UNIVERSITY OF WOLLONGONG IN DUBAI

Assignment Cover Sheet

Subject Code: CSIT321
Subject Name: Project
Submission Type: Turnitin
Assignment Title: Grad Project Report
Student Name: Harman Singh, Abeera Amir, Ayushi Agnihotri, Eman Mulla, Muhammad Shaad
Student Number: 6618704 , 6621612, 6617037, 6636767, 6670428
Student Phone/Mobile No. 056-7805891,054-4354257,054-7057391,050-7589351,0563370497
Student E-mail: hs797 ,aa757,aa401,ems110,ms546 @uowmail.edu.au
Lecturer Name: Dr.Mai
Due Date:
Date Submitted:

PLAGIARISM:	**DECLARATION:**
The penalty for deliberate plagiarism is FAILURE in the subject. Plagiarism is cheating by using the written ideas or submitted work of someone else. UOWD has a strong policy against plagiarism. The University of Wollongong in Dubai also endorses a policy of non-discriminatory language practice and presentation. **PLEASE NOTE:** STUDENTS MUST RETAIN A COPY OF ANY WORK SUBMITTED	I/We certify that this is entirely my/our own work, except where I/we have given fully-documented references to the work of others, and that the material contained in this document has not previously been submitted for assessment in any formal course of study. I/we understand the definition and consequences of plagiarism. **Signature of Student:**

Optional Marks:

Comments:

Lecturer Assignment Receipt (To be filled in by student and retained by Lecturer upon return of assignment)

Subject: **Assignment Title:**
Student Name: Harman Singh **Student Number:** 618704
Due Date: **Date Submitted:**
Signature of Student:

Student Assignment Receipt (To be filled in and retained by Student upon submission of assignment)

Subject: **Assignment Title:**
Student Name: Harman Singh **Student Number:** 6618704
Due Date: **Date Submitted:**
Signature of Lecturer

LAWFLOW

FINAL REPORT

Spring'22 – Autumn '22 Batch

This technical document has been prepared as part of requirements for the capstone graduation project subject CSIT321

From University of Wollongong in Dubai

Developed by

SHAAE-256

Abeera Amir.....6621612
Ayushi Agnihotri.....6617037
Eman Mulla.....6636767
Harman Singh.....6618704
Muhammad Shaad.....6670428

CERTFIICATION

We, Team SHAAE-256, declare that this technical document, submitted in fulfilment of the requirements for the completion of the graduation project subject CSIT321, in the Faculty of Engineering & Information Sciences, University of Wollongong in Dubai, is wholly our own work unless otherwise referenced or acknowledged. The document has not been submitted for qualifications at any other academic institution.

Team Members:

Abeera Amir *Abeera Amir*

Ayushi Agnihotri *Ayushi Agnihotri*

Eman Mulla *Eman Mulla*

Harman Singh *Harman Singh*

Muhammad Shaad *Muhammad Shaad*

EXECUTIVE SUMMARY

This report focuses on the development of the Lawflow Desktop Application. Based on research and background study, the idea behind the application is discussed in depth i.e., the problem that is being solved, the importance of developing this application and how it will be beneficial to our main stakeholders. The project goals, budget, risks, limitations, and future enhancements are discussed in detail along with WBS that shows the project's development process. In consideration to the market gap, as competitor analysis was done to compare our product to other products from the legal-tech field.

The designing of the Lawflow desktop application was divided into different stages such as design constraints, methodology, architecture, high-level and low-level design as well the User Interface Design. The report further investigates the crucial stage of implementation testing of the application where the testing approach, test case and system maintenance are discussed.

ACKNOLWEDGEMENTS

We would like to extend our gratitude to Dr. Farhad Orumchian , our faculty mentor for his continued support and encouragement for the development of our application Lawflow.

A special thanks to our capstone project coordinators : Dr May El Barachi and Dr Zeenath Khan, for their help and support in guiding the team in achieving our goal as well as their encouragement to maintain our progress in track. We would also like to acknowledge with much appreciation, the crucial role of the WISP Mentors – Nayab Nadeem and Abeer Haroon for their assistance in several technical areas such as Workflow and the general development of our application. A special thank you to Asma Damankesh for her guidance and input on the project's development.

Finally, we would like to express our gratitude to our family and friends for their trust and support, as well as the panel for their feedback that helped us improve our application. Thank you very much for always guiding us and supporting us in achieving our objectives.

TABLE OF CONTENTS

Figures

Tables

I. INTRODUCTION

1. Innovation

There has been an increase in the use of communication systems and information systems in various fields and their use has expanded in all directions. Judicial Systems in particular are no exception to this. There is no democracy without a system of swift and transparent justice, therefore it is necessary to introduce technology to the system as it can save both time and decrease the number of pending cases thus boosting the efficiency of the services provided to the citizens and the society. Having technology-assisted systems can improve efficiency, decrease the budget and increase trust between society and the judiciary(Rose J. et. Al., 2013).

The lawyers are the primary stakeholders of the judicial system and therefore need to assist in order to create a smooth workflow for them over all so as to improve the current condition of the system. An approach can be investigated to resolve this issue. Creating a system that caters to the needs of lawyers and assists them in carrying out day to day tasks with ease.

Our product focuses on this issue by targeting the prime conditions that 11 lawyers need assistance with every day. It will be installed on the lawyer's desktop, and it will be designed with features such as file management, document translations and many more. The system uses implement Artificial Intelligence, Workflow Engine, duplicate document detection, document categorization, as well as a feature that would provide document translations and encryption for all confidential data stored on the desktop application.

2. Problem Statement & Background study

In today's judicial world, the number of cases has been rapidly increasing and this means the vast number of cases have become hard to manage and categorize due to the system being technologically handicapped. As of 31 January 2021, there are currently 37,251,615 pending cases in India, and this has created a substantial backlog of cases that can eventually lead people to lose faith in the judicial system. (https://njdg.ecourts.gov.in/njdgnew/?p=main/pend_dashboard). Based on conducting several surveys and interviews with a group of lawyers and barristers about the situation, it was brought to our attention that the challenges they face in their everyday tasks can be eased and made simpler.

LawFlow intends to address the needs of the law sector by providing a system that would allow to efficiently manage day-to-day tasks (such as file management) using a workflow engine, while also making sure professionals in the field are spending less time administrating menial tasks and more time fostering better client relations.

As compared to a simple file management system, our system uses advanced features to solve various issues. The system will implement Artificial Intelligence, Workflow Engine, duplicate document detection, document categorization, as well as a feature that would provide document translations, ensuring better understanding between the attorneys and their clients, and potentially cutting down on miscommunications. Keeping in mind the need for attorney-client

privilege and the nature of this profession, the system will aim to encrypt all private and confidential file using AES/RSA methods of encryption so that the access is limited, and the information stored within the system is secure.

3. Our solution

LawFlow intends to address the needs of the law sector by providing a system that would allow to efficiently manage day-to-day tasks (such as file management) using a workflow engine, while also making sure professionals in the field are spending less time administrating menial tasks and more time fostering better client relations

Our project aims at organizing the work of lawyers as well as using advanced features to enhance it. The objective is to use current technologies to offer the best possible outcome, with the highest quality within our ability as well as regarding and being careful of the limited time and budget availabilities.

The aim of the proposed project is:

- Provide a solution to handle multiple files
- Automate filing tasks
- Have an advanced search bar to search up documents with ease
- Translation of documents to various languages
- Workflow Engine

II. FEASIBILITY

1. Project Charter

Project Name	LAWFLOW	**Start Date**	April 27,2022
Project Manager	Eman Shahid Mulla	**Target Date**	TBD

Project Team			
Name	**Department**	**Student ID**	**Email**
Abeera Amir	Backend/Frontend	6621612	aa757@uowmail.edu.au
Ayushi Agnihotri	Backend/Frontend	6617037	aa401@uowmail.edu.au
Eman Shahid Mulla	Backend/Frontend	6636767	esm110@uowmail.edu.au
Harman Singh	Backend/Frontend	6618704	hs797@uowmail.edu.au
Muhammad Shaad	Backend/Frontend	6670428	ms546@uowmail.edu.au

Project Purpose	**Milestones**		
	Milestone Number	**Milestone Name**	**Due Date**
In a legal system, every lawyer must work with a lot of cases and with it comes a multitudinous number of tasks and responsibilities. This puts a lot of pressure on the lawyers and managing so many cases can be tiresome. Technology is slowly becoming more intertwined with the legal industry and by investing in new solutions, it can make the system more efficient and therefore increase the firm's productivity and have a better connection with their clients.	1	Project Idea Proposal	Week 1
	2	Project Finalization	Week 2
	3	Background Researchand course completion	Week 3 to Week 6

(Mulla, et al.)

LawFlow aims to provide a solution that will inculcate Artificial Intelligence, Workflow Engine, duplicate document detection, documentcategorization and even a translation feature to help make day to day tasks more efficient.	4	Feasibility Report	Week 3
Business Objective	5	Finalizing the initial technology and becoming aware of other alternatives.	Week 4
Every lawyer roughly manages or works on 50 to200 cases a year and that's excluding the contract clients.[1] Handling so many cases sometimes at once or one after the other gets hard and tiresomewhich can also indirectly affect the lawyer's ability to perform well in court hearings there making them inefficient. LawFlow's aim is to help organize the day-to-day tasks and responsibilities for lawyers and uses advanced technology and features to enhance it. We plan to implement current technologies suchas AI, workflow engines and many more and bring out a product that will help make the judicial system more efficient in the long run while keeping in mind the time constraints and budget.	6	Researching on Key system factors	Week 4
	7	Planning Report	Week 4
	8	Project Requirements Document	Week 6
	9	Design Report	Week 9
	10	Proof of concept presentation	Week 10

Deliverables	Stakeholders
Research project attributes UI design and wireframes for application Login Module Document scanning and categorization by folder Advanced search with Artificial Intelligence Translation feature Workflow Engine Encryption	Government OfficialsLaw Firms Lawyers Team SHAAE-256
Risks	**External Dependencies**
Privacy Issues Datasets Security Issues Database Leaks	***Faculty Mentor*** Dr. Farhad Oroumchian ***WISP Mentor/s*** Abeer Haroon & Nayab Nadeem

Table 1: Project Charter

2. Scope Statement

2.1 Scope Background and Justification

Our team approached multiple individuals in the law profession with the intent of identifying their day-to-day concerns in terms of technology and software, and how their workload could be handled more efficiently. The feedback received was mainly centered around the need for an application that would be able to organize documents and an automated filing system, which is what LawFlow is keen on delivering.

2.2 Scope Description

With several documents for each court case, it becomes difficult to organize and keep track of documents required for each case. Furthermore, searching for documents within documents of several other cases becomes a task and could further lead up to the mixing of different court documents.

Our project aims at resolving these issues by creating a desktop application which uses a workflow engine and a file management system with several advanced features. These features include an advanced Artificial Intelligence (AI) search which helps the users search for documents by entering keyword or by clicking an image of the document they were searching for and matching it with an existing document. Another notable feature is eliminating document duplication by scanning its contents. To add to this, the documents get filtered on their own into categories. Before a client meeting, the workflow engine will help the lawyer determine what documents are needed for that hearing, the application will have a scheduling system, which would help in recognizing who the client is and when the meeting is.

The system would also be incorporate a translation feature, which would translate both documents and translation as the person speaks. The system would also provide a translation option that would allow the documents stored to be translated into any language of choice for better review and understanding. To add to this, the system will have encryption over important document so as to prevent access to everyone and limiting access to the legal team/lawyer.

2.3 Business Objectives

Our project aims at organizing the work of lawyers as well as using advanced features to enhance it. The objective is to use current technologies to offer the best possible outcome, with the highest quality within our ability as well as regarding and being careful of the limited time and budget availabilities.

The aim of the proposed project is:
- Provide a solution to handle multiple files
- Automate filing tasks
- Have an advanced AI search bar to search up documents with ease
- Translation of documents to various languages
- Speech to text translation

2.4 Deliverables

Deliverable Number	Deliverable Details	Expected Completion
1	Research on different project attributes: The features of ‘Lawflow’were decided via research and consulting different software engineers and mentors to ensure that the system had a viable scopein the limited timeframe.	2 weeks (Completed)

2	Create a UI design and wireframes for this application: using different available applications such as Adobe XD and proto.io	3 days
3	Login Module: Backend will have to be created to help store thedetails onto the database, front end for entering the details, .NET(C#)/ python for backend	6 days
4	Document scanning and sending to folders: OCR apis: googlevision/ azure Read	2 weeks
5	Advanced search using AI: Tensorflow with C# will be used todevelop the advanced AI search	2 weeks
6	Translation option: Google translate/ AWS translate/ Libra translate	4 days
7	Workflow engine to determine upcoming meetings: Apache airflowfor workflow engine/Process maker can be used for easier integration of workflow onto our system	2 weeks
8	Documents encryption: AES/RSA to encrypt the files, a symmetrickey will be created	1 week

Table 2:Deliverables

2.5 Future Enhancements

Due to limited time and scope constraints, the current system would be limited to the above specified deliverables. In the future the system could:

- Be designed in a way that would allow it to be tailored to cater to other professions and occupations outside law.
- The system could be developed even further to be made into a mobile application

3. Goals and Objectives

The main aim of LawFlow is to reduce generic and day-to-day tasks of Lawyers/Barristers. Some of the tasks even require business firms to hire assistants thus leading to added costs or even the Lawyers doing tasks themselves which is time consuming. LawFlow also aims to reduce the time taken between legal processes by speeding up the time taken for smaller tasks and consequently increasing efficiency and productivity. Tasks such as document translation, search, organizing documents, and following a workflow that are essential to any law firms or individual lawyers in their tasks are all managed by LawFlow. 7 Furthermore, LawFlow is a software that considers market gaps identified during the interview process by our main stakeholders. Therefore, it aims to finish this market gap by launching an application that fits the needs of our main stakeholders. It consists of features that may not readily be available in the field of law. With LawFlow, these features can be personalized, and a safe space is created for the confidential documents.

4. PESTEL Analysis

Political

Political factors may include the information saved by the law profession (lawyers/ paralegals) within the application.

Environmental

Physical environment conditions such as weathers will have no effect on the system itself. However, the database of the system will be hosted by a third party, depending on the weather condition in the place of hoisting, the database will work efficiently or may have some breaks due to unforeseen weather conditions.

Social

Since our application caters to the personal needs of the lawyer, the lawyer may be hesitant about the information he/she is sharing as the information can be extremely confidential. While the application is created to ensure the safety of the users data is prioritized.

Technological

There have been several technological advancements in a short period of time. During the process of development of this application, different advancements could take place in the technologies that are being used by us. For instance, a package could release a better version, or there could be significant changes in the programming language used by use (version/ null safety etc). While developing the application, the best interest of users must be kept in mind hence using the most stable and recent version of different technologies which is also compatible with a majority of desktop devices.

Economical

While currently the application prototype will be free of cost, in the future the application could be released on a larger scale, making certain features paid hence enabling a revenue.

Legal

Legal issues mainly stem from the misuse of information entrusted by the user within the application. Confidential data will be stored in the application, the legal laws differ from country to country and will have to be heavily regarded in the process of development of the application.

5. Competitor Analysis

Features	Lawflow	Clio	Filevine	Mycase
Workflow Engine	☑	☑	☑	☑
Document Translation	☑	☒	☒	☒
Document Categorization	☑	☒	☒	☒
Appointment Schedular	☑	☑	☑	☑
Advanced Search	☑	☒	☒	☒

Table 3: Competitor Analysis

6. Work Breakdown Structure

Semester I

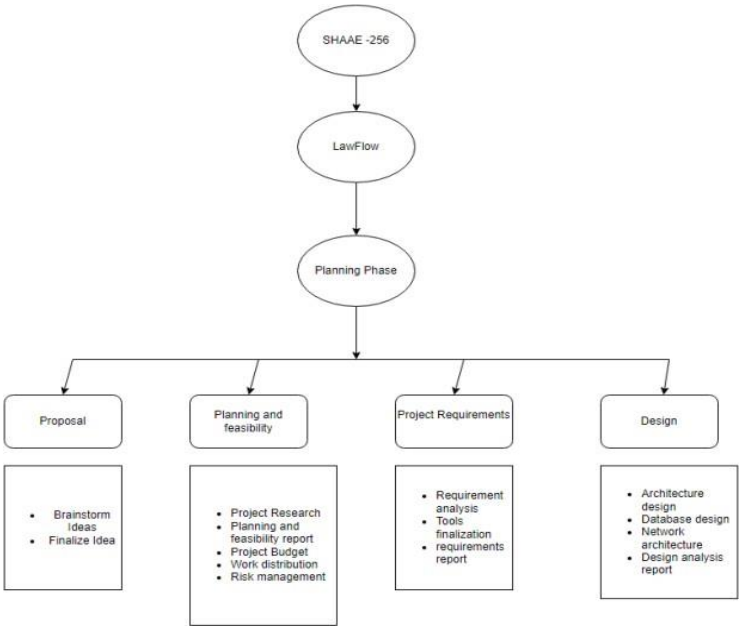


Figure 1: WBS Planning Phase

Semester II

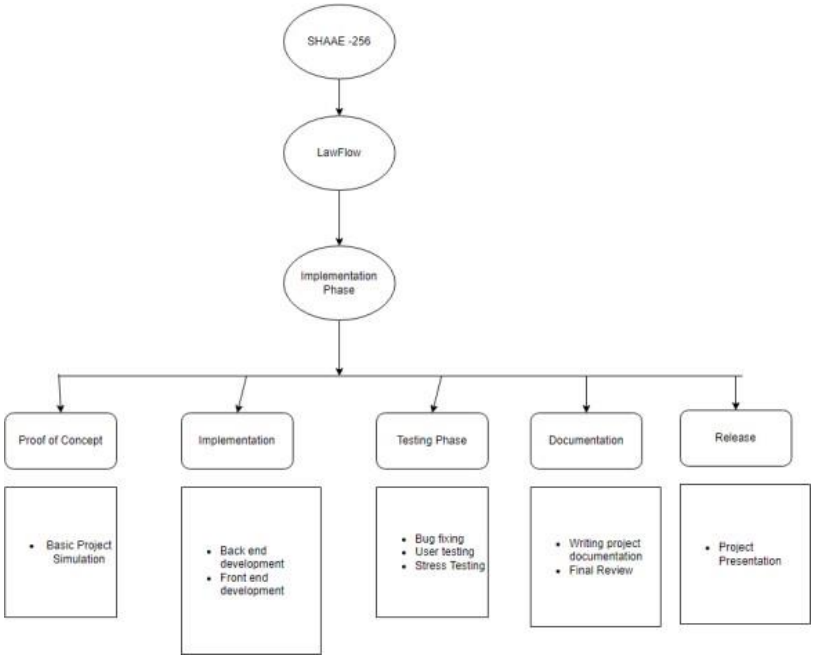


Figure 2: WBS Implementation Phase

7. Project Budget, including Time and Cost Estimates

7.1 Cost and Time estimate Per Feature

Deliverable	Time estimate (Days)	Cost Estimate (AED)	Assumption
UI framework	2	0	All group members have necessary and efficient knowledge about UI creation apps (proto.io / adobe XD)
Login module	4	0	N/A
Document scanning and sending to folders	14	0	N/A
Advanced Search Bar	7	0	Group mates have well researched the different options available and their ability to be used in the project efficiently.
Translation option	7	0	N/A
Workflow engine	20	200 Apache airflow: free, budget kept in case a different technology is used.	Group mates have well researched the different options available and their ability to be used in the project efficiently.

