

A PROJECT REPORT
ON
“AUTOMATED JURISDICTION SUPPORT SYSTEM”

Submitted to
UNIVERSITY OF MUMBAI

In Partial Fulfilment of the Requirement for the Award of

BACHELOR’S DEGREE IN
COMPUTER ENGINEERING

BY

Sardar Sobiya Mohd.Ibrahim Shagufta	16DCO69
Adhikari Shahzeen Rafique Masooda	16DCO44
Sayyed Rehan Zahir Ahmed Ruksana	16DCO72
Naje Aqeel Rahil Shahida	16DCO64

UNDER THE GUIDANCE OF
Prof. Apeksha Gopale



DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam's Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY

Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206

2018-2019

AFFILIATED TO
UNIVERSITY OF MUMBAI

**A PROJECT II REPORT
ON**

“AUTOMATED JURISDICTION SUPPORT SYSTEM”

**Submitted to
UNIVERSITY OF MUMBAI**

In Partial Fulfilment of the Requirement for the Award of

**BACHELOR’S DEGREE IN
COMPUTER ENGINEERING**

BY

Sardar Sobiya Mohd.Ibrahim Shagufta	16DC069
Adhikari Shahzeen Rafique Masooda	16DC044
Sayed Rehan Zahir Ahmed Ruksana	16DC072
Naje Aqeel Rahil Shahida	16DC064

**UNDER THE GUIDANCE OF
Prof. Apeksha Gopale**

**DEPARTMENT OF COMPUTER ENGINEERING
Anjuman-I-Islam’s Kalsekar Technical Campus
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206**

**2018-2019
AFFILIATED TO**



UNIVERSITY OF MUMBAI

Anjuman-i-Islam's Kalsekar Technical Campus

Department of Computer Engineering
SCHOOL OF ENGINEERING & TECHNOLOGY
Plot No. 2 3, Sector - 16, Near Thana Naka,
Khandagaon, New Panvel - 410206



CERTIFICATE

This is certify that the project entitled

“Automated Jurisdiction Support System“

submitted by

Sardar Sobiya Mohd.Ibrahim Shagufta	16DCO69
Adhikari Shahzeen Rafique Masooda	16DCO44
Sayyed Rehan Zahir Ahmed Ruksana	16DCO72
Naje Aqeel Rahil Shahida	16DCO64

is a record of bonafide work carried out by them, in the partial fulfilment of the requirement for the award of Degree of Bachelor of Engineering (Computer Engineering) at *Anjuman-I-Islam's Kalsekar Technical Campus, Navi Mumbai* under the University of MUMBAI. This work is done during year 2018-2019, under our guidance.

Date: / /

(Prof. Apeksha Goaple)
Project Supervisor

(Prof. Kalpana Bodke)
Project Coordinator

(Prof. Tabrez Khan)
HOD, Computer Department

DR. ABDUL RAZAK HONNUTAGI
Director

External Examiner

Acknowledgements

I would like to take the opportunity to express my sincere thanks to my guide **Prof. Apeksha Gopale**, Assistant Professor, Department of Computer Engineering, AIKTC, School of Engineering, Panvel for her invaluable support and guidance throughout my project research work. Without her kind guidance & support this was not possible.

I am grateful to her for her timely feedback which helped me track and schedule the process effectively. Her time, ideas and encouragement that she gave helped me to complete my project efficiently.

We would like to express deepest appreciation towards **DR. ABDUL RAZAK HONNUTAGI**, Director, AIKTC, Navi Mumbai, **Prof. Tabrez Khan**, Head of Department of Computer Engineering and **Prof. Kalpana Bodke**, Project Coordinator whose invaluable guidance supported us in completing this project.

At last we must express our sincere heartfelt gratitude to all the staff members of Computer Engineering Department who helped me directly or indirectly during this course of work.

Sardar Sobiya Mohd. Ibrahim Shagufta

Adhikari Shahzeen Rafique Masooda

Sayyed Rehan Zahir Ahmed Ruksana

Naje Aqeel Rahil Shahida

Project I Approval for Bachelor of Engineering

This project entitled *Automated Jurisdiction Support System* by *Sardar Sobiya Mohd.Ibrahim Shagufta, Adhikari Shahzeen Rafique Masooda, Sayyed Rehan Zahir Ahmed Ruksana, Naje Aqeel RAhil Shahida* is approved for the degree of *Bachelor of Engineering in Department of Computer Engineering*.

Examiners

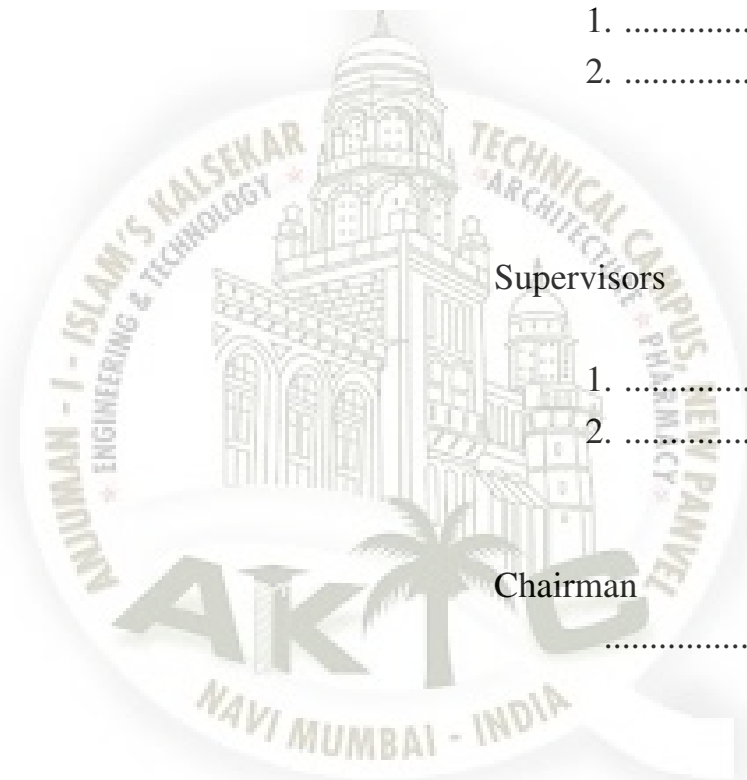
1.
2.

Supervisors

1.
2.

Chairman

.....



Declaration

I declare that this written submission represents my ideas in my own words and where others ideas or words have been included, I have adequately cited and referenced the original sources. I also declare that I have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any idea/data/fact/source in my submission. I understand that any violation of the above will be cause for disciplinary action by the Institute and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been taken when needed.



Sardar Sobiya Mohd.Ibrahim Shagufta
16DCO69

Adhikari Shahzeen Rafique Masooda
16DCO44

Sayyed Rehan Zahir Ahmed Ruksana
16DCO72

Naje Aqeel Rahil Shahida
16DCO64

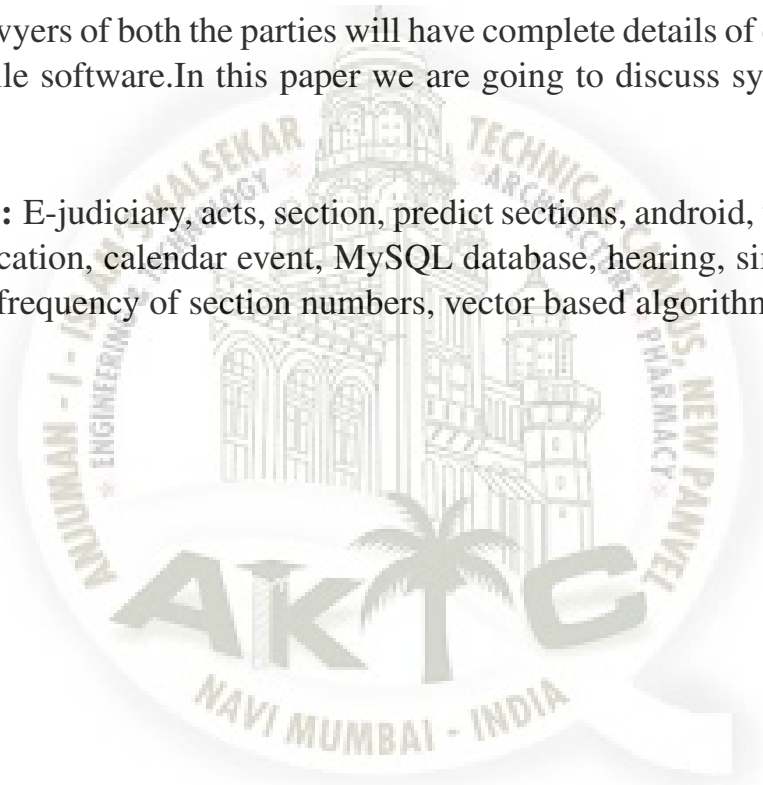
ABSTRACT

Title : Automated Jurisdiction Support System

Every work of Judiciary System is done manually. From filing FIR to the last verdict of judge too much time is consumed. We are proposing a system that is trying to automate Indian judiciary system, so as to minimize the wastage of time of people involved in the case. The FIR and the charge-sheet will store online in our system which will make sure they are secure, also the system will predict act and section on the basis of keyword while creating charge-sheet..

Judge and lawyers of both the parties will have complete details of case in the form of web or mobile software. In this paper we are going to discuss system's working scenario.

Keywords: E-judiciary, acts, section, predict sections, android, website, scheduling, email notification, calendar event, MySQL database, hearing, similar cases, cosine similarity, frequency of section numbers, vector based algorithm

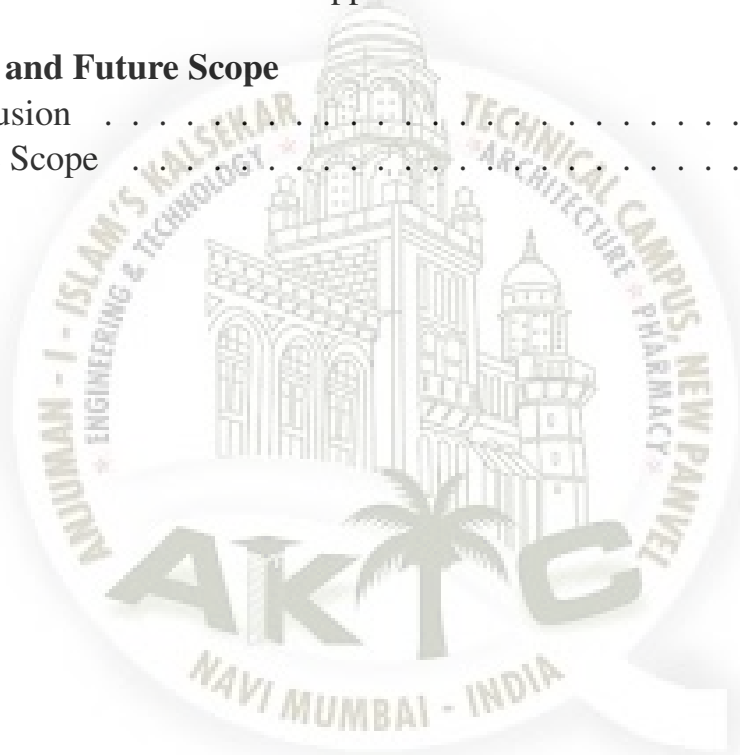


Contents

Acknowledgement	iii
Project I Approval for Bachelor of Engineering	iv
Declaration	v
Abstract	vi
Table of Contents	ix
1 Introduction	2
1.1 Purpose	2
1.2 Project Scope	3
1.3 Project Goals and Objectives	3
1.3.1 Goals	3
1.3.2 Objectives	3
1.4 Organization of Report	3
2 Literature Survey	5
2.1 E-Court Technology Diffusion in Court Management	5
2.1.1 Advantages of Paper	5
2.1.2 Disadvantages of Paper	5
2.1.3 How to overcome the problems mentioned in Paper	6
2.2 ICT in Indian Court Challenges Solution	6
2.2.1 Advantages of Paper	6
2.2.2 Disadvantages of Paper	6
2.2.3 How to overcome the problems mentioned in Paper	6
2.3 A secure E-Court System for India	7
2.3.1 Advantages of Paper	7
2.3.2 Disadvantages of Paper	7
2.3.3 How to overcome the problems mentioned in Paper	7
2.4 Technical Review	7
2.4.1 Advantages of Technology	8
2.4.2 Reasons to use this Technology	8
3 Project Planning	9
3.1 Members and Capabilities	9
3.2 Roles and Responsibilities	9

3.3	Assumptions and Constraints	9
3.3.1	Assumptions	9
3.3.2	Constraints	9
3.4	Project Management Approach	10
3.5	Ground Rules for the Project	10
3.6	Project Budget	10
3.7	Project Time-line	11
4	Software Requirements Specification	12
4.1	Overall Description	12
4.1.1	Product Perspective	12
4.1.2	Product Features	12
4.1.3	User Classes and Characteristics	13
4.1.4	Operating Environment	13
4.1.5	Design and Implementation Constraints	13
4.2	System Features	13
4.2.1	System Feature	13
4.3	External Interface Requirements	15
4.3.1	User Interfaces	15
4.3.2	Hardware Interfaces	16
4.3.3	Software Interfaces	16
4.3.4	Communications Interfaces	17
4.4	Nonfunctional Requirements	17
4.4.1	Performance Requirements	17
4.4.2	Safety Requirements	17
4.4.3	Security Requirements	17
5	System Design	18
5.1	System Requirements Definition	18
5.1.1	Functional requirements	18
5.1.2	System requirements (non-functional)	23
5.2	System Architecture Design	24
5.3	Sub-system Development	26
5.3.1	Module 1 Police	26
5.3.2	Module 2 Judge	28
5.3.3	Module 3 Lawyer	29
5.3.4	Module 4 Steno-typist	30
5.4	Systems Integration	31
5.4.1	Class Diagram	31
5.4.2	Activity Diagram	32
5.4.3	Component Diagram	33
5.4.4	Deployment Diagram	34

6	Implementation	35
6.1	Police	35
6.2	Judge	37
6.3	Lawyer	39
6.4	Steneo-typist	41
7	System Testing	42
7.1	Test Cases and Test Results	42
7.2	Sample of a Test Case	43
7.2.1	Software Quality Attributes	50
8	Screenshots of Project	51
8.1	Web Module And Android App Module	51
9	Conclusion and Future Scope	63
9.1	Conclusion	63
9.2	Future Scope	63
	References	63
	Achievements	64



List of Figures

3.1	Gantt Chart	11
5.1	Usecase Diagram	19
5.2	Level0 DFD Diagram	20
5.3	Level1 DFD Diagram	21
5.4	Level2 DFD Diagram	22
5.5	ER Diagram	24
5.6	System Architecture	25
5.7	Police model diagram	27
5.8	Judge model diagram	28
5.9	lawyer model diagram	29
5.10	Steno-typist model diagram	30
5.11	Class diagram	31
5.12	Activity Diagram	32
5.13	Component diagram	33
5.14	deployment diagram	34
6.1	Police Home page	35
6.2	Judge Home Page	37
6.3	Lawyer Home Page	39
6.4	Steneo-typer Form	41
8.1	Police Login Form	51
8.2	Police Home Page	52
8.3	FIR Form	52
8.4	Case Acceptance	53
8.5	Charge-sheet Form	53
8.6	All Cases	54
8.7	Steno-typist Login Form	55
8.8	Steno-typist Form	55
8.9	Court's Home Page	56
8.10	Case Detail with Case History	56
8.11	Judge's Home Page and New Cases	57
8.12	Reject Case and Accept Case	57
8.13	Calendar Event and Email Notification of Hearing Detail	58

8.14	Running Cases of Judge	58
8.15	Reschedule Case and Calendar Event for new date	59
8.16	Hearing Detail and Closed Cases of Judge	59
8.17	Lawyer's Home Page and New Case for lawyer	60
8.18	Accept Case by Registering Client Detail	60
8.19	Payment detail of new case and List of Running Cases	61
8.20	Similar Cases and Hearing details for all Running Cases	61
8.21	List of Closed Cases handle by lawyer	62
8.22	Get similar cases by giving case id	62



List of Tables

1.1	Statistic Table	2
3.1	Table of Capabilities	9
3.2	Table of Responsibilities	9



Chapter 1

Introduction

The Indian judicial system, a part of world's largest democracy, is very old to follow. Right from the monarchical rule to the British era and the modern system of the independent India. More than 3 crore cases pending in different courts of India. Many of these cases are pending for more than 10 years. Below is the table that Shows approximate number of cases pending in Supreme Court, High Courts and District and Subordinate courts in India.

Table 1.1: Statistic Table

SR. No	Court	pending Course
1	Supreme Court	Around 60 Thousand
2	High Court	Around 42 Lakh
3	District and Sub-Ordinate Court	Around 2.7 Crore

This is one of the primary issues of Indian jurisdiction system. The main reason of this is delay in hearing and old manual work. Like the other pillars of democracy, the judiciary too has been found to engage in corruption. there are higher chances that evidence are intentionally or accidentally miss placed.

If there is some solution that can solve these kinds of problems then it is beneficial for judiciary. Court-Cal (Automated Judiciary System) is one of the solution for these problems.

1.1 Purpose

To put the paper work of this government section to whole new technical level, this project is going to be the most efficient for judicial purpose.

To provide an interface that can predict section. To decrease study load of lawyers by providing similar case details. To provide case history details to judge and lawyers.

1.2 Project Scope

A website that will register police and steno-typist in our system. In this website steno-typist can type entire conversation. Police can submit FIR and charge-sheet online. Lawyers and judge will have android app which will contain all the details of case. Case assignment and scheduling will be done through app.

1.3 Project Goals and Objectives

1.3.1 Goals

The main goal of our project is to automate the manual work of jurisdiction system and provide a single integrated digital platform to all the main roles of jurisdiction system.

1.3.2 Objectives

The main objectives of our system are:

- To reduce the load of jurisdiction system.
- To provide an interface that can predict sections at the time of FIR and Charge-sheet registration.
- To decrease study load of lawyers by providing similar case details.
- To provide case history details to judge and lawyers.
- Also system will notify about about case hearing dates to manage and schedule case.

1.4 Organization of Report

In Chapter 1: We have considered Project overview under which we have explained various important terminologies like Introduction of the project, purpose to make this system, goals and objectives of project and scope of the project.

In Chapter 2: we have discussed about various paper that we have referred for our project. We have mentioned the description, pros and cons and how the overcome the problem under every paper. Total of three literature and one technical paper have been referred.

In Chapter 3: we have discussed about the requirement analysis under it we have consider about the platform requirements, software and hardware requirement along

with the feasible study.

In Chapter 4: we can see the system design and architecture various diagram are seen in this chapter which represent the software , diagram including our system architecture ,use-case diagram, DFD, class diagram, activity diagram, deployment diagram and component diagram.

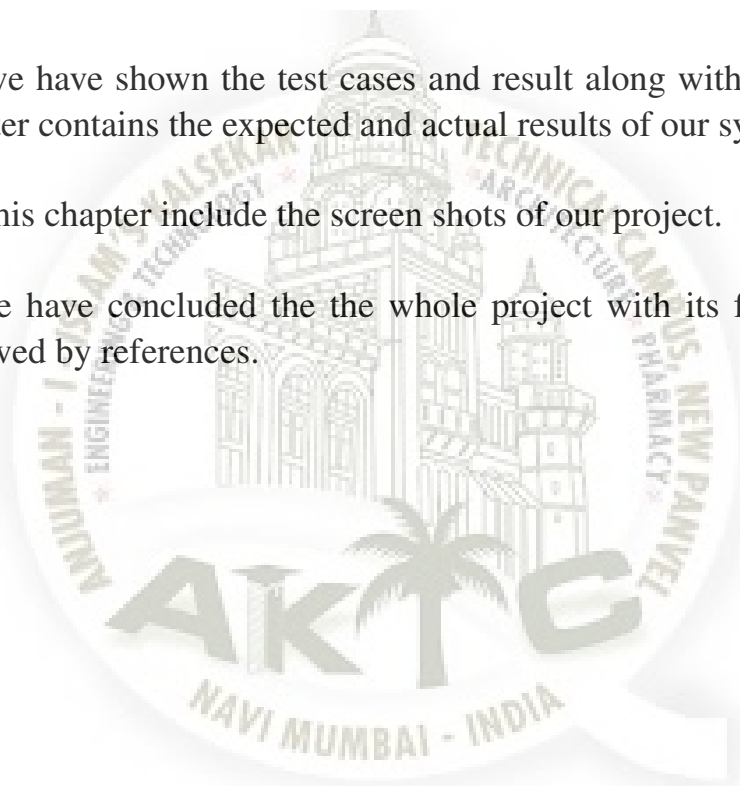
In Chapter 5: we have explain the project in detail by dividing into modules. Various modules of system are Police, Steno-typist, Judge and Lawyer.

In Chapter 6: we have discussed about the implementation details including assumption and constrains.

In Chapter 7: we have shown the test cases and result along with analytic discussion. This chapter contains the expected and actual results of our system.

In Chapter 8: This chapter include the screen shots of our project.

In Chapter 9:we have concluded the the whole project with its future scope and limitation followed by references.



