# Hire a lawyer

**Final Year Project**

**Session 2017-2019**

A project submitted in partial fulfillment of the degree of

MCS **2nd Year Student**

Department of Computer Science

Faculty of Computer Science & Information Technology

The Superior College, Lahore

Spring 2019

| | |
|---|---|
| Type (Nature of project) | [ ✓] **D**evelopment [ ] **R**esearch [ ] **R&D** |
| Area of specialization | Web development |

**Project Group Members**

| Sr.# | Reg. # | Student Name | Email ID | *Signature |
|---|---|---|---|---|
| (i) | Mcsm-F17-013 | Saba Munir | mcsm-f17-013@superior.edu.pk | |
| (ii) | Mcsm-F17-020 | Aqsa Razaq | mcsm-f17-020@superior.edu.pk | |
| (iii) | Mcsm-F17-045 | Snober Isra | mcsm-f17-045@superior.edu.pk | |

# Plagiarism Free Certificate

This is to certify that, I **Saba Munir** daughter of Munir Ahmad, group leader of FYP under registration no **Mcsm-F17-013** at Computer Science Department, The Superior College Lahore. I declare that my FYP proposal is checked by my supervisor and the similarity index is 9% that is less than 20%, an acceptable limit by HEC. Report is attached herewith as Appendix D.

Date: 22-08-2019

Name of Group Leader: Saba Munir

Signature: _____

Name of Supervisor: Muhammad Fiaz

Designation: Junior Lecturer

Signature: _____

HOD: Dr. Irfan ud din

Signature: _____

# APPROVAL

**PROJECT SUPERVISOR**

Comments: _____

_____

Name:_____

Date:_____ Signature:_____

**PROJECT MANAGER**

Comments: _____

_____

Date:_____ Signature:_____

**HEAD OF THE DEPARTMENT**

Comments: _____

_____

Date:_____ Signature:_____

# Dedication

This work is dedicated to our families, friends and everyone who showered us with their love and support. We are especially thankful to those who presented challenges to our work and we pray for them because without their resistance we would never have been able to achieve what we did.

# Acknowledgements

We are really thankful to our supervisor who has guided us well throughout the semester. As well as the people that provided us with their honest opinions and reviews. Who made time for us for conducting their interview from their busy schedules' .We would also like to thank our friends and families that provided us with the opportunity and support for developing this project.

# Executive summary

We are the citizens of an Islamic country and justice should be our priority .Not only for rich one, there should be an equality. Our project is about the judiciary system of Pakistan. For this purpose we will develop a website which is helpful for those people who don't have any idea about how to choose a best lawyer to defend their case. In this system we will also provide the information of those lawyers and reconciliation agencies who defend the cases of financially poor people. This software will also helpful for students who recently completed their degree and further they are looking for a best mentor to guide them for practice.

People can also interact with those lawyers which are related to their case and also have a specialization in that field through using this software. People don't have to visit courts to collect details about their case because our system will also provide the current information of every case.

# Table of Content

# List of Figures

# List of Tables

# Chapter 1
## Introduction

# Chapter 1: Introduction

Targeted audience for this software solution will be those people who won't understand the judicial system or those who don't know how to hire the best lawyer for their relevant cases. We want to make a solution which generally provide the basic information about different categories of lawyers. We will also provide the best lawyers for the certain cases at the client's desired location.

## .1. Background

Our idea is not fully new. There are already few web sites which are working on it but they only work on web side and they only tell about lawyers but we are going to enhance it by adding many features which are not included in existing websites e.g. current information of every case (hearing date of case).Our software will also helpful for those students who are looking for a best mentor for practice.

## .2. Motivations and Challenges

World is changing, technology involves in every kind of field like in travelling, buying cloths, in hospital or in hotels. Now we are going to develop a lawyer's management system in which we want to save time and money of those people who want to save their time .We provide information of the lawyers in their area and they can search easily for their desired type of case.

## .3. Goals and Objectives

We wants to interact the lawyers with their clients at one platform .The public or the clients can meet their relevant lawyers by using our website .This may save the time and money. Clients can get all the information about their cases through our website. We also provide a platform for newly graduated law students to search their mentor for their better future.

## .4. Literature Review/Existing Solutions

Such websites are existing and working on this idea but lawyers register themselves after paying dues and can further communicate with their clients but we will provide these all without any charges and also enhance the features in our website and application .We add new features in our system both for lawyers and newly graduate law students.

## .5. Gap Analysis

Clients face problems related to their cases they don't know how to hire a best lawyer, they don't know all about their cases. So we introduce a software for clients and lawyers to solve are their judiciary problems by using our platform.

## .6. Proposed Solution

- We make a website which provides the basic information of the different lawyers available at different locations across the Pakistan.
- Lawyers can register themselves and they can verify their credibility through contacting with us directly.
- People using our software can find different information about cases and find their relevant lawyer to handle the case for them at their location.
- We are providing a platform for students to find out the best mentor for their counselling.
- We provide the links of that reconciliation agencies which sort out the cases without any fee by the willingness of plaintiff and responded.

## .7. Project Plan

Main components of our project are written below.

1. Basic information

   This section will contains every type of geranial information about constitution and everything about judiciary system, etc.

2. Lawyers information

   Everything about a lawyer comes in this section finding, hiring and communication information.

3. Accounts management

   Our customers can create and maintain their accounts. Login system and password reset, etc.

4. Website for lawyers

   i. For lawyers

This will all the features for lawyers to manage their accounts and to manage their customers, etc.

ii. For public/clients

This will have different features like from finding appropriate lawyers to contacting them and hiring them it will also provide general information for the public/clients. They can manage their accounts, cases, etc.

## .7.1. Work Breakdown Structure

### 1 Analysis

1.1 Functional Requirement's

1.2 Non-functional Requirement's

1.3 Other requirements

### 2 Design

2.1 Business logic design

2.2 Database design

2.3 Front-end design

### 3 Implementation

3.1 Database implementation

3.2 Business logic implementation

3.3 Front-end implementation

### 4 Test

### 5 Deployment

## .7.2. Roles & Responsibility Matrix

The purpose of roles & responsibility matrix is to identify who will do what.

*Table*

*1.Roles & Responsibility Matrix*

| WBS # | WBS Deliverable | Activity # | Activity to Complete the Deliverable | Duration (# of Days) | Responsible Team Member(s) & Role(s) |
|---|---|---|---|---|---|
| 0 | Proposal | 1 | | 10 | Saba |
| 1 | Analysis | 2 | | 15 | |
| | | | | | |
| 1.1 | Functional Requirement's | | | 7 | Saba |
| 1.2 | Non-functional Requirement's | | | 7 | Aqsa |
| 1.3 | Other requirement's | | | 2 | Snobar |
| 2 | Design | 3 | | 30 | |
| 2.1 | Business logic design | | | 10 | Saba,Aqsa,Snobar |
| 2.2 | Database design | | | 12 | Aqsa |
| 2.3 | Front-end design | | | 8 | Saba |
| 3 | Implementation | 4 | | 90 | |
| 3.1 | Database implementation | | | 15 | Aqsa,Snobar |
| 3.2 | Business logic implementation | | | 35 | Saba |
| 3.3 | Front-end implementation | | | 40 | Saba,Aqsa,Snobar |

| | | | | | |
|---|---|---|---|---|---|
| 4 | Test | 5 | | 15 | Saba,Aqsa |
| 5 | Deployment | 6 | | 15 | Saba,Snobar |

### .1.1. Gantt chart

| Task | Start Date | End Date | Duration |
|---|---|---|---|
| **Proposal** | 10/18/2018 | 10/23/2018 | **5** |
| **Analyses** | | | **15** |
| Funtionnal requirments | 10/24/2018 | 11/1/2018 | 8 |
| non-function requirements | 11/2/2018 | 11/8/2018 | 6 |
| other requirements | 11/9/2018 | 11/15/2018 | 6 |
| **Designe** | | | **55** |
| business logic | 11/16/2018 | 12/10/2018 | 24 |
| database logic | 12/11/2018 | 12/28/2018 | 18 |
| front-end design | 12/29/2018 | 1/20/2019 | 22 |
| **Implimentation** | | | **160** |
| database | 1/21/2019 | 2/10/2019 | 20 |
| business logic | 2/11/2019 | 3/5/2019 | 25 |
| fornt-end | 3/6/2019 | 4/10/2019 | 34 |
| **Test** | 4/11/2019 | 5/1/2019 | **21** |
| **Deployment** | 5/2/2019 | 6/10/2019 | **38** |
| **Total Days** | | | **294** |

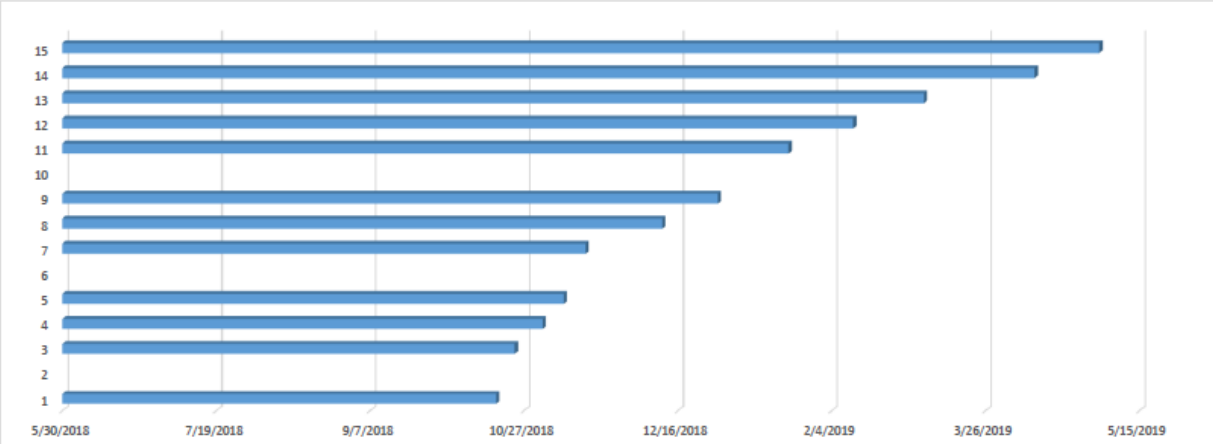



*Figure 1.Gantt chart*

## .2. Report Outline

In this chapter we discuss about problem which we face and give you its solution, background of proposed system, why we introduce and what problem this system is solving, project plan of our proposed system how we make and gather information, what kind of task we have to make this project, what kind of challenges we face to make this project prototype, background and assign tasks to group members and their role models.

# Chapter 2
# Software Requirement Specifications

# Chapter 2: Software Requirement Specifications

## .1. Introduction

### .1.1. Purpose

The main or important objective behind developing this software is to facilitate the lawyers and public. Basically we are developing the lawyer's management system. The main purpose of this documentation is to clearly identify and defining the requirements providing by clients. This software requirement specifications document will serve as a reference for the project team when architecting and constructing the software.

### .1.2. Document Conventions

This Software Requirements document is intended for: − Developers who can review project's capabilities and more easily understand where their efforts should be targeted to improve or add more features to it. Project testers can use this document as a base for their testing strategy as some bugs are easier to find using a requirements document. This way testing becomes more methodically organized. − End users of this application who wish to read about what this project can do.

### .1.3. Intended Audience and Reading Suggestions

### .1.4. Product Scope

- Basic information

  All about judiciary system at this platform.

- Lawyer's information

  Information of registered lawyers and their relevant cases

- Management system

  Lawyers can manage their cases by using this. Can store all details of their cases and clients can also get information about their cases. Lawyers can also update the case history and detail

# .2. Overall Description

We are the citizens of an Islamic country and justice should be our priority .Not only for rich one, there should be an equality. Our project is about the judiciary system of Pakistan. For this purpose we will develop a website which is helpful for those people who don't have any idea about how to choose a best lawyer to defend their case. In this software we also provide the information of those lawyers and reconciliation agencies who defend the cases of financially poor people. This system also helpful for students who recently completed their degree and further they are looking for a best mentor to guide them for practice.

People can also interact with those lawyers which are related to their case and also have a specialization in that field through using this system. People don't have to visit courts to collect details about their case because our website will also provide the current information of every case.

Targeted audience for this software solution will be those people who won't understand the judicial system or those who don't know how to hire the best lawyer for their relevant cases. We want to make a solution which generally provide the basic information about different categories of lawyers. We will also provide the best lawyers for the certain cases at the client's desired location.

We make a website and which provides the basic information of the different lawyers available at different locations across the Pakistan. Lawyers can register themselves and they can verify their credibility through contacting with us directly. People through using our software can find different information about cases and find their relevant lawyer to handle the case for them at their location. We also provide a platform for students to find out the best mentor for their counselling. We will also provide the links of that reconciliation agencies which sort out the cases without any fee by the willingness of plaintiff and responded.