Documents encryption	10	50 – 150 Use of RSA/ AES libraries could have some cost involved.	N/A
Integration testing	3	0	N/A
Unit testing (Each feature)	6	0	All group members are aware about how to create unit test programs
System testing	4	0	N/A
Database manipulation	Entire duration of phase 2	150 – 300 Hosting the database on a cloud service	N/A
Total	82 days	~650	

Table 4: Cost and Time estimate per feature

7.2 Cost and Time estimate Per Phase

- Planning Phase

Phase	Time estimate (Days)	Cost Estimate
Proposal	20	$0
Planning and feasibility	14	$0
Project requirements	14	$0
Design	21	$0

Total	83 days	$0

Table 5: Cost and Time estimate planning phase

- Implementation Phase

Phase	Time estimate (Days)	Cost Estimate (AED)	Assumptions
Implementation	69	300 Use of paid libraries, paid services for cloud	N/A
Testing phase	14	$0	N/A
Documentation	69	$0	Will be done as the implementation is being done.
Release	7	150-300	N/A
Total	83 days	~650	N/A

Table 6:Cost and time estimate implementation phase

8. Performance Measurement Baselines

8.1 Scope baseline

The main objective of LawFlow is to aid its users, i.e., employees in the legal system (paralegals) increasing the efficiency of their day-to-day tasks. The project aims at incorporating a workflow engine and AI to implement a document recognition system. It is necessary that the team stays within the scope during the entirety of the project in order to achieve the project goals within the specified time period.

To satisfy this performance metric, the team should focus on the following:

- Outline the purpose of the project
- Meet deliverables
- Create a proper workflow by assigning work and documenting changes.

8.2 Cost baseline

The cost estimate provided in section 5.2 provides insights on how to stay within budget as a team. It is necessary that the teams work cost-efficiently to avoid any overhead costs. It must be noted that while adhering to the cost baseline, the quality of the product should not deteriorate. Quality and cost control must be considered when deploying the system.

8.3 Schedule baseline

The project charter and the work breakdown structure include several tasks and important milestones mentioned that the team must complete within the specified deadline to ensure timely delivery of the product. Tracking and measuring the process of the team weekly and its members daily along with continuous meetings to meet the project goals in a timely manner.

9. Milestones and Associated Dates

Milestone number	Milestone name	Due date	Description	Status
1	Project Proposal.	Week 1	A project proposal with 3ideas, ranked from 1 to 3based on preference.	Completed
2	Project Idea Finalization.	Week 2	The ideas are presented to the subject coordinator, and the project with highest priority is chosen.	Completed
3	Literature review.	Week 3-6	Research concerning as to how the technology will be implemented is taking place by team members.	Completed
4	Feasibility Report.	Week 3	A study conducted to gather the strengths, weaknesses, areas for improvement and open issues posed against theproject.	Completed

(Mulla, et al.)

5	Finalizing the initial technologyand becoming aware of other alternatives.	Week 4	The technology to be used throughout the project as well as alternatives and backup plans are finalized.	Completed
6	Researching on key system factors.	Week 4	Key ideas of the systemare finalized such as workflow engine, document recognition system and voice to textsystems.	Completed
7	Planning Report	Week 4	This report consists ofthe project plan and itsprocesses.	Completed
8	Project Requirement Document	Week 6	The requirements document defines whatthe product needs. It states the purpose of the product and what it mustdo. It does not define how to provide or build what you need.	Completed
9	Design Report	Week 9	The design report focuses on the design of the project, key features and major deliverables ofthe project. The design report includes variationsin design allowing the team to prefer one over the other.	Completed

10	Proof of concept presentation	Week 10	The project will be simulated in front of a panel of judges to determine the potentialof the project.	Completed

Table 7:Milestones

10. Staffing

Name	Type	Availability	Cost
Harman Singh	Team Member	Full Time	$0
Eman Mulla	Team Member	Full Time	$0
Abeera Amir	Team Member	Full Time	$0
Muhammad Shaad	Team Member	Full Time	$0
Ayushi Agnihotri	Team Member	Full Time	$0
Dr. Farhad Oroumchian	Faculty Mentor	By Appointment	$0
Abeer Haroon	WISP Mentor	By Appointment	$0
Nayab Nadeem	WISP Mentor	By Appointment	$0

Table 8:Staffing

(Mulla, et al.)

11. Risk Management Plan

Category	Risk	Rank	Impact	Solution
Database	Database Leaks	Very High	Very High	Access controls, Two-Factor Authentication, User login systems etc.
	Data Loss	Very High	Very High	Automatic and timelybackup of data
Software	Unmet Expectations	Very High	Very High	Throw-Away prototyping, Testing andTeam discussion
Physical	Distance from thesystem may resultin issues with voice recognition	High	Low	The user must sit in a 2meter distance from themicrophone
Cost	Budget Variance	High	High	Create budget forecasts and plan out costs for each phase
Datasets	Lack of Diversity	High	Very High	To be able to properly train the artificial intelligence systems, datasets of voice samples must be obtained from various accents as the system will be used by a diverse set of people. Lack of proper training will result in a unreliable system
Time	Late delivery of project	Moderate	High	Timely discussions and deadlines of tasks.

				Detailed project schedules
				Online Database owned by third party companies pose an even higher level of risk ask it becomes more vulnerable to attacks and data losses

Table 9:Risk management plan

12. Open Issues

Physical distance will play a huge role in the accuracy of the voice recognition system. The further away the sound signals are coming from, the weaker the transcript and more error-prone it will be. To reduce this risk as it has a very high chance of occurring every time the translation feature is used, it is recommended to be in close proximity to the system for it to be able to understand and translate the speaker accurately.

The AI will be initially trained with English language only since it is a developing project and has a limited time scope it will be impossible to train it with multiple languages. As this makes the project scope limited, more languages will be added in the future further expanding the scope.

To ensure strict security of confidential files and data, strict security controls are implemented. Some of the controls include Access controls, Two-Factor Authentication, User login. However, this risk cannot be eliminated by a 100 per cent. Other factors such as lawyer's personal decisions will have a huge impact on the safety of data. The system, on its own, is designed to reduce the error of accidental data loss or sharing/mixing up client data by implementing a client-profile structure that is only visible to the lawyer.

III. REQUIREMENTS

1. Introduction

1.1 Overview

There has been a surge in the number of unresolved cases over the past few years. The judicial system is unable to handle a large number of cases due to a lack of technology or access. The 2022 Economic Times reported that "Over 4.70 crore cases are pending in various courts" in India as of now. (The Economic Times, 2022) This shows that there is a substantial amount of backlog, and this could lead to people losing faith in the judicial system.

The main concept of LawFlow focuses on easing the tasks for the judicial system by bringing attention to the needs of the lawyers and how simple tasks can be automated and help the overall workflow be smooth and easy-going. The system intends to address the needs of the law sector by providing a system that would allow the lawyer to efficiently carry out day-to-day tasks such as file management) using a workflow engine, while also making sure professionals in the field are spending less time administrating menial tasks and more time fostering better client relations.

Our system uses advanced features to solve various issues, it will implement Artificial Intelligence, Workflow Engine, duplicate document detection, document categorization, as well as a feature that would provide document translations, ensuring better understanding between the attorneys and their clients, and potentially cutting down on miscommunications. Keeping in mind the need for attorney-client privilege and the nature of this profession, the system will aim to encrypt all private and confidential file using AES/RSA methods of encryption so that the access is limited, and the information stored within the system is secure. This is system is not a solution to replacing all the tasks that a lawyer handles, but it will definitely be a way to improve the current situation of the judicial system by assisting the professionals in hand with everything that they do and thus improve the client-attorney relationships as well.

1.2 Scope

With several documents for each court case, it becomes difficult to organize and keep track of the documents required for each case. Furthermore, searching for documents within documents of several other cases becomes a task and could further lead to the mixing of different court documents.

Our project aims at resolving these issues by creating a desktop application that uses a workflow engine and a file management system with several advanced features. These features include an advanced Artificial Intelligence (AI) search which helps the users search for documents by entering keywords or by clicking an image of the document they were searching for and matching it with an existing document. Another notable feature is eliminating document duplication by scanning its contents.

To add to this, the documents get filtered on their own into categories. Before a client meeting, the workflow engine will help the lawyer determine what documents are needed for that hearing, the application will have a scheduling system, which would help in recognizing who the client is and when the meeting is. The system would also incorporate a translation feature, which would translate both documents and translation as the person speaks. The system would also provide a translation option that would allow the documents stored to be translated into any language of choice for better review and understanding. To add to this, the system will have encryption over important documents so as to prevent access to everyone and limit access to the legal team/lawyer.

1.3 Stakeholders

- **Lawyers**
  Lawyers are at the forefront of this application. LawFlow is intended to be used by lawyers effectively allowing them to maximize their potential and increasing efficiency while performing their day-to-day tasks.

- **Law firms**
  In order to maximize the lawyer's potential, law firms need to authorize lawyers to use this application. By investing in this application, law firms will be able to increase their employee's potential and therefore generating higher monetary gains.

- **Team SHAAE-256**
  Team SHAAE-256 is the development team of LawFlow and aims to provide lawyers with a better management system by integrating various services in one application.

- **Lawyer's Client**
  Lawyer's client is an indirect stakeholder in this project. By using law flow, lawyers will be able to increase their focus on the client's case therefore, increasing their chance in winning their respective case.

1.4 Definitions

Word	Definition
Artificial Intelligence	Problem solving and learning intelligence which is practiced bymachines.
Multifactor Authentication	An authentication method that allows a user to access a website or application only after presenting two or more pieces of evidence to theauthentication mechanism.

Document Translation	A translation service that translates an entire document into the preferredlanguage.
Workflow engine	A software application that helps organizations initiate and automatetasks.
Hybrid Storage	Storage method in which data is stored locally as well as on the cloud.

Table 10:Definitions

1.5 Document Conventions

***Introduction*:** This part is a brief introduction to the system for our readers. Included in the document is, System overview, goals and objectives, stakeholders, and detailed description of the scope of the system.

General design constraints: In this section, we will discuss the product environment and Characteristics of users using this system.

***Non-functional requirements*:** This section describes various non-functional requirements of the system.

System Features: This section provides insights into system features. Representing various functionalities of the system.

Version History: This section shows the version history of the document*.

**Subject to change.*

1.6 Assumptions

The device/laptop should be connected to the internet as the app is based on the client-server architecture.

2. Design Constraints

2.1 Product Environment

There has been an increase in the use of communication systems and information systems in various fields and their use has expanded in all directions. Judicial Systems in are no exception to this. There is no democracy without a system of swift and transparent justice, therefore, it is necessary to introduce technology to the system as it can save both time and decrease the number of pending cases thus boosting the efficiency

of the services provided to the citizens and the society. Having technology-assisted systems can improve efficiency, decrease the budget and increase trust between society and the judiciary.(Rose J. et. Al., 2013)

The lawyers are the primary stakeholders of the judicial system and therefore need to assist in order to create a smooth workflow for them over all so as to improve the current condition of the system. An approach can be investigated to resolve this issue. Creating a system that caters to the needs of lawyers and assists them in carrying out day to day tasks with ease.

Our product focuses on this issue by targeting the prime conditions that lawyers need assistance with every day. It will be installed on the lawyer's desktop, and it will be designed with features such as file management, document translations and many more. The system uses implement Artificial Intelligence, Workflow Engine, duplicate document detection, document categorization, as well as a feature that would provide document translations and encryption for all confidential data stored on the desktop application.

2.2 User Characteristic

Users: Our main users for the system would be the members of a legal team. They can download the application on their computers and store all their information and files in it, enabling the system to help them in their tasks further.

Human interactions that our system would require:

- Users need to provide their credentials to create a account and login.
- Users need to upload their client's documents and other crucial case information into the system allowing the system to sort them out and help the user in the upcoming tasks by automating their workflow.
- User would need to speak to or present audio to be able to convert the audio into a text format.
- User would need to schedule meetings and some information about then so that the system can remind the lawyer and also assist them in the documents they need for it.

2.3 Mandated Constraints

- **Document Recognition**
  The document that is sent to the system needs to clear and any smudged ink on the document can decipher results.

2.4 Potential System Evolution

- **Generic Version**

In the future, LawFlow can be designed to take into account different professions and have a more generic layout to fit the needs of various professions. A newer version of the application can be designed with generalized features and be implemented in the future.

- **Mobile Application**
  LawFlow can be further scoped down to be able to provide some services/features on a mobile application.

- **Additional Languages**
  Currently LawFlow only supports English. To be able to implement LawFlow on a global scale it is necessary for it to be able to support various languages thus more languages can be added in the future.

3. Non-functional Requirements

3.1 Usability Requirements

LawFlow desktop application is easy to use, and it provides efficiency for the lawyer since it helps reduce the times to manually carry out their tasks and they can use that time for other important tasks and since the system is automated, the risk of human errors will be minimal.

3.2 Operational Requirements

The user needs to have the desktop application installed on their system.
The recordings should be done in a quiet environment to prevent any hindrance in results and the user should be within 1m from the device while recording any audio.

3.3 Performance Requirements

This application aims to minimize latency not only in terms of data acquisition speed, but also in terms of its functionality. Therefore, the backup server and cloud will operate allday to improve access speed. The system should have a cold boot time of 5 seconds and a warm boot time of less than 2 seconds. If multiple users are using the system at the same time, 90% of the operations will be completed in less than 3 seconds such speeds owing to the hybrid-based storage LawFlow uses.

3.4 Security Requirements

To ensure security of confidential client data, encryption of documents will be implemented using RSA/AES methods. Furthermore, MFA (Multi-Factor Authentication) will be implemented in LawFlow to prevent unauthorized access or in the

case if credentials are compromised. The creation of accounts for each user will keep the relevant client data in a safe space and free from intruders.

3.5 Safety Requirements

LawFlow is a safe-to-use application and poses no life risk to the users.

3.6 Legal Requirements

Legal issues mainly stem from the misuse of information entrusted by the user within the application. Confidential data will be stored in the application, the legal laws differ from country to country and will have to be heavily regarded in the process of development of the application.

3.7 Documentation and Training

The software will provide an in-depth tutorial to the user whenever they sign up, showing them all the features of the system and how they work with each other. The software will also have a dedicated help page that includes a description of all the features and solutions for any common issues occurring with the system.

3.8 External Interface

- **User Interface**

In LawFlow, Placement of major functionalities will remain consistent throughout the user interface, creating an environment that allows users to be familiar with the controls. This will allow for increased usability, eliminating confusion which is necessary considering how crucial time management is for lawyers.

As mentioned above, LawFlow's user interface will be clear and easy to understand. The use of buttons, icons and colours have been carefully selected to facilitate integration and navigation within the platform. The following are some of the designs that will be implemented in LawFlow to ensure that the app is virtually appealing and convenient to use.

Colour coding to help highlight important tasks. For instance, upcoming meetings or other tasks that the user will have to perform will be highlighted or marked in red.

Additional notes can easily be added on every screen or within a particular section with the use of the note's icon.

You can quickly add a new client, task or calendar event using the + icon.

The three vertical lines mean "wait, there's more". You can easily access the menu by hovering over this icon.

- **Software Interface**

  Front End
  We will be developing our desktop application using REACT JS. This allows the developers to create a rich UI experience as mentioned in the paragraph above.

  Back End
  We will be using a server to host our back end which will be run using python. The server will host different services(Google vision etc.) that we wish to integrate in our application.