Chapter 2

Literature Survey

2.1 E-Court Technology Diffusion in Court Management

Description: This system is based on following 5 modules:

1. E- Filing system (EFS): Electronic Filing System (EFS) generally allows for case filing via the internet.
2. Queue Management System: It automatically schedule hearing for particular case number.
3. Case Management System (CMS): It performs assigning cases to judges, setting and tracking performance targets etc.
4. Court Recording and Transcribing (CRT): It record the whole process of hearing before judges in the open court, so that the whole court proceeding can be stored in audio video format.
5. Audio and Video conferencing: It is used when one or both lawyers representing a plaintiff and defendant could not come to the court for the case management session.

2.1.1 Advantages of Paper

- a. Efficiency in managing case files, retrieving case information within seconds
- b. Case can be easily registered online.
- c. Review the facial expression of the witnesses or the accused.

2.1.2 Disadvantages of Paper

- a. Audio/video syncing issues and low-resolution images are the major risk.
- b. System is costly.

2.1.3 How to overcome the problems mentioned in Paper

- a. Our project is free of cost
- b. For audio /video conferencing the hardware costly so we are not implementing this feature.

2.2 ICT in Indian Court Challenges Solution

Description:

- based access to authorized users
- Uploading the scanned files/evidence and adding appropriate meta data.
- Allows Judges to see recording of proceeding for review and why case was rescheduled last time.
- Making the knowledge and information content available in 24x7 online environments.
- Appropriate Searching of case records.
- Provision to provide case CD/DVD to authorized person.
- Provision for taking record backup at a specified backup site.
- Live web cast of case proceedings through web portal.
- Can be use by court reporters that missed a word or statement.

2.2.1 Advantages of Paper

- a. No need to carry Case File and evidences at each hearing in physical form as all these available online.
- b. Different courts are able to share the information online.

2.2.2 Disadvantages of Paper

- a. he amount of data that needs to be managed and protect will be a ultimate challenge for the judiciary.

2.2.3 How to overcome the problems mentioned in Paper

- a. To solve this Data compression is needed which is our future scope.

2.3 A secure E-Court System for India

Description: This system is based on following 4 modules:

- **Administrator Module:** Maintain the lawyer profile, random case id generation, maintain databases, user and lawyer authentication, case status, result of the cases.
- **User Module:** Select lawyers based on their requirements.
- **Lawyer Module:** Lawyer will have the authority to accept any case or reject any case, updating personal profile, upload the evidences.
- **Notification Module:** Notifications will be provided to the user regarding their passwords and hearing dates, status and result of case.

2.3.1 Advantages of Paper

- a. Less preparation
- b. Cost savings
- c. Greater revenue

2.3.2 Disadvantages of Paper

- a. A Video conferencing is not accurate because of slow network and it may possible there can be synchronization problem.

2.3.3 How to overcome the problems mentioned in Paper

- a. For audio /video conferencing the hardware costly so we are not implementing this feature.

2.4 Technical Review

Word Net and Cosine Similarity based Classifier of Exam Questions using Bloom's Taxonomy

Description:

The main challenge in the research problem is that the exam questions are not properly categorized and correct weights are not assigned for each category in the mid or final exam questions. To address this challenge this research followed its methodology using revised Bloom's taxonomy. There are few steps used to categorize the questions automatically as described below.

1. Question Extraction: Pypdf package was used to extract exam questions from PDF documents.
2. NLP Processing: Once questions were stored in MySQL database, each question is then tokenized.
3. Verb Extraction: After the completion of tokenization, word correction, limitation and tagging, verbs were extracted for each question and stored in the database.
4. Tag pattern Identification and cosine module: Extracted questions were broken into individual sentences.

2.4.1 Advantages of Technology

- a. Automatically categorization of questions in question paper.
- b. Correct weights are easily assigned to questions.
- c. Cosine similarity is accurate method of calculating similarity.

2.4.2 Reasons to use this Technology

- a. We are using this method to find similarities between cases. Cosine similarity produces a simple summary measure that can be easily used to differentiate between similar texts.
- b. Cosine similarity gives accurate result for text similarities as there are no distributional assumptions to cosine similarity, unlike other word frequency approaches such as Word scores.
- c. Cosine similarity checks semantic similarities between texts.

Chapter 3

Project Planning

3.1 Members and Capabilities

Table 3.1: Table of Capabilities

SR. No	Name of Member	Capabilities
1	Sardar Sobiya	Database, Android Development ,Python
2	Adhikari Shehzeen	UI Design,Web Design
3	Sayyed Rehan	Web Development ,UI Design,
4	Naje Aqeel	UI Design,Documentation

3.2 Roles and Responsibilities

Table 3.2: Table of Responsibilities

SR. No	Name of Member	Role	Responsibilities
1	Sardar Sobiya	Team Leader	Android&Web Development,Python,Database
2	Adhikari Shehzeen	Member	Android Development,Python
3	Sayyed Rehan	Member	Web Development,UI Design
4	Naje Aqeel	Member	UI Design, Web

3.3 Assumptions and Constraints

3.3.1 Assumptions

We assume that the data we provide is purely true because our project is highly depend on data. Data is stored online therefore we assume that data must be secure from the unauthorized user.

3.3.2 Constraints

In our project we are giving scheduling notification through email to judge and lawyer, the main constraint for this is user should give valid email address.

3.4 Project Management Approach

In our project we have used spiral model for implementing all the phases successfully. This model involves strategies, which is a combination of incremental and prototype models. This model is suitable for planning and implementing to achieve the goal of the project. It maintains a systematic step wise approach.

3.5 Ground Rules for the Project

- We treat each other with respect.
- We value constructive feedback. We will avoid being defensive and give feedback in a constructive manner.
- Additional meetings can be scheduled to discuss critical issues or tabled items upon discussion and agreement with the team leader.
- As team members, we will pitch in to help where necessary to help solve problems and catch-up on behind schedule work.
- When we pose an issue or a problem, we will also try to present a solution.

3.6 Project Budget

The budget for this project is very low as most of the tools we have use are open source. Following open source software are used.

- Operating System : Ubuntu 16.04(Open Source).
- IDE : Android Studio(Open Source)
- Database : MySQL(Open Source)

3.7 Project Time-line

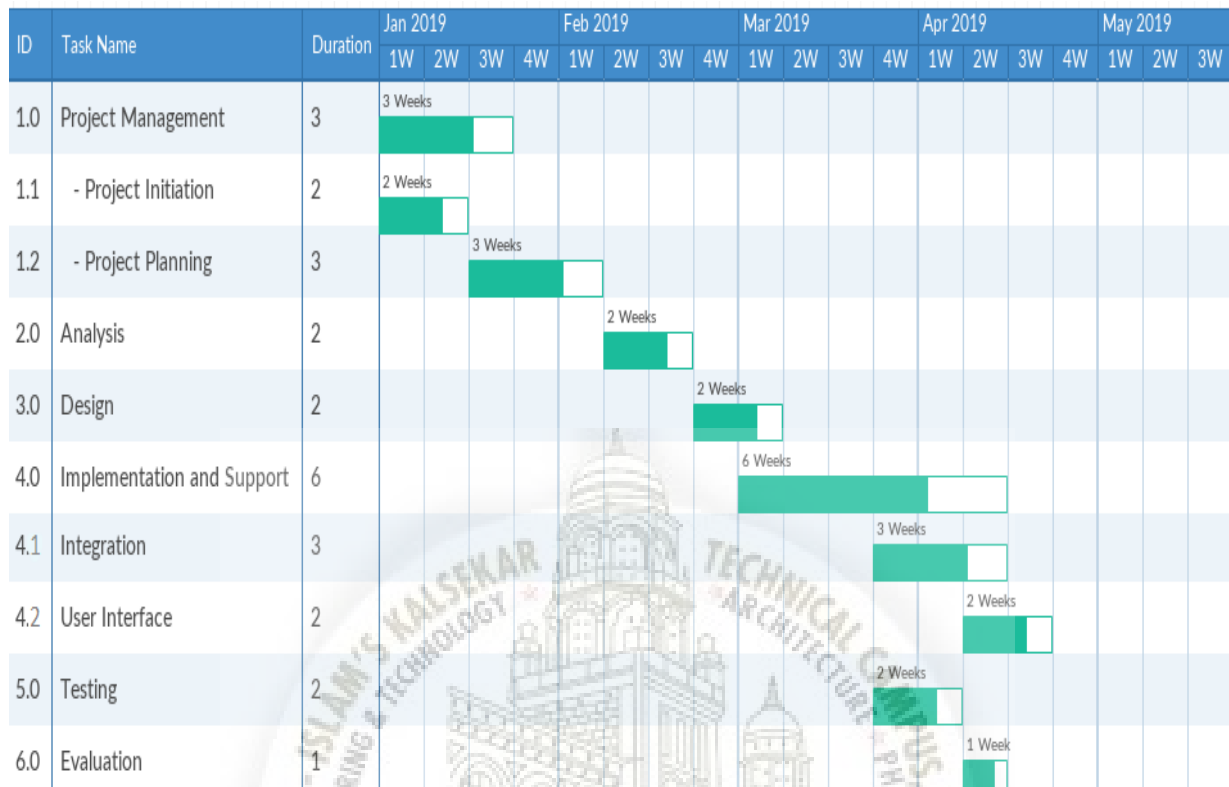


Figure 3.1: Gantt Chart

Chapter 4

Software Requirements Specification

4.1 Overall Description

This Software Requirement Specification is the requirement work product that formally specifies our system. The objectives of this document therefore is to formally describe the system's high level requirements including functional requirement, non-functional requirement business rules and constraints.

4.1.1 Product Perspective

The various system tools that have been used in developing the back-end and other tools of the project are being discussed in this section. The back-end is implemented using php, android, python and MySQL. MySQL is the world second most widely used open source relational database management system. PHP is a server side scripting language designed for web development. PHP code is interpreted by a web server with PHP processor module which generates the resulting web pages. Python is used for various operations here we are using for getting similarities between cases. Android development is one of the major part of our system.

4.1.2 Product Features

- Registration of Police, Steno-typist, Judge and Lawyer in the system.
- Online submission of FIR and Charge-sheet.
- Automatic generation of case id for each case.
- Maintain case history and hearing details online.
- Cases are assigned to judges through system.
- Scheduling for case hearing done through system.
- Scheduling Notification being given to Judges and Lawyers through email.

- Entire case detail including FIR, charge-sheet and history of case are available to judges, lawyers and court.
- Rescheduling of cases are being done easily through system.
- Lawyers are getting similar cases with the current case as strong study material.

4.1.3 User Classes and Characteristics

Different user will use the product differently and the user class related to the app will change according to the need of user. But the characteristics of the classes will remain the same and user will primary interact with three common class of product that is authentication, case detail and case history. These classes will uses by all 3 roles of android app.

4.1.4 Operating Environment

The website is environment friendly,the website run on all the updated browser. An android app must need android mobile to run.

4.1.5 Design and Implementation Constraints

For implementing android app developer must need android studio IDE. For storing data on server MySQL server used. For website implementation developer must need php into there device.

4.2 System Features

The main features of the website are to register FIR and charge-sheets online, fetch sections at the time of FIR and charge-sheet creation, maintaining case history and case detail. The main features of app are to provide case detail and case history to lawyers, judges and court's admin, automatic scheduling of case hearing, automatic assignment of cases to judges, scheduling notification and getting similar cases.

4.2.1 System Feature

1. Automatic generation of case id.
2. Case assignment.
3. Case scheduling.
4. Scheduling notification.

5. Fetch sections.
6. Get similar cases.

Description and Priority

1. **Automatic generation of case id :**

This feature will generate case id automatically. Priority: Medium

2. **Case assignment :**

In this feature cases will automatically assigned to judges through system. Priority: High

3. **Case scheduling :**

In this feature hearing date and time for the case will be automatically decides. Priority: High

4. **Scheduling notification :**

In this feature notification of hearing schedule will be given through email. Priority: High

5. **Fetch sections :**

In this feature sections for the crime will be fetched based on case titles at the time of FIR creation. Priority: High

6. **Get similar cases :**

In this feature lawyer will get details of similar cases with current case. Priority: High

Stimulus/Response Sequences

1. The police and steno-typist need to log-in to the website.
 - If police logged in then he can register FIR, charge-sheet online.
 - If steno-typist logged in then he will type entire conversation of hearing and store online.
2. The judge, lawyer and court's admin needs to log-in to android app.
 - If judge logged in then he can accept or reject any case, get details of cases, get scheduling notification and can reschedule case.
 - If lawyer logged in then he can accept case to handle, get details of cases, get scheduling notification and also similar case detail.
 - If court's admin logged in then he will get case detail and history of any case.

Functional Requirements

Functional requirements are the following:

1. The Automated Jurisdiction System(AJS) should store all information about Police,Lawyer,Judge and Steno-typist.
2. The AJS should store all information of FIR and Charge-sheet into the databases.
3. The AJS allow searching sections by title or keywords.
4. The AJS should allow judge to accept or reject the case.
5. The AJS should allow police to submit FIR into database for further investigation.
6. The AJS should generate case id when a case will arrived.
7. The AJS should create notification and send to judge and lawyer by e-mail automatically before hearing date.
8. The AJS should allow access only to the authorized person.
9. The AJS should allow lawyers to search previous cases which are similar to a particular case which he/she is currently handling.
10. The AJS should store conversation into database server at the time of hearing of case.

4.3 External Interface Requirements

4.3.1 User Interfaces

UI has two interfaces : 1 - Website 2- Android Application

Website : Website is created for two main users of AJS which are Police module and Steno-typist module.it registers police and Steno-typist into the system,creates FIR and charge-sheets whenever the complaint will arrived and store all on going conversation of court into the database.

Website user's menu consists:

1. Registration: here police and steno-typist can get register
2. FIR form: here FIR will be generate through FIR form
3. Search: here sections can be fetch using keywords

4. Charge-sheet form: here Charge-sheet will be generate through charge-sheet form
5. Home: return user to main window
6. Log Out: this function is intended for exit from user's settings.

Application : It is an Android application for two other users Lawyer and Judge. it registers Lawyer and Judge into the system,Assign and schedule cases to judge,notify judge and lawyer before hearing date and fetch previous similar cases for lawyer.

Application user's menu consists:

1. Registration: here judge and lawyer can get register
2. New Cases: information about New Cases
3. Running Cases: information about Running Cases
4. Closed Case: information about Closed Cases
5. Similar Cases: provide previous similar cases to lawyer
6. Home: return user to Home Activity
7. Log Out: this function is intended for exit from user's settings.

4.3.2 Hardware Interfaces

1. For User's Home PC :

- IBM-compatible PC with Pentium III and higher
- 50Bytes free space on HDD
- 32Bytes RAM
- Internet connection

2. For Server :

- IBM-compatible PC with Pentium III and higher
- 256Bytes RAM or higher
- 80Bytes free space on HDD

4.3.3 Software Interfaces

1. For User's Home PC :

- MS Windows 95/98/2000/NT/XP/7/8/10,Linux ubuntu16/ubuntu18/Mint
- Firefox,opera,google chrome,internet explorer

2. For Server :

- PhpMyAdmin server
- Android Studio
- Java Development Kit 1.2 and higher
- Development environment – php,html,css,python,ML,android

4.3.4 Communications Interfaces

The AJS system, consists of three main blocks, each of them is mandatory.

1. **Database :** Database is intended for storing different types of data such as FIR, Charge-sheet, case details, user details, case history, hearing detail etc.
2. **PhpMyAdmin Server :** This server is intended for Database management. It receives commands from AJS and according its demanding take data from database.
3. **API :** This is the core of our system .By means of this function API connects to database server (MySQL server) and generate requests for data issue, data renew, deleting data, fetching of data etc from database.

4.4 Nonfunctional Requirements

4.4.1 Performance Requirements

The performance of the system is based on the data stored and fetch from server. If server will respond slow or will not connect then the system will not work

4.4.2 Safety Requirements

The database should be robust, if any damage will occur in database whole data will be destroy therefore replica should maintain for safety.

4.4.3 Security Requirements

The major security requirements for the system will be the safeguarding of the user data from any kind of exploit. The server on which the Online Data is stored will have its own security to prevent unauthorized write/delete access.

Chapter 5

System Design

5.1 System Requirements Definition

Our system is based on website and android application. Police and steno-typist uses website for storing case information. Lawyer and judge will use android app for managing cases. All the information is stored in MySQL database server. Python is used to search for similar cases.

5.1.1 Functional requirements

Functional requirements are the following:

1. Our system should store all information about Police, Lawyer, Judge and Steno-typist on server.
2. System should store all information of FIR and Charge-sheet into the databases.
3. Sections should be fetch at the time of FIR creation.
4. Judge should accept or reject new case assigned to him.
5. System should generate case id when a case will arrived.
6. System should create notification and send to judge and lawyer by e-mail automatically before hearing date.
7. The AJS should allow access only to the authorized persons.
8. Lawyer should get details of similar cases for any case.
9. System should maintain history of case.

Use-case Diagram

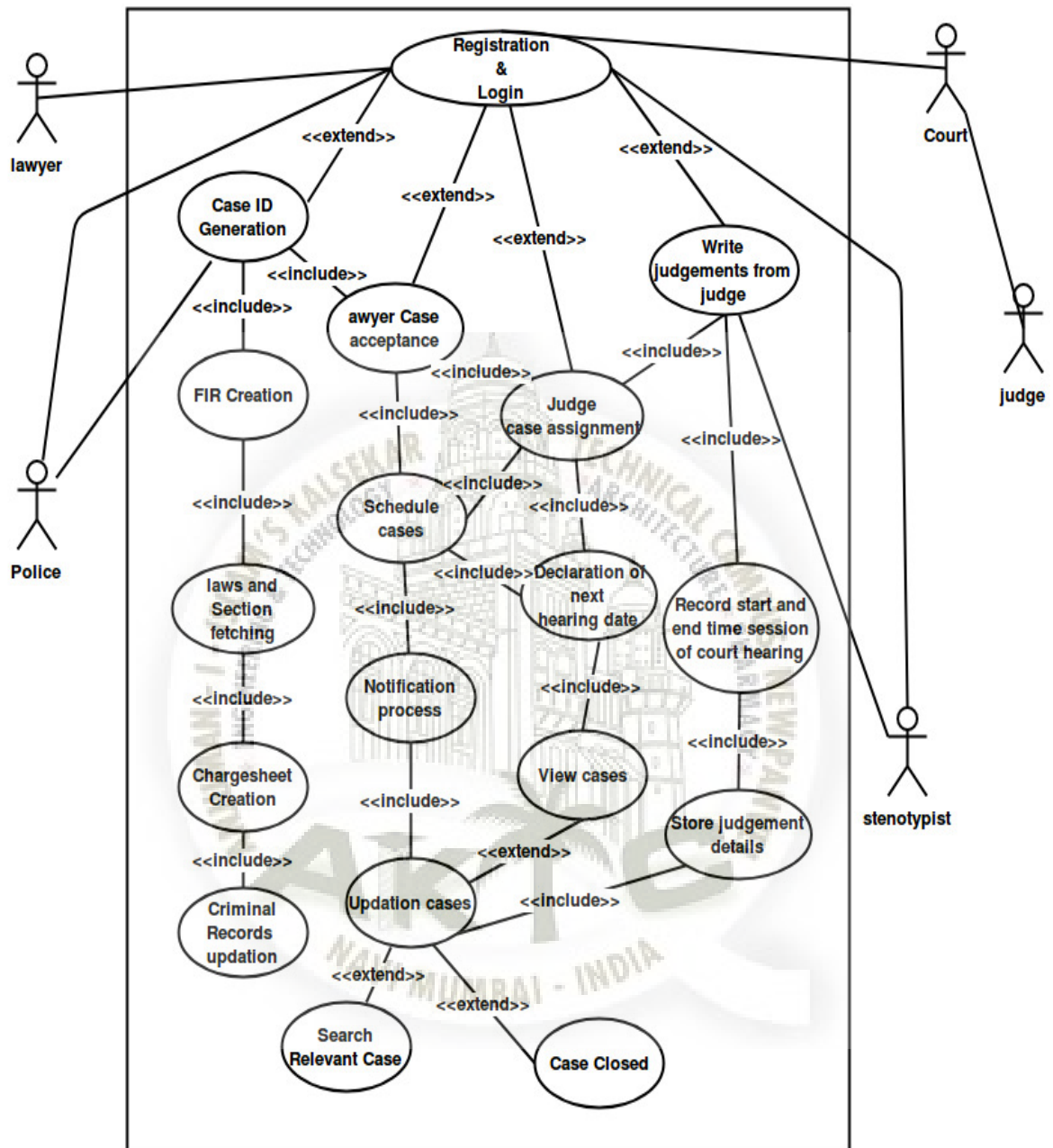


Figure 5.1: Usecase Diagram

Description: There are 4 roles; Police, lawyer, judge and steno-typist. User will register on our system and after log in based on role process will start.

- **Police** start typing FIR and charge-sheet on our website and store online. In police module, fetching of section will be done at the time of FIR and charge-sheet creation.

- **Lawyer** get details of cases and also get previous similar case details in android app. Lawyer will get notification from our system for case hearing.
- **Judge** will have details of all cases in app. Notification will be sent to judge for hearing. Judge will reschedule next hearing date through our system.
- **Steno-typist** type entire conversation/ discussion at the time of hearing on our website. Discussion along with hearing start and end time will be store online.