## .2.1. Product Perspective

"Hire a lawyer" is a very useful and time saving software for those who don't want to visit courts for time saving. Also useful for student who are looking for best mentor for practice. Helpful for those who want to contact with reconciliation agencies to solve their cases. Best plat form for lawyers to manage their cases online.

## .2.2. Product Functions

1. Users can get information of judiciary system.
2. Clients can search their desired lawyer relevant to their case.
3. Lawyers can also manage their cases and can save case history.

4. Lawyers and clients can also in contact with each other at this platform.

## .2.3. User Classes and Characteristics

## .2.4. Operating Environment

This software run on devices of upper version or equal to kit Kat and enable with GPS

To simplify the discussion, the statement made here is that most of the Linux 2.6 based devices are x86 based systems, whereas most mobile phones are ARM based products. While ARM represents a 32-bit reduced instruction set computer (RISC) instruction set architecture, x86 systems are primarily based on the complicated instruction set computer (CISC) architecture. In general, the statement can be made that ARM (RISC) is executing simpler (but more) instructions compared to an x86 (CISC) system. As already discussed, memory is at a premium in mobile devices due to size, cost, and power constraints.

ARM addresses these issues by providing a 2nd 16-bit instruction set (labeled thumb) that can be interleaved with regular 32-bit ARM instructions. This additional instruction set can reduce the code size by up to 30% (at the expense of some performance limitations). Ergo, from an overall systems perspective, the incorporation of the thumb instruction set can be considered as an exercise in compromises. Compared to x86 processors, the ARM design reveals a strong focus on lower power consumption, which again makes it suitable for mobile devices.

## .2.5. Design and Implementation Constraints

In this stage, prototype will be deployed and implemented in real environment. Selected users will test on the prototype and evaluate the system. First, users will test on the first prototype based on the test plan created in stage 3 and test result will be recorded for further evaluation. Evaluation for the first prototype will be performed based on the test result gathered from users. Interview and observation will be performed while testing on prototype. These activities are performed to verify the acceptance of the prototype by users. Observation is performed while the prototype is implemented in real environment to observe whether the functions and features provided in the prototype is satisfied the system requirements and users requirements discussed in early stage. If users are not satisfied with current prototype functionalities, interview will be done with users to acquire new requirements. Thus, system development process will loop back

to design stage and the new prototype is developed based on the new requirements provided by user.

### .2.6. User Documentation

**1.** User manual attached appendix.

**2.** Admin User Documentation attached as an appendix.

**3.** Testing Report attached as an appendix.

**4.** Final Report.

### .2.7. Assumptions and Dependencies

Hire a lawyer is a knowledge sharing portal due to web-based nature of project following Are the assumption and dependencies that could affect the system requirements and limits Functionality:

**1.** HAL will be deployed on a web server so due to unavailability of web server system may not be deployed live or not available during downtime or server maintenance activities.

**2.** Obsolete devices with outdated systems and digital certificate may not be able to Access the system.

**3.** Common internet issues may also cause system components unreachable.

**4.** Hardware and software not meeting the above requirements may not support the System functionality.

**5.** Application and other software components may not work properly on different Devices.

**6.** An unexpected change in the major component of the project may also affect system Functionality.

## .3. External Interface Requirements

### .3.1. User Interfaces

After studies on few existing judiciary based system, the design of “Hire a lawyer” had been chosen for proposed system interface design. The web page is divided into two main parts, one for lawyers and other for clients

The first page for users (clients) who can search a lawyer according to their requirements and can share all about his/her case with a relevant lawyer. User can send a request to the relevant lawyer to defend his/her case.

The second page for lawyers who can manage their case details after register themselves on this plat form. Lawyers can safe all the case details online. Lawyers contact and update their clients by using this software.

## .3.2. Hardware Interfaces

*Table 2.Hardware Interfaces*

| | |
|---|---|
| **CPU** | Pentium |
| **Phone** | Smartphone (android/ios/windows) |
| **internet connection** | 1 MB (Min) |
| **Storage** | 150MB(Min) |
| **Hard Drive** | 100 GB |
| **RAM** | 512 MB (Min) |

## .3.3. Software Interfaces

**Tools:**

**IDE:** Visual Studio Code, SQLite manager

**Languages:** PHP, sql, JavaScript, css3, html5, Ajax, json

**Database:** SQLite

**Frameworks:** larval, bootstrap

## .3.4. Communications Interfaces

Our software is divided into the three parts
1-Frontend (interface or layout)

2-Backend (business logic)

3-Database (data)

We uses Ajax at the frontend to communicate with the backend and your backend process the request and fetch the required information from the database and give us the response in json.

# .4. System Features

## .4.1. System Feature 1

Gets location of lawyers through android phone

#### 1.1.1.1. Description and Priority

It is of High priority feature, without this feature this app don't work. This feature work is to get area of lawyer from android phone.

**1.1.1.2. Stimulus/Response Sequences** This action performs when a user search his relevant lawyer from a specific area. This

Operation will perform at every search request.

#### 1.1.1.3. Functional Requirements

The system must be able to show information to user in real time. The system must be able to show all the available lawyers according to the search requirements.

- Android application should work properly.
- Internet should be connected.
- Database should be working fine.

## .4.2. System Feature 2

#### 1.1.1.4. Description and Priority

It is of High priority feature, without this feature lawyers cannot save the detail of any case. This feature work to update a case details.

#### 1.1.1.5. Stimulus/Response Sequences

This action performs simultaneously when the lawyer login and update the case details. When user use app only then this will action performs.

#### 1.1.1.6. Functional Requirements

- Android application should work properly.
- Internet should be connected.
- Database should be working fine

### .4.3. System Feature 3

- Functional requirement is referring to the functionalities must be apply to a system. The functional requirements of Judiciary system are stated below.
- The system must be able to show information to user in real time.
- The system must be able to process the location or area of all lawyers.
- The system must be able to show all related lawyers and their relevant cases.
- The system must be able to allow user to retrieve information from mobile device and computer.

## .1. Other Nonfunctional Requirements

### .1.1. Performance Requirements

The system should provide the accurate information about judiciary system and lawyers the system should reduce the paper work done lawyers.

The system should be able to increase the efficiency and performance of Case management system.

The system should reduce work done by lawyers by updating and saving records of their cases.

The system should allow user to access information in anywhere with anytime

### .1.2. Safety Requirements

All information of our clients will safe, not any single record of our clients and their client can be lost.

### .1.3. Security Requirements

Security was taken as a very serious nonfunctional requirement so that the data and information of each user is protected from visibility and possible alteration by the other users. Thus the “Hire

a lawyer" provides a user account which enables users to enter their respective profile as a meeting initiator or participant using a login id and password mechanism.

## .1.4. Software Quality Attributes

These are:

- Correctness
- Reliability
- Adequacy
- Learn ability
- Robustness
- Maintainability
- Readability
- Extensibility
- Testability ▪ Efficiency
- Portability

## .1.5. Business Rules

Our main user (the lawyer) should be license holders of high courts .lawyers should be specify their specialty of field e.g. (civil, family, taxes). Only those students are eligible to select the mentor for practice who are graduated by their law degree.

# Chapter 3
# Use Case Analysis

# Chapter 3: System Analysis

The public who needs to hire a lawyer to their relevant case not knows about the best relevant lawyer because of this a lot of time is wasted and opposition make stronger. This will help them to find their desired lawyer. Also the clients are informed about their hearing and about the remarks of court about their case because of this clients always knows about their cases on the other hand the lawyer's manage their whole system on our website.