4. System Features

4.1 Adding Documents via Scan

Feature	Adding document via scanning
Description And Priority	The user scans a hardcopy document, and itgets converted into a soft copy and gets stored. Cost: High Risk: Medium Value: Medium
Use Case	• User scans a handwritten letter of a client, and it gets converted into a softcopy and gets saved in the system
Additional Requirements	

Table 11Adding document via scanning

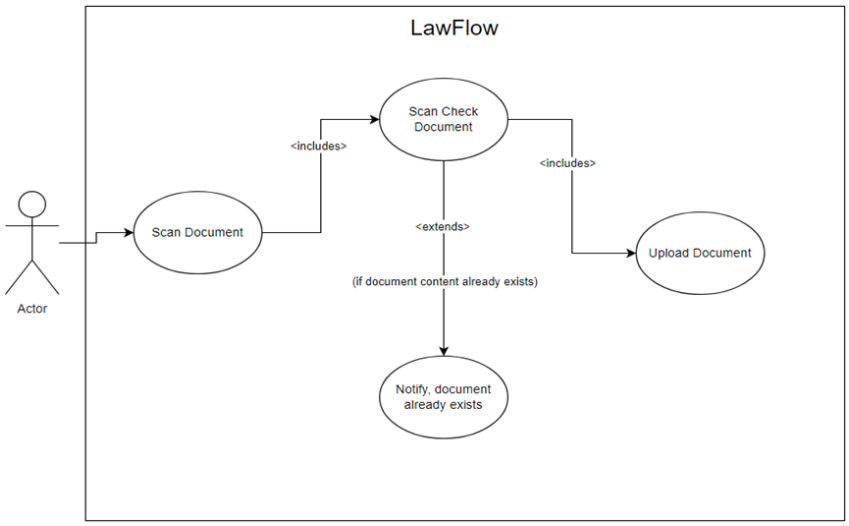


Figure 3:Adding document via scanning

4.2 Scan check existing documents

Feature	Scan check existing documents
Description And Priority	When the user scans a document or is uploading one, the system already checks other pre-existing documents to make sure itisn't already present in the system. Cost: Low Risk: High Value: Medium
Use Case	• User uploads documents, the systemnotifies the user that the same document already exists in the system.
Additional Requirements	

Table 12: Scan check existing documents

4.3 Next Court Stage for Clients

Feature	Next court stage for clients
Description And Priority	The system will notify the user of which is the next legal stage for each client and whatare the necessary things needed for it.
Use Case	• The system informs the user that Client X's court hearing is in 2 days and these documents need is needed for it.
Additional Requirements	

Table 13:Next court stage for clients

4.4 Cancel meeting

Feature	Cancel meeting
Description And Priority	Allows the user to cancel meetings which have been added to the calendar via a simplegui option. Meeting will get removed from the calendar, Time slots and workflow will be adjusted accordingly. Priority: High Cost: low Value: low
Use Case	• Cancel meeting via click of button
Additional Requirements	

Table 14:Cancel meeting

4.5 Calendar Link

Feature	Calendar Link
Description And Priority	Enables the meetings and important tasks to be added to the schedular as well adding thetasks to the calendar. Priority: High Cost: Low Value: medium
Use Case	• Scheduling important meetings
Additional Requirements	

Table 15:Calendar Link

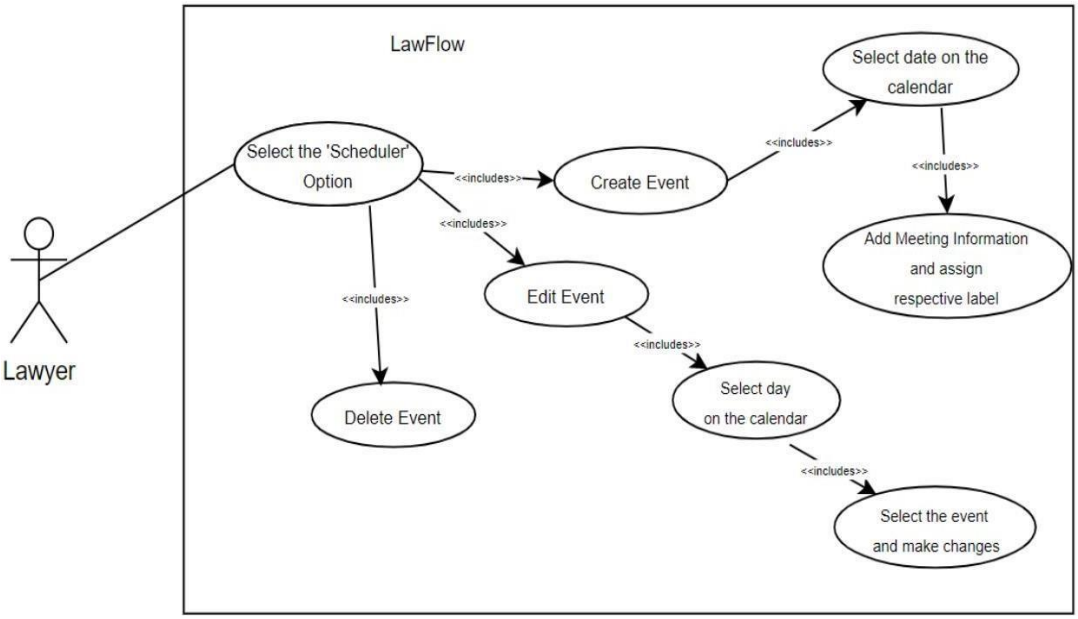


Figure 4: Use Case Schedular

4.6 Automatic folder creation based on keywords

Feature	Automatic folder creation based onkeywords
Description And Priority	When a user scans a document, the softwarewill detect important keywords. These keywords will enable the creation of automatic files. For instance: if the file contains the word "Evidence", if folder already does not exist under client name, folder gets created, else the file is simply added to the folder called "ClientID/ evidence" Priority: High Cost: Medium Value: High
Use Case	• Automatic folder creation accordingto keywords
Additional Requirements	

Table 16: Automatic folder creation based on keywords

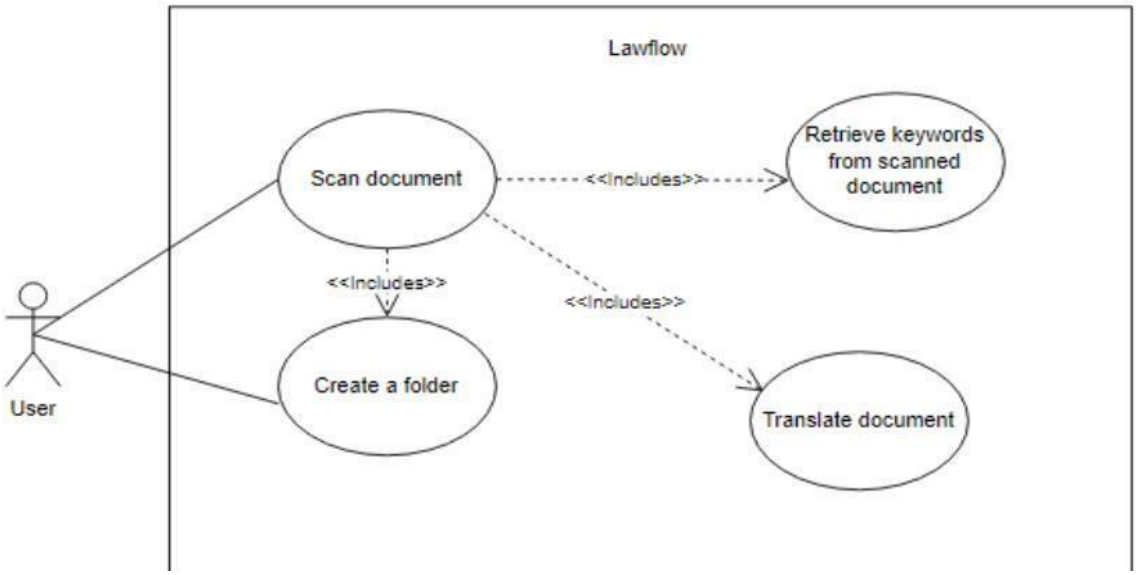


Figure 5: Use case Scan document

4.7 Meeting Tracker – Prevention of Clashes

Feature	Meeting Tracker – Prevention of Clashes
Description And Priority	Each meeting is set up with clients on an automated basis, the system must ensure thattwo client meetings never clash in order to prevent confusion. Priority: High Cost: Medium Value: High
Use Case	• System must get updated in order toincorporate prevention of meeting clashes
Additional Requirements	

Table 17: Meeting tracker

4.8 Login

Feature	Login
Description And Priority	To ensure safety and security of confidentialinformation of each client and lawyer, the lawyers will work on their own separate accounts. Lawyers will receive their own credentials to access their private accounts and then be able to only access data of their clients. Cost: Low Risk: Medium Value: High
Use Case	• Entering Credentials &Authentication
Additional Requirements	

Table 18:Login

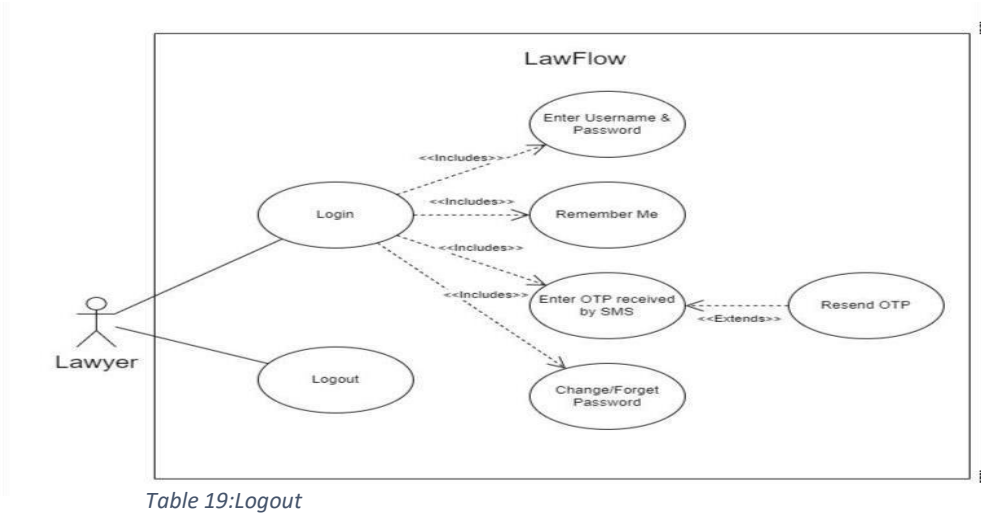


Table 19:Logout

4.9 Logout

Feature	Logout
Description And Priority	Adding to the safety feature, lawyers will beable to sign out from their account to prevent unauthorized access to confidential data. Cost: Low Risk: Medium Value: High
Use Case	N/A
Additional Requirements	

Table 20:Logout

4.10 Document Translation

Feature	Document Translation
Description And Priority	This feature consists of translating a givendocument from a language that the user selects to the language the user wants to translate it to. Cost: High Risk: Medium Value: High
Use Case	• Selecting translation languages &saving
Additional Requirements	

Table 21:Document Translation

4.11 Lawyer Scheduler

Feature	Lawyer Schedule
Description And Priority	This feature allows the lawyer to manually input their unavailable days in the scheduleso that no meeting is scheduled on that day.This will grey out the selected days on the calendar to avoid accidental meeting schedules.
	Cost: Low Risk: Low Value: Medium
Use Case	N/A
Additional Requirements	

Table 22:Lawyer Schedule

4.12 Search

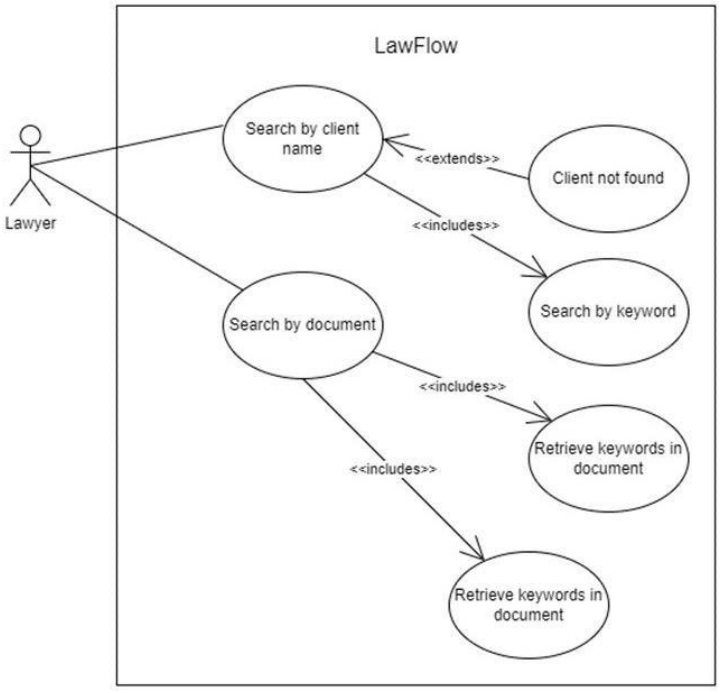


Figure 6:Search

4.12.1 Search By Client Name

Feature	Search by client name
Description And Priority	Allows the user to search for a specific client by entering their initials or full name,then displaying information about the client including ways to contact, schedule meeting. Cost: Low Risk: Low Value: Low
Use Case	• Search by client name
Additional Requirements	

Table 23:Search by client name

4.12.2 Search By Document

Feature	Search by document
Description And Priority	Allows lawyers to retrieve documentsinstantly using document search technology. Cost: Low Risk: Low Value: Low
Use Case	• Search by document
Additional Requirements	

Table 24:Search by document

4.12.3 Search By Keyword

Feature	Search by keyword

Description And Priority	Gives lawyers the ability to find informationthat they're looking for even when searchingwith incomplete information. Cost: Low Risk: Low Value: Low
Use Case	• Search by keyword
Additional Requirements	

Table 25: Search by keyword

4.12.4 Search by Image

Feature	Search by image
Description And Priority	Allows lawyers to upload an image, thensearch to find information similar to the contents of the image provided. Cost: High Risk: High Value: Low
Use Case	• Lawyer searches using an image
Additional Requirements	

Table 26:Search by image

(Mulla, et al.)

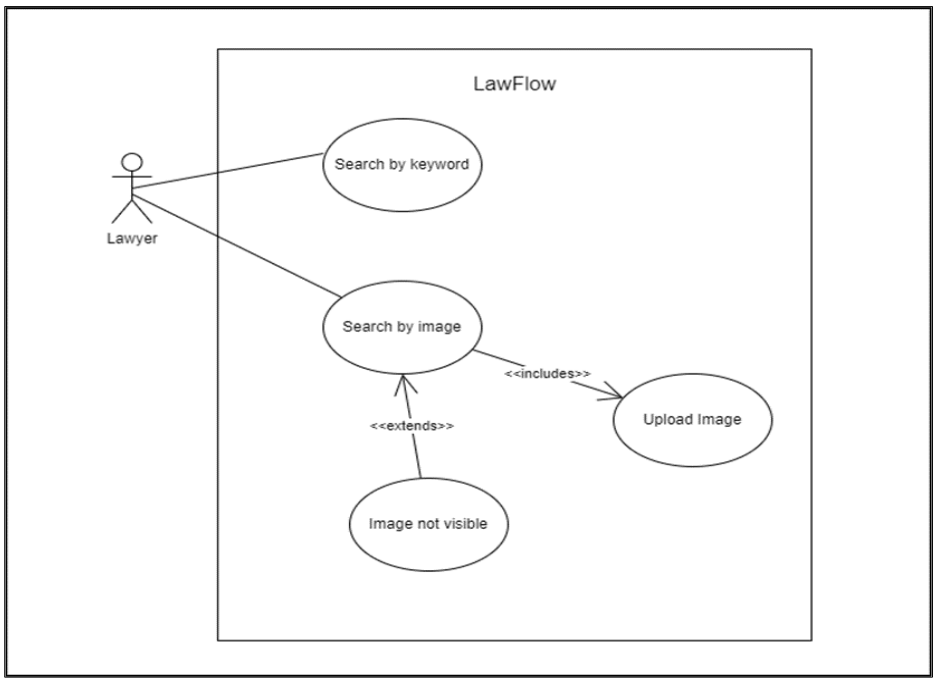


Figure 7: Search by image

4.13 Add Meetings

Feature	Add meetings
Description And Priority	Allows lawyers to schedule a meeting, afterspecifying the date, time and location giventhe schedule is not preoccupied. After the meeting is scheduled, electronic-mails are sent to the recipients. Cost: Medium Risk: Medium Value: High
Use Case	• Schedule a meeting
Additional Requirements	

Table 27:Add meetings

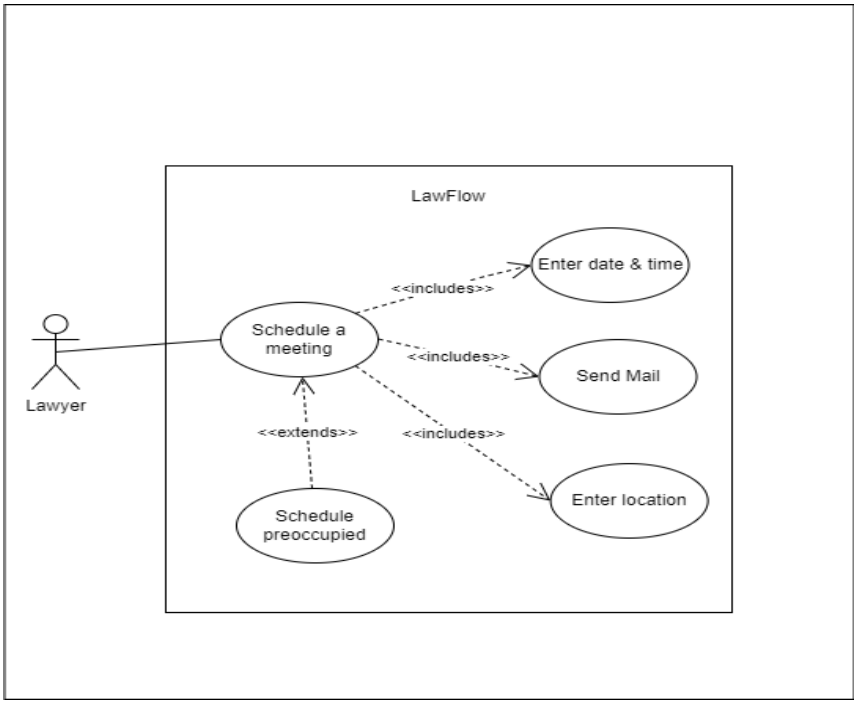


Figure 8:Meeting Schedular

4.14 Add new clients

Feature	Add new clients
Description And Priority	Allows lawyers to add new clients, after specifying the client details and personal information given that the client that does not already exist. After the client is created,the lawyer can view the client in the Client Overview Section on the Homepage. Cost: Medium Risk: High Value: High
Use Case	N/A
Additional Requirements	

Table 28:Add new clients

IV. DESIGN

(Mulla, et al.)

1. Introduction

1.1 System Overview

Over the past few years there has been an increase in the number of unresolved cases, and this has put the judicial systems under a lot of pressure because the lack of use of technology or its access. It is a necessary to create better and constructive solutions to help this cause.

Lawflow focuses on ensuring that the tasks are completed with ease, making sure that lawyer's needs are addressed by providing better and more efficient methods that make the workflow swift and easy going. The system aims to address the needs of the law sector by creating a platform that allows lawyers to conduct their everyday tasks by implementing workflow engine so that the lawyers can spend less time on completing menial tasks and more time focusing on their clients and cases. Some tasks require law firms to hire assistants to do the generic and day to day tasks of their lawyers, this leads to added costs and expenditure on more human capital or the lawyers themselves spend hours on paperwork which can be very time consuming. Lawflow focuses on reducing this gap between legal processes by increasing the overall productivity and efficiency. Tasks like document translation, search, organizing documents, and following a workflow that are essential to any law firms or individual lawyers in their particular tasks are all managed by LawFlow.

The system uses advanced features for completing tasks. Technology such as Artificial Intelligence, Workflow Engine, duplicate document detection, document categorization, document translation will be implemented to ensure more effective client relations and a better client-attorney relationship. Acknowledging the need of attorney-client privilege and the nature of this profession, the system will be encrypting all private and confidential files by implementing AES/RSA methods of encryption so to limit the access to assigned users and securing all personal information on the system.