Data-flow Diagram

LEVEL 0:

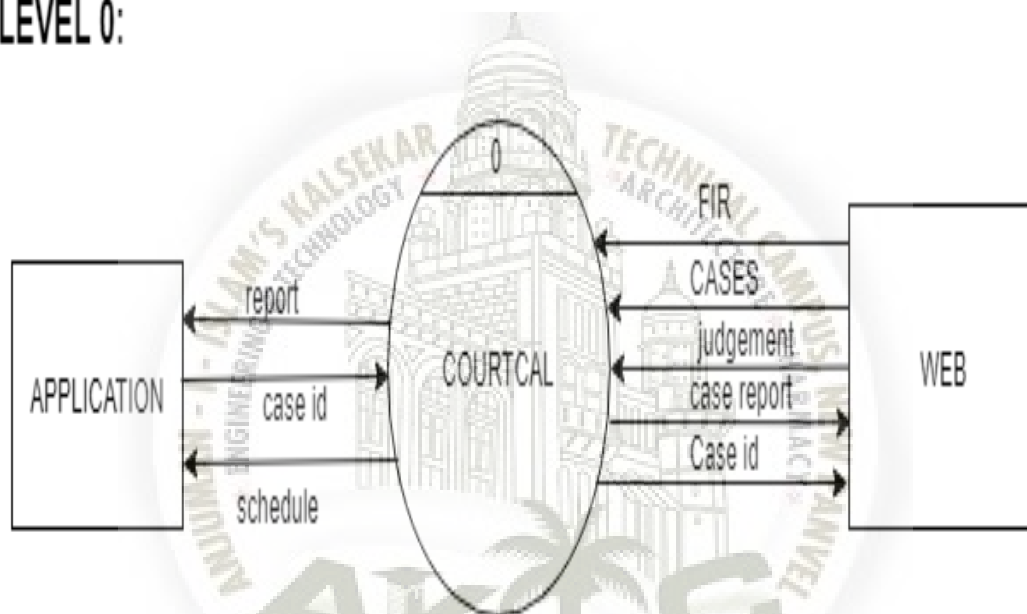


Figure 5.2: Level0 DFD Diagram

LEVEL 1

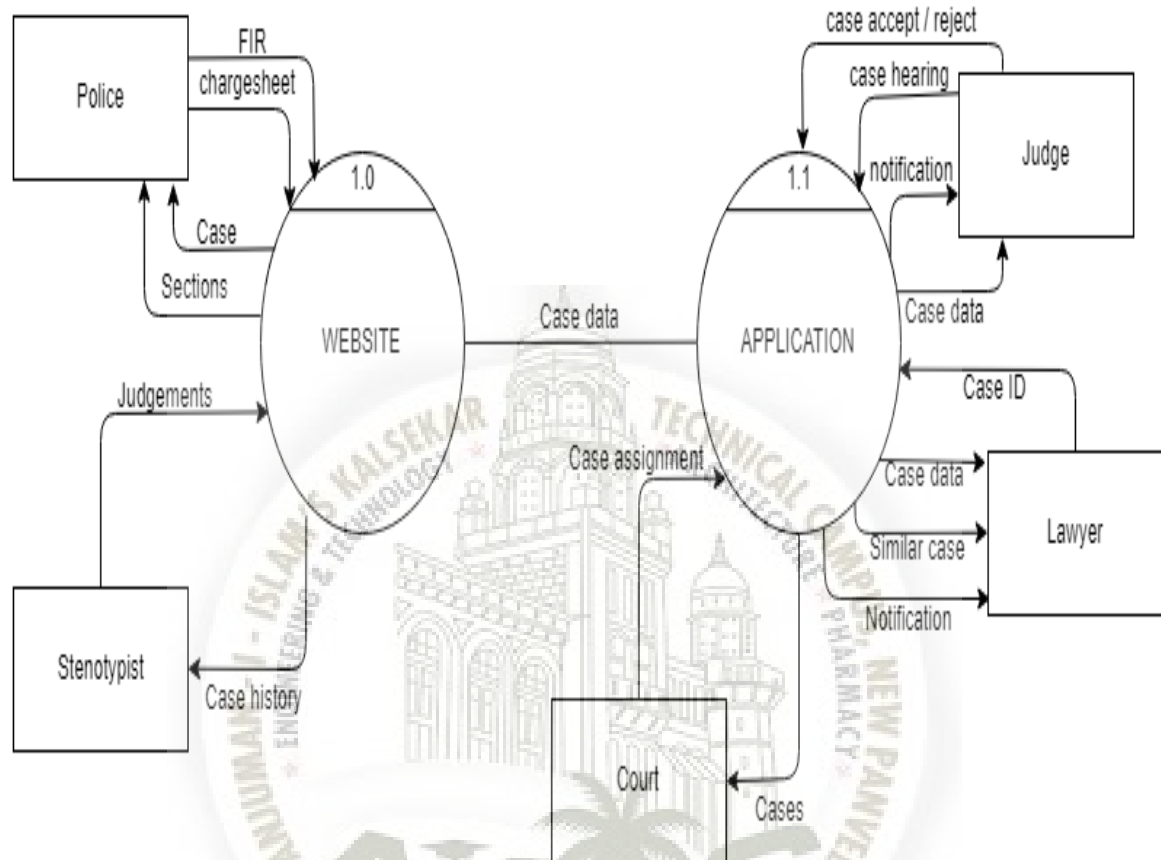


Figure 5.3: Level1 DFD Diagram

LEVEL 2

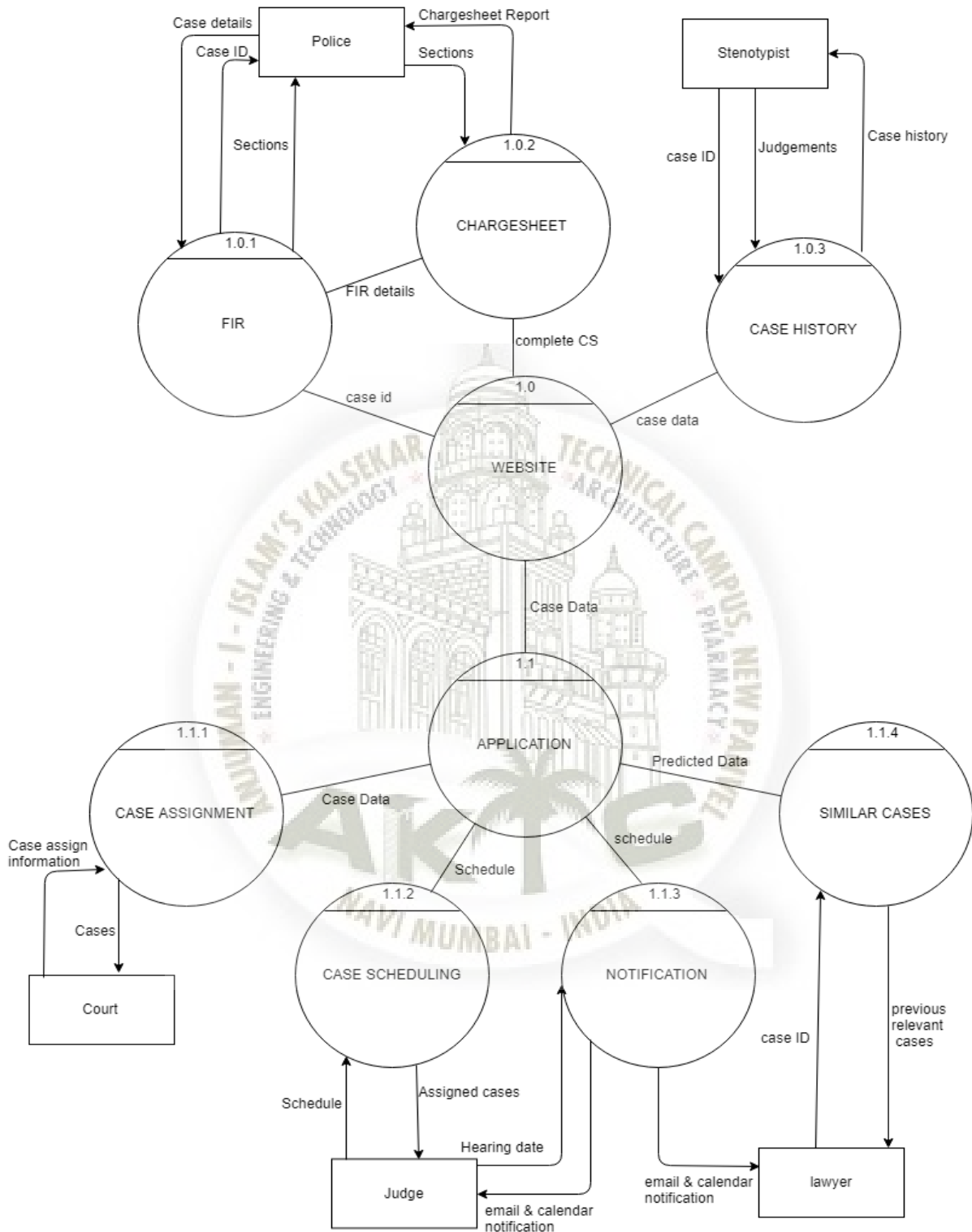


Figure 5.4: Level2 DFD Diagram

These are level-0, level-1 and level-2 DFD of our system which gives information about data flow in the system.

5.1.2 System requirements (non-functional)

Performance Requirements :

The performance of the system is based on the data stored and fetch from server. If server will respond slow or will not connect then the system will not work.

Safety Requirements :

The database should be robust, if any damage will occur in database whole data will be destroy therefore replica should maintain for safety.

Security Requirements :

The major security requirements for the system will be the safeguarding of the user data from any kind of exploit. The server on which the Online Data is stored will have its own security to prevent unauthorized write/delete access.



Database Schema/ E-R Diagram

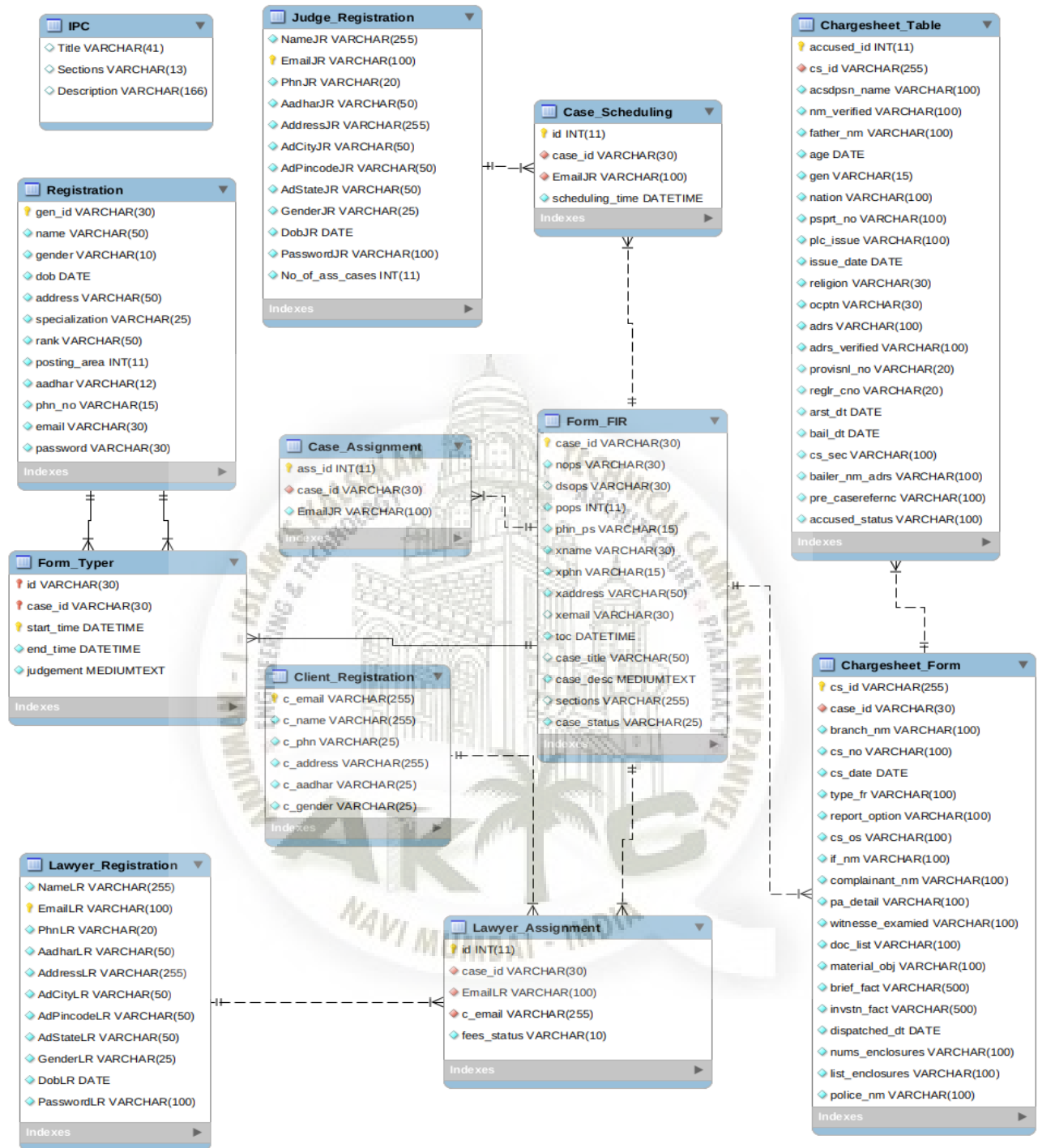


Figure 5.5: ER Diagram

5.2 System Architecture Design

After login police will register the FIR on our website which will store online. At the time of FIR creation case id for the case is automatically generated by the system. Similarly sections will be fetch based on title of case. Fetched section will be verified

by the police. This FIR will be given to judge. From android app judge will either accept or reject the case. If case is accepted by judge then charge-sheet for that case will be created by police on website. At the same time case will schedule for hearing. Hearing notification will be given to judge and lawyer. Case history is created by the steno-typist.

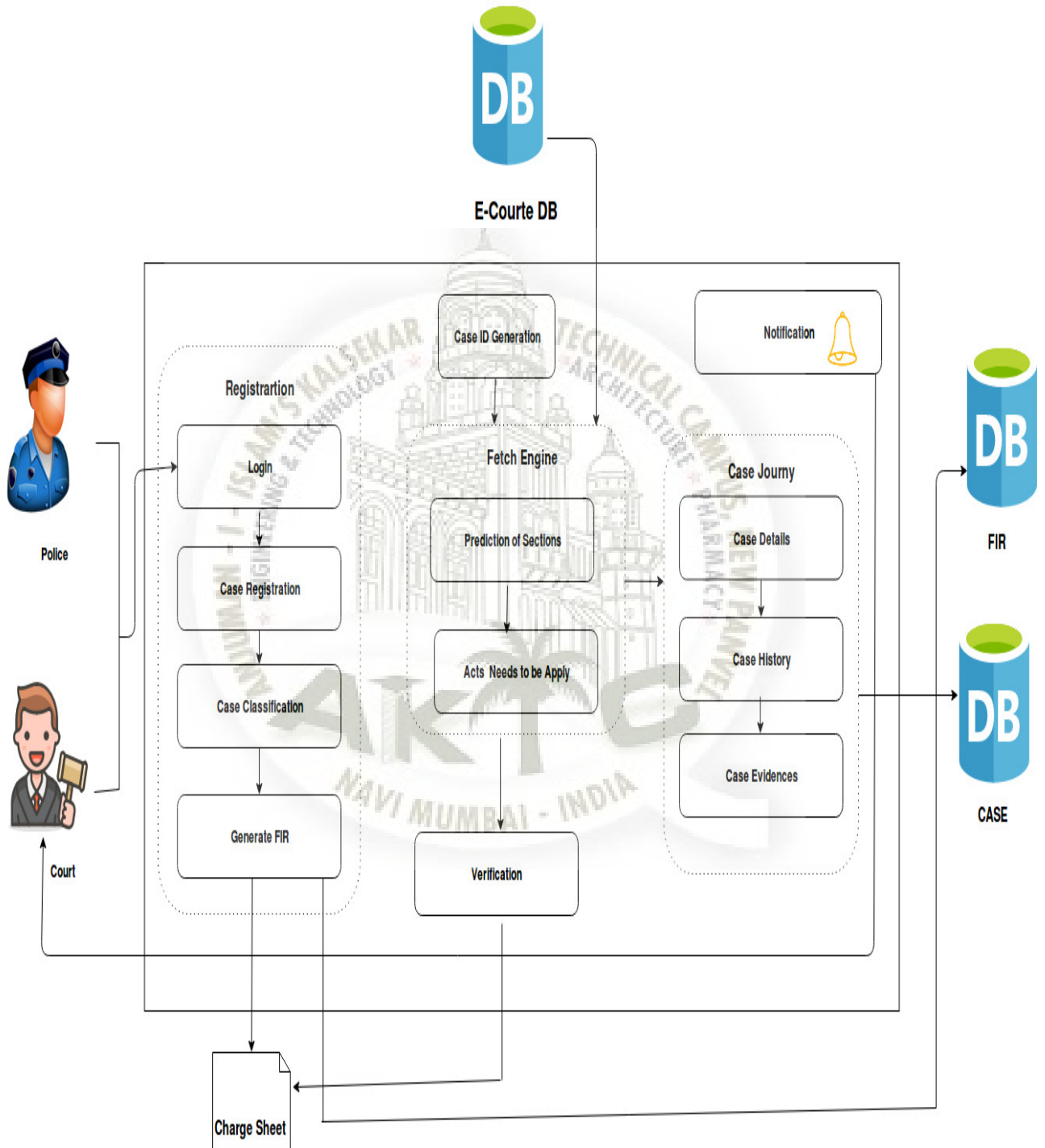


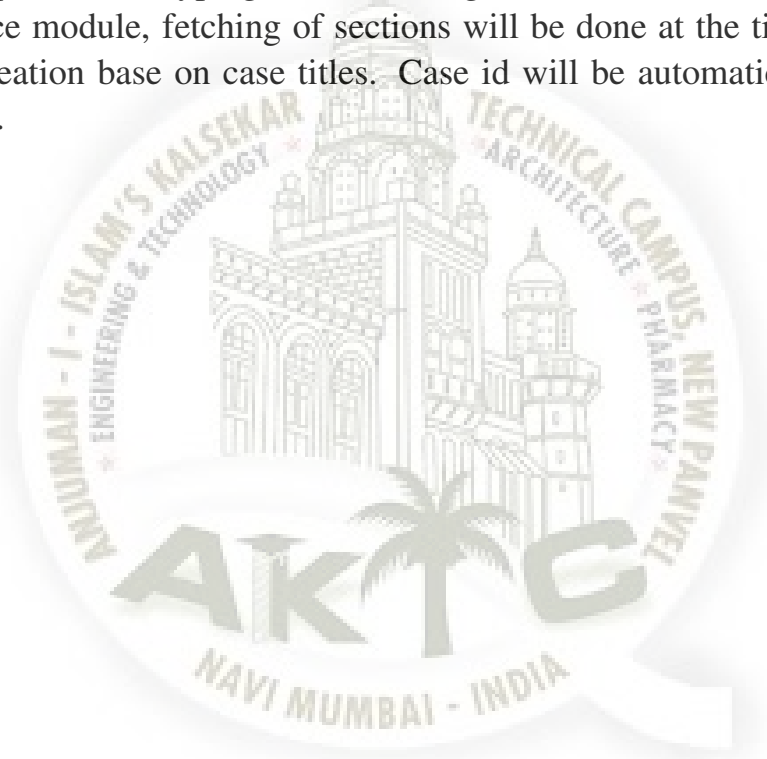
Figure 5.6: System Architecture

5.3 Sub-system Development

There are 4 modules; Police, Steno-typist, Judge and Lawyer. Police have right to register FIR and charge-sheet for the case through our system. Fetching of sections based on case title is done in this module. Steno-typist is responsible of typing entire conversation at the time of hearing which will act as case history for other modules. Judge can easily accept or reject any case assigned to him. After acceptance hearing date will be schedule and notification will be given to judge. Lawyer will have details of all cases. It will easily get similar cases detail as study material.

5.3.1 Module 1 Police

After log-in police start typing FIR and charge-sheet on our website and store it online. In police module, fetching of sections will be done at the time of FIR and charge-sheet creation base on case titles. Case id will be automatically generated through system.



