Related SQL:

```
"SELECT * FROM Lawsuit INNER JOIN LawsuitConciliator ON  
Lawsuit.lawsuitID = LawsuitConciliator.lawsuitID WHERE  
LawsuitConciliator.conciliatorID = '$conciliatorID' AND  
Lawsuit.currentStatus = 0";
```

Where currentStatus = 0 indicates that the case is not yet finalized, (i.e. it is an open case) and \$conciliator is the user's (conciliator) id.

5. ADVANCED DATABASE COMPONENTS

5.1 VIEWS

5.1.1 RELATED LAWSUITS VIEW

```
CREATE VIEW related_lawsuits AS
SELECT * FROM Lawsuit INNER JOIN LawsuitVictim ON Lawsuit.lawsuitID =
LawsuitVictim.lawsuitID WHERE LawsuitVictim.citizenID = '$userID AND
Lawsuit.currentStatus = 0
UNION
SELECT * FROM Lawsuit INNER JOIN LawsuitSuspect ON Lawsuit.lawsuitID
= LawsuitSuspect.lawsuitID WHERE LawsuitSuspect.citizenID = '$userID'
AND Lawsuit.currentStatus = 0
```

5.1.2 JUDGE'S ONGOING LAWSUITS VIEW

```
CREATE VIEW judge_ongoing_lawsuits AS
SELECT * FROM Lawsuit INNER JOIN LawsuitJudge ON Lawsuit.lawsuitID =
LawsuitJudge.lawsuitID WHERE LawsuitJudge.judgeNo = '$judgeNo' AND
Lawsuit.currentStatus = 0
```

5.1.3 TOP RATED CONCILIATORS

```
CREATE VIEW top_rated_conciliators AS
SELECT * FROM Conciliator ORDER BY rating DESC";
```

5.2 REPORTS

5.2.1 LAST 30-DAY OPEN CASE REPORT:

```
CREATE VIEW last30_open_case AS
```

```
SELECT lawsuitID, type, crimeCategory, city, district, courtHouse
FROM (Lawsuit JOIN Trial ON lawsuitID = Trial.lawsuitID) WHERE
Lawsuit.currentStatus = 0 AND (UNIX_TIMESTAMP() - Trial.date <=
2592000)
```

5.2.2 LAST 30-DAY TRIAL REPORT:

```
CREATE VIEW last30_trial AS
SELECT * FROM Trial WHERE (UNIX_TIMESTAMP() - date <= 2592000)
```

5.3 TRIGGERS

- When new account is created by any user or new account is provided by the administration, the user would be added on to the related table (user) by administration with considering its national ID.
- When new lawsuit is filed by lawyer the following lawsuit would be added on relevant table(lawsuit) as tuple with required entities. The system itself would provide the lawsuitID automatically after creation.
- When conciliator is assigned to a new specific case, the attributes of reconciliation change accordingly
- After the finalized conciliated case, the rating of the relevant conciliator would be updated with regarding the success or the failure of the conciliator on that specific case.
- After acception of a possible case by lawyer, the system would automatically consider that case is filed by the lawyer, so it would be added on “My Cases” page.

5.4 CONSTRAINTS

- A conciliator cannot participate in a lawsuit that is not in their expertise area.
- A lawyer, a conciliator or a judge cannot be represented as suspect and/or victim associated in a lawsuit in which they participate in their respective roles.

- A statement date or a trial date related to a lawsuit cannot be later than the stated lawsuit's date of finalization.

6. IMPLEMENTATION PLAN

We will use PHP, HTML, CSS and Javascript in our website. To maintain responsiveness in our design, we intend to utilize Bootstrap. JQuery will be used with Javascript to enhance our development. For database management, we intend to use MySQL. The datas obtained from MySQL server will be manipulated and prepared to use in the views and interfaces of the website.

7. WEBSITE

The project information and the source code will be maintained on the following link:

Wiki Page: egehatirnaz.github.io/Judiciary-Informatics

Repository: github.com/egehatirnaz/Judiciary-Informatics