## 3.1. Use Case Model

### 3.1.1 Use Case of lawyer

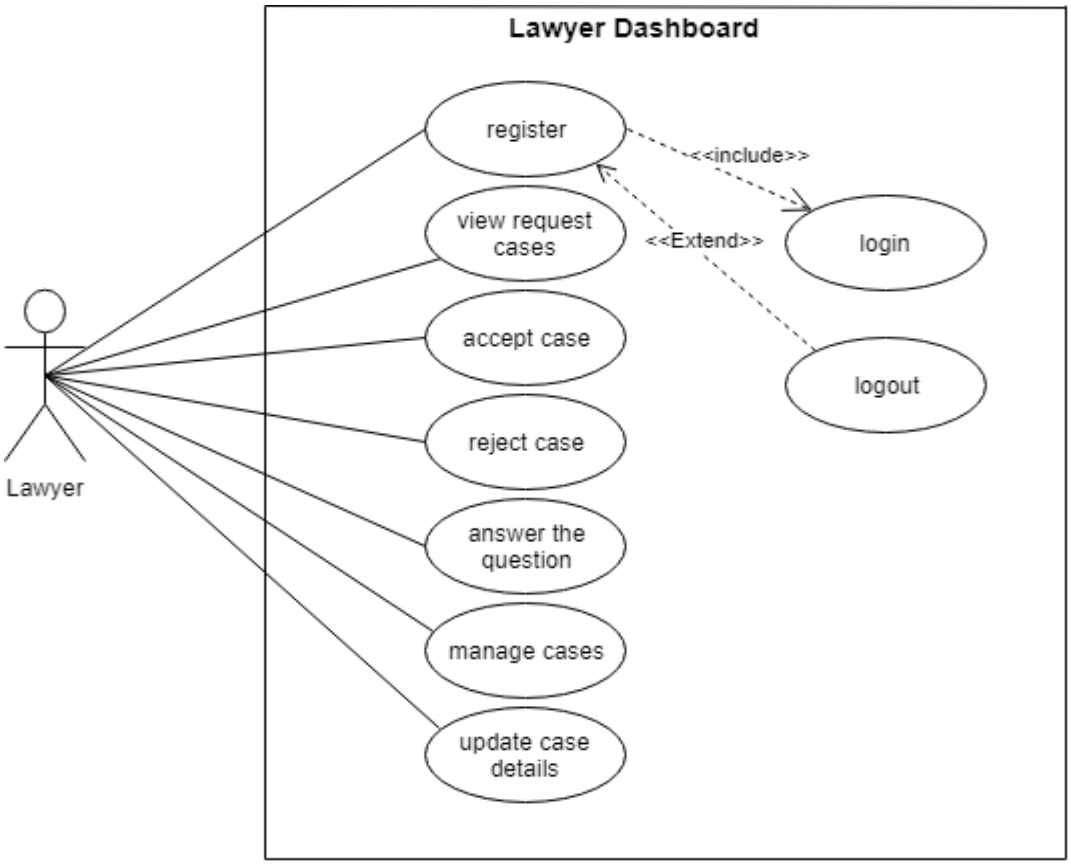



*Figure 2.Use case of lawyer*

### 3.1.2 Use Case of client

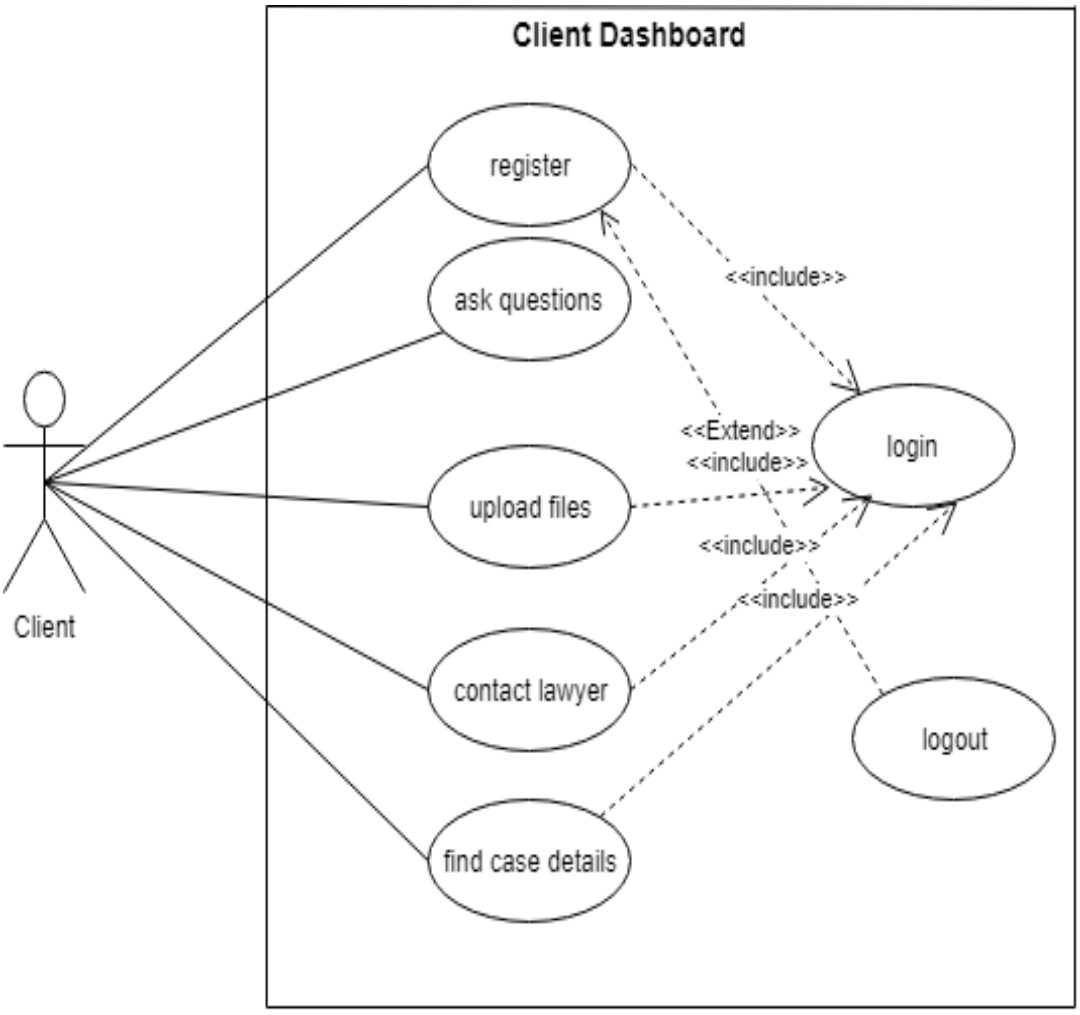



*Figure 3.Use case of client*

### 3.1.3 Fully dressed use case

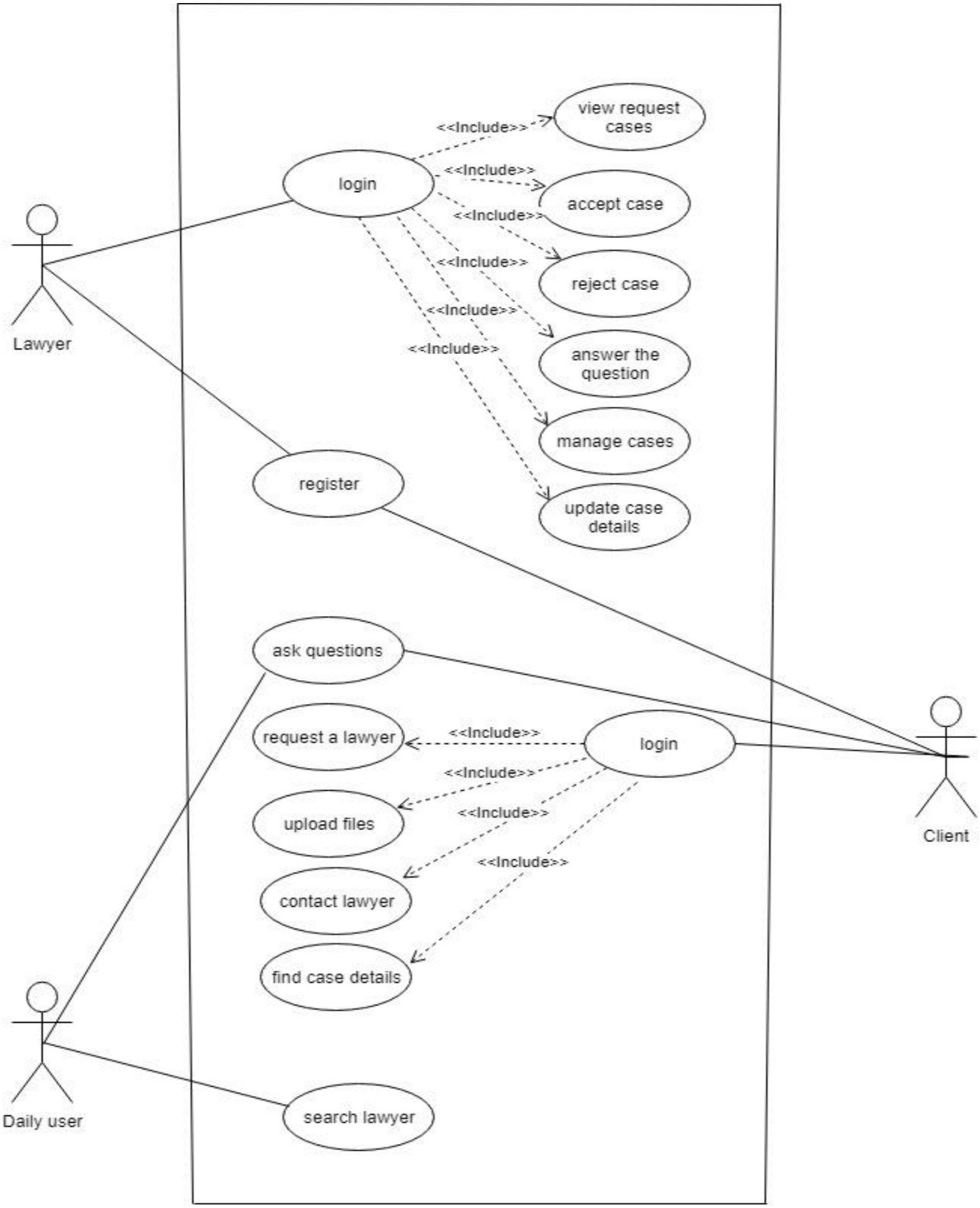



*Figure 4.Fully dressed use case*

## 3.2. Use Case Descriptions

In this stage, analyses on existing systems have been made. Few online lawyers' management systems available had been studied and the main system requirements had been founded. User want to know where is the best lawyers for their relevant cases

Besides, observation on current judiciary system has been made. Users can know about the valid information of all registered lawyers and then further contact with them through our software. Students can also search their mentor for practice in courts. Furthermore, interview has been done with the student and his mentor. The result of interview is student want to know that the selected lawyer is best or not as a mentor. Thus, functional and nonfunctional requirements are gathered from observation and interview.

Next, literature reviews in term of technologies, suitable programming language, platforms and methodologies are made. This is done to ensure that the proposed system delivered matches the user's requirements and expectations. A study on technology needed is to know how the technology work and make use of it in proposed system. Study on suitable programming language and different platforms are important to ensure that the programming language is sufficient to build the proposed system and how to integrate different modules with different platforms into a meaningful system to provide useful information to user. Finally, study on suitable methodology is to understand more about the chosen methodology in aiding the proposed system development processes.

Deliverables: Requirements analysis, functional and non-functional requirements, literature reviews (in term of technology needed, suitable programming language, platforms and methodology).

*Table 3.Use case description*

| | |
|---|---|
| **Tittle** | Hire A Lawyer |
| **Primary Actors** | Client |
| **Secondary Actors** | Lawyer |
| **Stake Holders** | Manager, Coder, website users |

| | |
|---|---|
| **Pre-conditions** | • The user must possess the website by typing URL.<br>• The users must login to move on dashboards. |
| **Post-conditions** | • The dashboards must be handled by login users. |
| **Status** | Developed |
| **Basic Flow** | 1. The home screen appears after entering URL.<br>2. The user use the system in website environment.<br>3. Users can login after registration.<br>4. Users can use the dashboards after login.<br>5. Users can update the case details.<br>6. Lawyer can accept or reject the requested cases. |

# Chapter 4
# System Design

# Chapter 4: System Design

This chapter contains the key-diagrams and models to visually represent the system.

Software is about to work with client server architecture. This is about the models and tools which is used to designing our system or our product.

## 4.1. Architecture Diagram

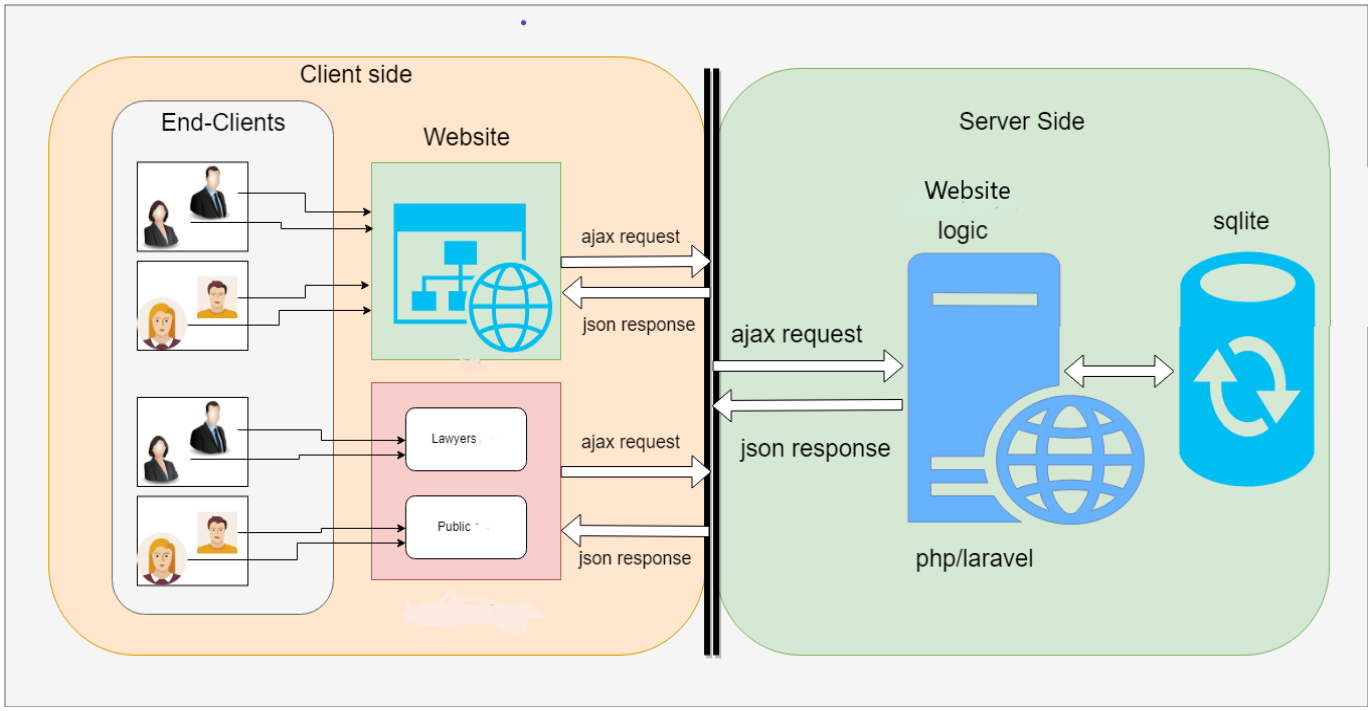



*Figure 5.Architecture Diagram*

## 4.2. Deployment Diagram

The following diagram shows the structure of our judiciary project in its deployed environment i.e.