Police Flow Diagram or Modular Diagram

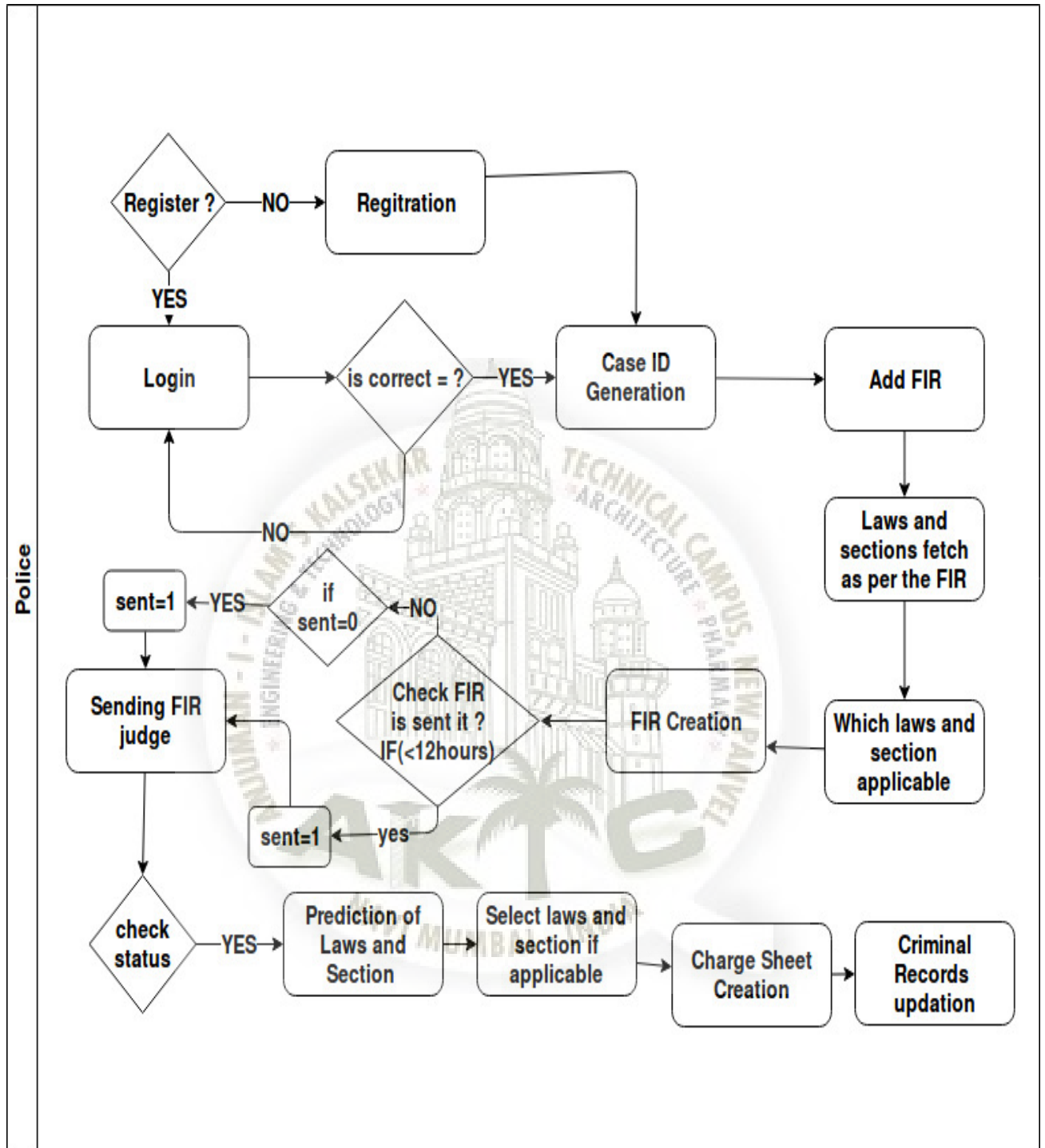


Figure 5.7: Police model diagram

5.3.2 Module 2 Judge

Judge will have details of all cases in app. Judge can accept or reject the case assigned to him. Once case is accept notification will be sent to judge for hearing. Judge can easily reschedule and close the case through app.

Judge Flow Diagram or Modular Diagram

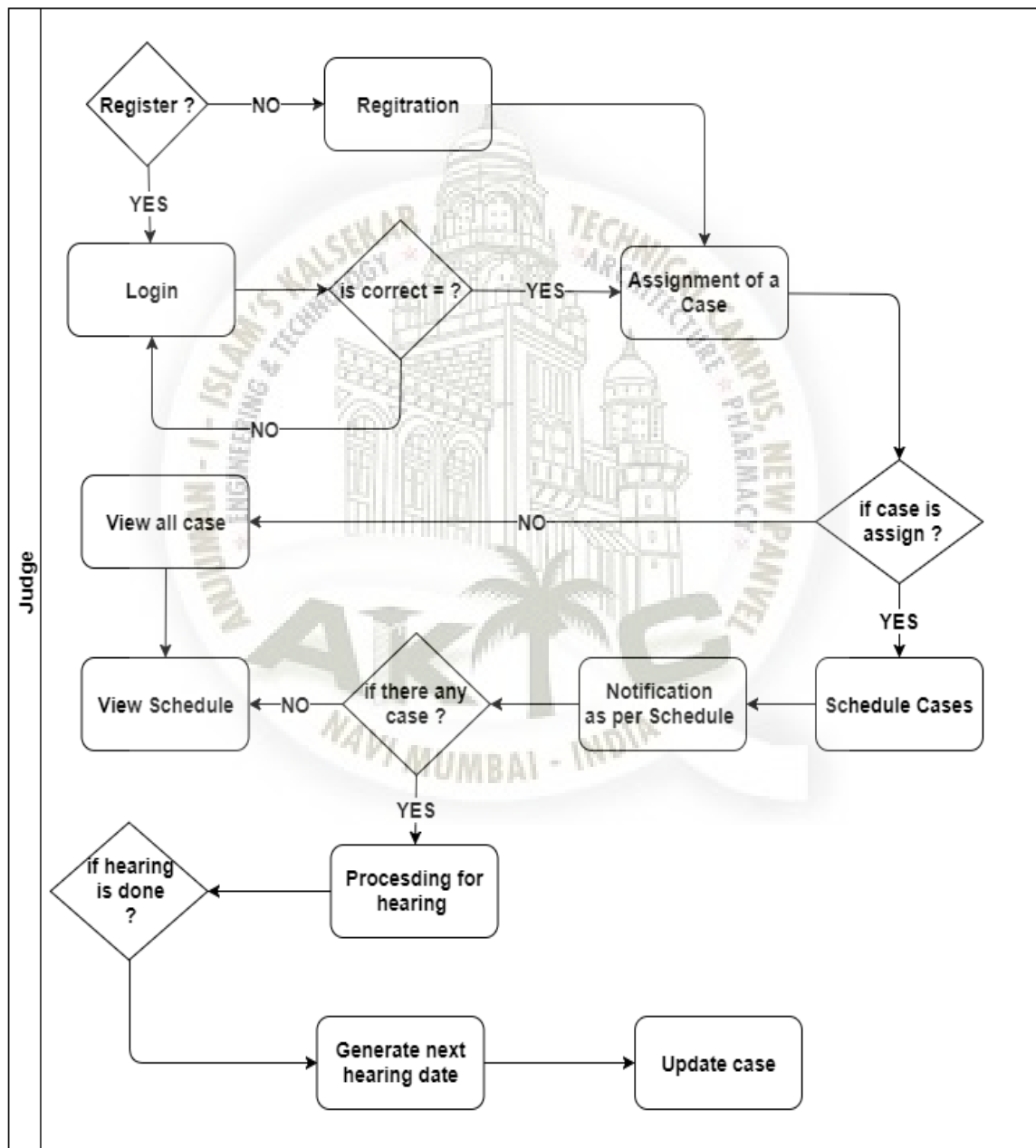


Figure 5.8: Judge model diagram

5.3.3 Module 3 Lawyer

Lawyer will get details of cases and also get previous similar case details in an-droid app. Lawyer will get notification from our system for case hearing.

Lawyer Flow Diagram or Modular Diagram

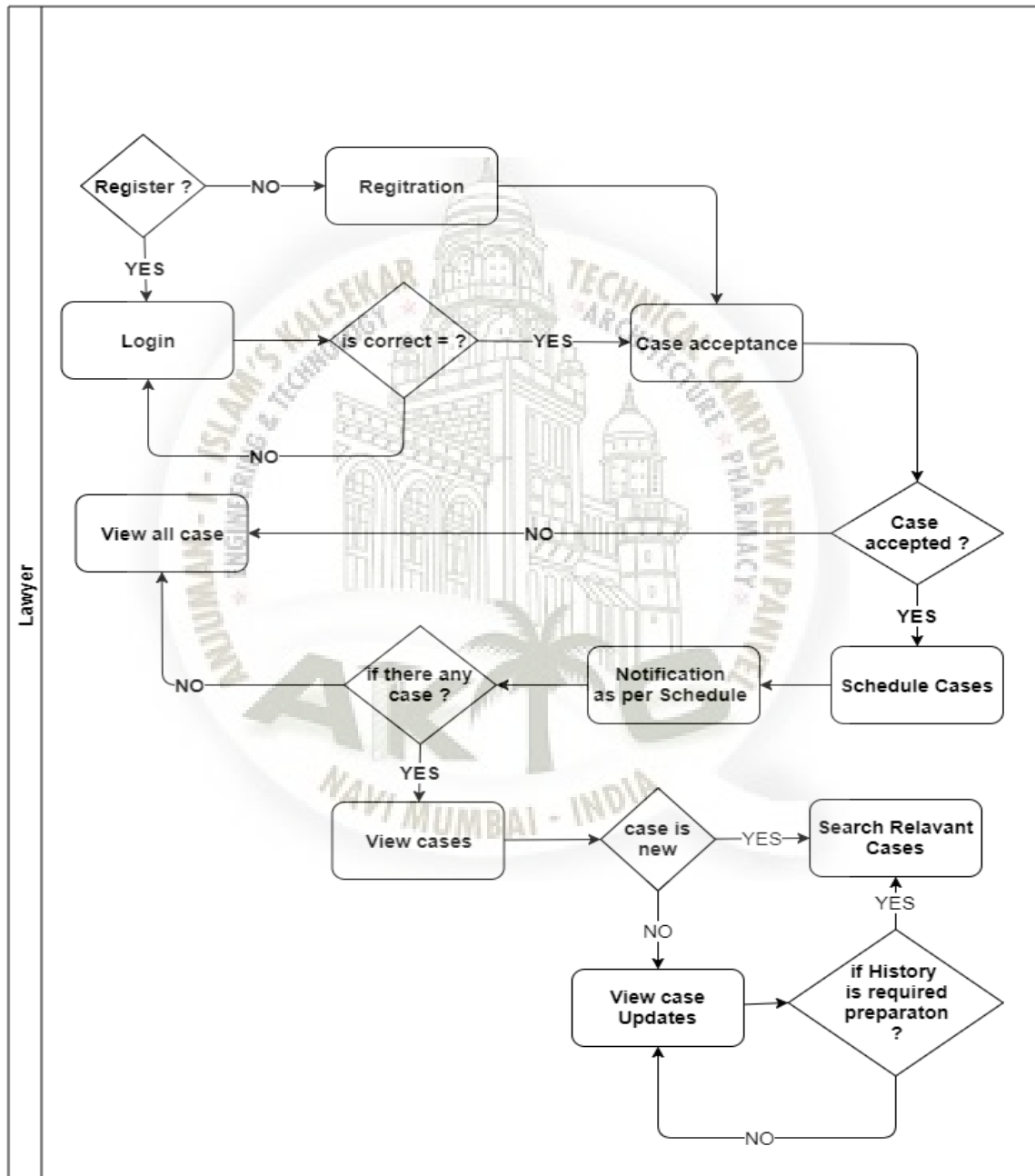


Figure 5.9: lawyer model diagram

5.3.4 Module 4 Steno-typist

Steno-typist type entire conversation/ discussion at the time of hearing on our website. Discussion along with hearing start and end time will be store online.

Steno-typist Flow Diagram or Modular Diagram

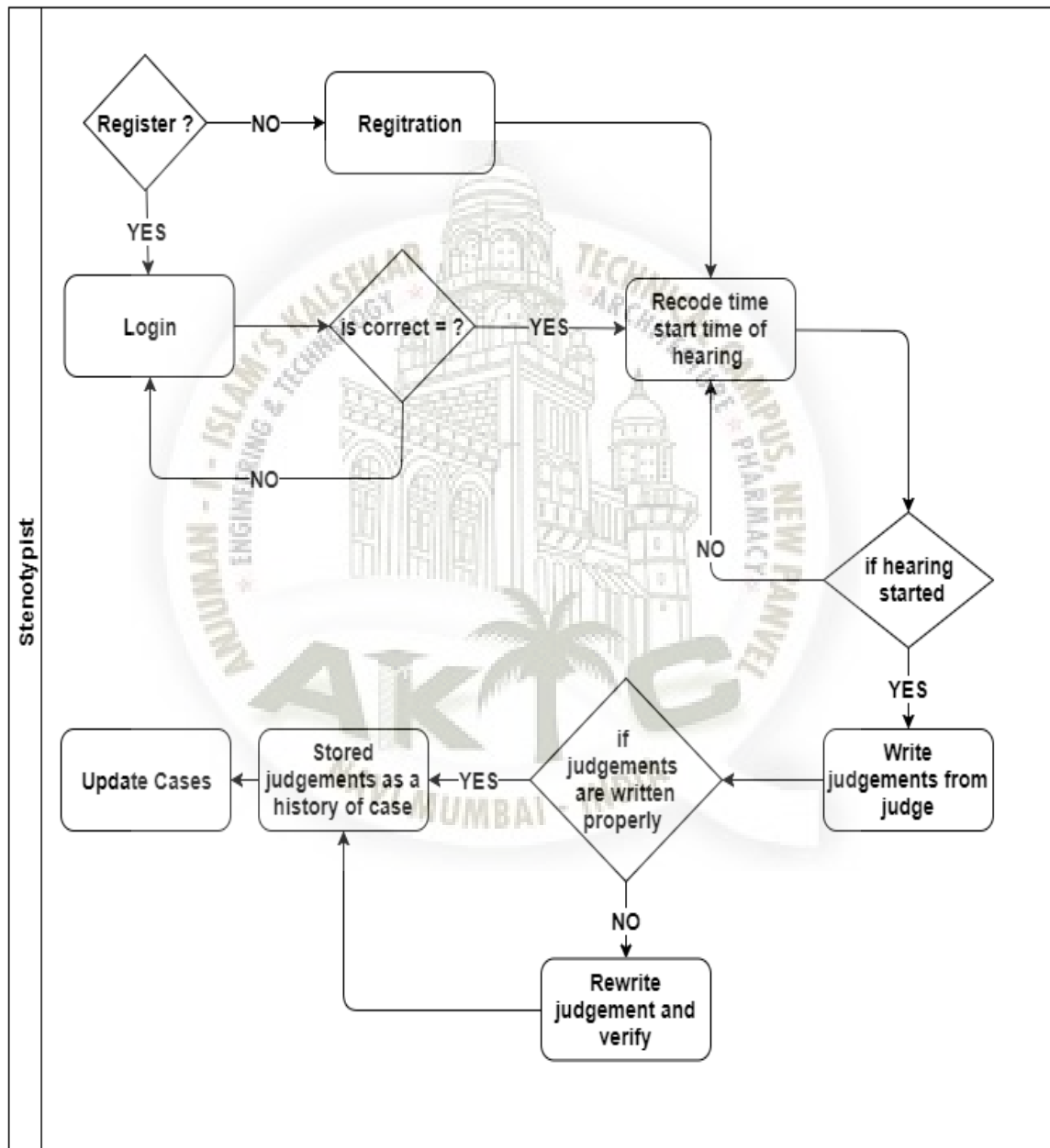


Figure 5.10: Steno-typist model diagram

5.4 Systems Integration

Android and web is integrated with each other and connected with database.

5.4.1 Class Diagram

Class diagram for our system is given below.

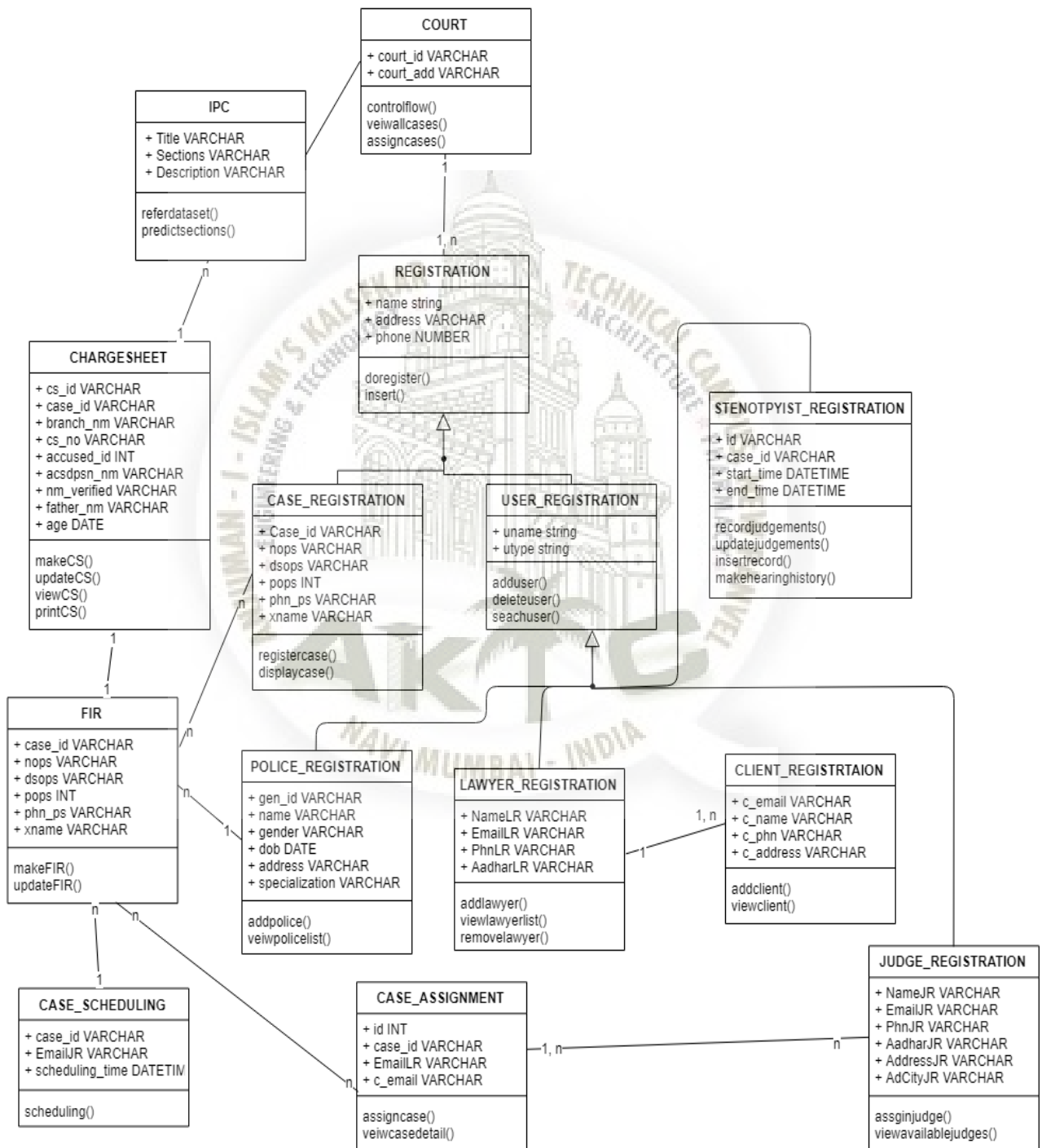


Figure 5.11: Class diagram

5.4.2 Activity Diagram

Activity diagram of the system is given below.

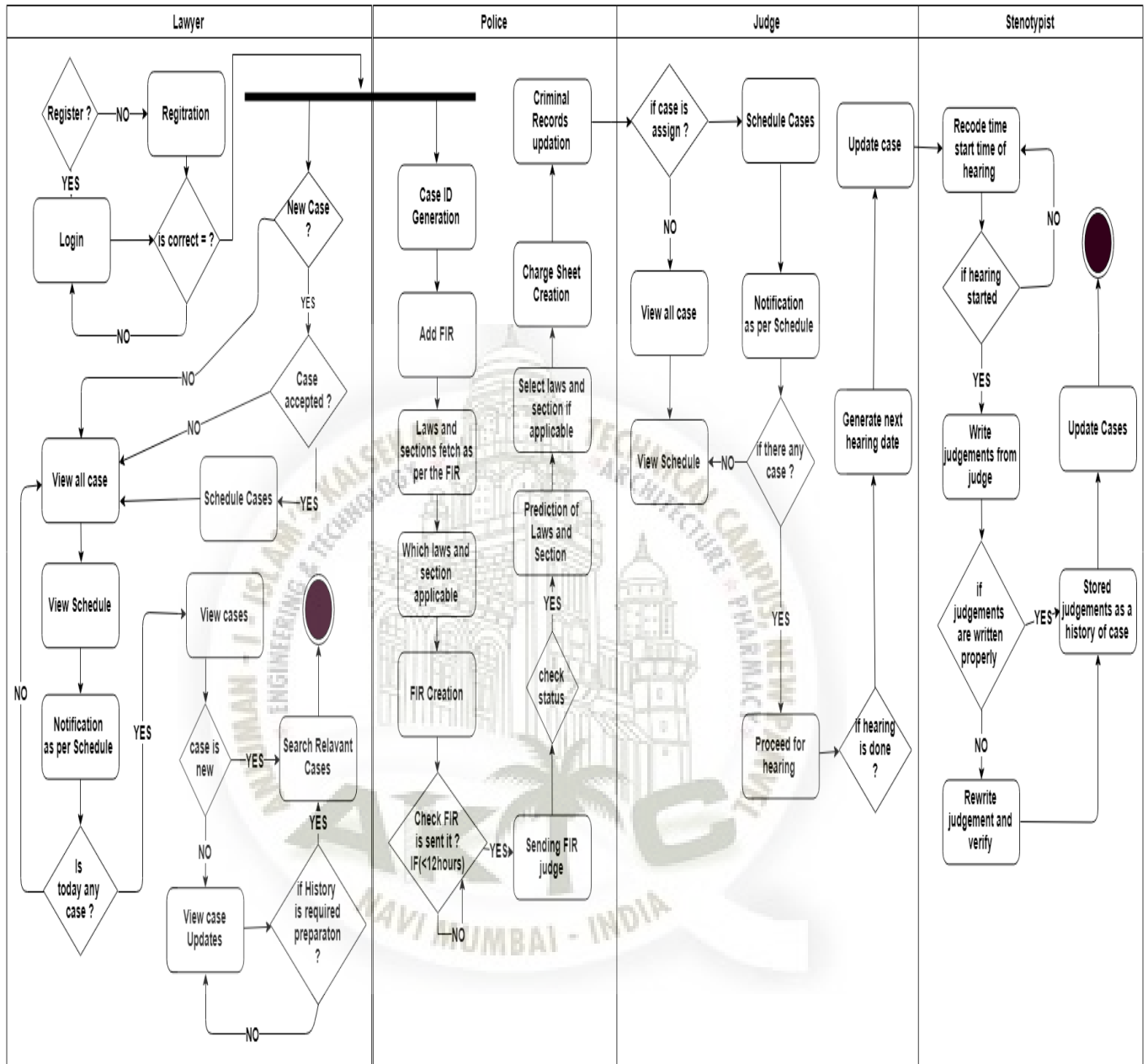


Figure 5.12: Activity Diagram

5.4.3 Component Diagram

Below is component diagram of system.