Lawflow is not a replacement to the tasks handled by lawyers, but it is a solution to improve the current situation of the judicial system by providing assistance to lawyers in managing the tasks better and therefore fostering better client-attorney relationships. It is a software that considers market gaps identified during the interview process by our main stakeholders. Therefore, it aims to finish this market gap by launching an application that fits the needs of our main stakeholders. It consists of features that may not readily be available in the field of law. With LawFlow, these features can be personalized, and a safe space is created for the confidential documents.

1.2 Design Map

The design report focuses on how the development team will work on the system's design and how the system's human-computer interaction will take place. Diagrams are included to help understand the system design better.

***Architecture*:**

- *Network Architecture*: This includes the structure of the network that shows how the application connects to cloud containing servers that has the APIs and the database.
- *System Architecture:* This consists of the connection between the hardware to the software and APIs to the user interface.

High- Level Design

- *Sequence Diagram*: It shows the sequence of the interaction between the classes for different use cases.
- *Class Diagram*: Shows the classes where the data is stored and how different classes connect to one another.
- *Database Diagram*: This diagram is related to the class diagram, and it shows the relations between classes and can be used to extract the data.
- *Deployment Diagram*: It shows the execution of the system architecture and how the elements will be implemented during system development.

Low-Level Diagram

- This contains the algorithms used in the system development and the wireframes used. The elements developed are the final result and it describes the functionality of the feature on the user interface.
- *Pseudocode:* It is used to design the algorithms by making a structure of the program before coding.

User Interface

- *UI Screens*: Provides an overview of how the application will be designed for the lawyer's use.
- *Flow charts:* Shows the user interaction with the system and the system's response.
- *Usability Guidelines*: Guidelines followed during the UI design of the system.
- *Usability Heuristics*: Heuristics followed during the UI design of the system.

1.3 Definitions and Acronyms

Word	Definition
Artificial Intelligence	Problem solving and learning intelligence which is practiced by machines.

Multifactor Authentication	An authentication method that allows a user to access a website or application only after presenting two or more pieces of evidence to the authentication mechanism.
Document Translation	A translation service that translates an entire document into the preferred language.
Workflow engine	A software application that helps organizations initiate and automate tasks.
Hybrid Storage	Storage method in which data is stored locally as well as on the cloud.
Google Vision	A software that uses machine learning to understand images. Can detect objects, faces and hand-written text.
Cloud Vision – Text Detection & OCR	A software that detects text in images and performs extraction if necessary.
Data privacy	Data protection against those who should not have access to it, as well as the capacity of individuals to control who has access to their data.
Data Encryption	A security method in which information is encrypted and can only be accessed or decrypted by a user with the corresponding privilege.
NoSQL DB	Non-table databases that store data differently rather than the use of relational tables. There are different types of NoSQL databases, depending on their data model. They offer flexible schemas and can be easily scaled for large amounts of data and heavy user loads.

Table 29:Definitions and Acronyms

(Mulla, et al.)

2. Design Considerations

2.1 Constraints

Following are the constraints that were found during the system's development and that remains currently in the design phase of the system:

Document Recognition

- The document sent to the system should be clear and any smudged ink on the document can decipher results.

Generic Version

- In the future, LawFlow can be designed to have a more generic layout to fit the needs of various professions. A newer version of the application can be designed with generalized features and be implemented in the future.

Mobile Application

- LawFlow can further be more inclusive to be able to provide some services/features on a mobile application.

Additional Languages

- Currently LawFlow supports English. To be able to implement LawFlow on a global scale it is necessary for it to be able to support different languages.

2.2 Risks and Volatile Areas

Some of the risks involved in the project include:

- Internet speed
- Image quality for scanning and translation (if necessary)
- Audio Quality for speech to text recognition

Internet speed: Internet speed is essential since the application is not completely local, it is a hybrid between cloud and local storage, while some features maybe available locally, to have access to the entirety of the app, and for the working of the app to its full potential, a stable internet speed will be required.

Image Quality for scanning and translation: The image should be of reasonable quality in order to ensure smooth scanning, each document will be scanned for the required keywords in order to determine the folder that it must be stored in, the words should be of clarity in order to do so. Furthermore, the same thing would apply for translation of documents.

2.3 System Environment

System Environment will be dynamic. It can be used at work, at home or in public places. For the system to work, it requires:

- User permissions
- User's microphone access
- User's camera access

2.4 Design Methodology

The design methodology used by us for this project will be agile. Instead of delivering all of the required results together, the team will work on delivering smaller portions in increments.
The main reason for choosing agile is having customers involved in each development process of the app this in turn will lead to the development of a product with higher quality.

Each module or iteration will have the following phases:

- Initiate
- Plan
- Build and Manage
- Close Phase & Review

In the initiation phase the module gets identified, followed by planning where the end goal is understood in more depth, the modules are divided amongst the developers. The developers then begin the process of building the required module and managing it. The module in then tested and is reviewed by the necessary audience in order to initiate immediate change if required.

3. Architecture

3.1 System Architecture

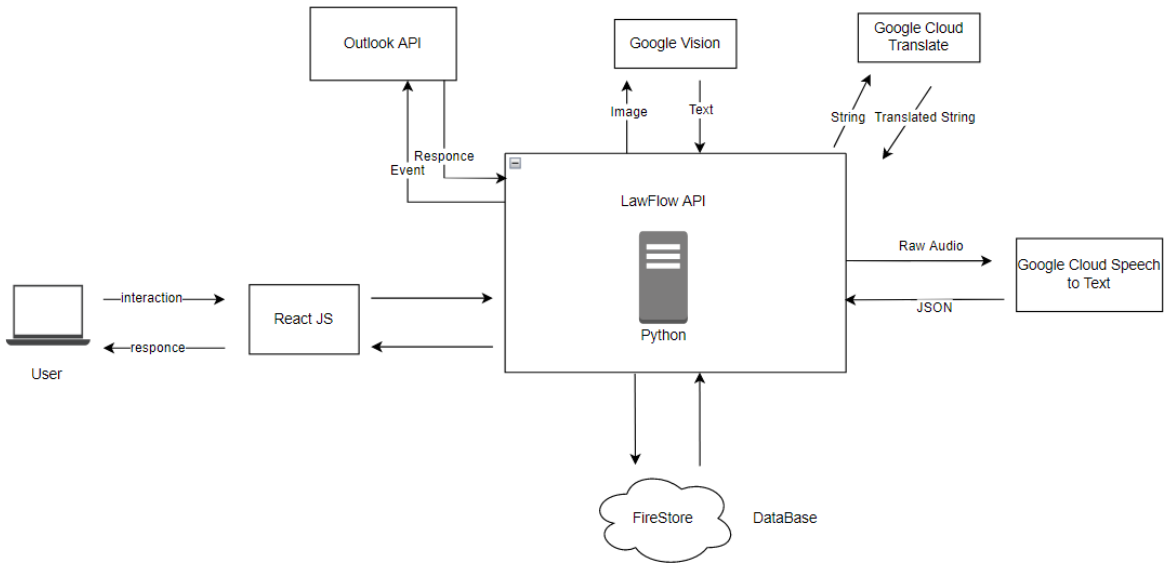


Figure 9: System Architecture

(Mulla, et al.)

The system architecture diagram displays the different uses of third-party APIs, as well as code which will be stored within the server. For speech to text, translation and OCR google cloud will be used, the data is provided in the required form to each of the APIs which are hosted by google and the response is recorded, this is done through the back end of the lawflow application. All the other code within the application is present on the server.

The API is then hosted on a cloud service, in this case google cloud. The front end sends requests to the API to obtain the relevant information. Finally, this information is viewed by the end user.

3.2 Architecture of the code within the server

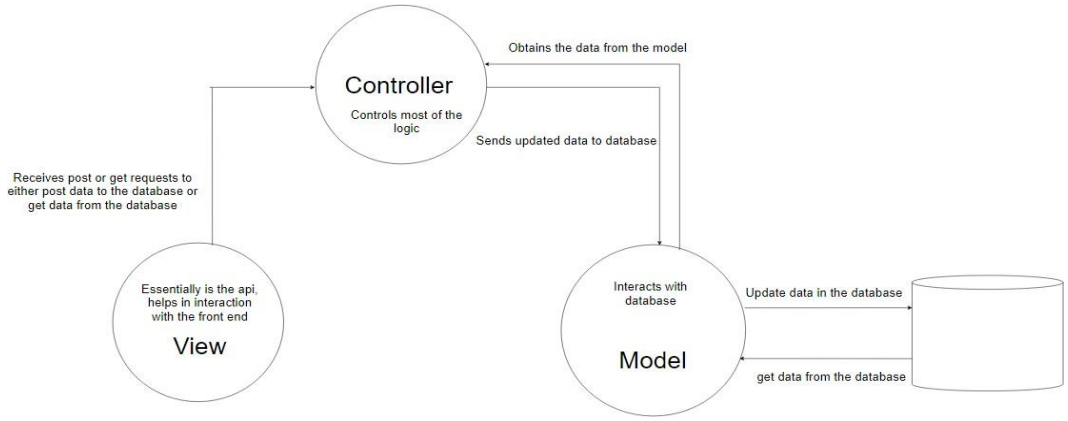


Figure 10:Architecture of code

All the code present within the server follows a Model, Routes, controller architecture. Each modules contains a model, view, and controller. The different modules are login, case structuring, contacts, OCR, translation, speech to text, encryption, appointments, to do list and workflow.

3.3 Network Architecture Diagram

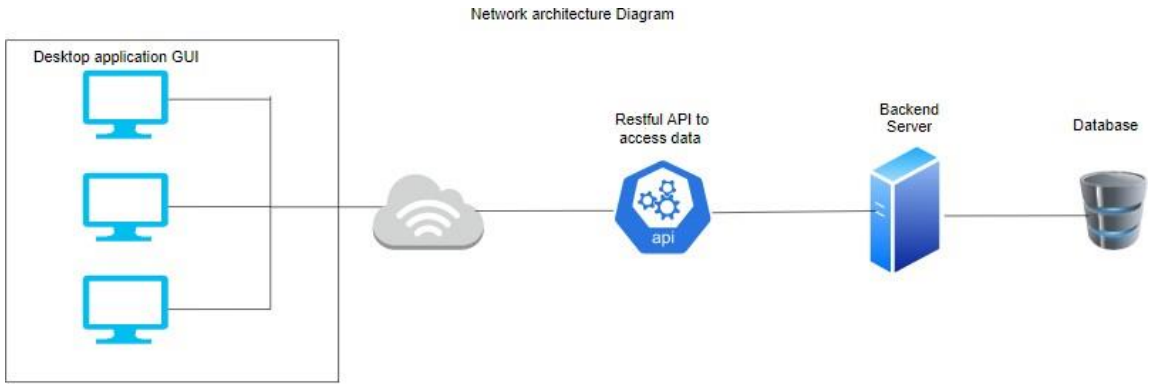


Figure 11:Network Architecture

The server will store all the logic required for the functionality to work properly. The database will have the data stored in JSON format. A RESTful API will be implemented to ensure ease in access of data from the server to the GUI. For the front end react node GUI will be used, API calls will be made to the API which will be deployed over an online hosting service (google cloud), the data depending upon the request will then be added or removed from the database.

4. High Level Design

4.1 Sequence Diagrams

Sign up

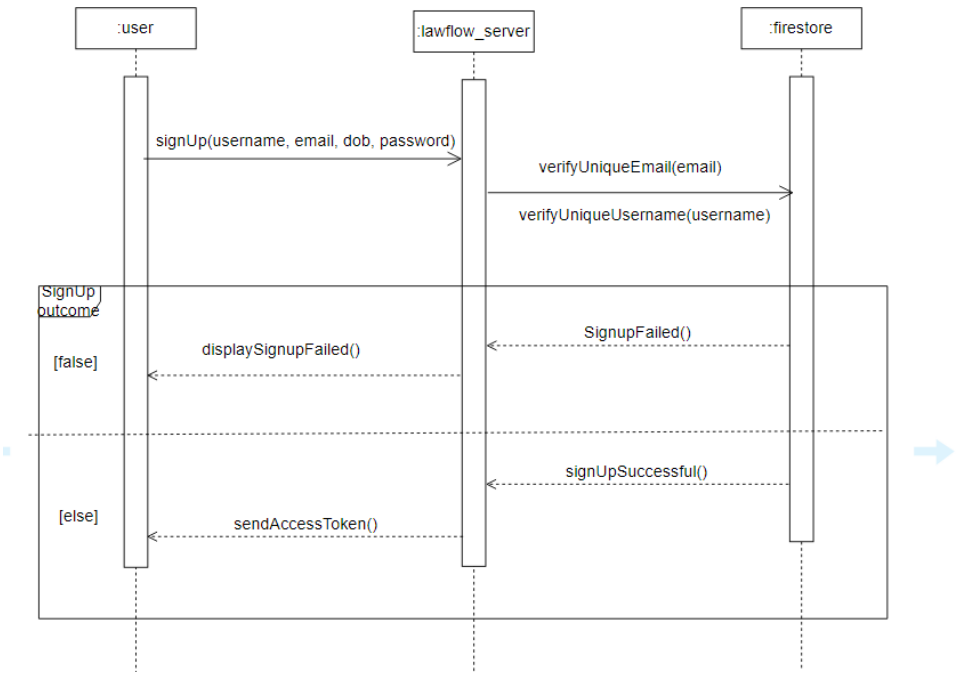


Figure 12:Sequence Diagram, Sign up

Explanation: User signs up using email, username date of birth and a password, the username is supposed to be unique, else the user is asked to re-enter the username till the username is unique. Once the user is accepted into the system a token is sent which includes the time to be logged in along with details like username and email.

Login

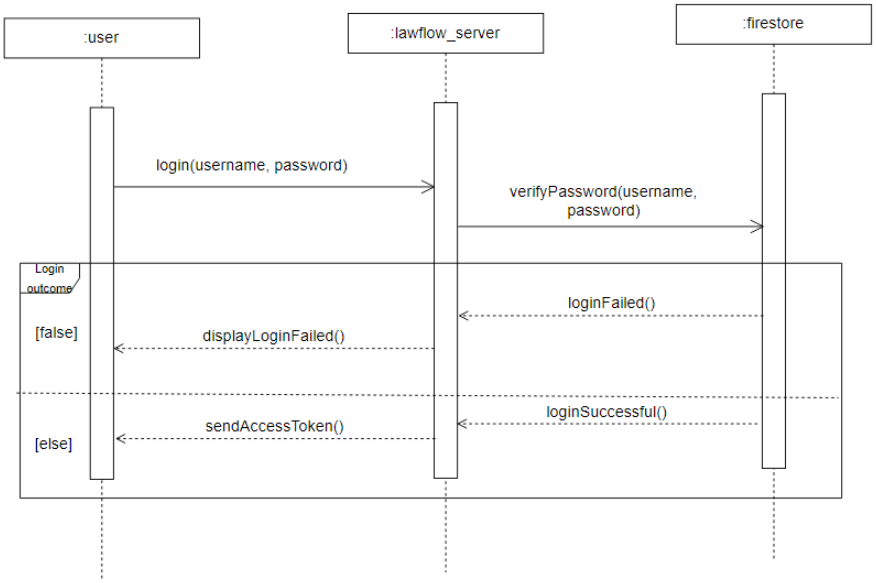
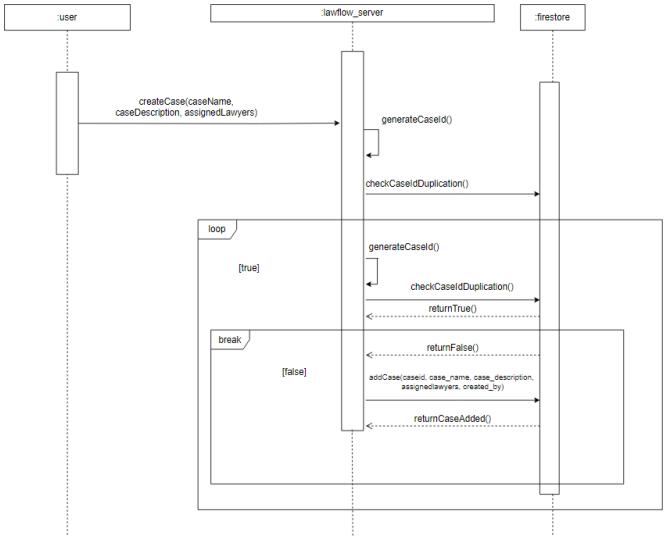


Figure 13:Sequence diagram login

Explanation: User logs in using email, and password. Once authenticated with the right password and an existent email within the database a token is sent which includes the time to be logged in along with details like username and email.

Adding a case



Explanation: User selects add a case option, the id with which the case is stored on the database is automated, the system verifies if the id is unique, if it is unique, the system proceeds to create a case within the database. The lawyer can assign other users onto a case. Doing so gives access to the other users assigned to the files present within the case.

To-do List add:

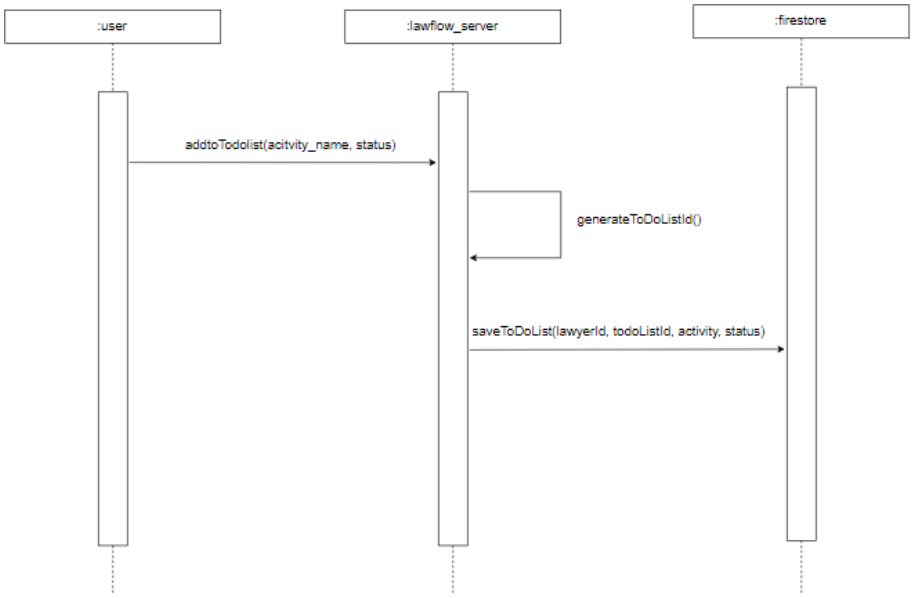


Figure 15:Sequence diagram, add to do list

Explanation: User adds a task to the to-do list along with the status which would be a drop down on the UI.