The physical deployment of the components

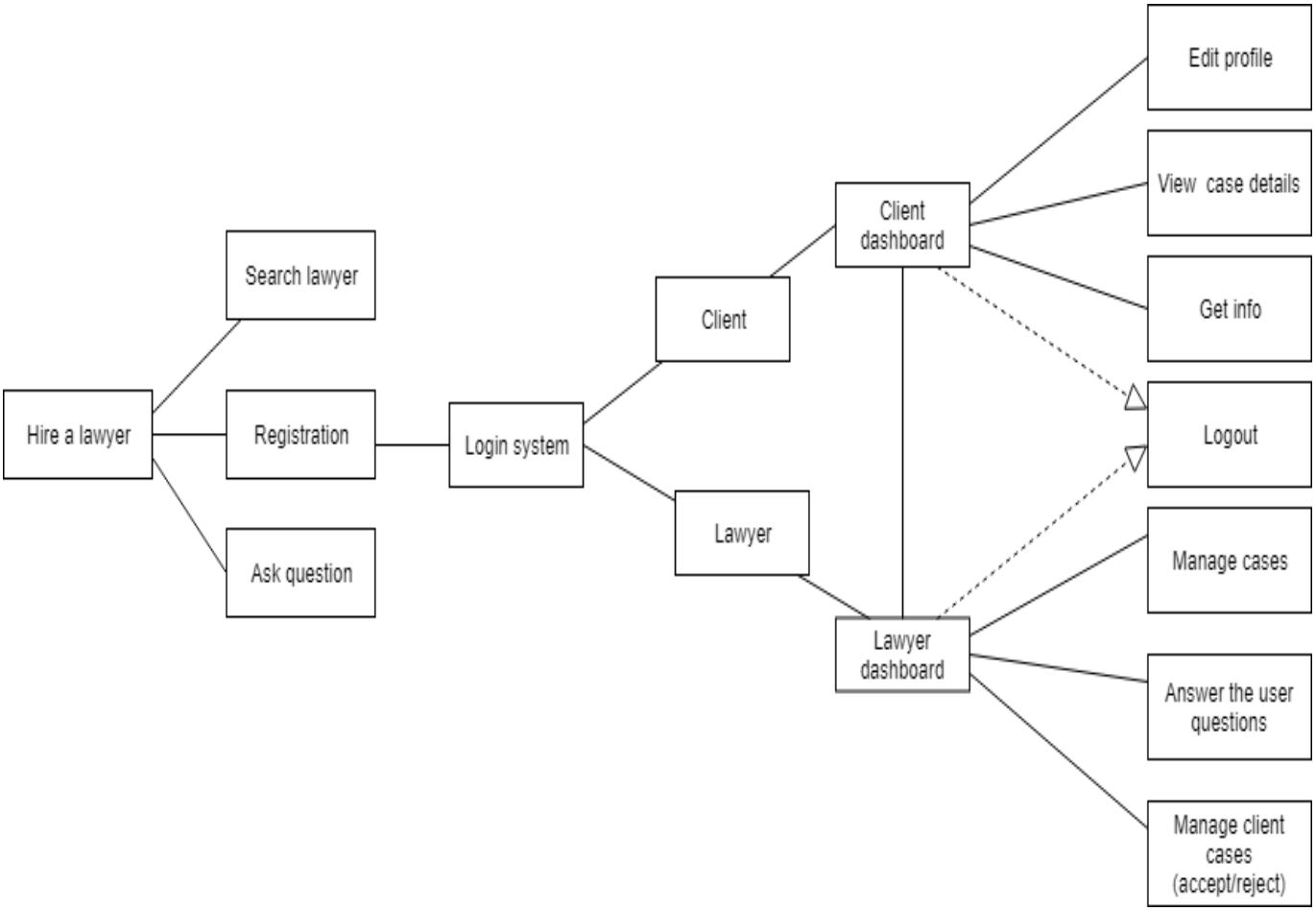



*Figure 6.Deployment Diagram*

## 4.3. Entity Relationship Diagram

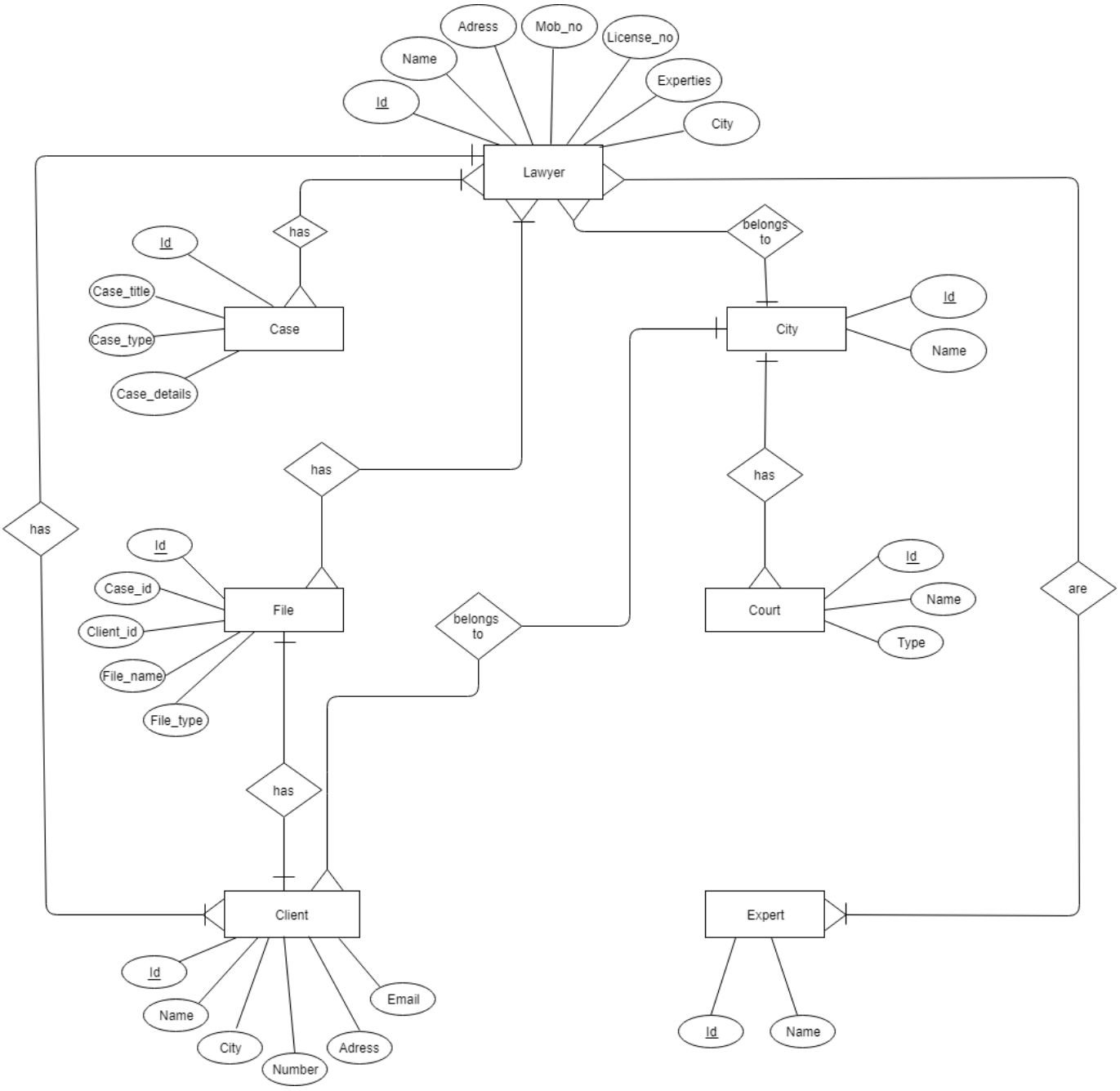



*Figure 7.Entity relationship Diagram*

## 4.4. Class Diagram

The following Class Diagram of our project gives an overview of *Hire a lawyer* by displaying its classes, attributes and operations:

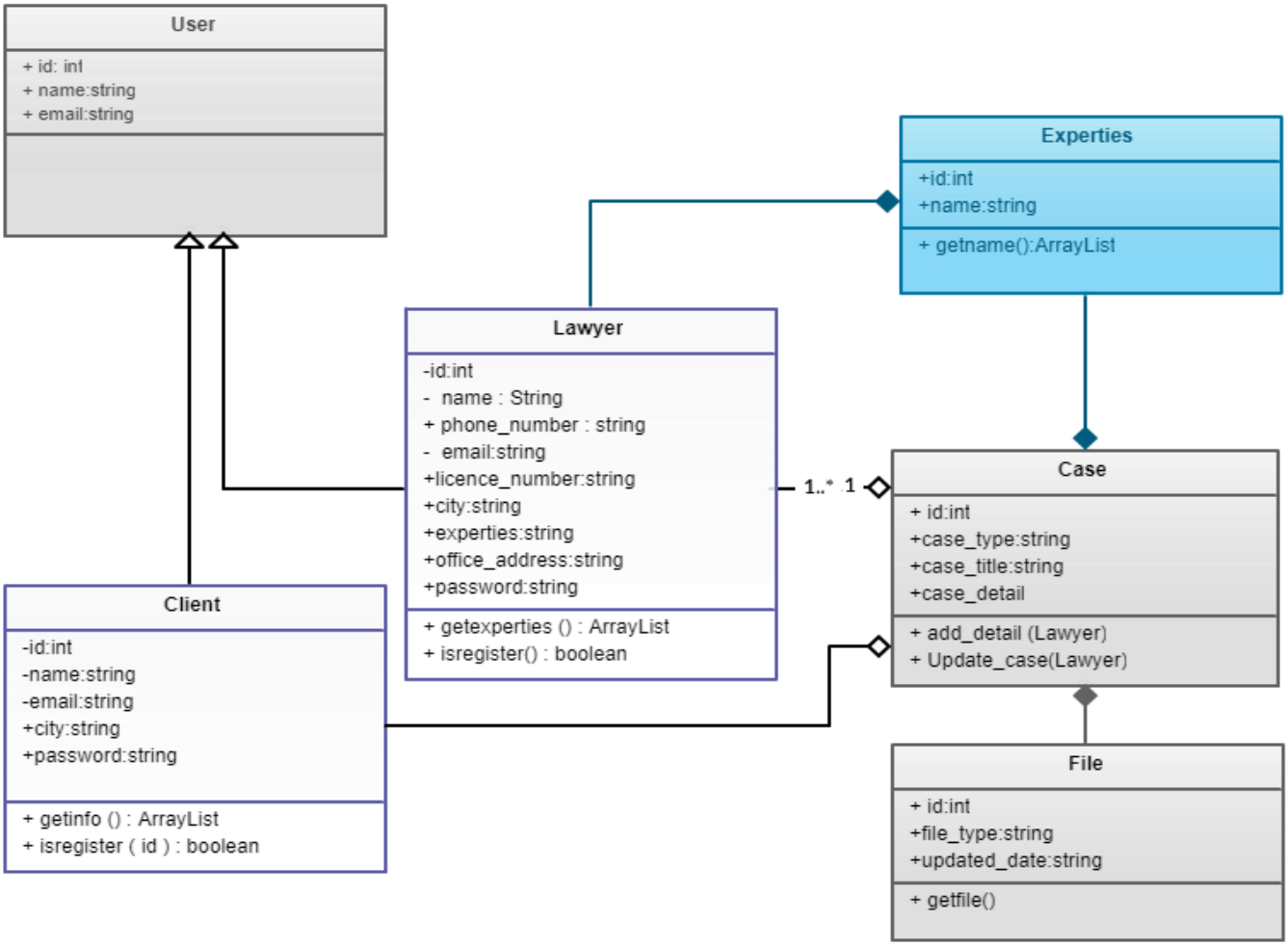



*Figure 8.Class Diagram*

# 4.5. Sequence /Collaboration Diagram

The following Sequence Diagram shows system objects interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the **sequence** of messages exchanged between the systems objects needed to carry out the functionality of the website.