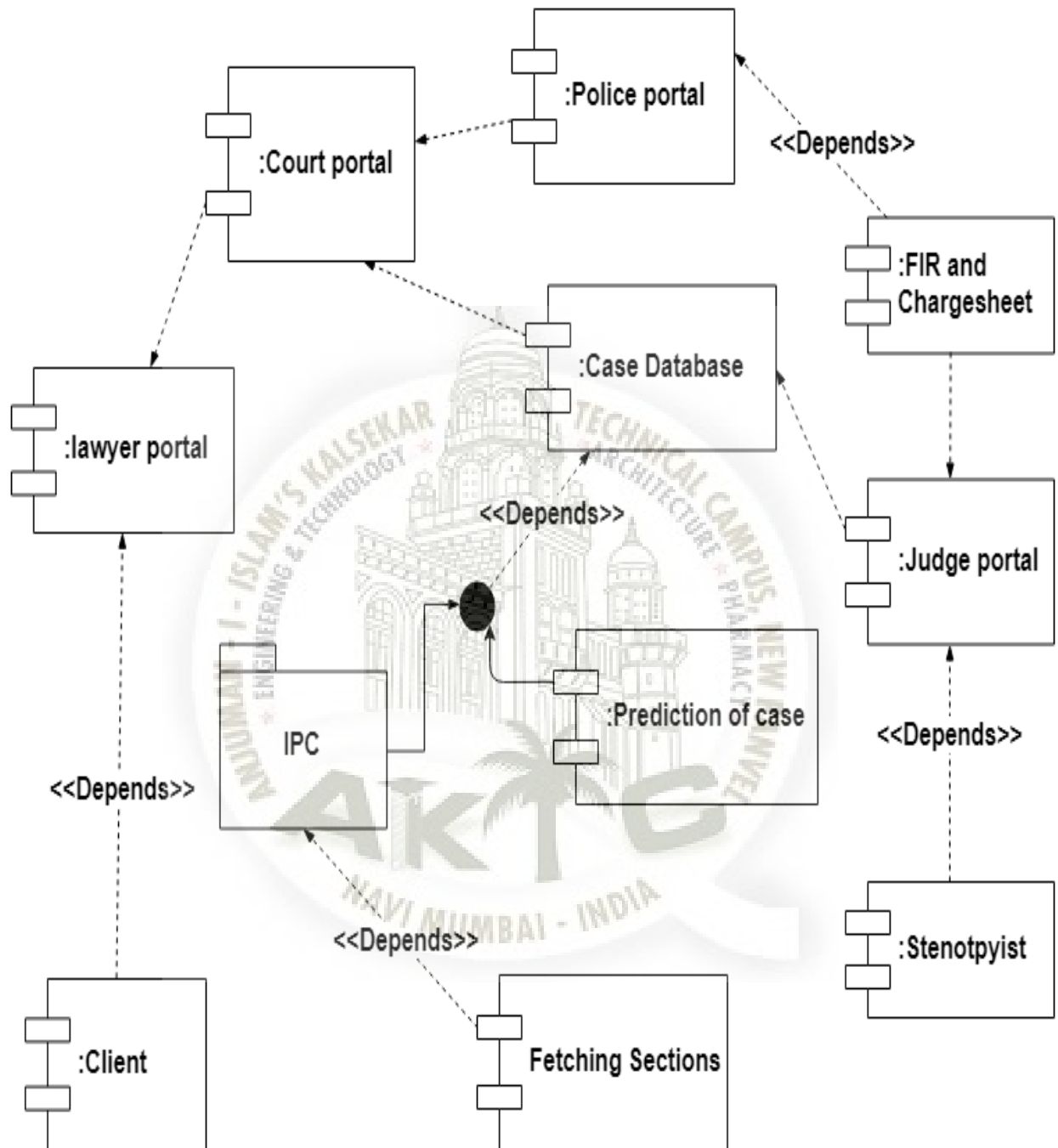


Figure 5.13: Component diagram

5.4.4 Deployment Diagram

Deployment diagram of the system is given below.

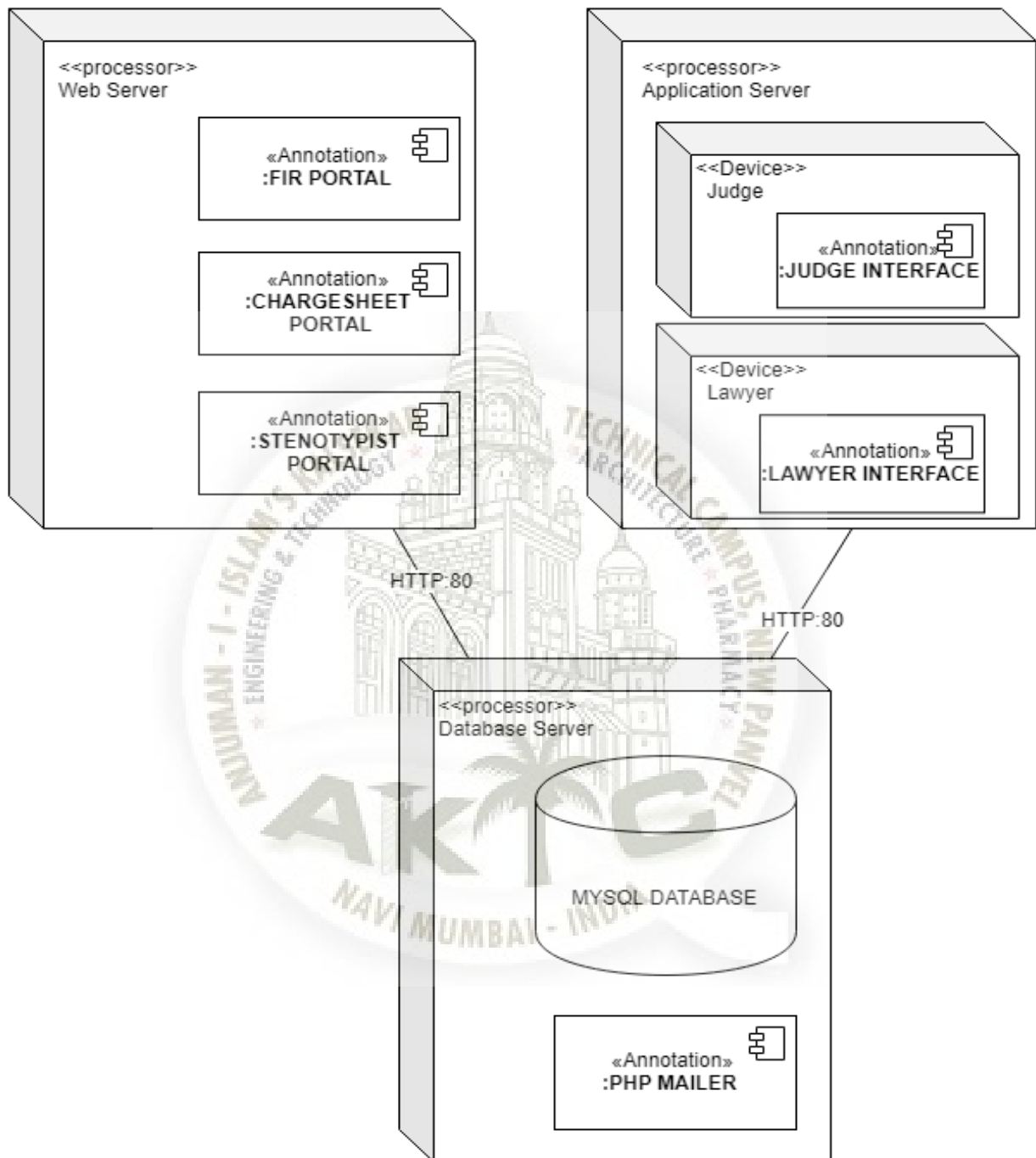


Figure 5.14: deployment diagram

Chapter 6

Implementation

6.1 Police

After log-in police has 3 functions to do. He can register FIR. View the cases which are accepted by judge. Charge-sheet will be generated only for accepted cases. Police can get detail of all accepted and pending cases.

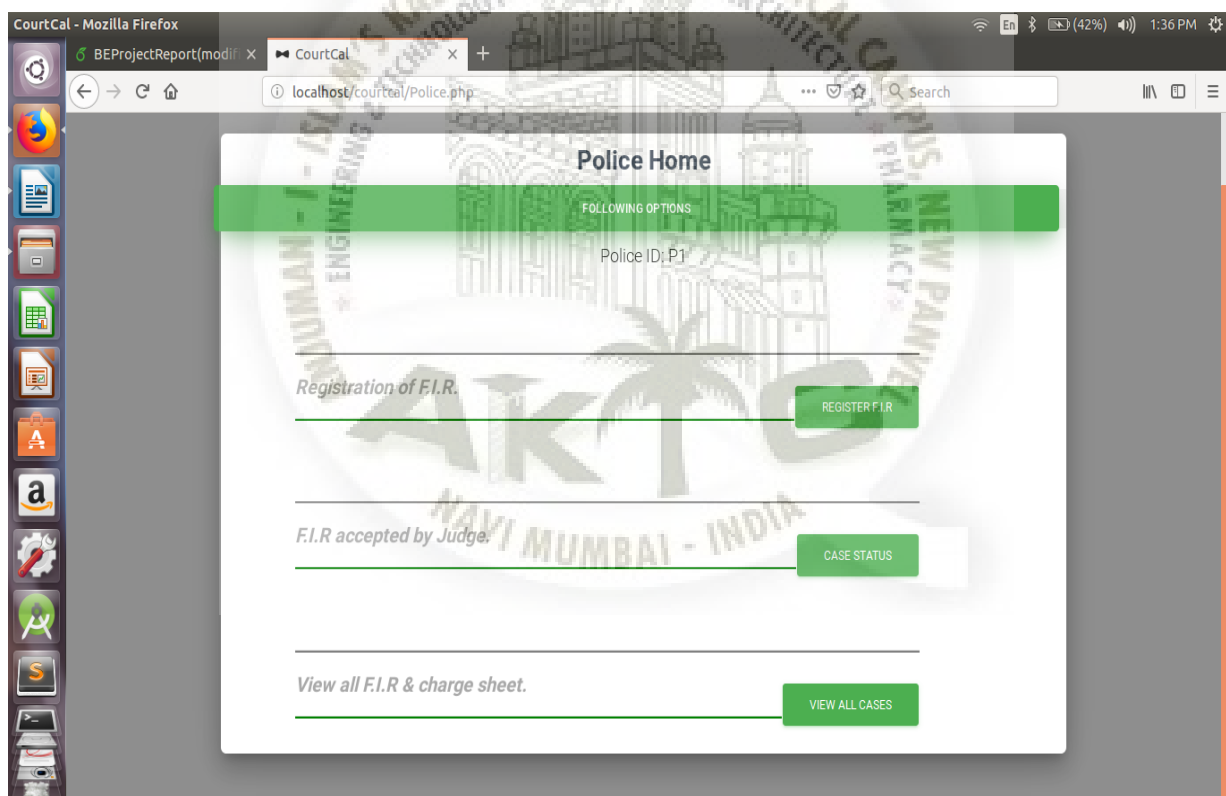


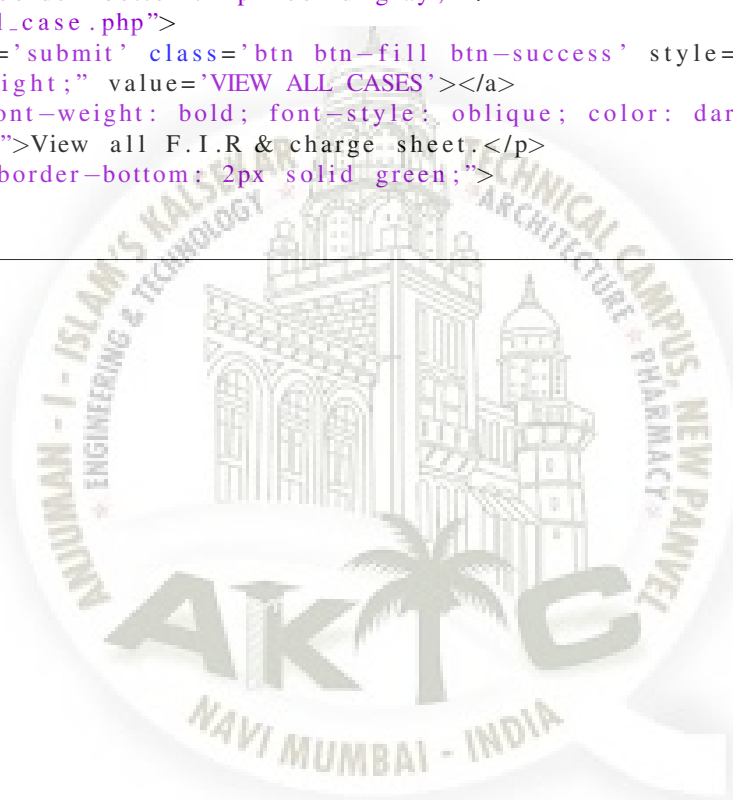
Figure 6.1: Police Home page

```

1 <div class="col-sm-9 col-sm-offset-1">
2   <div class="form-group label-floating"><br/>
3   <hr style="border-bottom: 2px solid gray;">
4   <a href="FIR_Form.php"><input type='submit' class='btn btn-fill btn-success
5     ' style="color: white; float: right;" value='REGISTER F.I.R.'></a>
6   <p style="font-weight: bold; font-style: oblique; color: darkgray; font-
    size: 20px; ">Registration of F.I.R.</p>
  <hr style="border-bottom: 2px solid green;">

```

```
7     </div>
8 </div>
9
10 <div class="col-sm-9 col-sm-offset-1">
11     <div class="form-group label-floating"><br/>
12     <hr style="border-top: 2px solid gray;">
13     <a href="Case_status.php"><input type='submit' class='btn btn-fill btn-
14         success' style="color: white; float: right;" value='Case Status'></a>
15     <p style="font-weight: bold; font-style: oblique; color: darkgray; font-size
16         : 20px;">F.I.R accepted by Judge.</p>
17     <hr style="border-bottom: 2px solid green;">
18     </div>
19 </div>
20 <div class="col-sm-9 col-sm-offset-1">
21     <div class="form-group label-floating"><br/>
22     <hr style="border-bottom: 2px solid gray;">
23     <a href="all_case.php">
24     <input type='submit' class='btn btn-fill btn-success' style="color: white;
25         float: right;" value='VIEW ALL CASES'></a>
26     <p style="font-weight: bold; font-style: oblique; color: darkgray; font-size
27         : 20px;">View all F.I.R & charge sheet.</p>
28     <hr style="border-bottom: 2px solid green;">
29     </div>
30 </div>
```



6.2 Judge

Judge need to log-in to android app to perform his functions. Judge get details of New cases, running cases, closed cases. He will also get scheduling detail.

Judge can accept or reject the case assigned to him by navigating to New cases. Once case is accept notification will be sent to judge for hearing. Judge can easily reschedule and close the case by navigating to Running cases.

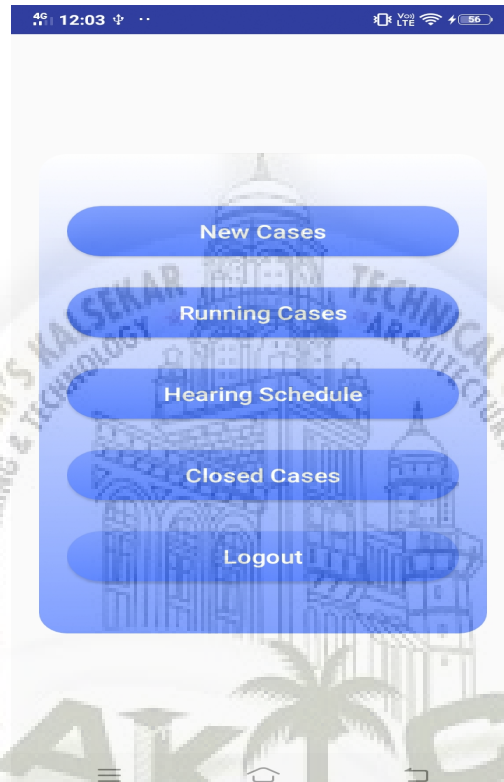


Figure 6.2: Judge Home Page

```

1 public class JudgeHome extends AppCompatActivity {
2     Button judgeLogout, btnJNewCases, btnJRunningCases, btnJUpcomingCases,
3     btnJClosedCases;
4     private Session session;
5
6     @Override
7     protected void onCreate(Bundle savedInstanceState) {
8         super.onCreate(savedInstanceState);
9         setContentView(R.layout.activity_judge_home);
10
11         judgeLogout = (Button) findViewById(R.id.judgeLogout);
12         btnJNewCases = (Button) findViewById(R.id.btnJNewCases);
13         btnJRunningCases = (Button) findViewById(R.id.btnJRunningCases);
14         btnJUpcomingCases = (Button) findViewById(R.id.btnJUpcomingCases);
15         btnJClosedCases = (Button) findViewById(R.id.btnJClosedCases);
16
17         session = new Session(JudgeHome.this);
18
19         judgeLogout.setOnClickListener(new View.OnClickListener() {

```

```
19         @Override
20         public void onClick(View view) {
21             session.delPref();
22             Intent in = new Intent(getApplicationContext(), FrontPage.class)
23                 ;
24             startActivity(in);
25         }
26     });
27
28     btnJNewCases.setOnClickListener(new View.OnClickListener() {
29         @Override
30         public void onClick(View view) {
31             Intent in = new Intent(getApplicationContext(), Judge_NewCases.
32                 class);
33             startActivity(in);
34         }
35     });
36
37     btnJRunningCases.setOnClickListener(new View.OnClickListener() {
38         @Override
39         public void onClick(View view) {
40             Intent in = new Intent(getApplicationContext(),
41                 Judge_RunningCases.class);
42             startActivity(in);
43         }
44     });
45
46     btnJUcomingCases.setOnClickListener(new View.OnClickListener() {
47         @Override
48         public void onClick(View view) {
49             Intent in = new Intent(getApplicationContext(),
50                 Scheduling_Detail.class);
51             startActivity(in);
52         }
53     });
54
55     btnJClosedCases.setOnClickListener(new View.OnClickListener() {
56         @Override
57         public void onClick(View view) {
58             Intent in = new Intent(getApplicationContext(),
59                 Judge_ClosedCases.class);
60             startActivity(in);
61         }
62     });
63
64     @Override
65     public void onBackPressed() {
66         moveTaskToBack(true);
67     }
68 }
```

6.3 Lawyer

Lawyer need to log-in to android app to perform his functions. lawyer will get details of any new case before accepting it, running cases, closed cases. He will also get scheduling detail and details of similar previous cases.

Lawyer can accept the case by navigating to New cases. Once case is accept notification will be sent to lawyer for hearing. Lawyer can easily get similar cases details for running cases and also for other cases by giving case id.