OCR

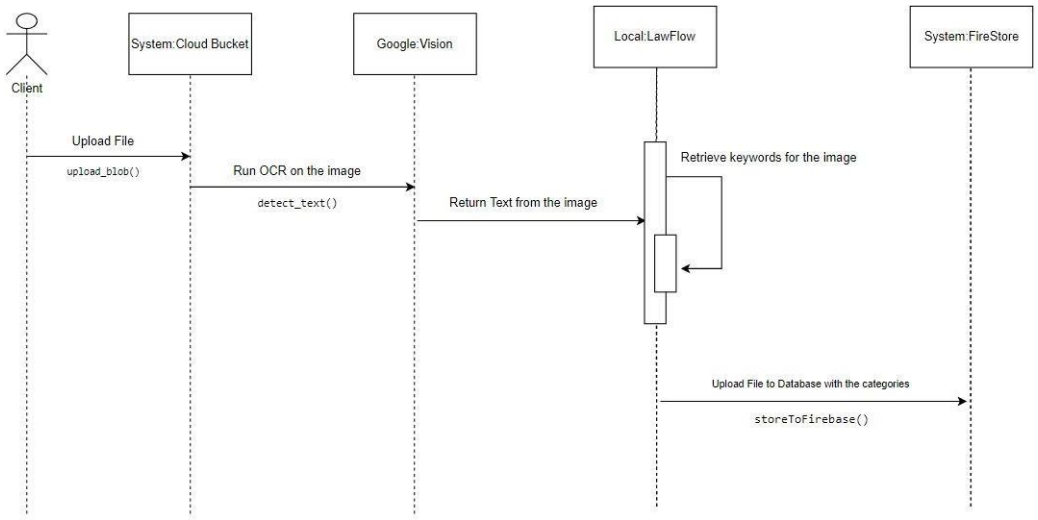


Figure 16: Sequence diagram OCR

Explanation: The user uploads an image/pdf which is then uploaded to the google bucket, the file is then taken from the bucket and read using google vision or pdf reader, key words are taken from the read file and used as categories while storing it in the Firestore database.

Translation

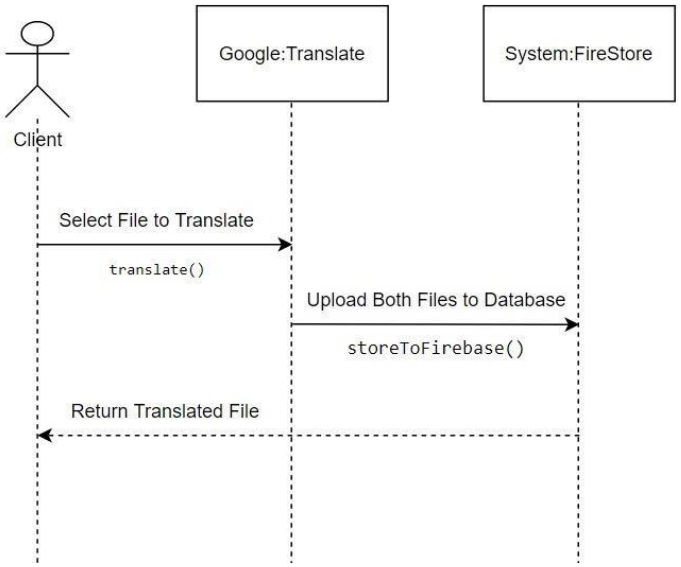


Figure 17: Sequence Diagram, Translation

Explanation: The user selects the file he wants to translate, the file if in any other language is then translated to English and the file is uploaded both to Firestore and google bucket.

Encryption

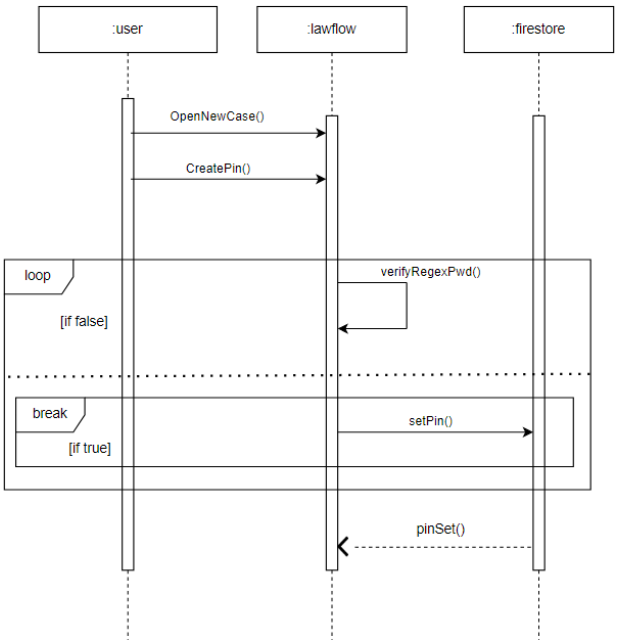


Figure 18: Sequence Diagram, Encryption

Explanation: User opens a new case and creates a pin for the case, this pin is verified using regex, if the pin satisfies the regex, all documents are encrypted with this pin.

4.2 Package Diagram

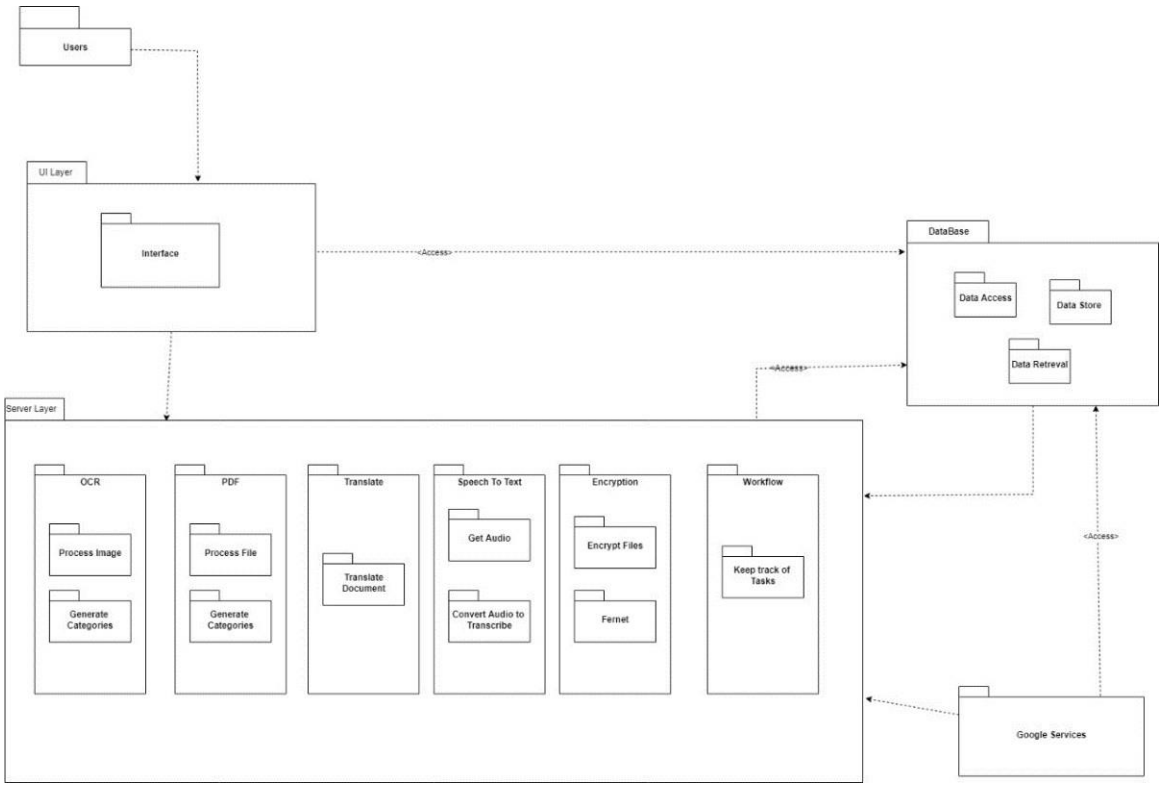


Figure 19:Package Diagram

4.3 Class Diagram

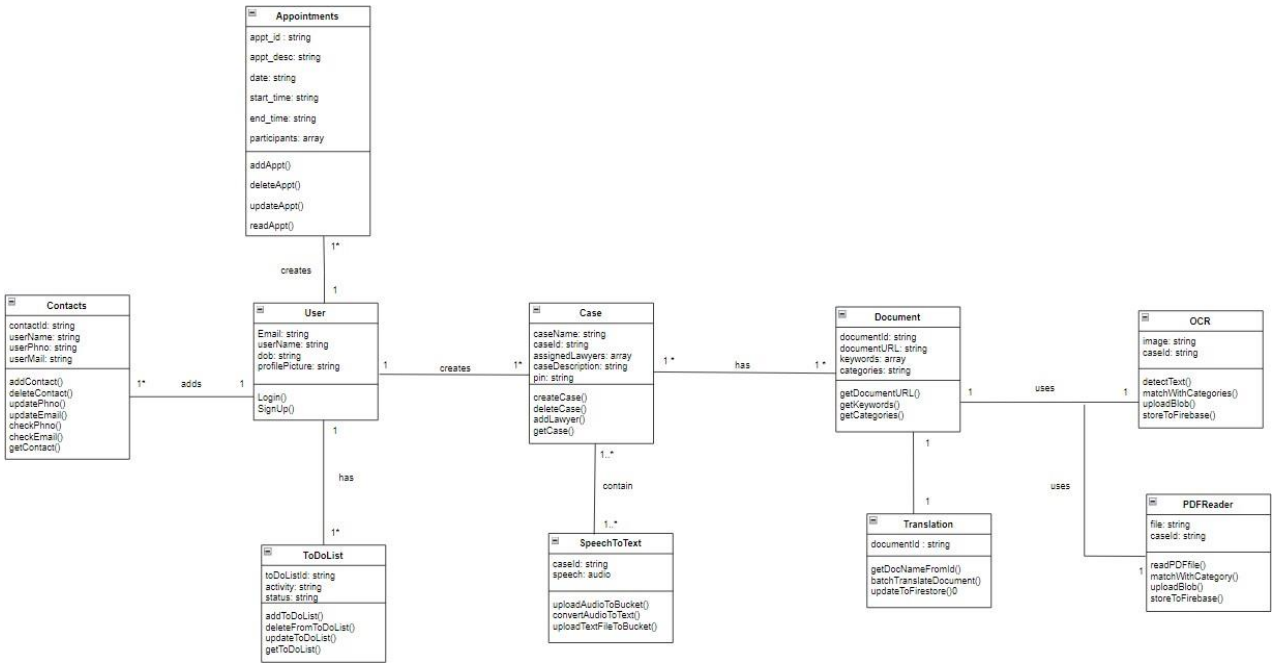


Figure 20:Class Diagram

4.4 Database Diagram

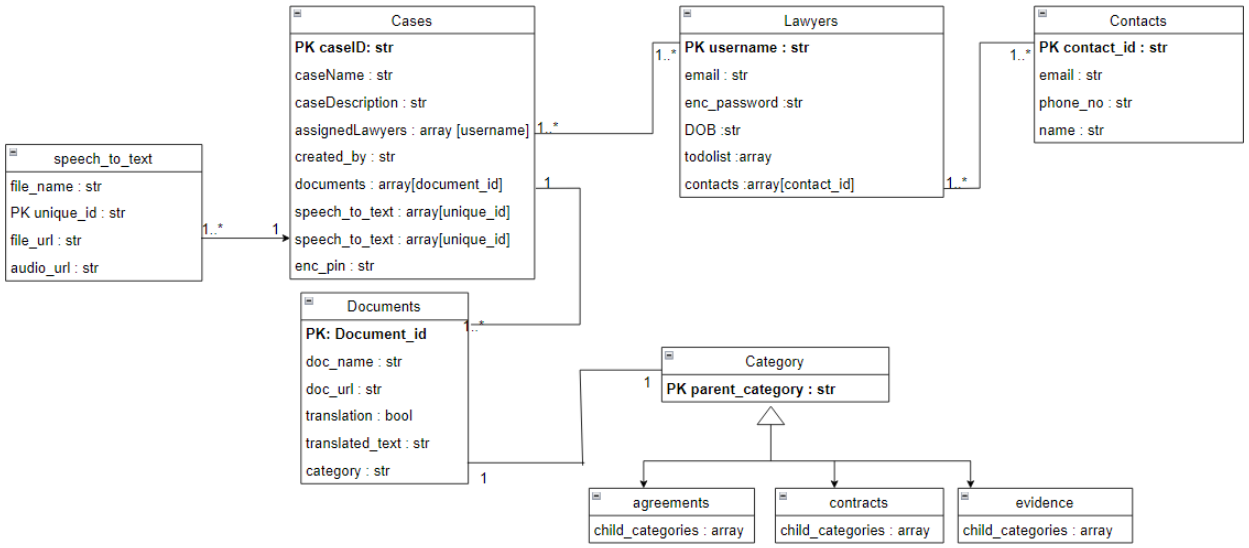


Figure 21:Database Diagram

4.5 Deployment Diagram

Deployment Diagrams are used to portray the implementation of a system. The diagram usually consists of various nodes that depict the physical components of the system, which help in visualizing the hardware and software of the system.

(Mulla, et al.)

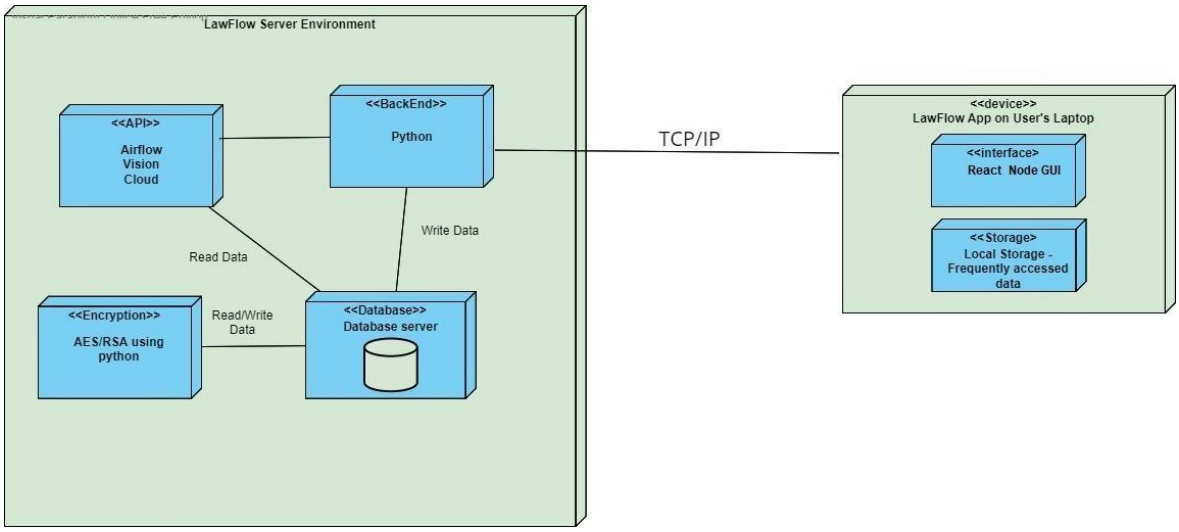


Figure 22:Deployment Diagram

The desktop application i.e., the front end will be designed using Electron, which is compatible with various Operating Systems and uses Node.js as a dependency. Furthermore, Electron makes use of JavaScript on the front-end, making it extremely powerful.

Python will be used to code the back end of LawFlow. Python provides various libraries and is compatible with all the APIs that are required to develop the features of LawFlow.