## 4.5.1 Sequence Diagram of registration system

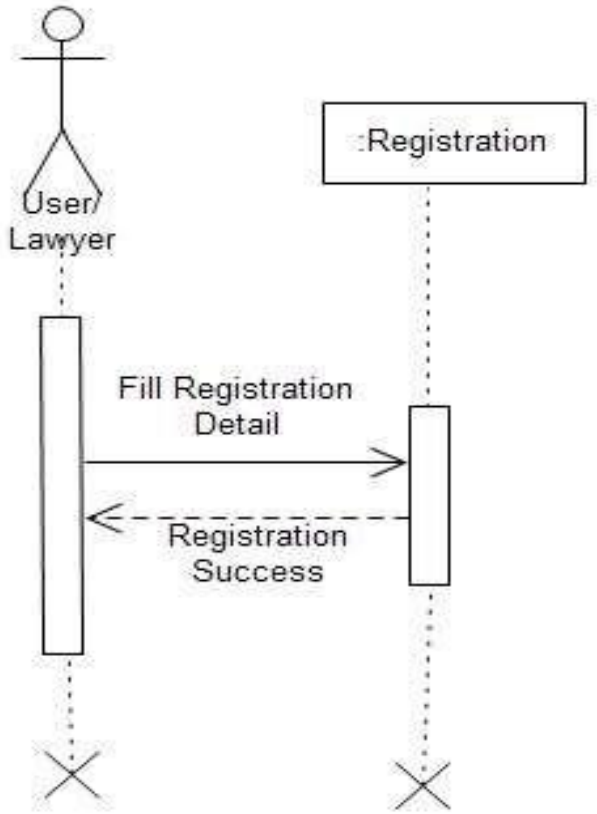



*Figure 9.Sequence Diagram of registration system*

## 4.5.2 Sequence Diagram of lawyer

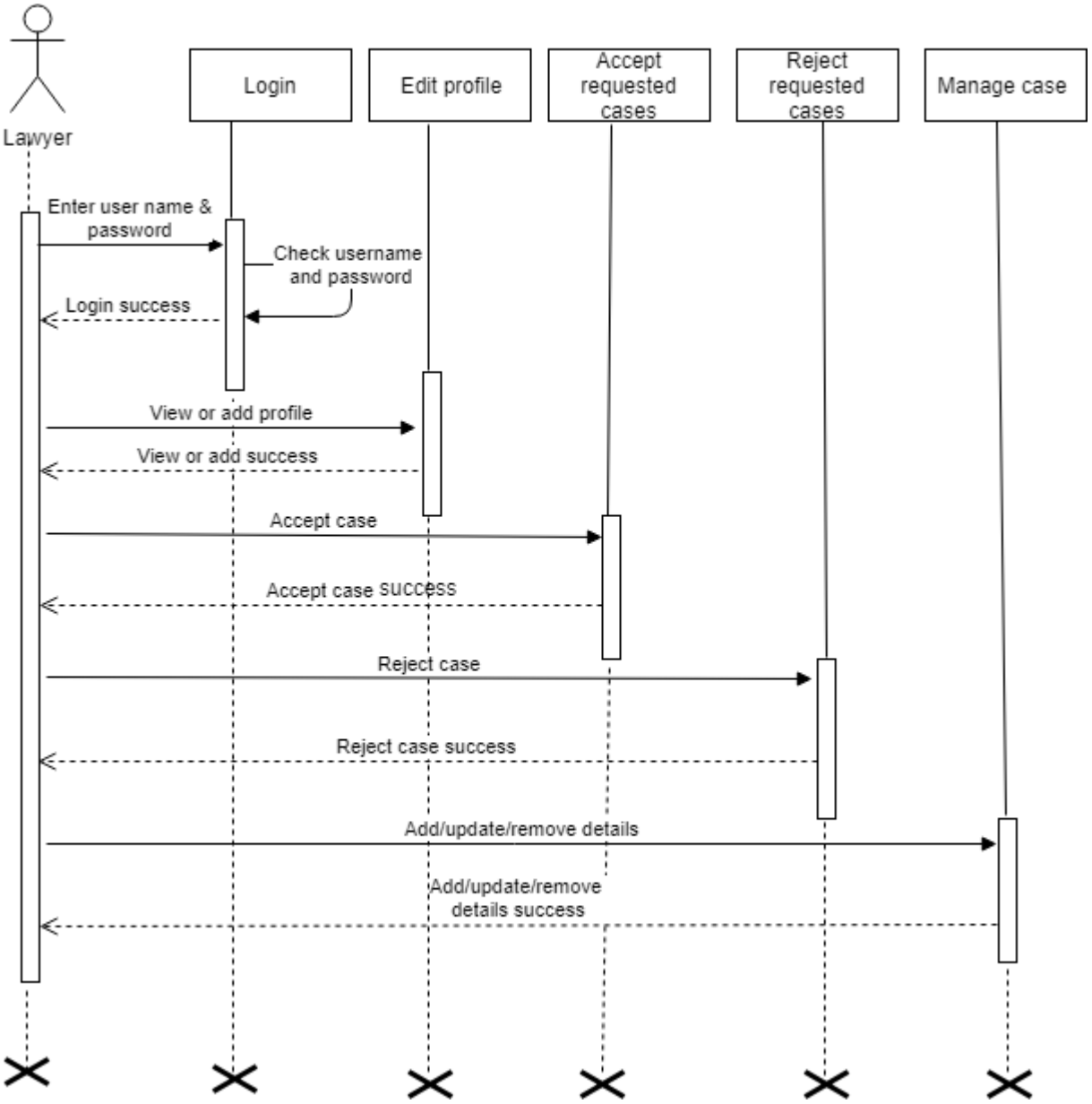



*Figure 10.Sequence Diagram of lawyer*

### 4.5.3 Sequence Diagram of user

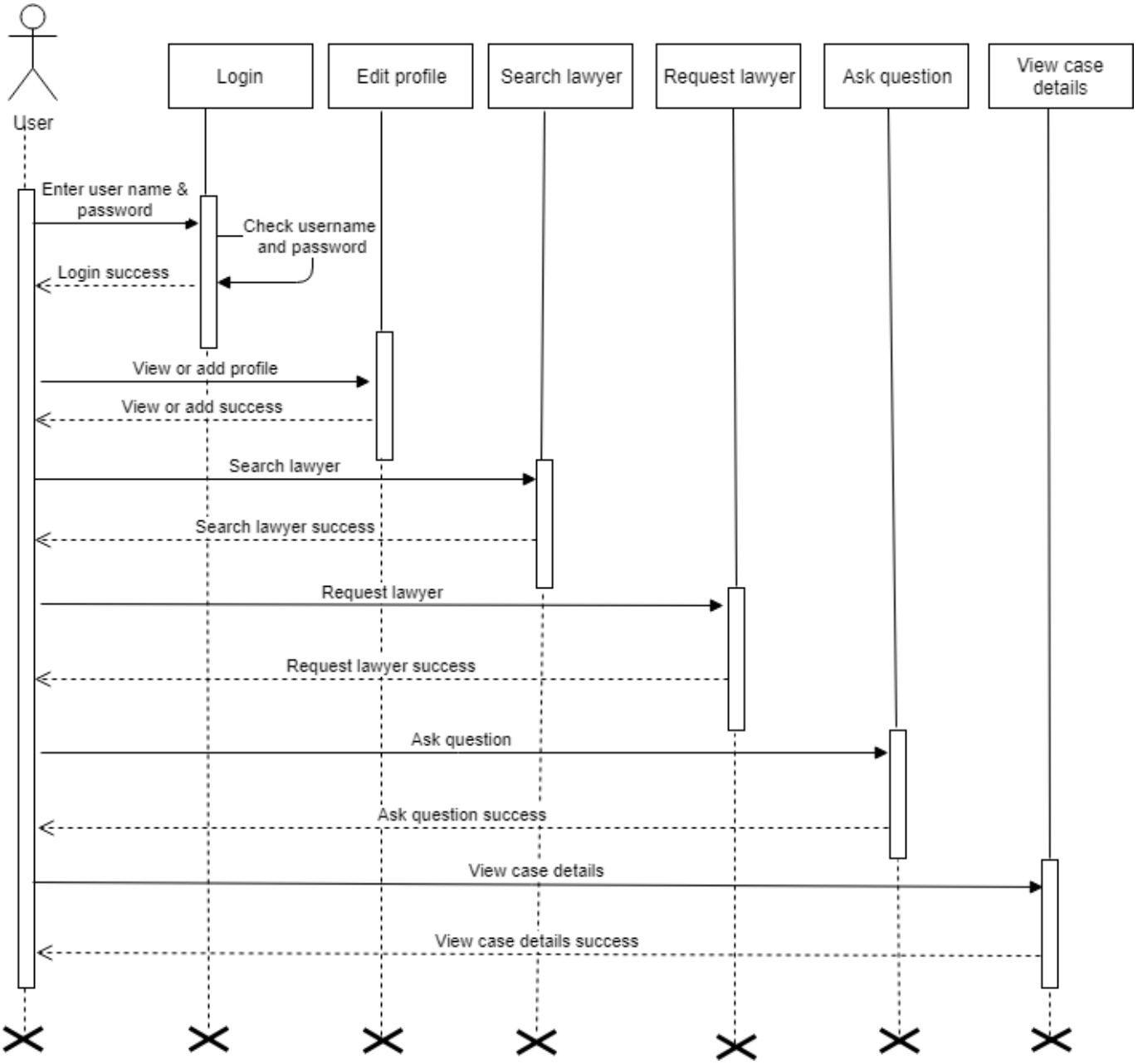



*Figure 11.Sequence Diagram of user*

## 4.6. State Transition Diagram

The following State Transition Diagram explains the states of the website from searching a lawyer and the case handling till the end.

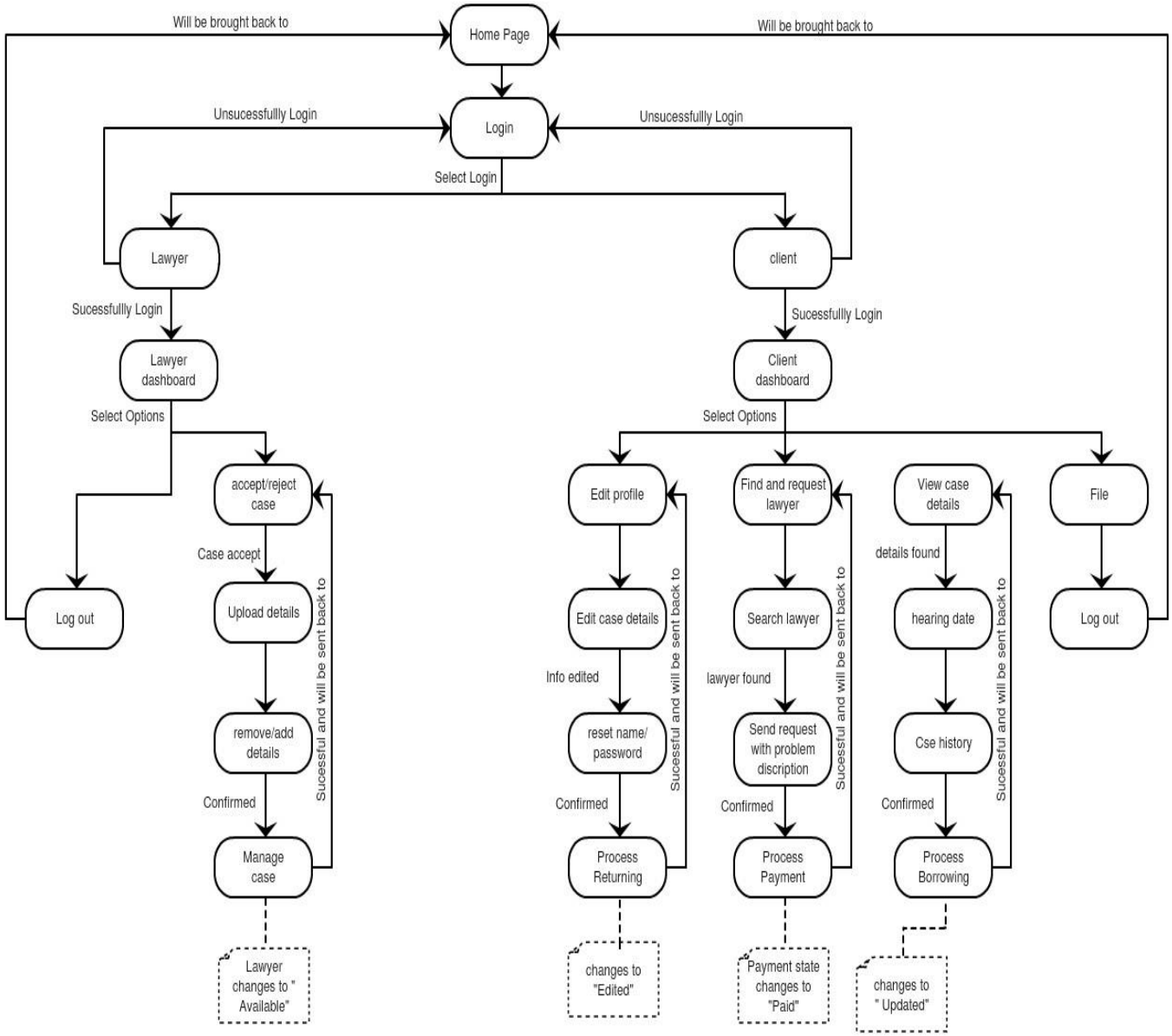



*Figure 12.State Transition Diagram*

## 4.7. Component Diagram

The following Component Diagram shows the dependencies and interactions between hardware and software components used in our project *Hire a lawyer.* Every component represents a modular part of our system, for example, dashboard of lawyer and client, user interface with website etc.

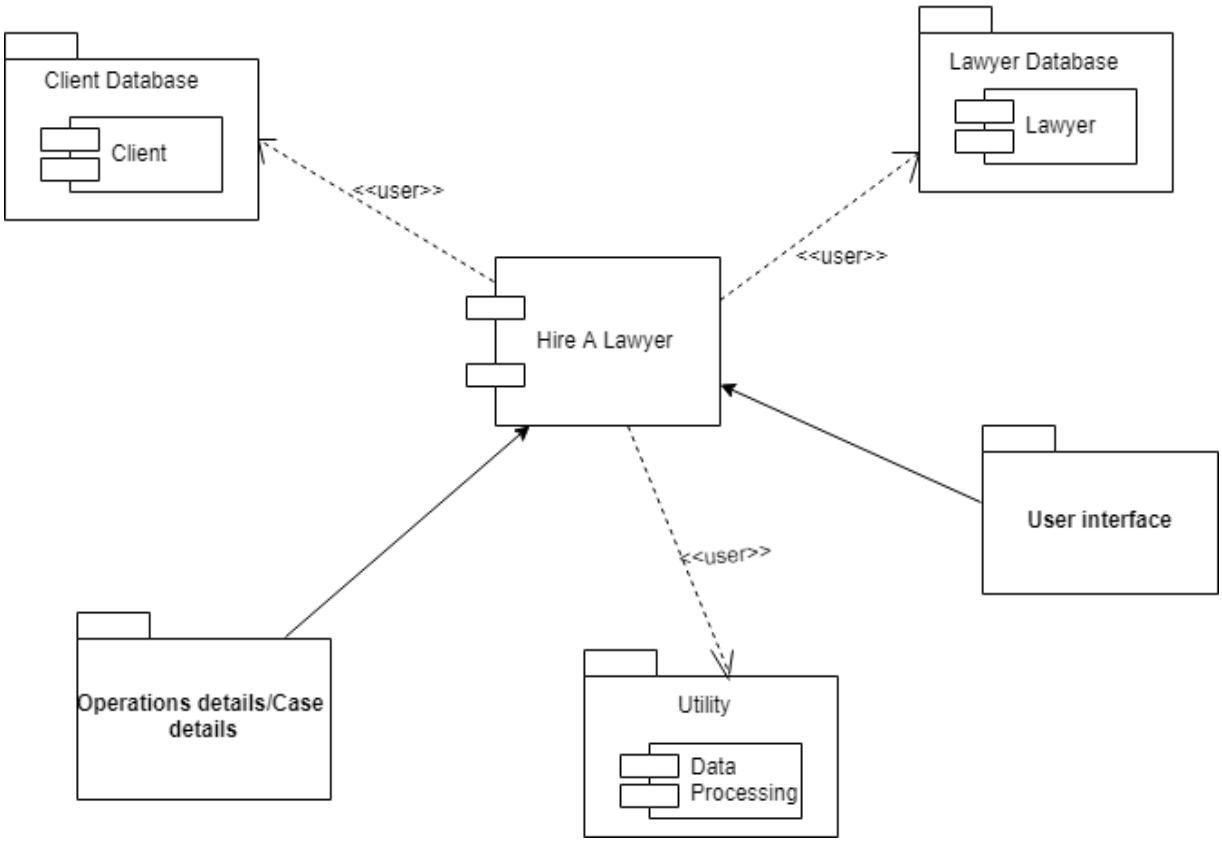



*Figure 13.Component Diagram*

## 4.8. Activity Diagram

The following **activity diagram** visually presents a series of major actions and the flow of control of our website. It explains the dynamic behavior of our system. The figure of Activity Diagram of *Hire a lawyer* is given below:

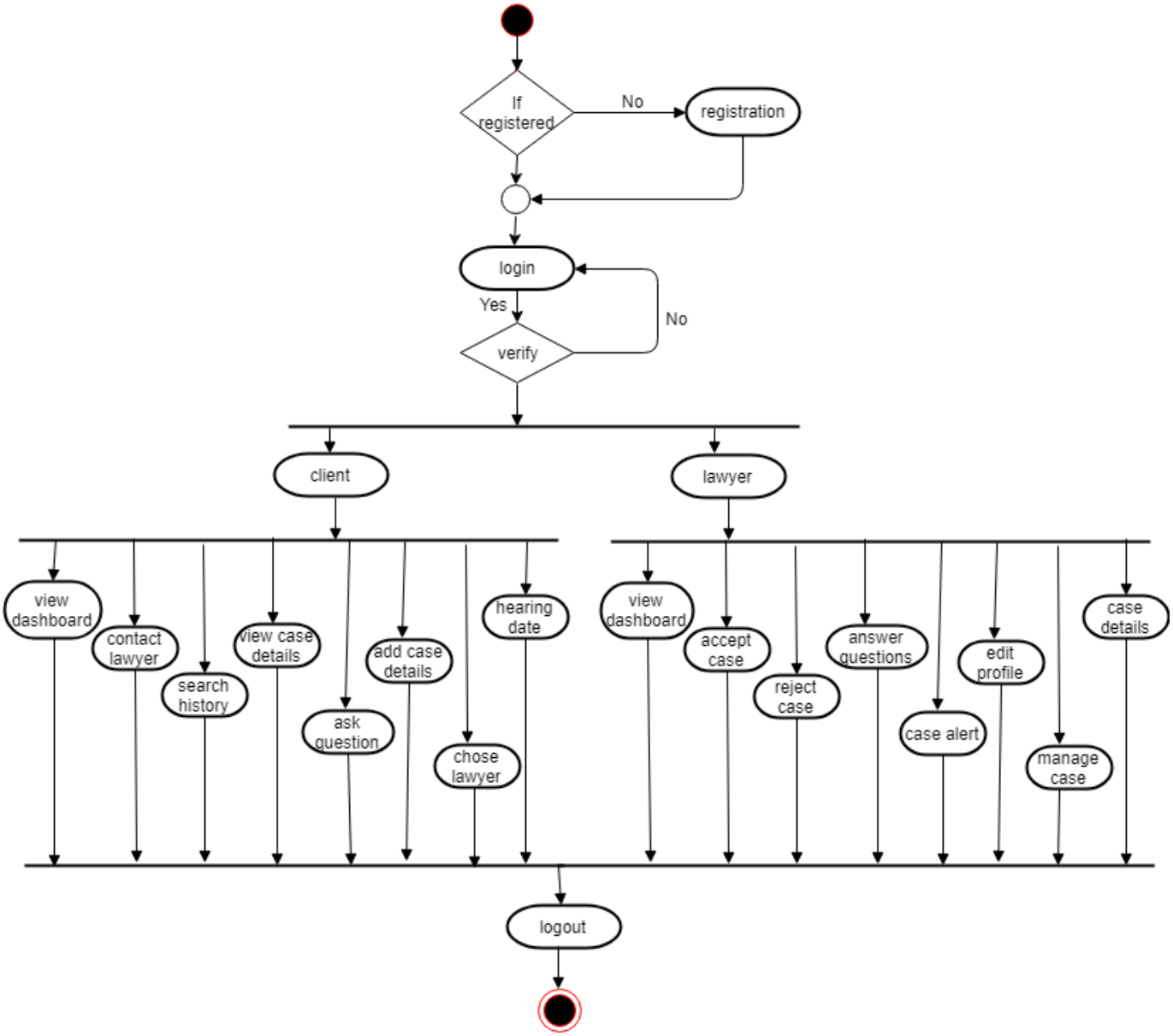



*Figure 14.Activity Diagram*

# 4.9. Data Flow diagram

The following Data Flow Diagram captures the flow of data in our website. The diagram will help in developing and understanding of system functionality and what are the different sources of data for judiciary system, what different transformations take place on data and what are final outputs generated by these transformations.

## 4.9.1 Data Flow Diagram Level0:

The following Data Flow Diagram Level0, also known as context diagram is containing one process that generalizes the function of the system:

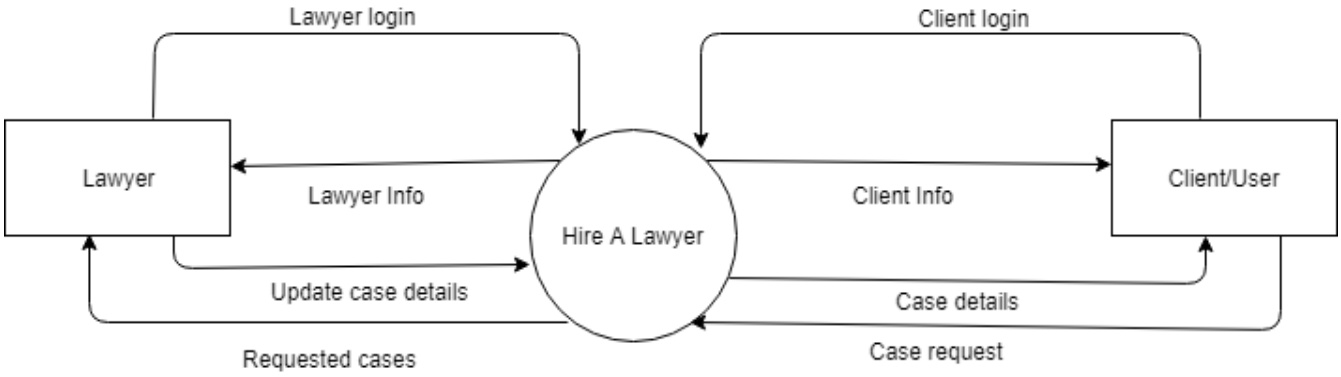



*Figure 15.Data Flow Diagram level0*

### 4.9.2 Data Flow Diagram Level1:

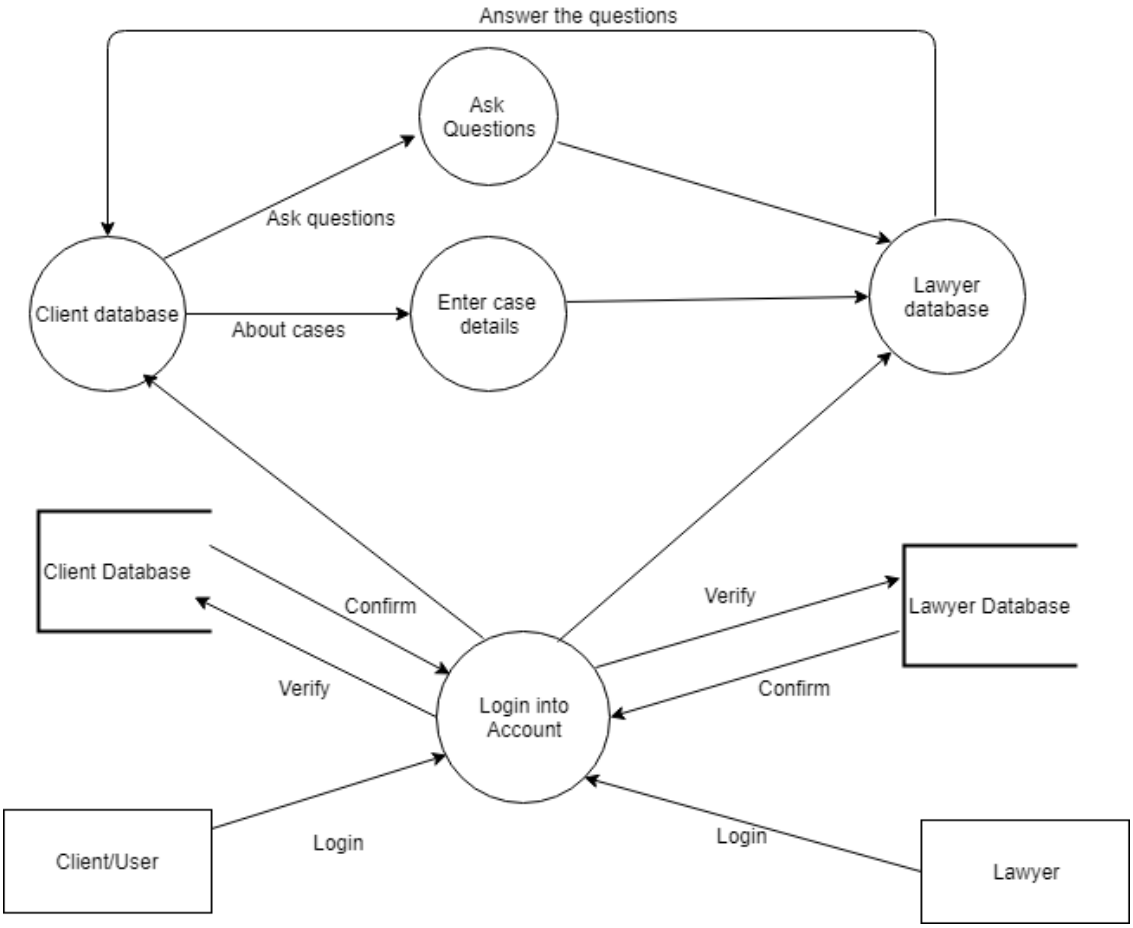



*Figure 16.Data flow Diagram level1*

### 4.9.3 Data Flow Diagram Level2:

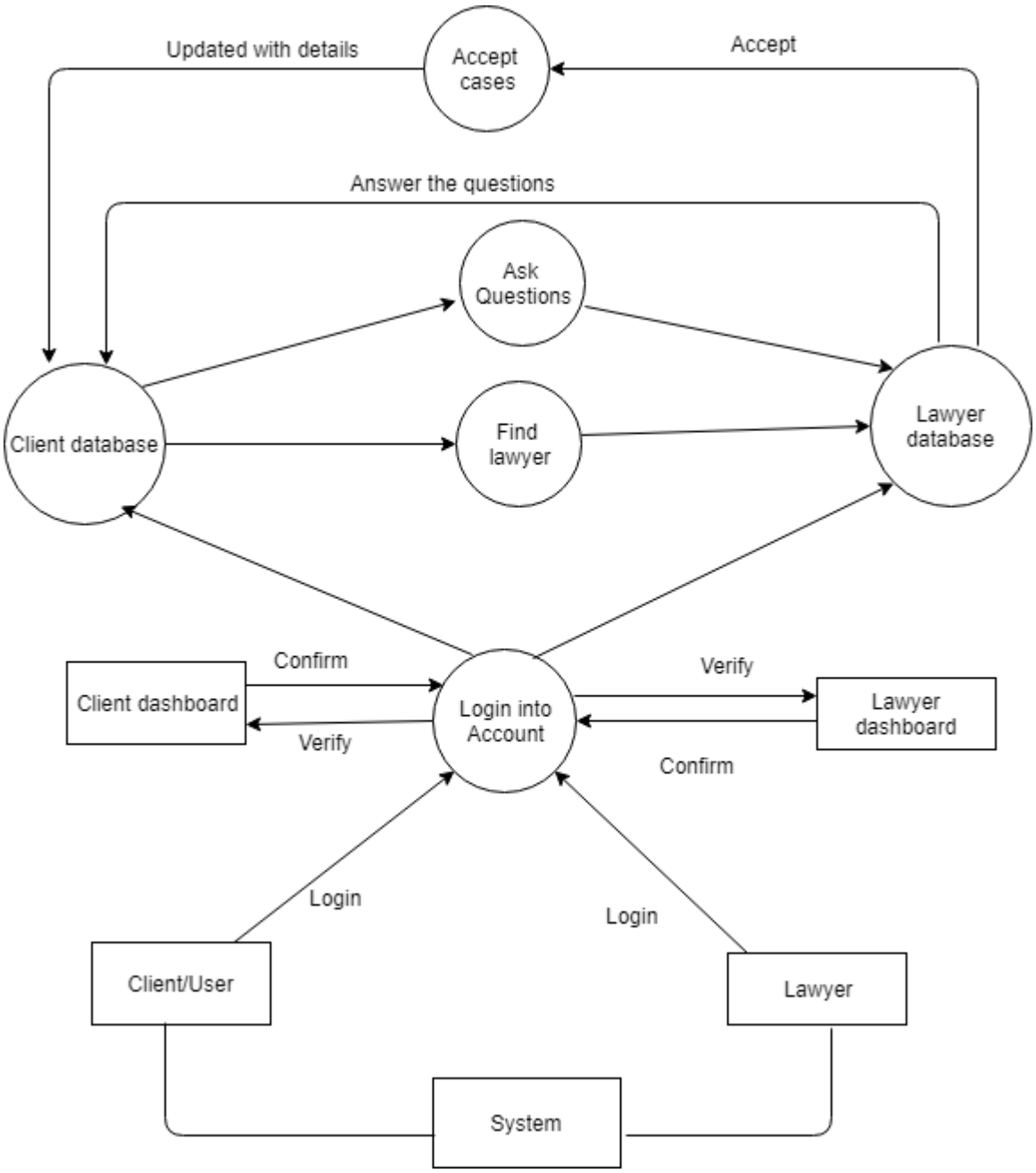



*Figure 17.Data Flow Diagram level2*

# Chapter 5
# Implementation

# Chapter 5: Implementation

In this stage, prototype will be deployed and implemented in real environment. Selected users will test on the prototype and evaluate the system. First, users will test on the first prototype based on the test plan created in stage 3 and test result will be recorded for further evaluation. Evaluation for the first prototype will be performed based on the test result gathered from users.