Figure 6.3: Lawyer Home Page

```

1 public class LawyerHome extends AppCompatActivity {
2     Button lawyerLogout, btnLNewCases, btnLRunningCases, btnLUpcomingCases,
3     btnLClosedCases, btnLSimilarCases;
4     private Session session;
5
6     @Override
7     protected void onCreate(Bundle savedInstanceState) {
8         super.onCreate(savedInstanceState);
9         setContentView(R.layout.activity_lawyer_home);
10
11         lawyerLogout = (Button) findViewById(R.id.lawyerLogout);
12         btnLNewCases = (Button) findViewById(R.id.btnLNewCases);
13         btnLRunningCases = (Button) findViewById(R.id.btnLRunningCases);
14         btnLUpcomingCases = (Button) findViewById(R.id.btnLUpcomingCases);
15         btnLClosedCases = (Button) findViewById(R.id.btnLClosedCases);
16         btnLSimilarCases = (Button) findViewById(R.id.btnLSimilarCases);
17
18         session = new Session(LawyerHome.this);

```

```
18
19     lawyerLogout.setOnClickListener(new View.OnClickListener() {
20         @Override
21         public void onClick(View view) {
22             session.delPref();
23             Intent in = new Intent(getApplicationContext(), FrontPage.class)
24                 ;
25             startActivity(in);
26         }
27     });
28
29     btnLNewCases.setOnClickListener(new View.OnClickListener() {
30         @Override
31         public void onClick(View view) {
32             Intent in = new Intent(getApplicationContext(), Lawyer_NewCase.
33                 class);
34             startActivity(in);
35         }
36     });
37     btnLSimilarCases.setOnClickListener(new View.OnClickListener() {
38         @Override
39         public void onClick(View view) {
40             Intent in = new Intent(getApplicationContext(), SimilarCaseHome.
41                 class);
42             startActivity(in);
43         }
44     });
45     btnLRunningCases.setOnClickListener(new View.OnClickListener() {
46         @Override
47         public void onClick(View view) {
48             Intent in = new Intent(getApplicationContext(),
49                 Lawyer_RunningCases.class);
50             startActivity(in);
51         }
52     });
53     btnLUpcomingCases.setOnClickListener(new View.OnClickListener() {
54         @Override
55         public void onClick(View view) {
56             Intent in = new Intent(getApplicationContext(),
57                 Lawyer_Scheduling_Detail.class);
58             startActivity(in);
59         }
60     });
61     btnLClosedCases.setOnClickListener(new View.OnClickListener() {
62         @Override
63         public void onClick(View view) {
64             Intent in = new Intent(getApplicationContext(),
65                 Lawyer_ClosedCases.class);
66             startActivity(in);
67         }
68     });
69
70     }
71     @Override
72     public void onBackPressed() {
73         moveTaskToBack(true);
74     }
75 }
```

6.4 Steno-typist

Steno-typist type entire conversation/ discussion at the time of hearing on our website. Discussion along with hearing start and end time will be store online. This conversation act as case history for particular case.

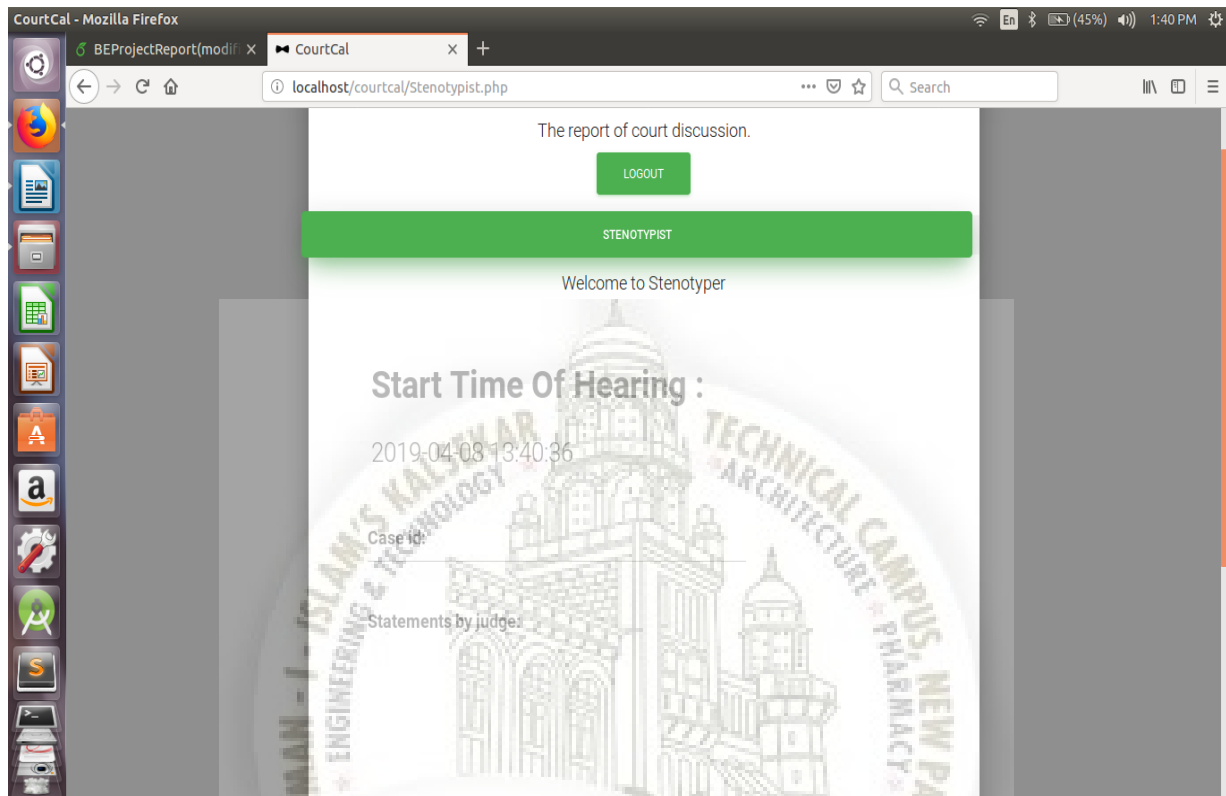


Figure 6.4: Steno-typer Form

```

1  if(isset($_POST['btn_stenotypist'])){
2      $case_id = $_POST['case_id'];
3      $start_time = $_POST['start_time'];
4      $judgement = $_POST['judgement'];
5      $gen_id = $_SESSION['gen_id'];
6
7      $sql = "insert into Form_Typer (id, case_id, start_time, judgement) values
8          ('$gen_id', '$case_id', '$start_time', '$judgement')";
9      $result = mysqli_query($con, $sql);
10     if(!$result){
11         echo "error ";
12         echo mysqli_error($con);
13     } else
14         header('Location: Stenotypist.php');

```

Chapter 7

System Testing

We have done unit testing as well as integration testing. We have done Alpha testing by the person apart from our group. Our product is tested on different PCs and android mobiles.

7.1 Test Cases and Test Results

Test ID	Test Case Title	Test Condition	System Behavior	Expected Result
T01	Login Page (App)	Should be registered user.	Connect to database and check for authentication.	A home page based on selected role.
T02	Login Page (Web)	Should be registered user.	Connect to database and check for authentication.	A home page based on id.
T03	Court's Home Page	Court's admin should be login.	Connect to database and fetch registered cases. Assign new cases to judges.	A page displaying all registered cases.
T04	Accept Case	Judge should be login.	Connect to database and assign date and time for case hearing.	Scheduling notification should given to judge. Case should shift from 'New Cases' to 'Running Cases'.
T05	Reject Case	Judge should be login	Connect to database and set status of case as 'NOT ASSIGNED'.	A page displaying all cases which are assigned to judge except current case.

T06	Reschedule Case	Judge should be login	Connect to database and assign new date and time for case hearing.	New scheduling notification should given to judge. Calendar event for new date should create.
T07	Close Case	Judge should be login	Connect to database and set status of case as 'Closed Case'	Case should be shift from 'Running Cases' to 'Closed Cases'.
T08	Case Detail	Lawyer, Court's admin or Judge should be login	Connect to database and fetch case detail.	A page displaying detail of case.
T09	Case History	Lawyer, Court's admin or Judge should be login	Connect to database and fetch history of case.	A page displaying history of case.
T10	Similar Cases	Lawyer should be login	Run python script and search for similar cases.	A page displaying list of similar cases for current case.

7.2 Sample of a Test Case

Title: Login Page (App) – Authenticate Successfully on courtcal app.

Description: A registered user should be able to successfully login based on there role.Role can be Court, Judge or Lawyer.

Precondition: The user must already be registered with an email address and password.

Test Steps:

1. Start CourtCal Android App.
2. In the 'id' field, type the email of the registered user.
3. In the 'password' field, type the password.
4. Select registered role.
5. Click 'Login'

Expected Result: A home page of court/judge/lawyer based on selected role.

Actual Result: After pressing on 'Login' button user is redirecting to his home page.

Title: Login Page (Web) – Authenticate Successfully on courtcal website.

Description: A registered user should be able to successfully login.

Precondition: The user must already be registered with an id and password.

Test Steps:

1. Navigate to CourtCal Website.
2. In the 'id' field, type the generated id of user.
3. In the 'password' field, type the password.
4. Click 'Login'

Expected Result: A home page of Police/Stenographer based on id.

Actual Result: After pressing on 'Login' button user is redirecting to his home page.

Title: Court's Home Page – All registered cases displaying on admin page.

Description: All the cases which are registered by police through website should be able to display on court's home page.

Precondition: The cases must already be registered.

Test Steps:

1. Navigate to CourtCal Website.
2. Login with police's id and password.

3. Register case by filling FIR form.
4. Click 'Submit' Button.
5. Open CourtCal android app.
6. Login with court's id and password.

Expected Result: A page displaying all registered cases. Allowing user to view the case detail including case history.

Actual Result: After login through court's id and password user is redirecting to his home page where all cases which are registered by police are displaying. For each case, buttons for viewing case detail and case history are available.

Title: Accept Case – Scheduling notification through email is giving to judge. Calendar event before one day of hearing is creating into judge's mobile.

Description: Judge should be able to accept or reject cases which are assigned to him. If judge accept any case then email notification should be given to judge on its registered email address and calendar event for the same should be created. The case should be remove from 'New Cases' list and automatically add into 'Running Cases' list.

Precondition: The judge must already be login into system.

Test Steps:

1. Start CourtCal android app.
2. Login with judges's id and password.
3. Press on 'New Cases'.
4. Select 'Accept' from 2 radio buttons.
5. Click 'Apply' Button.

Expected Result: Email notification of scheduling detail should given to judge and calendar event should create on judge's device. Case

should shift from 'New Cases' to 'Running Cases'.

Actual Result: After pressing Apply button email is sending to judge on registered email address as well as calendar event for the same is creating on judge's device. Now the current case is considering as Running case.

Title: Reject Case – Case is removing from judge's assigned cases.

Description: Judge should be able to accept or reject cases which are assigned to him. If judge reject any case then case should be remove from judge's assigned cases and consider as 'NOT ASSIGNED' case.

Precondition: The judge must already be login into system.

Test Steps:

1. Start CourtCal android app.
2. Login with judge's id and password.
3. Press on 'New Cases'.
4. Select 'Reject' from 2 radio buttons.
5. Click 'Apply' Button.

Expected Result: A page displaying all cases which are assigned to judge except case which is now rejected by judge.

Actual Result: After pressing Apply button the current case is removing from judge's assigned cases therefor removing from 'New Case' list.

Title: Reschedule Case – Scheduling notification through email is giving to judge which include new date of hearing. Calendar event before one day of hearing is creating into judge's mobile.

Description: If previous scheduling date for the case is expired or judge want to change the hearing date then rescheduling is happen. If

judge reschedule any case then email notification should be given to judge on its registered email address and calendar event for the same should be created.

Precondition: The judge must already be login into system.

Test Steps:

1. Start CourtCal android app.
2. Login with judges's id and password.
3. Press on 'Running Cases'.
4. Select 'Reschedule' from 2 radio buttons.
5. Click 'Apply' Button.

Expected Result: Email notification of scheduling detail should given to judge and calendar event should create on judge's device.

Actual Result: After pressing Apply button email is sending to judge on registered email address as well as calendar event for the same is creating on judge's device.

Title: Close Case – Case is shifting from 'Running Cases' to 'Closed Cases'.

Description: If status for the case in the hearing was given as closed then judge can easily set the status as 'Close' from app. If judge close any case then case should be remove from 'Running Cases' list and automatically add into 'Closed Cases' list.

Precondition: The judge must already be login into system.

Test Steps:

1. Start CourtCal android app.
2. Login with judge's id and password.
3. Press on 'Running Cases'.

4. Select 'Close' from 2 radio buttons.
5. Click 'Apply' Button.

Expected Result: Case should shift from 'Running Cases' to 'Closed Cases'.

Actual Result: After pressing Apply button the current case should shift from 'Running Cases' to 'Closed Cases'.

Title: Case Detail – Submitted detail of case is displaying.

Description: By pressing 'VIEW' Button for any case through any of users(judge, lawyer or court), case detail submitted by police is displaying.

Precondition: The user must already be logged in.

Test Steps:

1. Navigate to CourtCal Website.
2. Login with police's id and password.
3. Fill the FIR form.
4. Click 'Submit' Button.
5. Open CourtCal android app.
6. Login with user's id and password.
7. Press on 'VIEW' button.

Expected Result: A page displaying complete case detail.

Actual Result: After login user is redirecting to his home page. After pressing 'VIEW' button, a page is displaying with detail of case.

Title: Case History – Hearing statements of case are displaying.

Description: At the time of hearing, stenotypist type the entire conversation which will act as history. By pressing 'HISTORY' Button

for any case through any of users(judge, lawyer or court), case history submitted by steneotypist is displaying.

Precondition: The user must already be logged in.

Test Steps:

1. Navigate to CourtCal Website.
2. Login with steneotypist's id and password.
3. Type entire conversation for case.
4. Click 'Submit' Button.
5. Open CourtCal android app.
6. Login with user's id and password.
7. Press on 'HISTORY' button.

Expected Result: A page displaying complete case history.

Actual Result: After login user is redirecting to his home page. After pressing 'HISTORY' button, a page is displaying with entire history of case.

Title: Similar Cases – List of cases which are similar to current case is displaying.

Description: Lawyer can get detail of all the cases which are similar to current case. After pressing this button list of similar cases are showing. Allowing lawyer to view detail and history of all similar cases.

Precondition: The lawyer must already be login into system.

Test Steps:

1. Start CourtCal android app.
2. Login with lawyer's id and password.

3. Press on 'Similar Cases' or 'Running Cases'.
4. After pressing 'Similar Cases' button type the case id and then press on 'Get Similar Cases' button Or After pressing 'Running Cases' press on 'SIMILAR CASES' button for particular case.

Expected Result: A page displaying all cases which are similar with the current case.

Actual Result: After pressing 'SIMILAR CASES' or 'Get Similar Case' button list of similar cases are displaying on page.

7.2.1 Software Quality Attributes

1. **AVAILABILITY:** The system should not be down, whenever the user uses the system the specific data should be available to the user.
2. **CORRECTNESS:** As per the user selection the correct data should be shown to the user like if user press view button of case id '2/PW/2019' then case detail of that case should be visible to user.
3. **EXTENSIBILITY:** The system is capable to be modified by changing some modules or by adding some features to the existing system.