5. Low Level Design

5.1 Pseudocode:

OCR

```
OCR () {
Collect image from the user
Upload image to google bucket
Run image through google vision
Get main keywords from the image text
Upload document id and categories to firestore database
}
```

PDF Reader

```
PDFread () {
Collect file from the user
Upload File to google bucket
Read the uploaded file
Get main keywords from the image text
Upload document id and categories to firestore database
```

```
}
```

Translation

```
Translate () {
Get the document id that the user wants to translate
Translate the document using google cloud translate
Upload translated document id to firestore database
}
```

Login

```
Login () {
User enters password and email
Check if the email and password match using the firestore
database
If authenticated:
A token is sent back which contains the email address,
username and how long the user can remain logged in
}
```

Signup

```
Signup () {
User enters the email password dob and username
Checks if username already exists in the firestore
If the username already exists, user is prompted to enter a
new username
Once the user has signed up, a token is sent back
}
```

Document Encryption

```
documentEncryption () {
An encryption key is generated when the user enters their own
PIN.
Ask user to enter PIN for the caseID the user wants to encrypt
document for.
After all uploaded documents pass through OCR, retrieve all
documents from the bucket that are related to the caseID.
Create a temporary folder in a specific directory and download
blobs into it.
Encrypt all documents in the temporary folder.
Delete original blobs from the bucket.
Upload the encrypted blob with the same ID of the original
blob.
Delete all documents in the temporary folder.
}
```

Document Decryption

```
documentDecryption () {
Click on the CaseID and enter the PIN.
Read all documents in the directory, and decrypt clicked
document on the frontend for view.
}
```

Contacts

```
Contacts () {
User enters the name, phone number, email, lawyer username
User selects add contact and contact ID is generated
Checks if contact already exists in the firestore
If contact does not exist, then add contact
User deletes contact by passing contact id and username
Use updates phone number/email by passing the new values
Search feature to check if a phone number/email/name exists.
}
```

5.2 Wireframes

Home page:

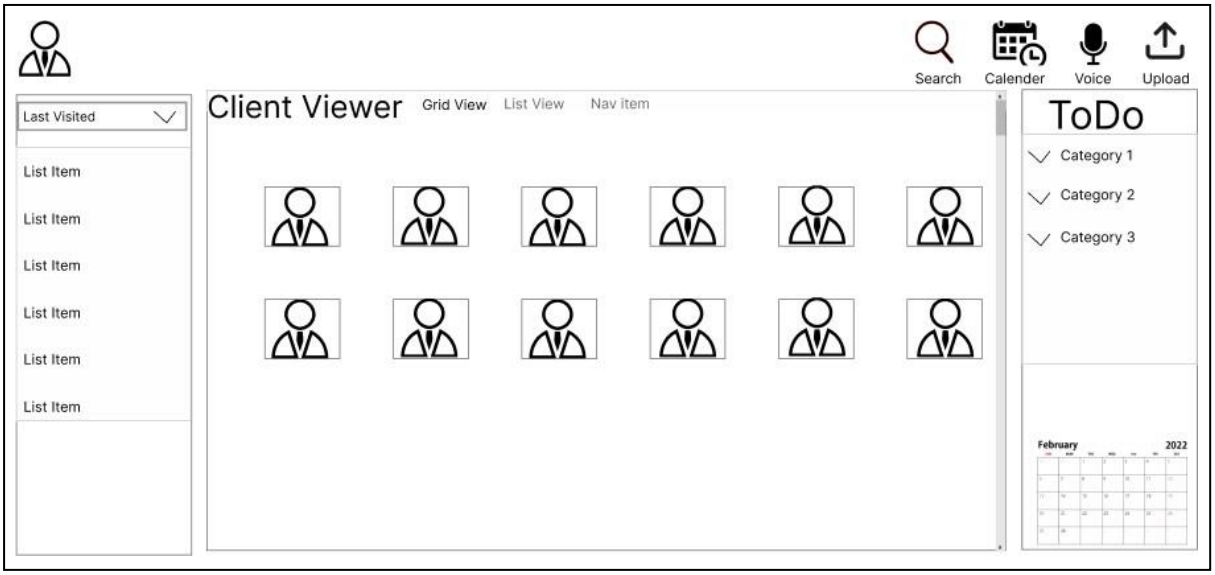


Figure 23:Wireframe homepage

Client Page:

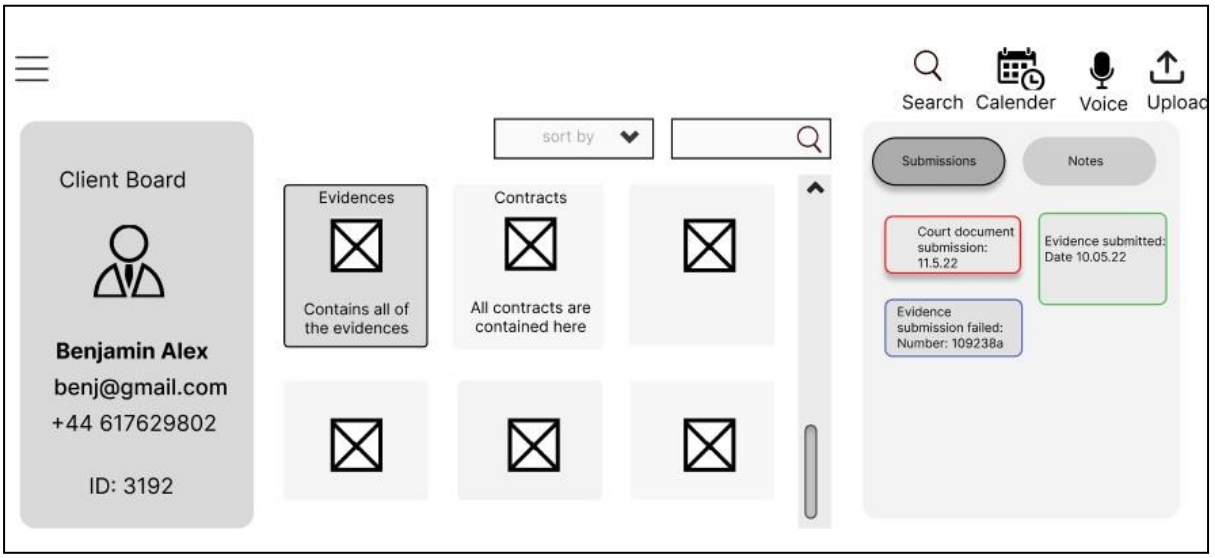


Figure 24:Wireframe Client page

Search Page:

Figure 25:Wireframe Search Page

(Mulla, et al.)

Document Translation:

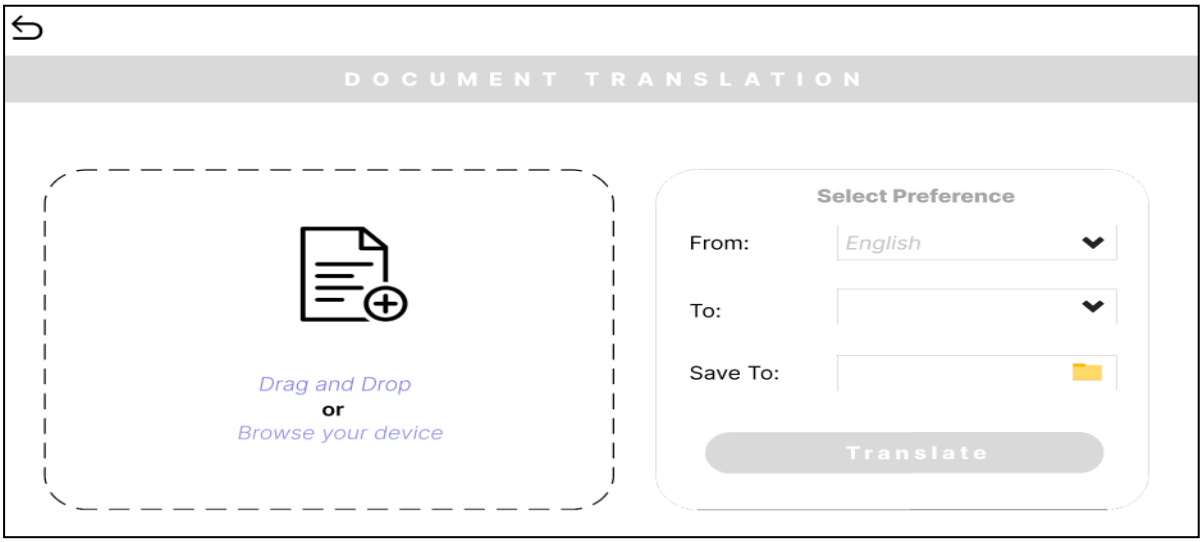


Figure 26:Wireframe Document translate

Scheduler:

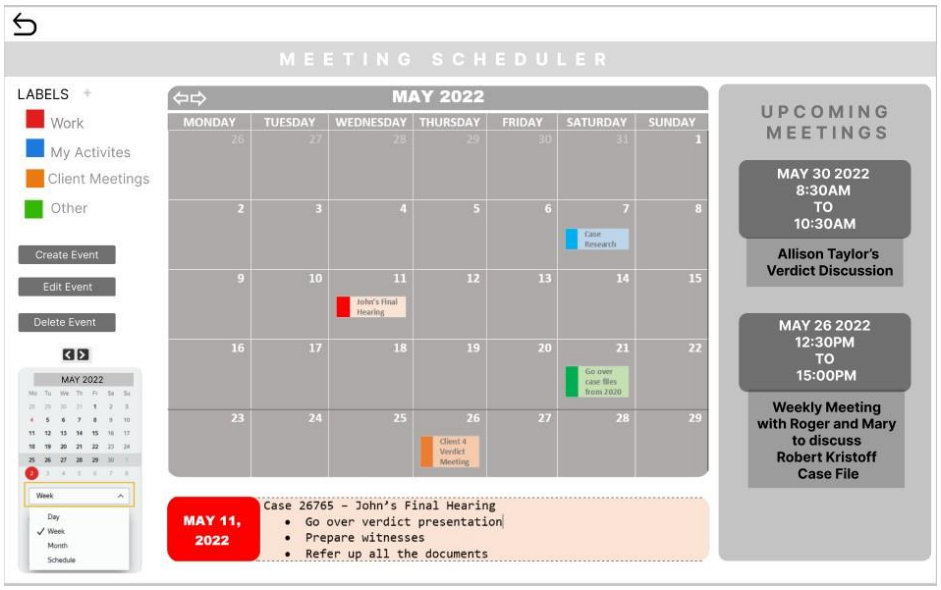


Figure 27:Wireframe Scheduler

5.3 Common Features

Home Screen

Each user starts by viewing the lawflow splash screen. After which the user is directed to go to either the login or sign up. The user is then taken to the home page, which consists mainly of the dashboard which in turn has the favourited items, the to do list and the recently viewed files.

Side Bar

Each screen has a side bar which is fixed, this side bar consists of various options that the user can use.

Top Bar

The top bar with the tag “Hello [name of the user]” along with an options profile picture and a logout button is common to all the screens, this is done so as to know that the user is logged in correctly and does not have to switch between different screens if he/she decides to log out.

Bread Crumb Navigation

The type of navigation being used within the desktop application is breadcrumb navigation. It is essential that the user at each point knows which page he/she is on and what the path of the page is. This makes it easier for the user to work with the application as well as ensures that the user is not lost as to where they are within the application at any point.

5.4 Design Principles

Visibility

Each button looks like its clickable, each item is labelled properly so as to ensure that the user knows what each button/ function of the application does.

Feedback

The user receives feedback after each function, for instance on clicking a button the user would receive feedback indicating if the task performed was successful or not.

Mapping to real world

Most of the icons used within the application are like icons used in other apps, this makes it easier for the user to adjust to the application, the user is also aware what each icon does. The application also looks and feels like most other desktop applications. The language used within the application will also be like languages in other applications.

Recognition rather than recall

The system will make it easy for the users to rely on recognition rather than having users remember the functions and where each function lies within the application. All the different options will always be visible to the user from the dashboard which is present persistently on each screen.

Aesthetic and minimalist design

The application is designed keeping in mind the visibility of each function. The design is kept simple to ensure the users find it easy to view everything. The design avoids the use of unnecessary graphics, animations, or high colour contrasts.

Consistency and standards

The colours used on all the pages and screen are of the same hex numbers. The fonts used within the entire application are similar depending on where it is used, for instance all the headings are made bold and of a larger font size as compared to the

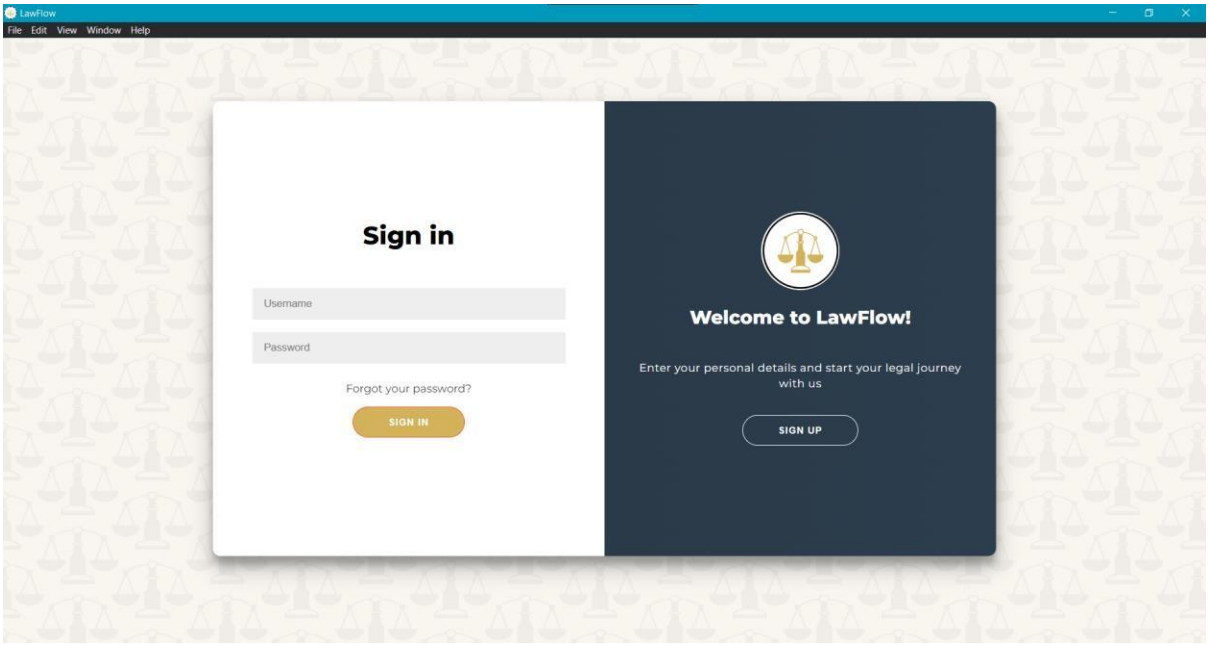
content font. Error messages on all screens will use the same language, furthermore if there are similar icons on different pages, they all map to the same function.

Help and documentation

Help and documentation will always be available to users in case they need guidance on a certain topic, there will be a collection of FAQs to make it easier for the user to ask/ search for help.

6. Screens of the Application

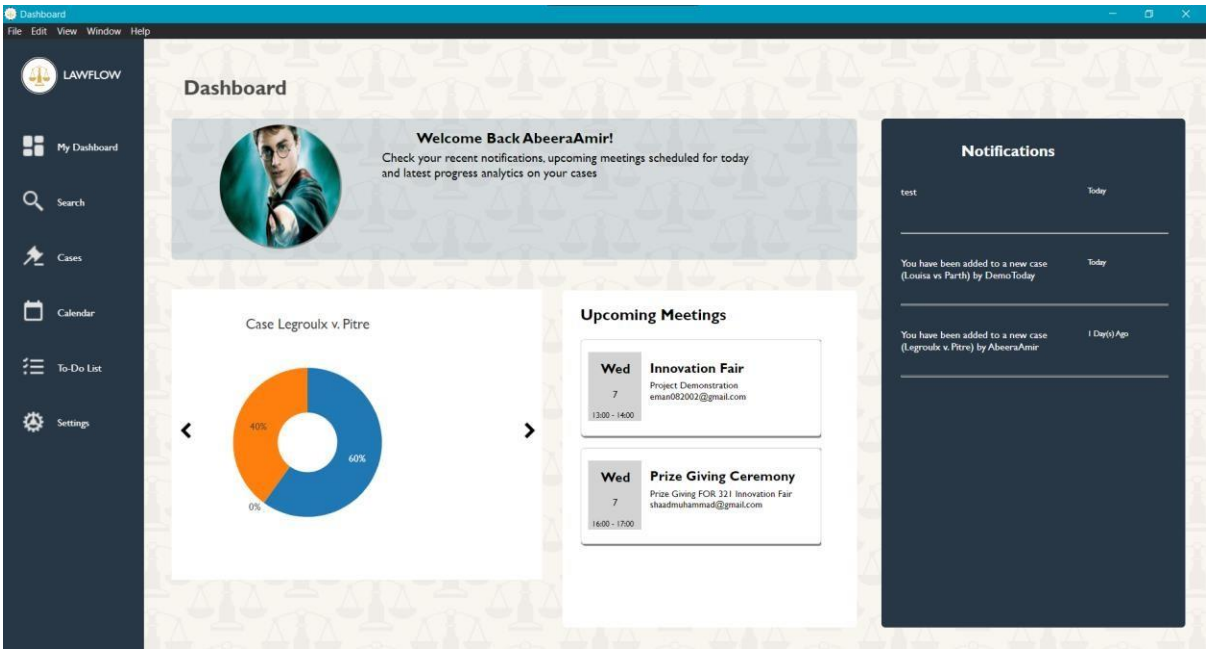
Log in:



For log in the user needs to enter the username and password, the username and password is authenticated, then the user is allowed to go to request the OTP, this is done for security purposes.

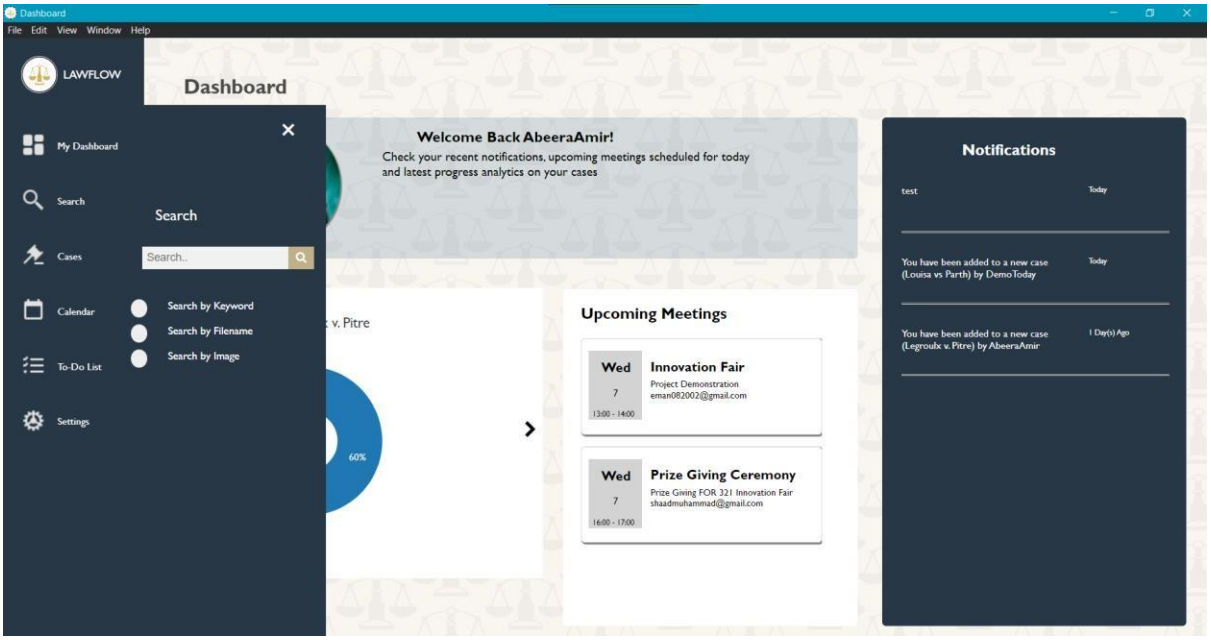
(Mulla, et al.)