Interview and observation will be performed while testing on prototype. These activities are performed to verify the acceptance of the prototype by users. Observation is performed while the prototype is implemented in real environment to observe whether the functions and features provided in the prototype is satisfied the system requirements and users requirements discussed in early stage. If users are not satisfied with current prototype functionalities, interview will be done with users to acquire new requirements. Thus, system development process will loop back to design stage and the new prototype is developed based on the new requirements provided by user.

## 5.1. Components, Libraries, Web Services and stubs

Components:

We will use MVC for the development of our project.

Modal:

Models are mainly used to control the data manipulation in our software.

View:

Views will be used to control the interface of the website everything of the client end.

Controller:

Controllers will control the logic of our website.

## 5.2. Deployment Environment

Windows 10
PHP Server
MySQL Server
Visual studio code

Adobe Photoshop CC
Chrome, Opera, Edge

## 5.3. Tools and Techniques

### Tools:

IDE: Visual Studio Code

Languages: PHP, sql, JavaScript, css, Ajax, json

Database: SQLite

Frameworks: larval, bootstrap

### Methodology:

Model: Spiral model

## 5.4. Best Practices / Coding Standards

1- Commenting & Documentation. IDE's (Integrated Development Environment) have come a long way in the past few years. ...
2 - Consistent Indentation. ...
3 - Avoid Obvious Comments. ...
4 - Code Grouping. ...
5 - Consistent Naming Scheme. ...
6 - DRY Principle. ...
7 - Avoid Deep Nesting. ...
8 - Limit Line Length.

## 5.5. Version Control

We will use GIT as our version control system

# Chapter 6
# Testing and Evaluation

# Chapter 6: Testing and Evaluation

This chapter contains the testing of the software after implementations. In this chapter we will discuss about all types of testing for the purpose of checking the credentials of the software. Testing is the process to check the working are requirements of the user that are fulfilled or not. On the basic of the testing we will be able to evaluate the performance our software.

## 6.1. Use Case Testing

Use Case Testing is defined as a software testing technique that helps identify test cases that cover the entire system, on a transaction by transaction basis from start to the finishing point. In our system there are two major users, lawyer and client. In use case testing we will check that both the users are working properly and their dashboards are in active state.

**Test case 01:** The user must be authorized to access system.

**Test case 02:** To verify that the user enters a valid password.

**Test case 03:** To verify that upon clicking the login button system must move to dashboard of lawyer or client.

**Test case 04:** To verify that the lawyer and client can update and delete Files.

## 6.2. Equivalence partitioning

Equivalence partitioning is a software testing technique that divides the input data of a software unit into partitions of equivalent data from which test cases can be derived. In principle, test cases are designed to cover each partition at least once. This technique tries to define test cases that uncover classes of errors.

*Table 4.Equivalence Partitioning*

| **Equivalence partitioning** | | |
|---|---|---|
| | **Valid** | **Invalid** |
| Lawyer/client | Registered | Not registered |
| **Login Status** | Login | Retry |

## 6.3. Data flow testing

*Table 5.Data Flow Testing Results*

| Data Flow Testing Results | | |
|---|---|---|
| **Data Unit Name** | **Data Flow Method** | **Test Results** |
| Lawyer/client Registration Form | Interface-->Database, Database-->Interface | Pass |
| Login Form | Interface-->Database, Database-->Interface | Pass |
| Upload File Function | Interface-->Database, Database-->Interface | Pass |
| Lawyer dashboard | Interface-->Database, Database-->Interface | Pass |
| Client dashboard | Interface-->Database, Database-->Interface | Pass |

## 6.4. Unit testing

*Table 6.Unit testing of Software Units*

| Unit Testing of Hire a lawyer Software Units | | |
|---|---|---|
| **Software Unit Name** | **Testing Method** | **Test Results** |
| Lawyer/client Registration Form | Manual testing on local host | Pass |
| Login Form | Manual testing on local host | Pass |
| Upload File Function | Manual testing on local host | Pass |
| Lawyer dashboard | Manual testing on local host | Pass |
| Client dashboard | Manual testing on local host | Pass |

## 6.5. Integration testing

*Table 7.Integration testing of units*

| Integration testing of Hire a lawyer units | | | |
|---|---|---|---|
| **Software Unit Name** | **Test with Unit** | **Testing Method** | **Test Results** |
| Lawyer/client Registration Form | HAL Interface and relevant feature | Manual testing on local host | Pass |
| Login Form | HAL Interface and relevant feature | Manual testing on local host | Pass |
| Upload File Function | HAL Interface and relevant feature | Manual testing on local host | Pass |

## 6.6. Performance testing

In performance testing we check the performance of our system depending on hardware components. Time required to fulfill a user request and response it.

Performance metrics commonly include:

**Memory:** The working storage space available to a processor or workload.

**Throughput**: How many units of information a system processes over a specified time.

**Response Time:** The amount of time that a lawyer or client send a request to sign up or login and the system response to the request.

## 6.7. Stress Testing

Stress testing refers to hardware to determine whether its performance is satisfactory under any extreme and unfavorable conditions, which may occur as a result of heavy network traffic, process loading, under clocking, overclocking and maximum requests for resource utilization. In our system we will check the performance of our software when a bundle of cases will be handled against a single lawyer and when lawyers will manage their cases at this platform at the same time.

# Chapter 7
# Summary, Conclusion and Future Enhancements

# Chapter 7: Summary, Conclusion & Future Enhancements

## 7.1. Project Summary

Targeted audience for this software are those people who won't understand the judicial system or those who don't know how to hire the best lawyer for their relevant cases. We want to make a solution which generally provide the basic information about different categories of lawyers. We will also provide the best lawyers for the certain cases at the client's desired location.

We make a website and which provides the basic information of the different lawyers available at different locations across the Pakistan. Lawyers can register themselves and they can verify their credibility through contacting with us directly. People through using our software can find different information about cases and find their relevant lawyer to handle the case for them at their location. We also provide a platform for students to find out the best mentor for their counselling.

## 7.2. Achievements and Improvements

Our achievements throughout the project are given below:

- On time project delivery with complete functionality.
- Completely tested and problem-solving solution for users.
- Satisfaction of serving the humanity with knowledge.
- Satisfaction of meeting case details.
- Confidence to solve a problem that we have faced in our judiciary system.

## 7.3. Critical Review

### Problem

- Getting information about constitution of Pakistan
- Mentor selection for LLB students for practice
- Register or defend a case
- Difficult to find lawyer

**Objectives**

To provide a dedicated platform to lawyers to solve the cases. We wants to interact the lawyers with their clients at one platform .The public or the clients can meet their relevant lawyers by using our website .This may save the time and money. Clients can get all the information about their cases through our website. We also provide a platform for newly graduated law students to search their mentor for their better future.

**Methodology**

Spiral model

**Conclusions**

The project Hire a lawyer is developed with the aim to provide a dedicated platform to lawyers and clients to handle cases.

## 7.4. Lessons Learnt

Following are the learning outcome that we have learned during the development of the project:

- Learn how to apply software engineering techniques on a project.
- Learn about the importance and role of documentation in a software project.
- Learning of new technology and tools.
- Learn to meet deadlines.
- Learn to be flexible while developing a software project.
- Team building skill and responsibilities.
- Learn to manage the project effectively.
- Learn how to tackle the changes to the software scope during development.
- Learn how to develop use full and live working project.
- Learn communication skill for better project communication management.
- Learn new testing and development tools.
- Learn to develop better documentation.

## 7.5. Future Enhancements/Recommendations

Following are the future recommendations for our project HAL.

- Add new features and functionalities if needed and requested by the lawyers and clients.
- Improve the performance of the website recommended by automated tools such as Google website insights and on the request of the users.
- Perform the data warehousing of data of websites after two to five years to make knowledge archives.
- Any other changes and improvements for positive improvement for the project are welcome in future.