Chapter 8

Screenshots of Project

8.1 Web Module And Android App Module



Figure 8.1: Police Login Form

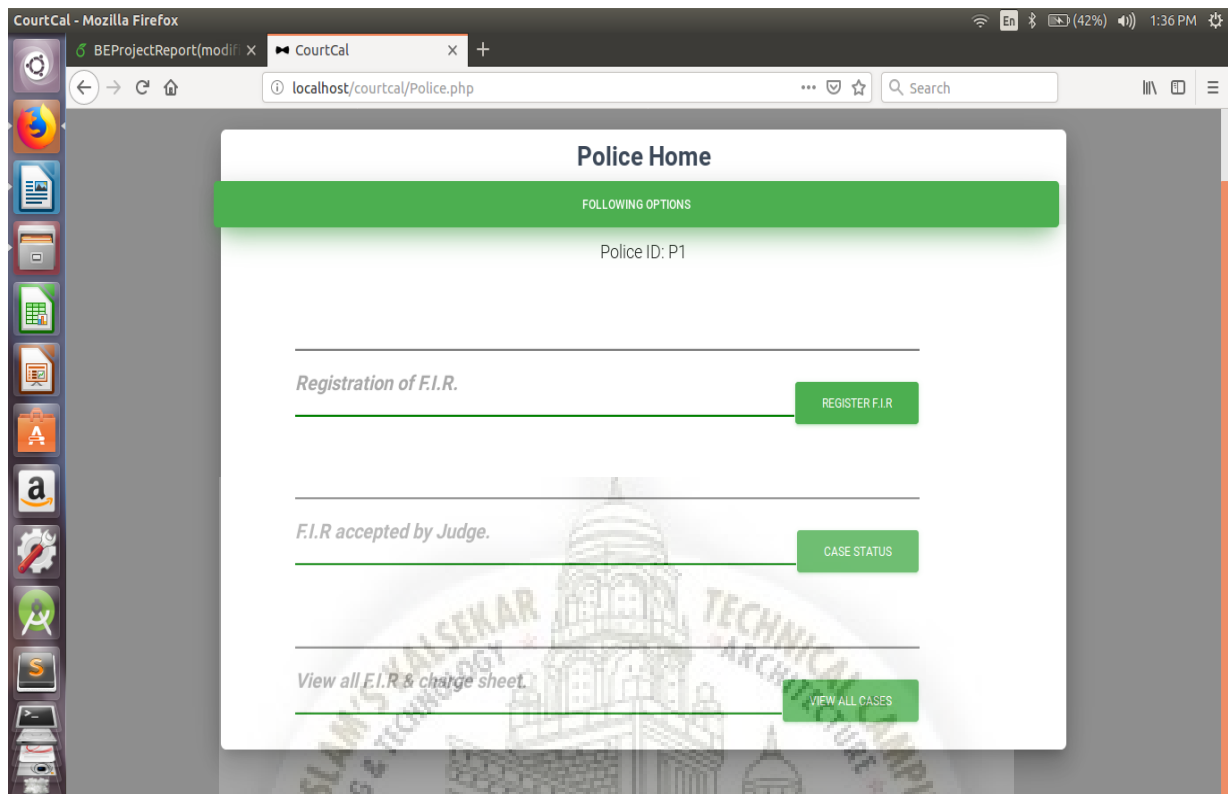


Figure 8.2: Police Home Page

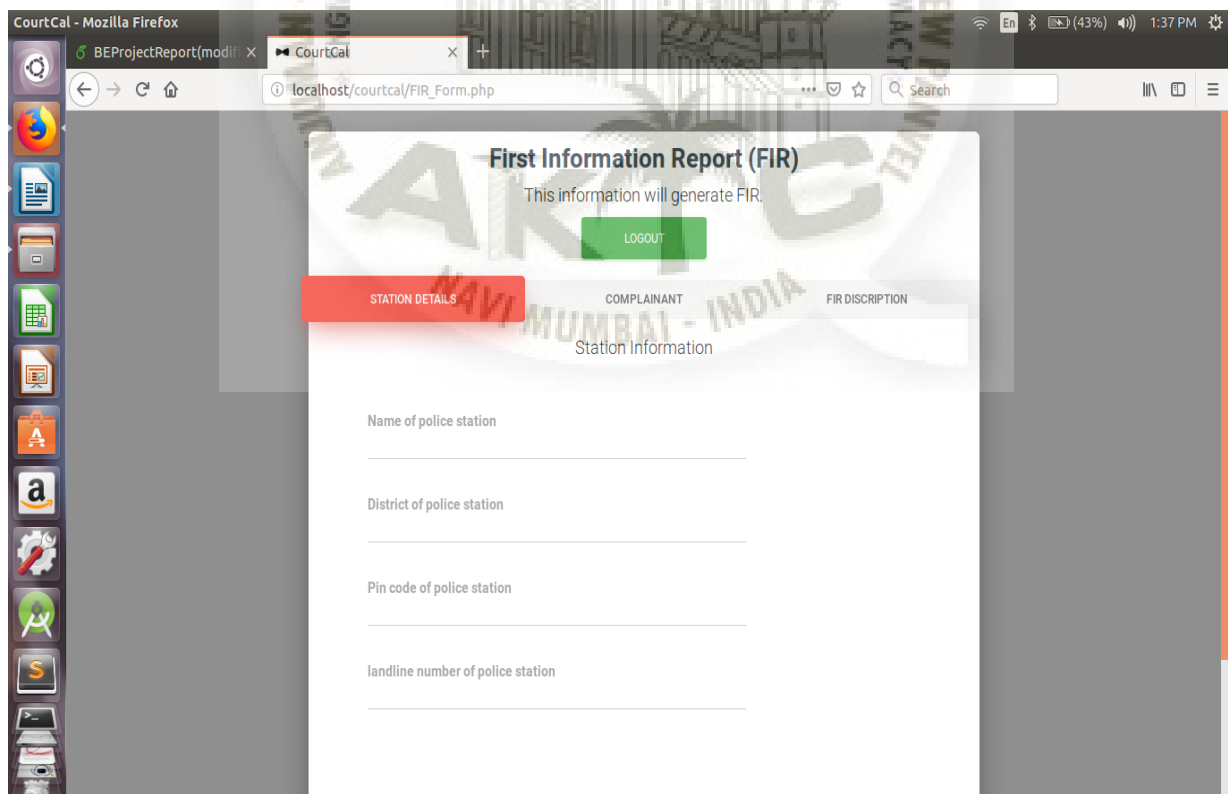


Figure 8.3: FIR Form

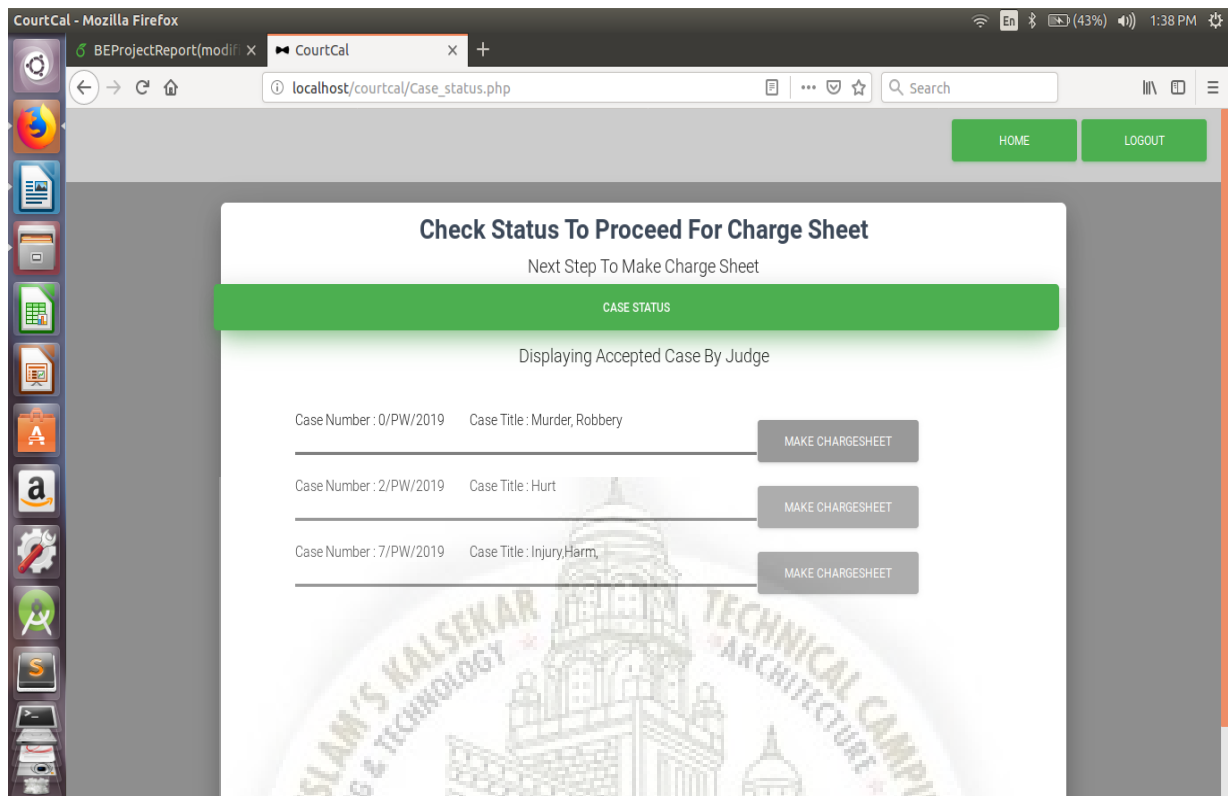


Figure 8.4: Case Acceptance

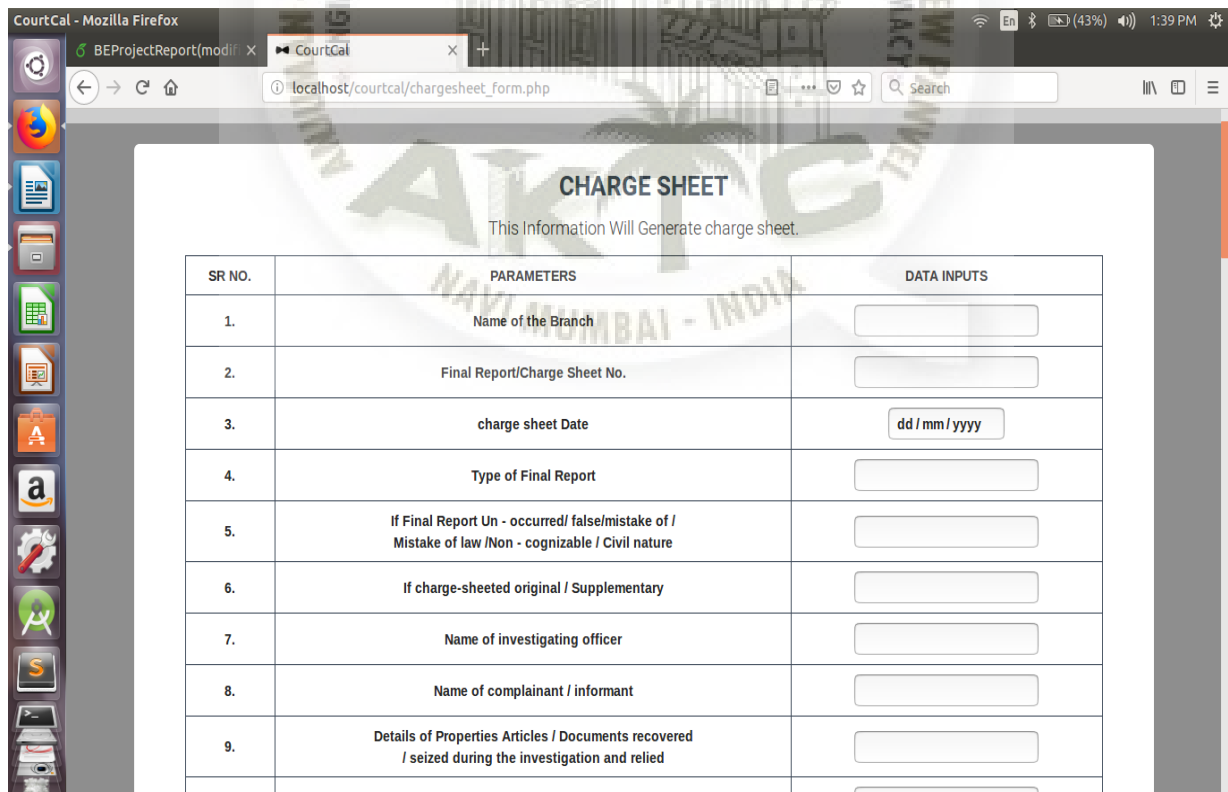


Figure 8.5: Charge-sheet Form

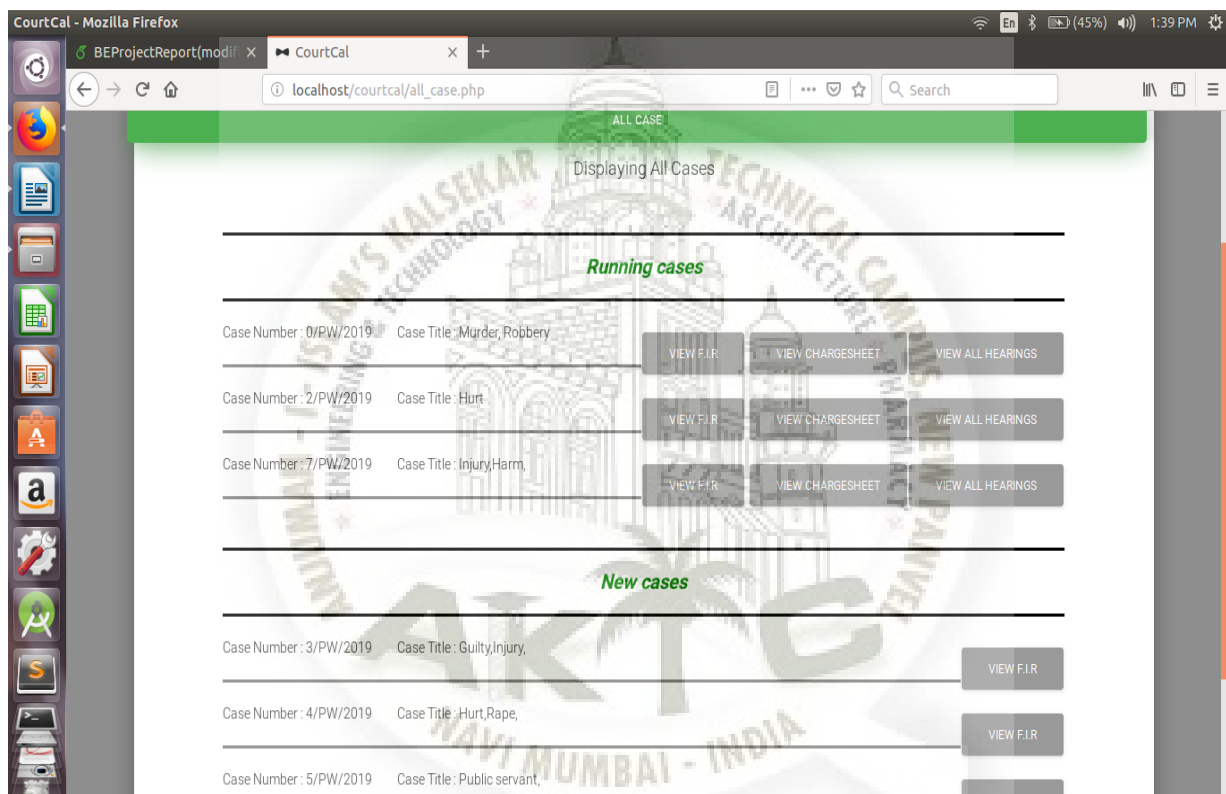


Figure 8.6: All Cases

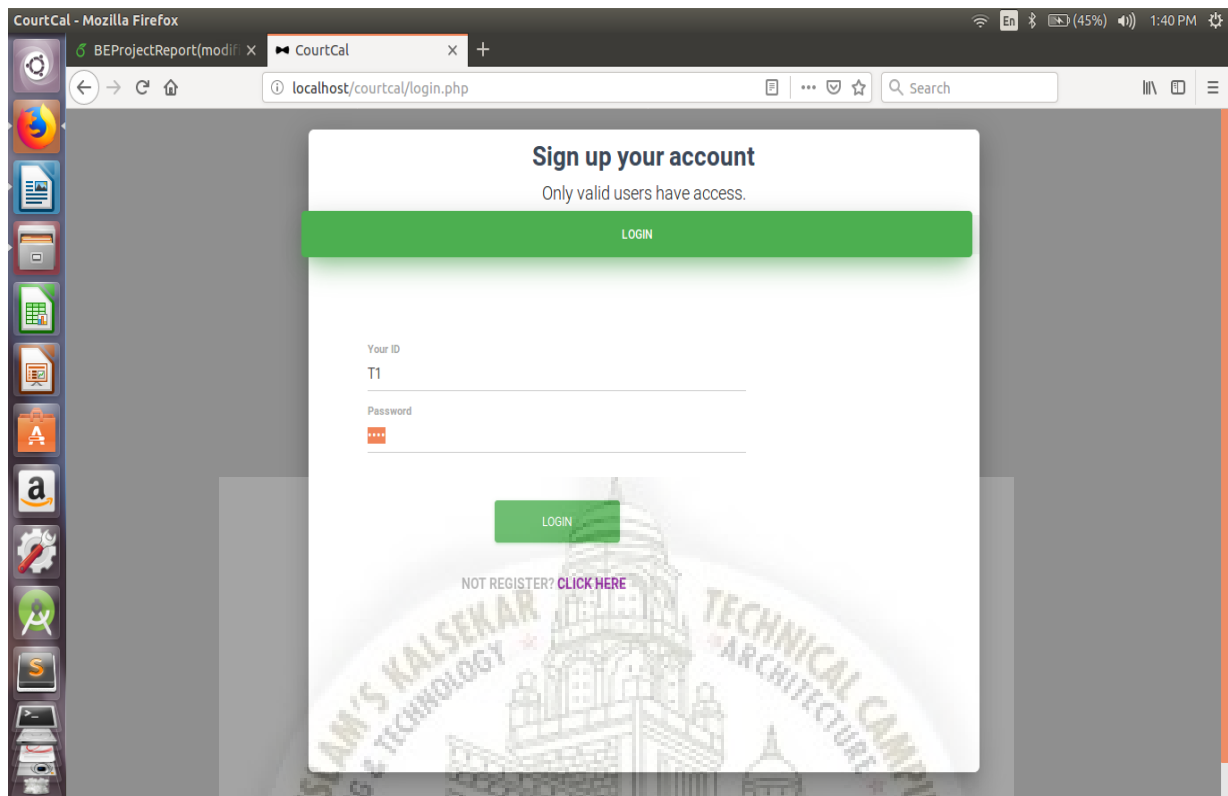


Figure 8.7: Steno-typist Login Form

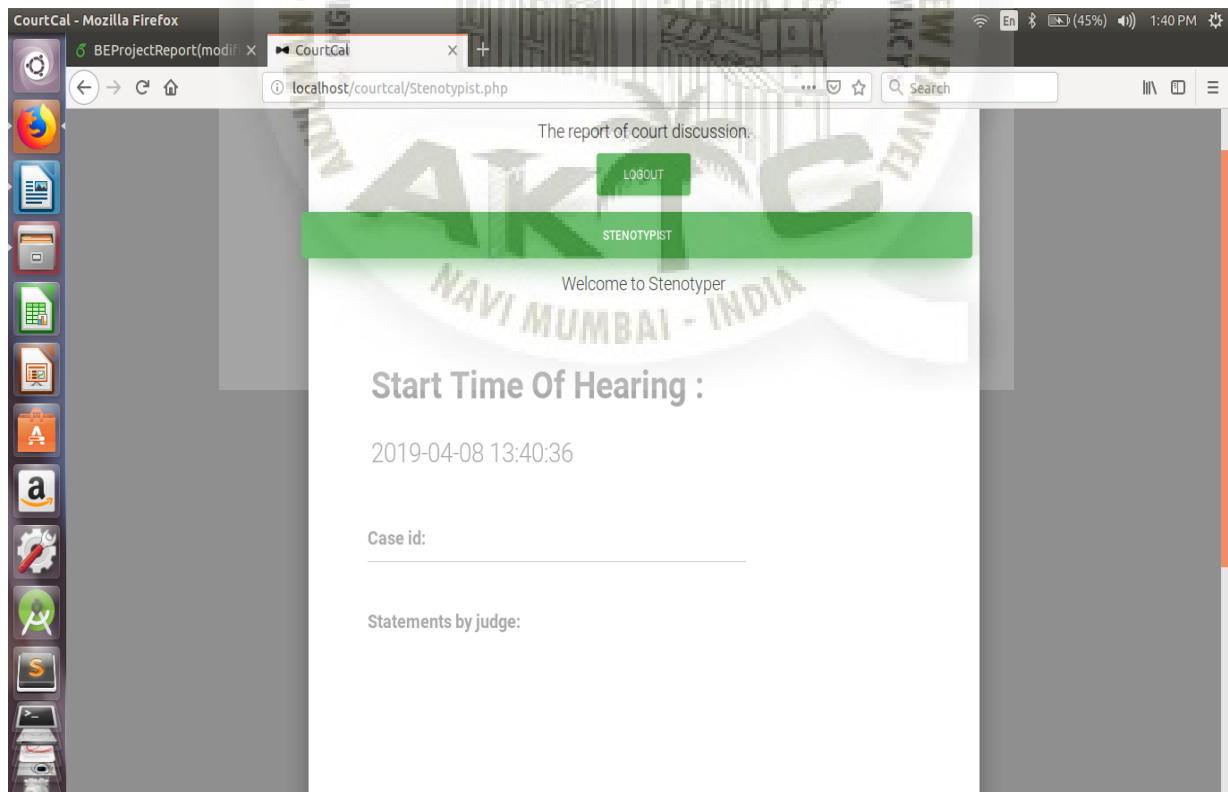


Figure 8.8: Steno-typist Form

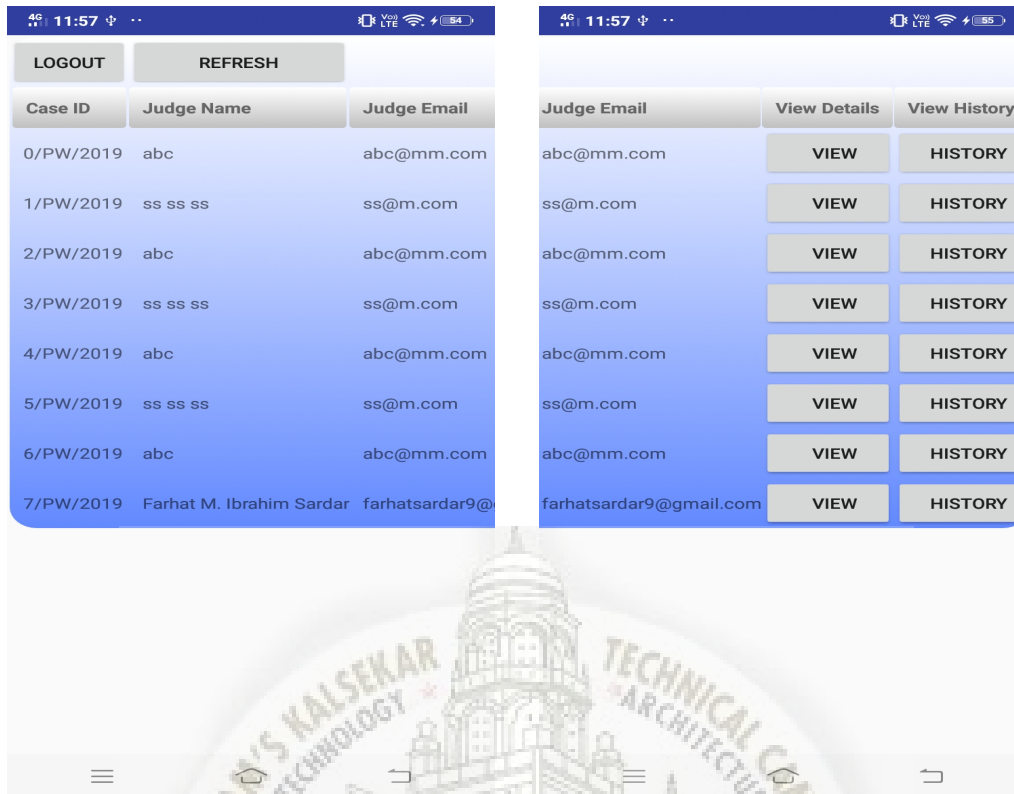


Figure 8.9: Court's Home Page

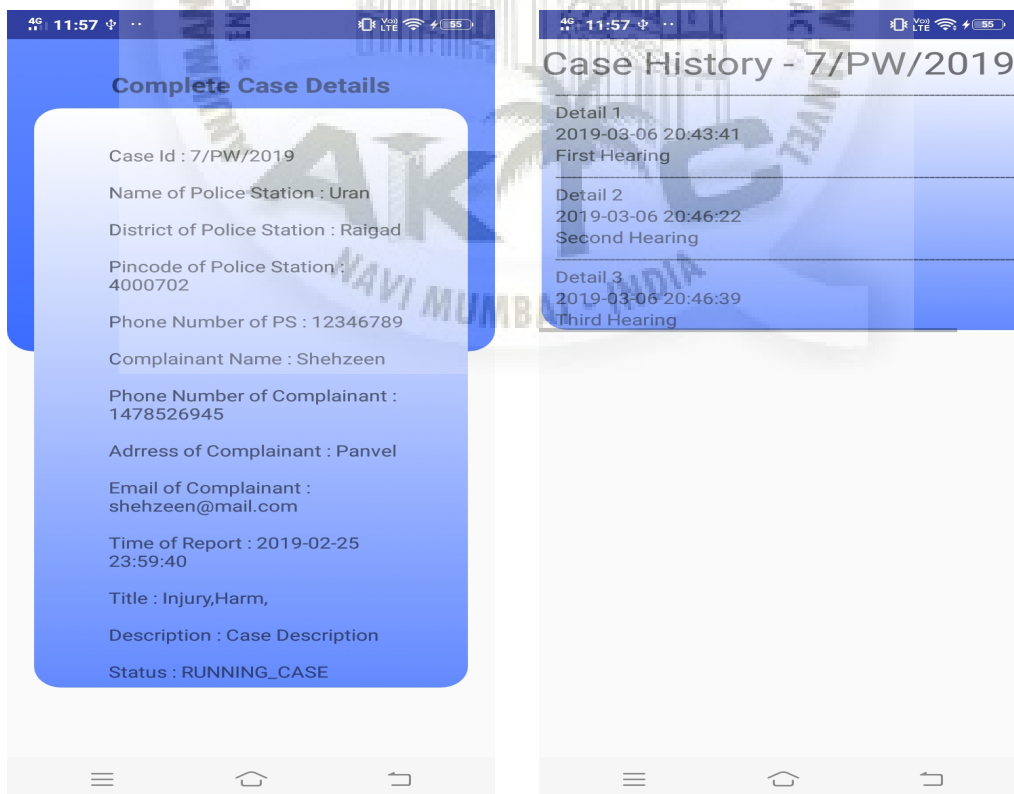


Figure 8.10: Case Detail with Case History



Figure 8.11: Judge's Home Page and New Cases

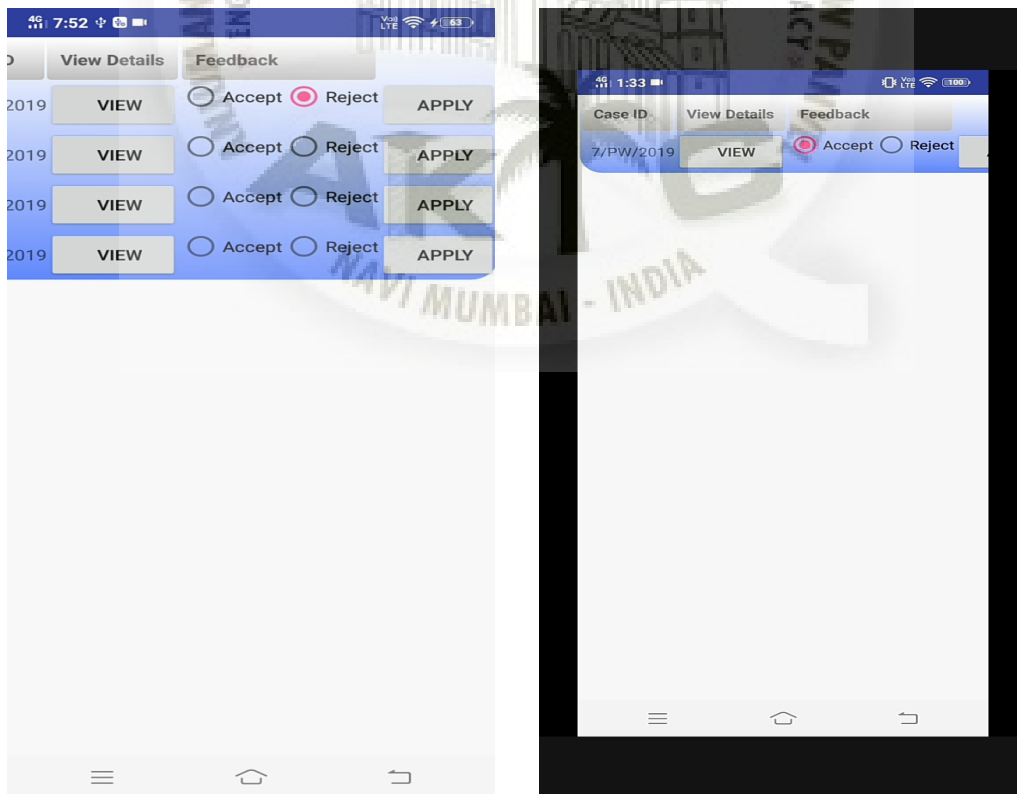


Figure 8.12: Reject Case and Accept Case

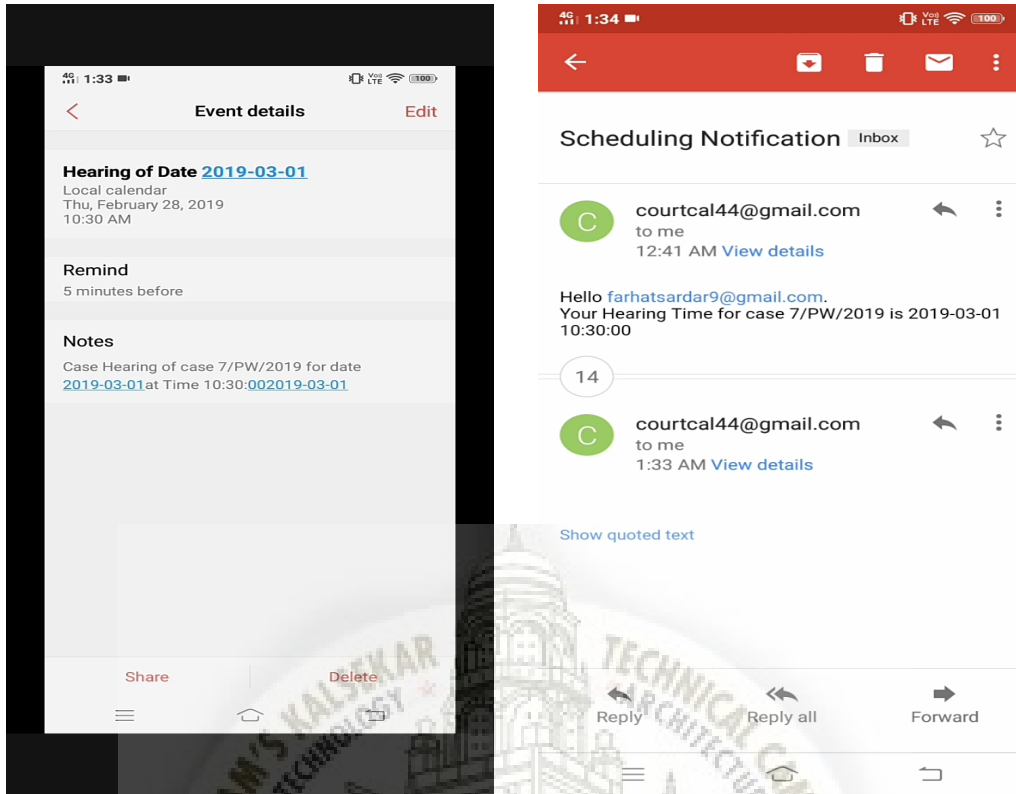


Figure 8.13: Calendar Event and Email Notification of Hearing Detail

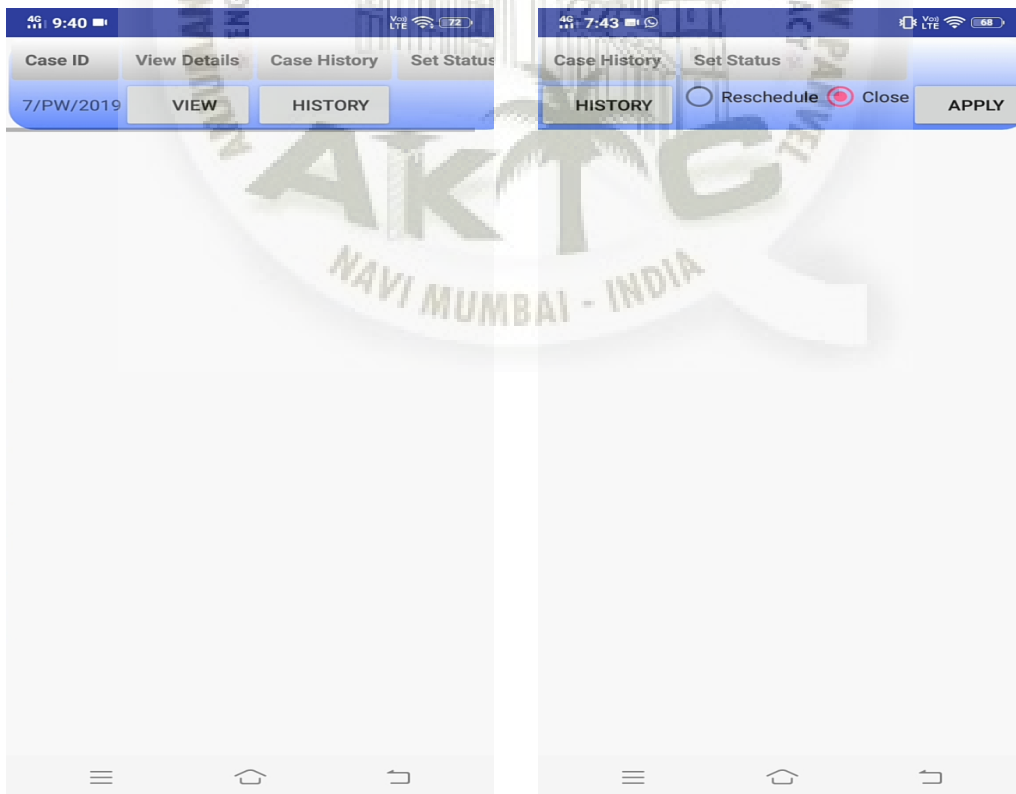


Figure 8.14: Running Cases of Judge

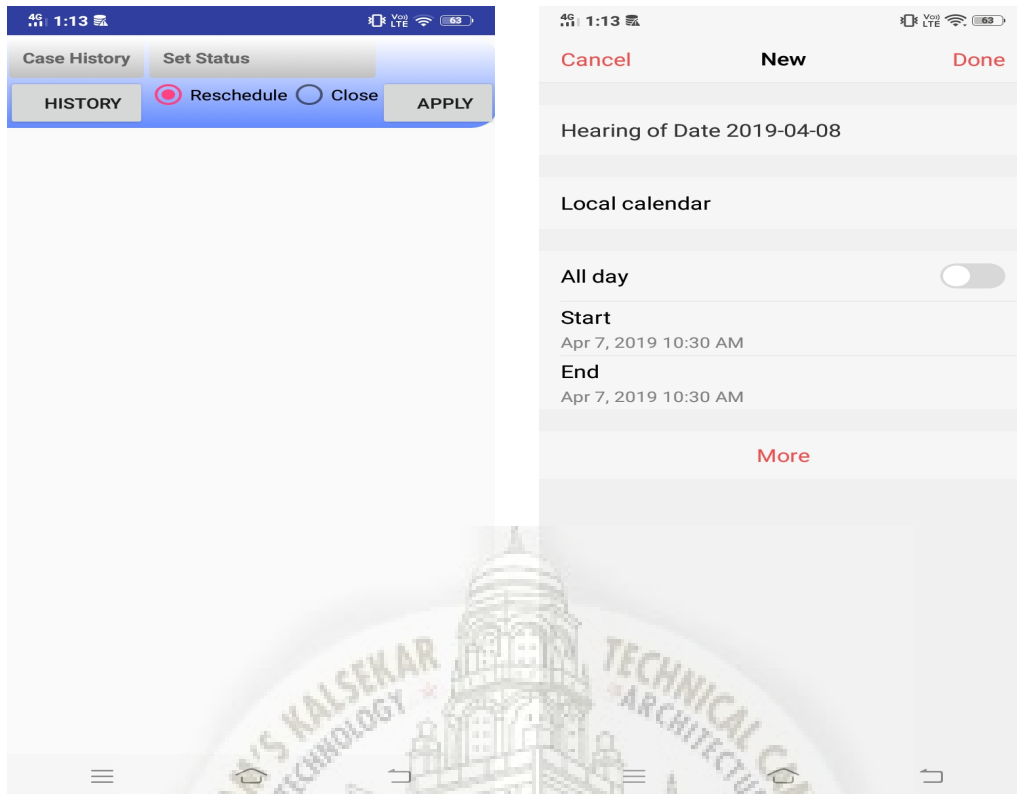


Figure 8.15: Reschedule Case and Calendar Event for new date



Figure 8.16: Hearing Detail and Closed Cases of Judge

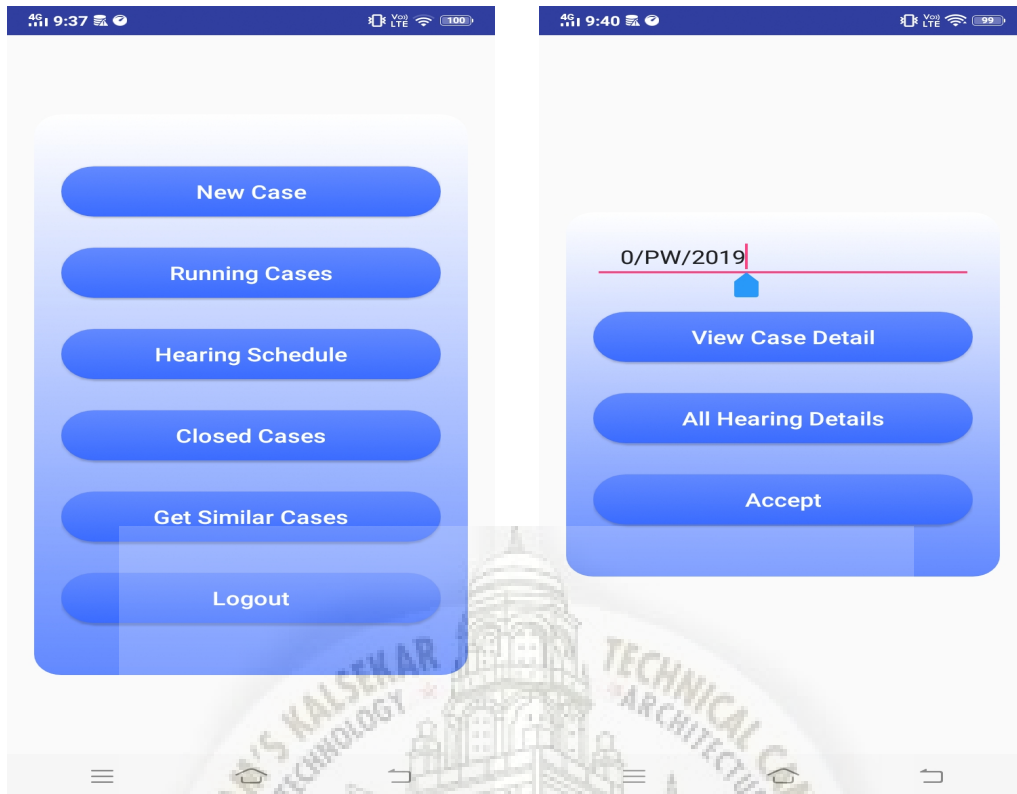


Figure 8.17: Lawyer’s Home Page and New Case for lawyer

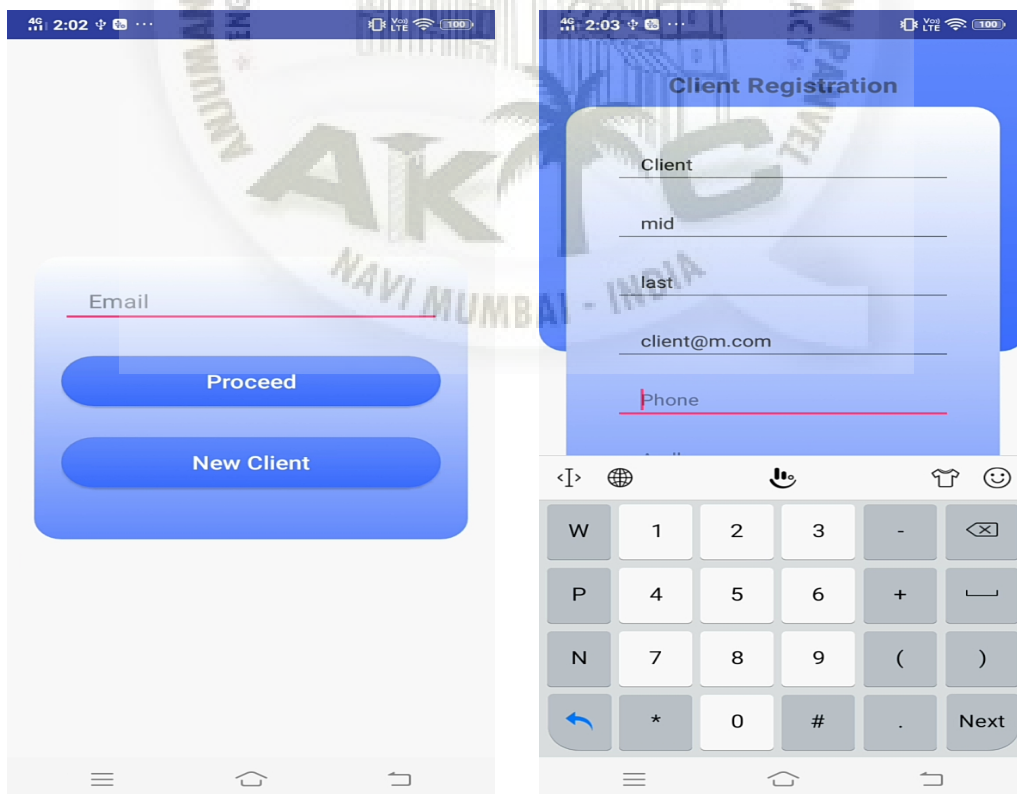


Figure 8.18: Accept Case by Registering Client Detail

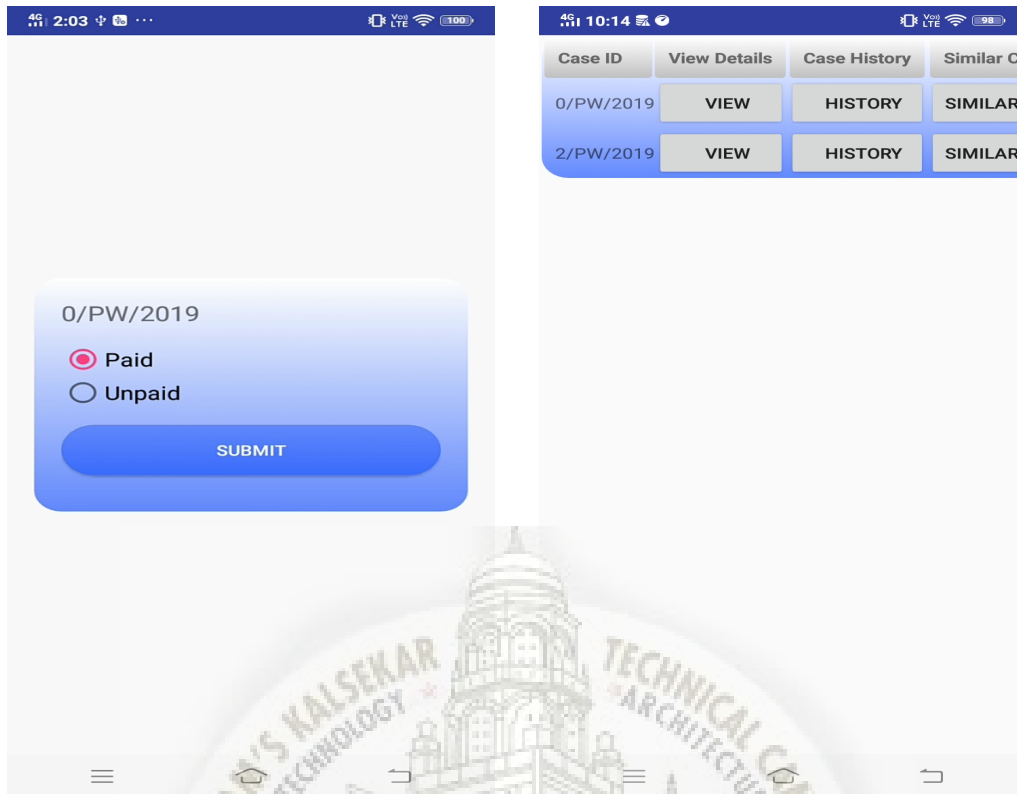


Figure 8.19: Payment detail of new case and List of Running Cases



Figure 8.20: Similar Cases and Hearing details for all Running Cases

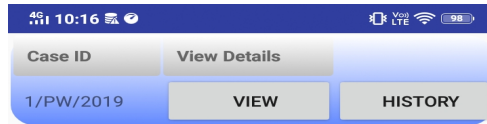


Figure 8.21: List of Closed Cases handle by lawyer



Figure 8.22: Get similar cases by giving case id

Chapter 9