Dashboard:



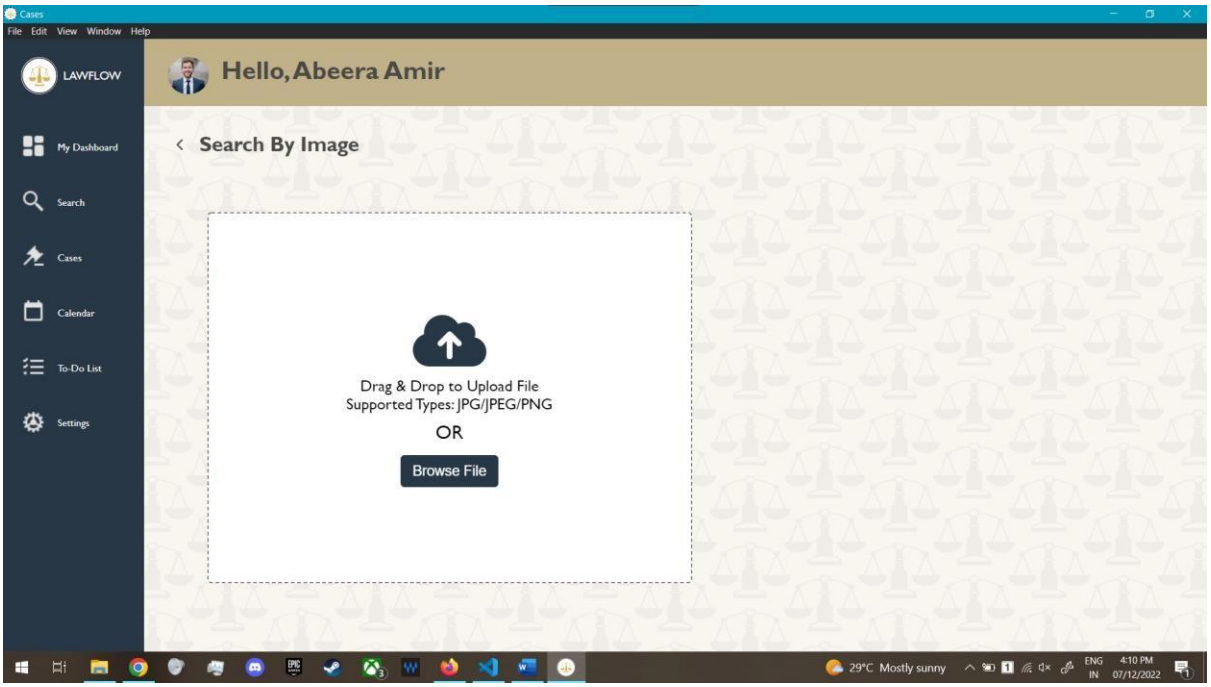
The dashboard consists of all the main items the user has, the favourites, notes created by the user and upcoming meetings. On the side a consistent side bar is always present.

Search:

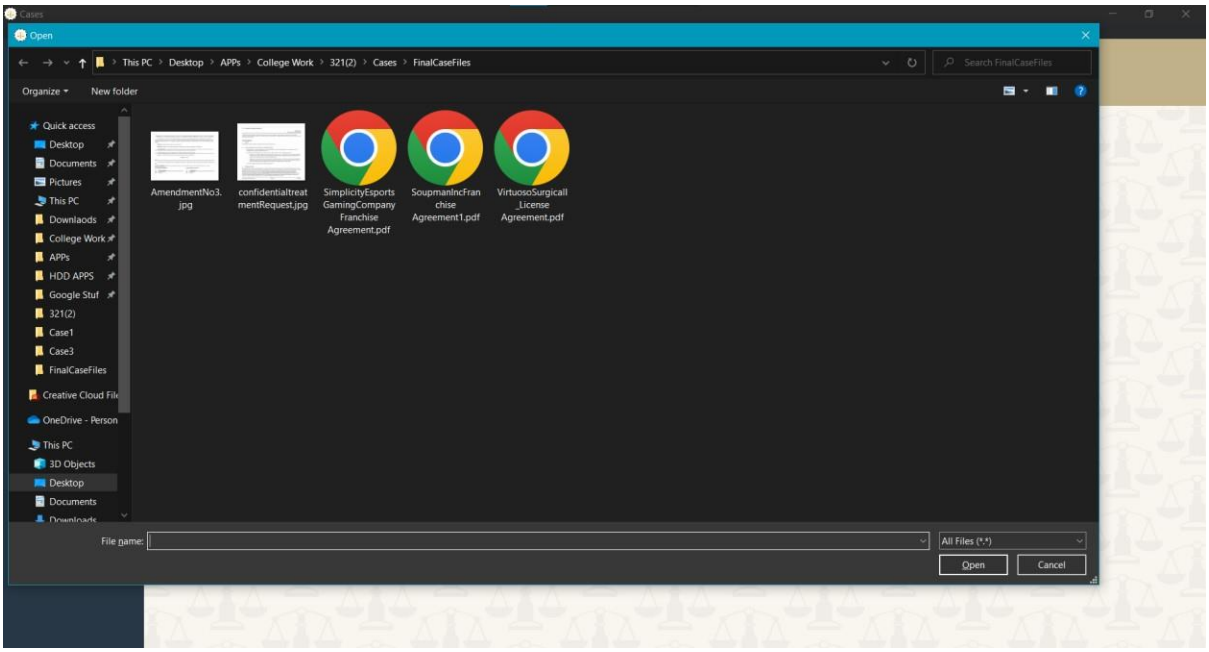


On clicking search, three options appear, search by image, search by filename or texts present within the file and search by contacts.

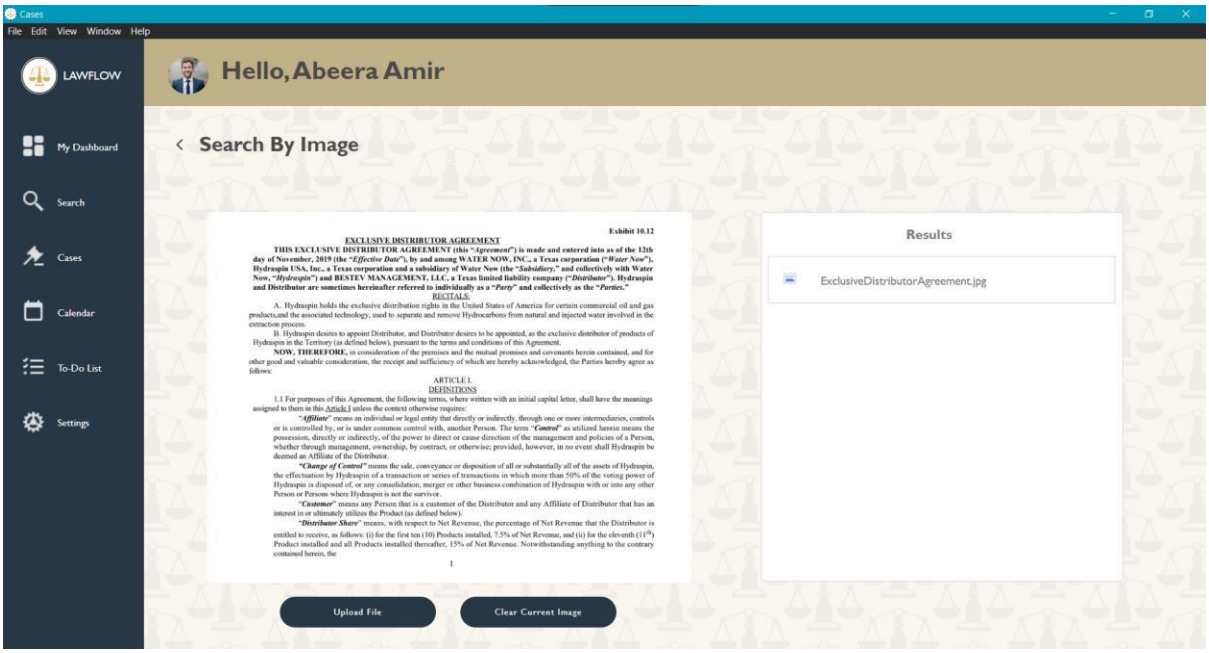
Search – Search by Image:



Search – Search by image:

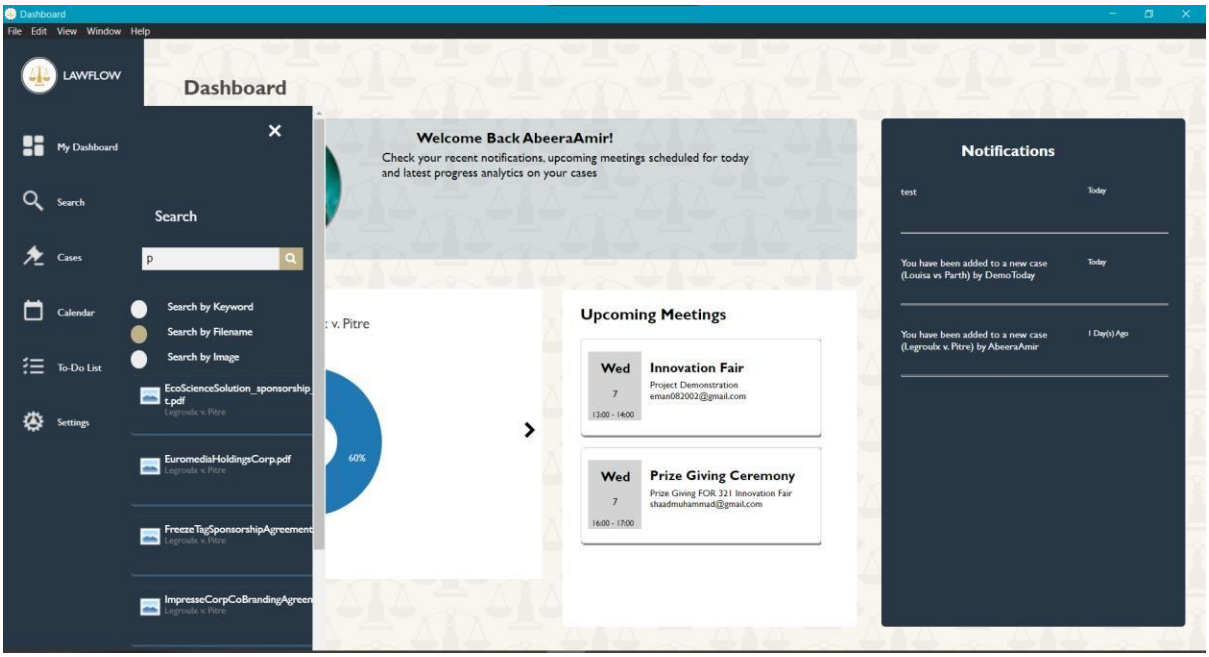


Search – search by image:



For search by image, the user uploads or drags and drops an image first into the drop box or browses an image, the image is scanned and the documents which identify the closest to the uploaded image are displayed under the field “Closest Matched documents”. At this point the user can choose to either view the document, go back, or upload another image.

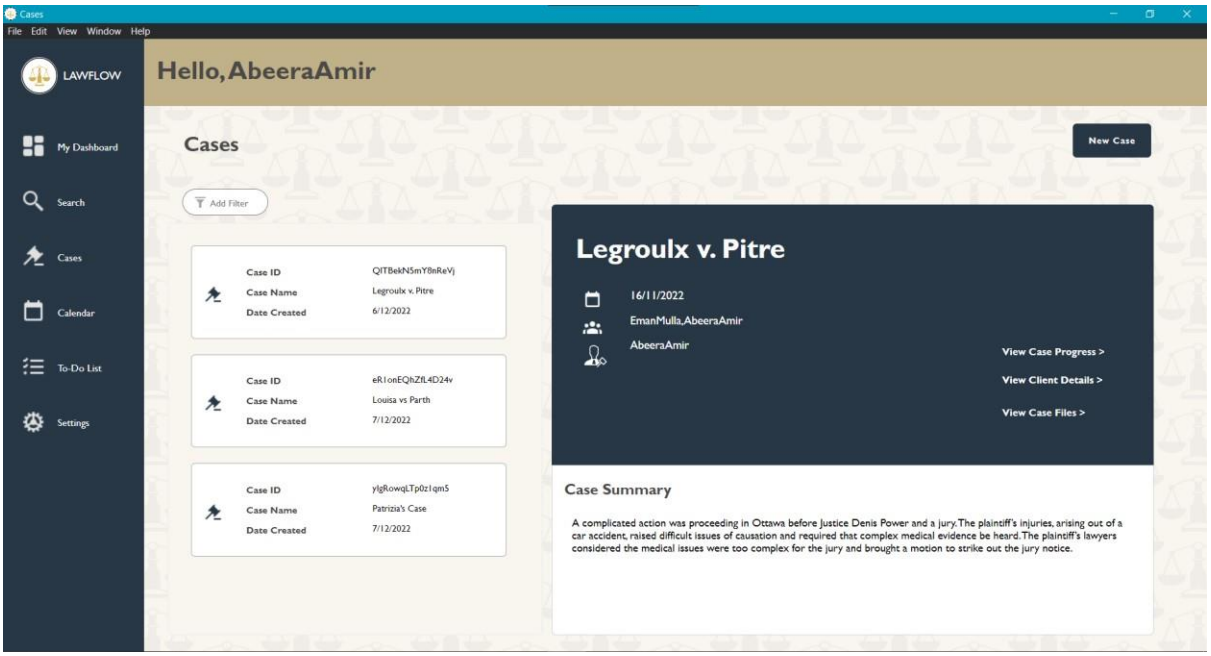
Search by tag/ file name:



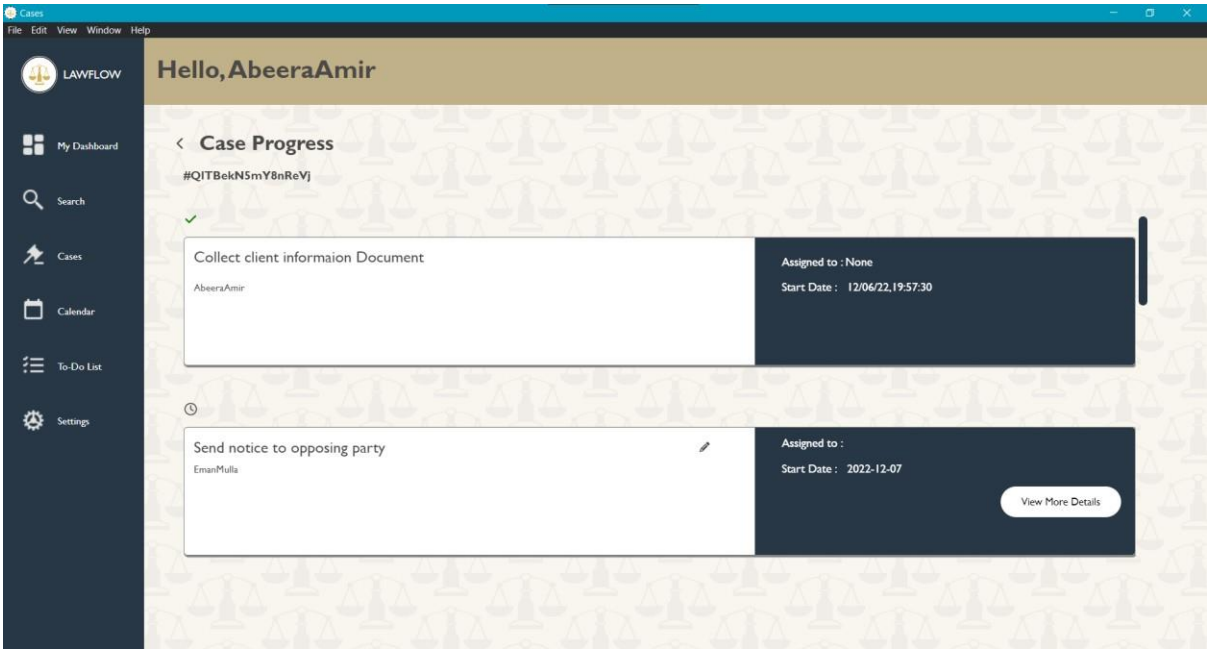
To search by a file name, keyword the user enters the keyword in the search bar, any file or directory similar to the name is displayed in a scrollable list.

(Mulla, et al.)