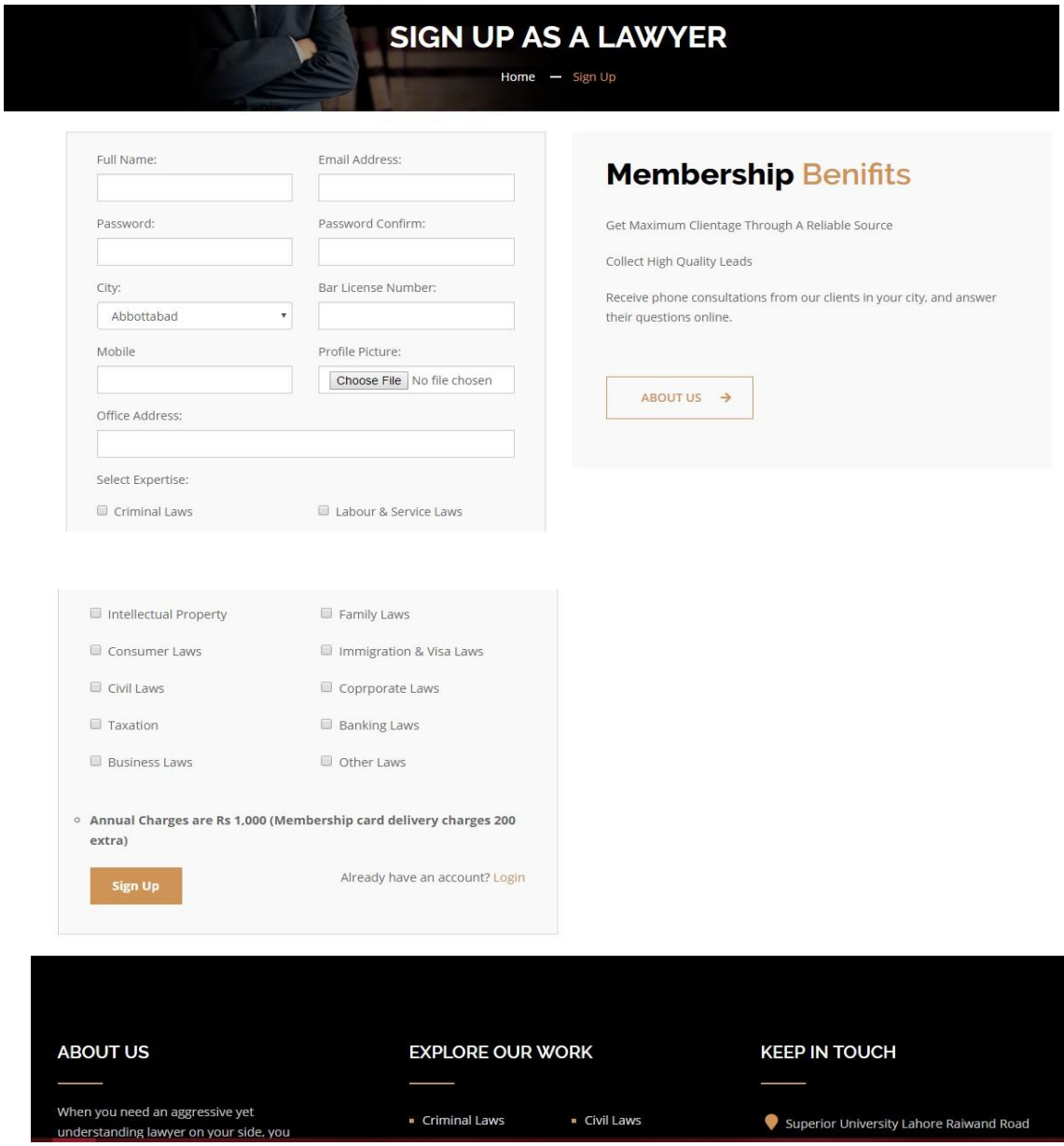
# Appendices

# **Appendix A:** User Manual

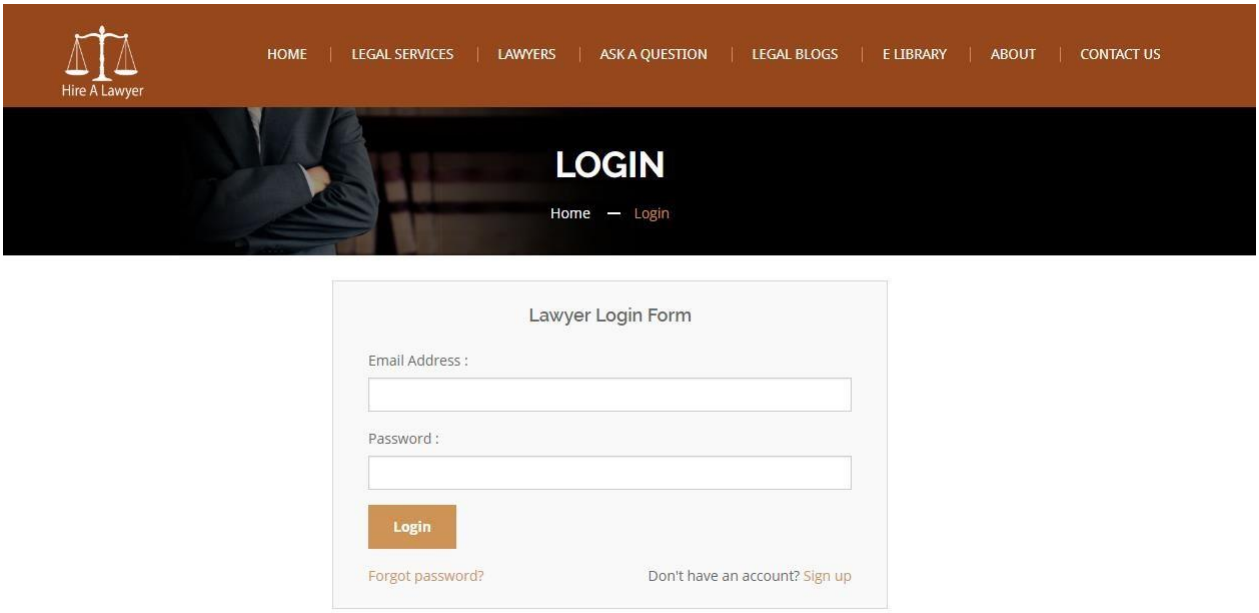
User Manual explains the step by step activities that should be performed to use the system. It provides all steps to carry out system tasks. It provides screenshots of the system functionality for lawyers and clients.

# Appendix A: User Login

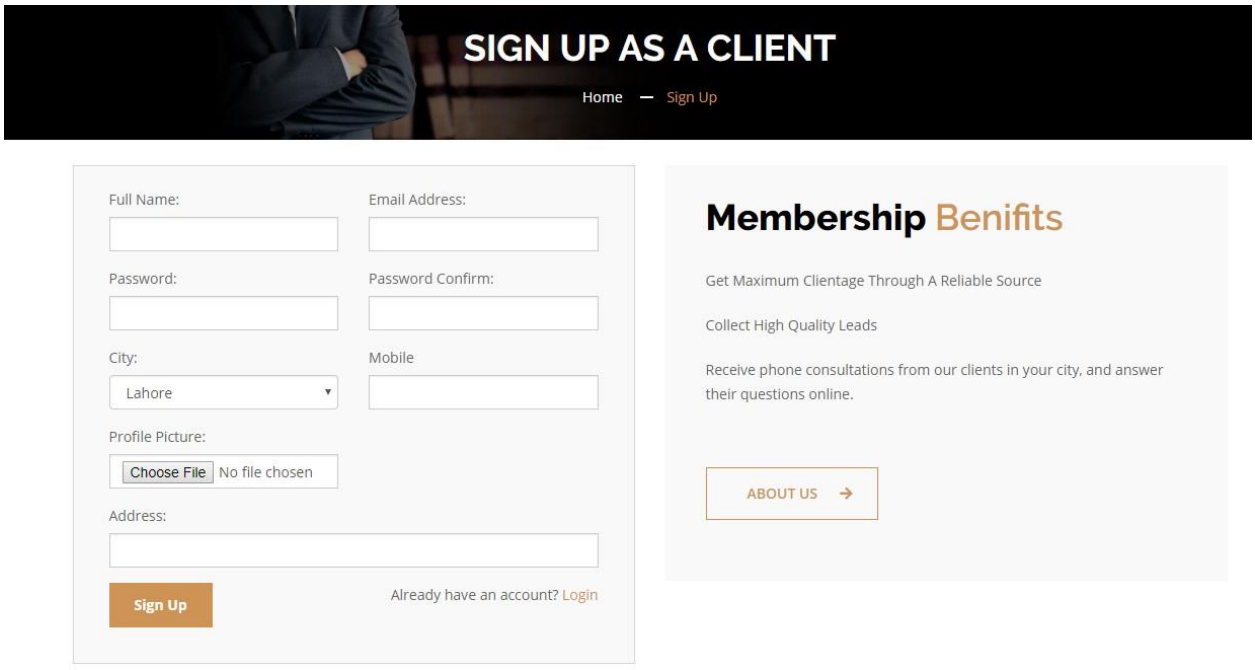
## A.1. Lawyer registration



## A.1.1. Lawyer Login



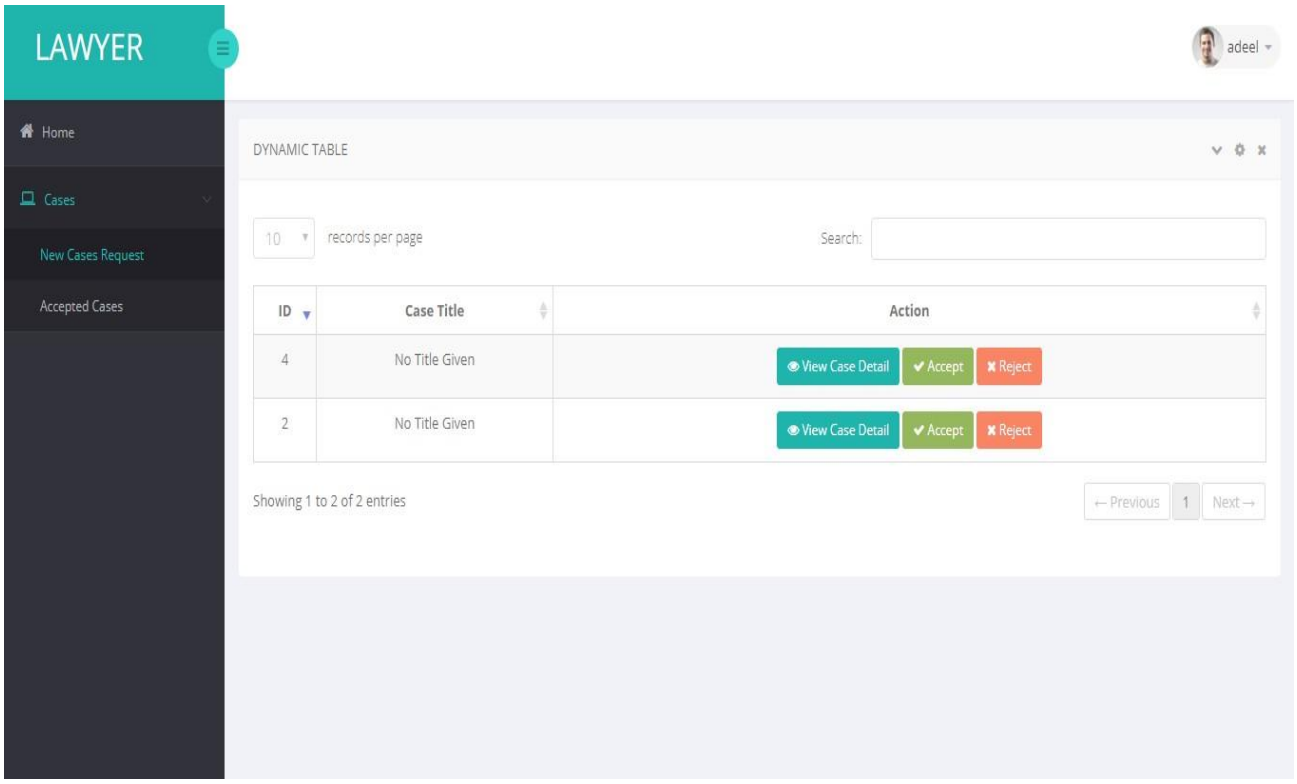

### A.1.1.1. Client registration



# **Appendix B:** Dashboard of users

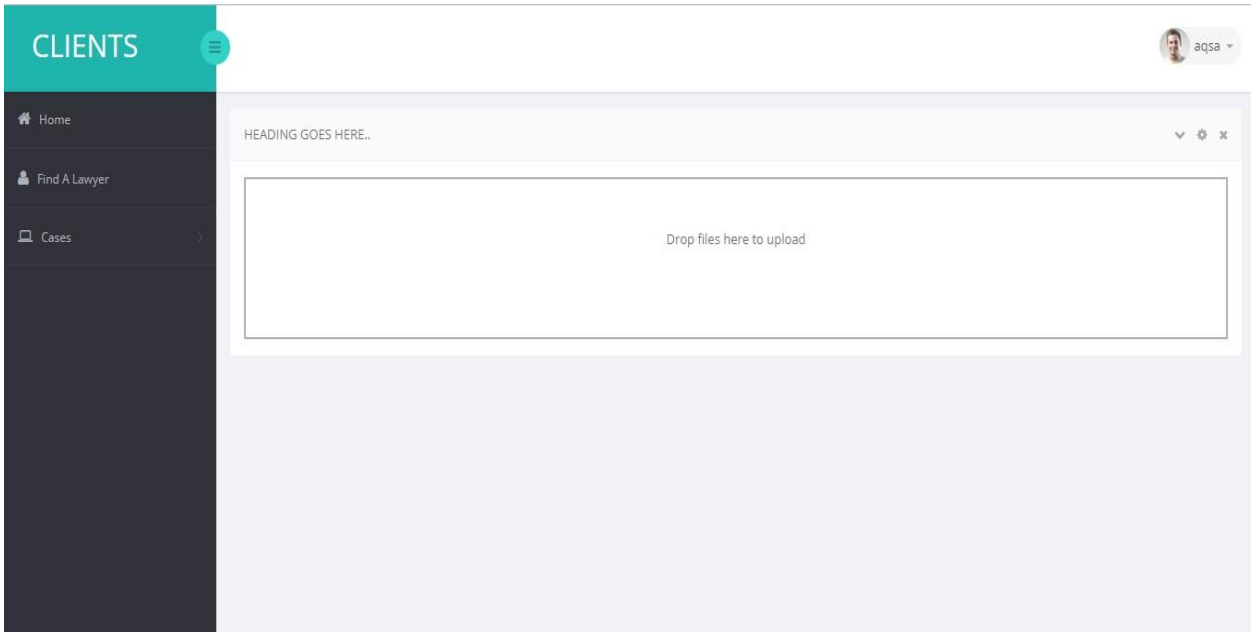
Dashboard of both user's lawyer and client explain the step by step activities that should be performed to use the system. It provides all step to carry out system tasks. It provides screenshots of the system functionality.

## B.1. Dashboard of lawyer




- Lawyer can view requested cases.
- Lawyer can accept or reject the requesting case.
- Lawyer can upload case details.
- Lawyer can search cases in accepted cases.

### B.1.1. Dashboard of client




- Client can edit profile.
- Client can upload any file related to his/her case.
- Client can request a lawyer.
- Client can find a lawyer.

# Reference and Bibliography

# Reference and Bibliography

[1] Black's Law Dictionary by Bryan A. Garner, Editor in Chief
Call Number: URI Reserves KF156 B53 2009
ISBN: 9780314199492
Standard legal authority. Citations to court decisions included when a term has been affected by legislation, court rules, or court decisions.

[2] Encyclopedia of Criminological Theory by Edited by Francis T. Cullen, Pamela Wilcox
Call Number: URI Reference HV6017 E527 2010
SBN: 9781412959186
Concise, to-the-point definitions of key concepts, ideas, schools, and concepts in criminological theory.

[3] Criminal Justice by Edited by Phyllis B. Gerstenfeld
Call Number: URI Reference KF9214.5 C75 2006
ISBN: 1587652188