Conclusion and Future Scope

9.1 Conclusion

After login police will register the FIR on our website which will store online. At the time of FIR creation case id for the case is automatically generated by the system. Similarly sections will be fetched based on title of case. Fetched section will be verified by the police.

This FIR will be given to judge. From android app judge will either accept or reject the case. If case is accepted by judge then chargesheet for that case will be created by police on website. At the same time case will schedule for hearing. Hearing notification will be given to judge and lawyer. Case history is created by the stenotypist.

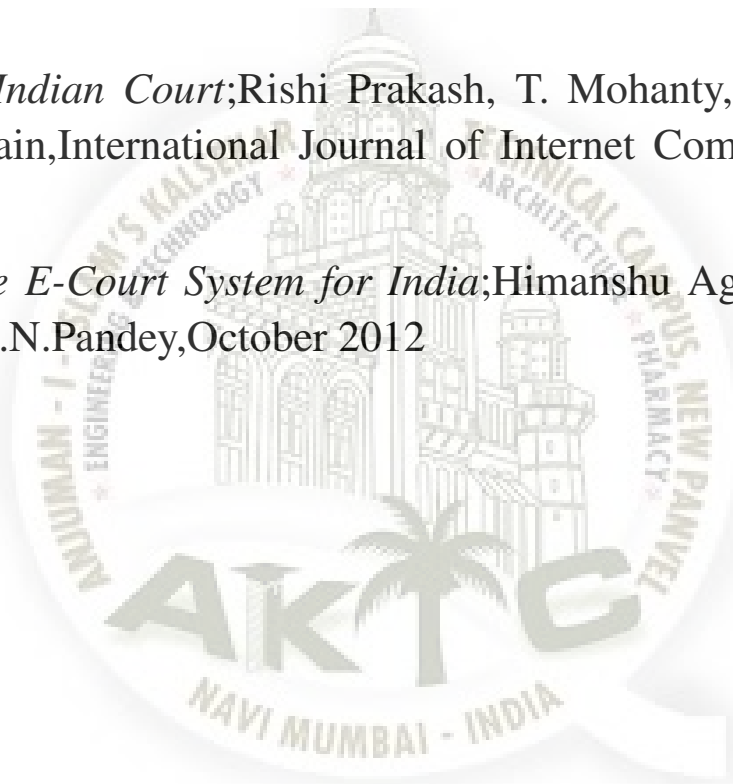
9.2 Future Scope

Following are the future scope of our project

- Audio & video conferencing
- Data compression

References

- [1] *E-Court : Technology Diffusion in Court Management*;Dr. Abrar Haider,Wan Satirah Wan Mohd Saman,Proceedings of the Nineteenth Americas Conference on Information Systems,August 2013
- [2] *ICT in Indian Court*;Rishi Prakash, T. Mohanty, Ramji Gupta Vinay Jain,International Journal of Internet Computing (IJIC), 2011
- [3] *A secure E-Court System for India*;Himanshu Agarwal,Harshal Singh,G.N.Pandey,October 2012



Achievements

1. Conferences

- (a) *Automated Jurisdiction Support system*; Sardar Sobiya, Adhikari Shahzeen, Sayyed Rehan, Naje Aqeel, International Conference on efficiency of software tools for mathematical modeling , April 2019 of attend(Venue :Thadomal shahani engineering college(TSEC))
- (b) *Automated Jurisdiction Support system*; Sardar Sobiya, Adhikari Shahzeen, Sayyed Rehan, Naje Aqeel, International Conference on efficiency of software tools for mathematical modeling , January 2019 of attend(Venue :Kalsekar Technical Campus engineering college(AIKTC))