Cases:



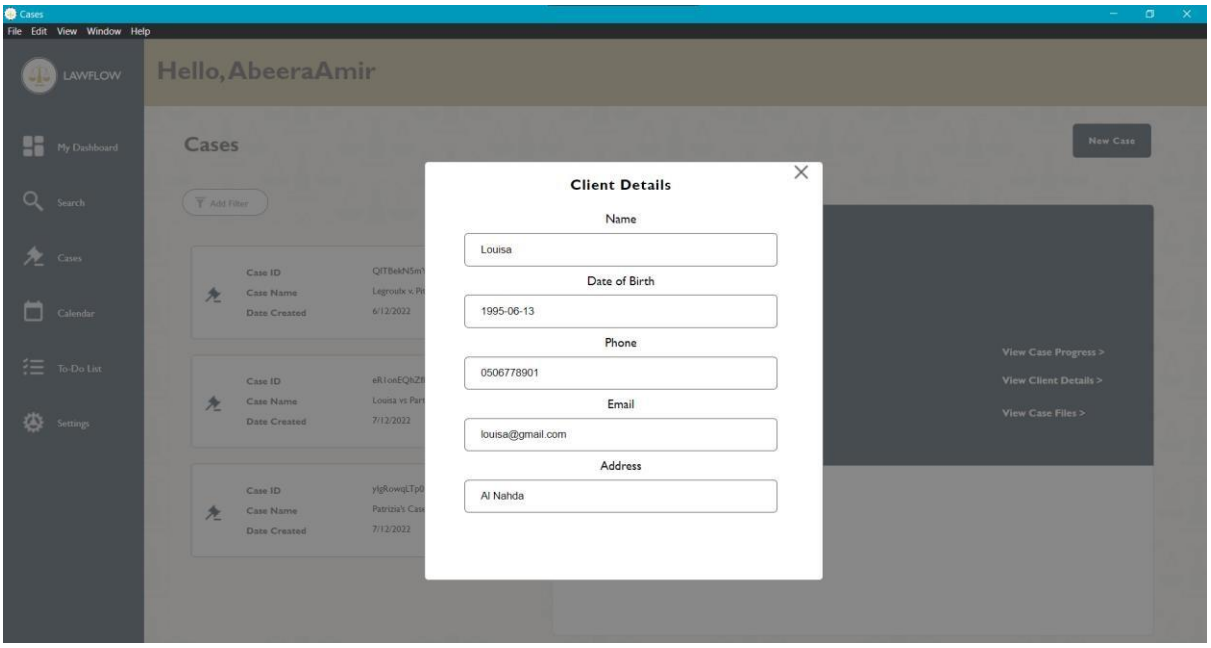
Case 2:



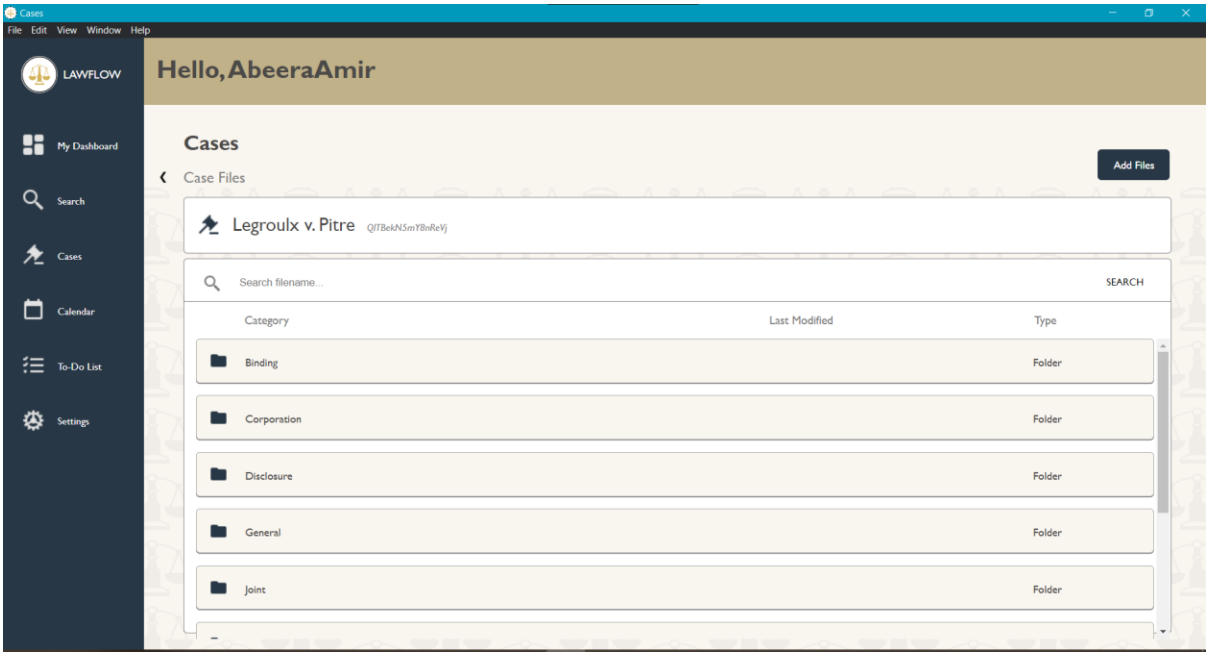
On selecting cases from the side bar, the user can view all the cases that he/she has access to in a scrollable list along with some basic information which acts like an identifier for the case. Once the user clicks on the particular case to be selected

(Mulla, et al.)

Case 3:

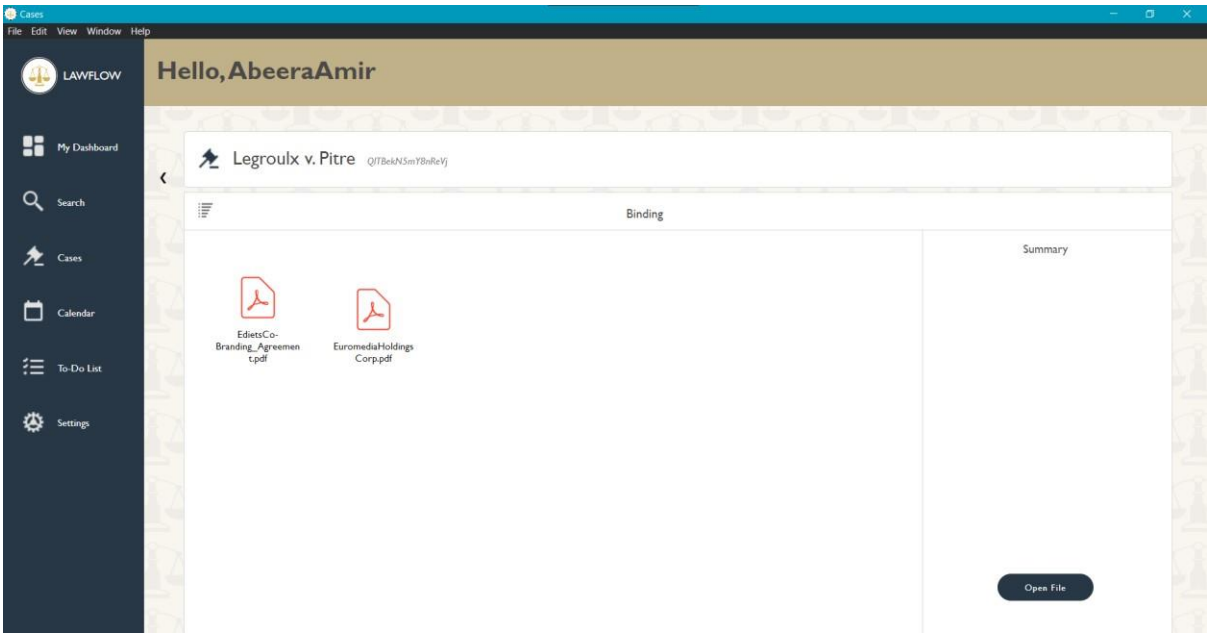


Cases 4:



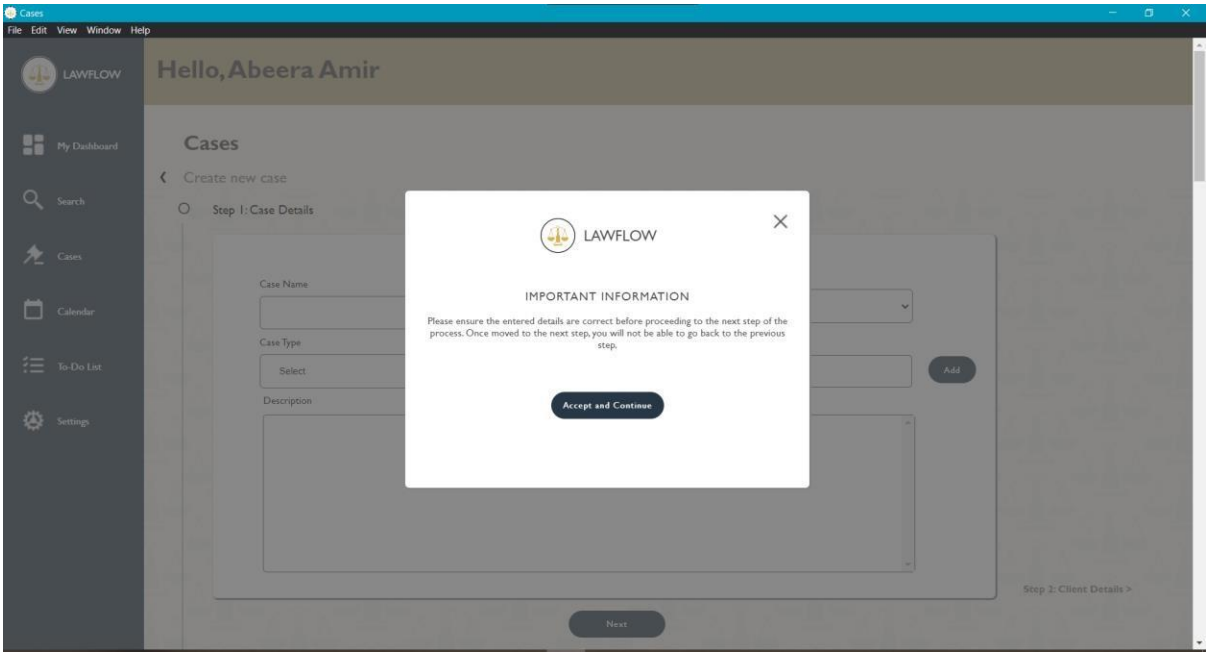
(Mulla, et al.)

Cases 5:

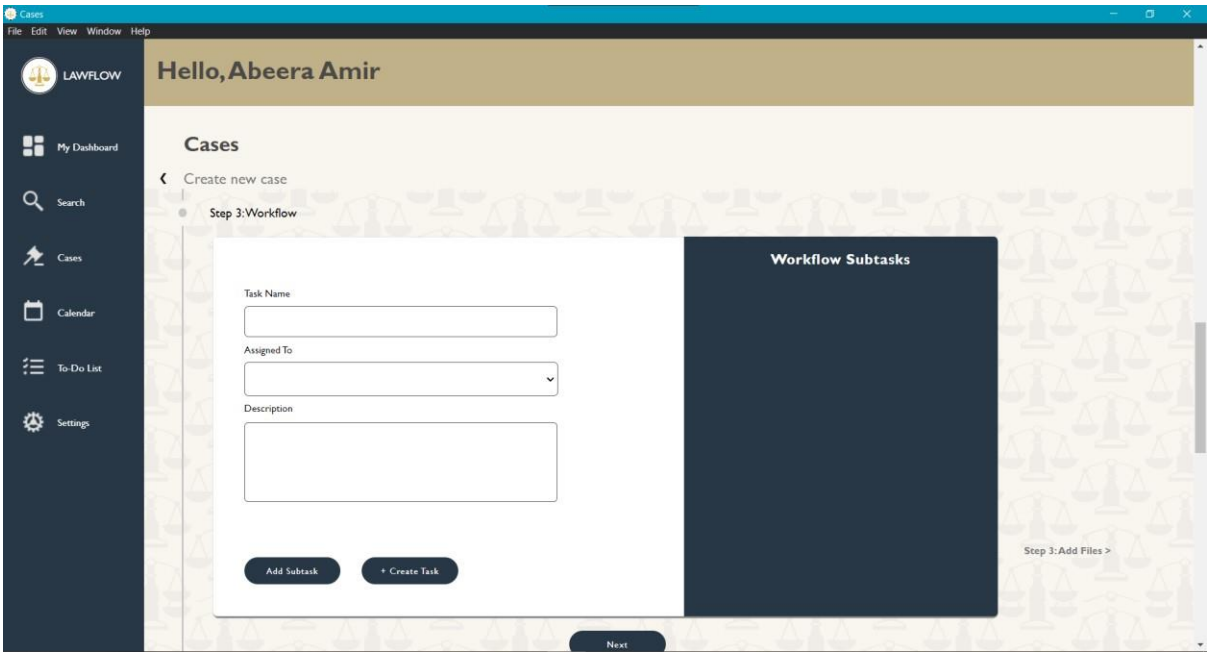


On opening a particular case, all directories present within the case are presented to us, user can then go within sub directories and access different files.

Creating a new case:

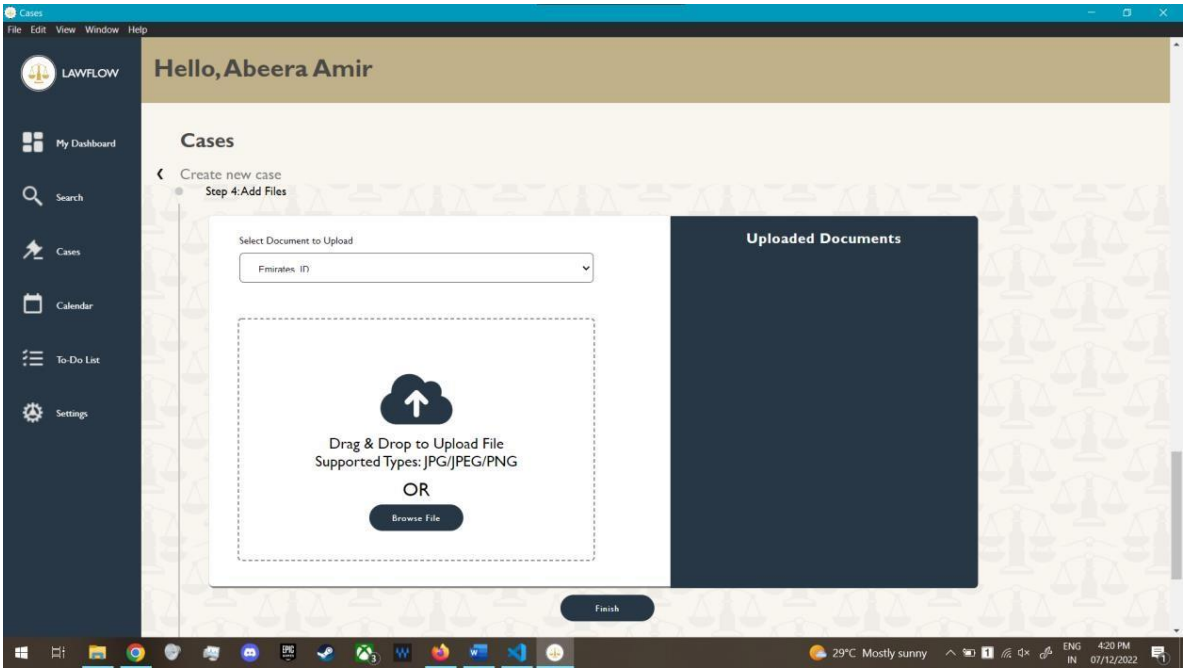


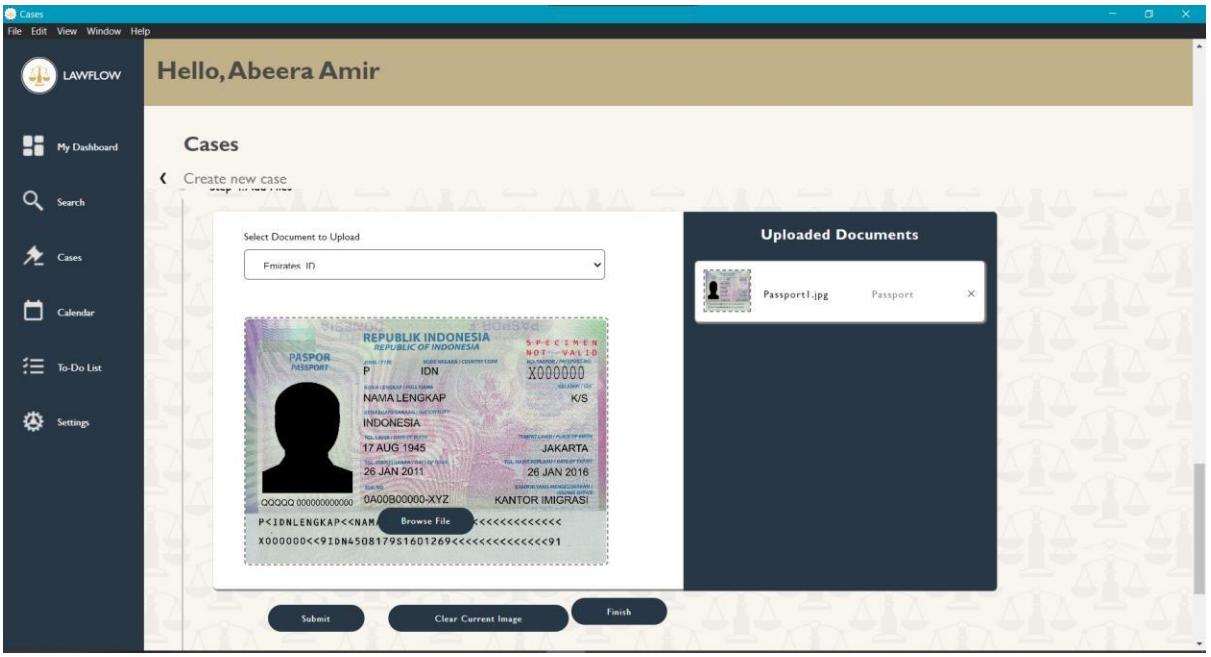
(Mulla, et al.)



On creating a new case a user is asked to assign a case name, lawyers, case description, and a pin for security. Initially as the case is opened no files are present so the screen provides a prompt saying, no files added to this case.

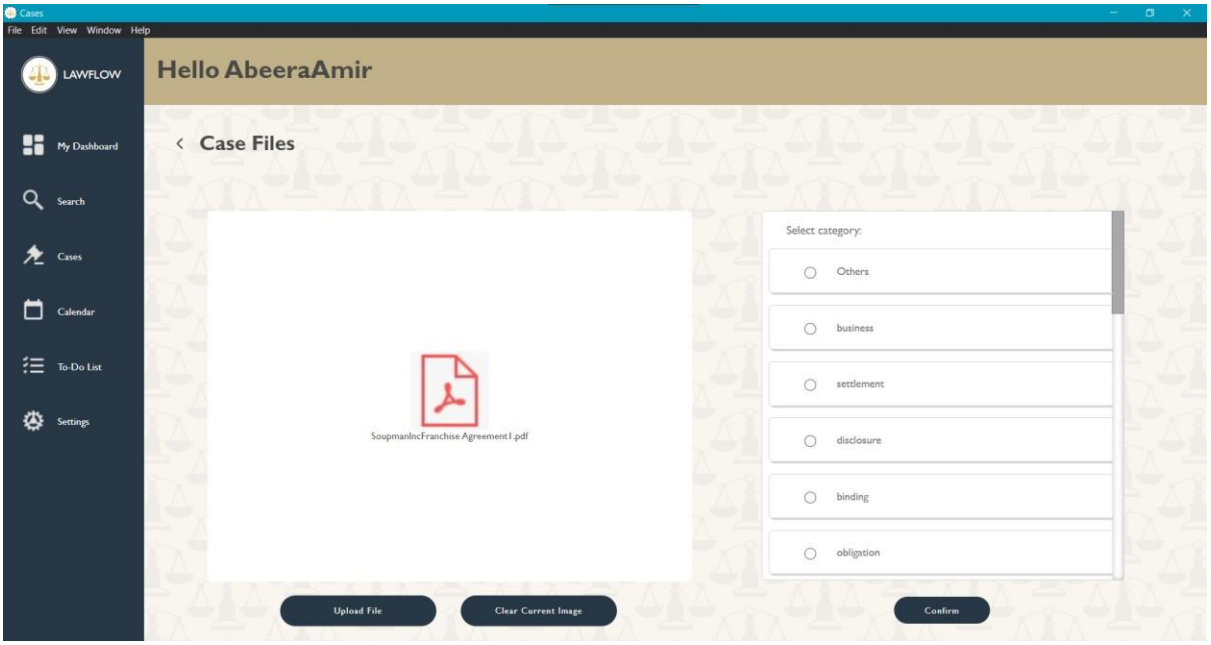
Adding a file to the new case:

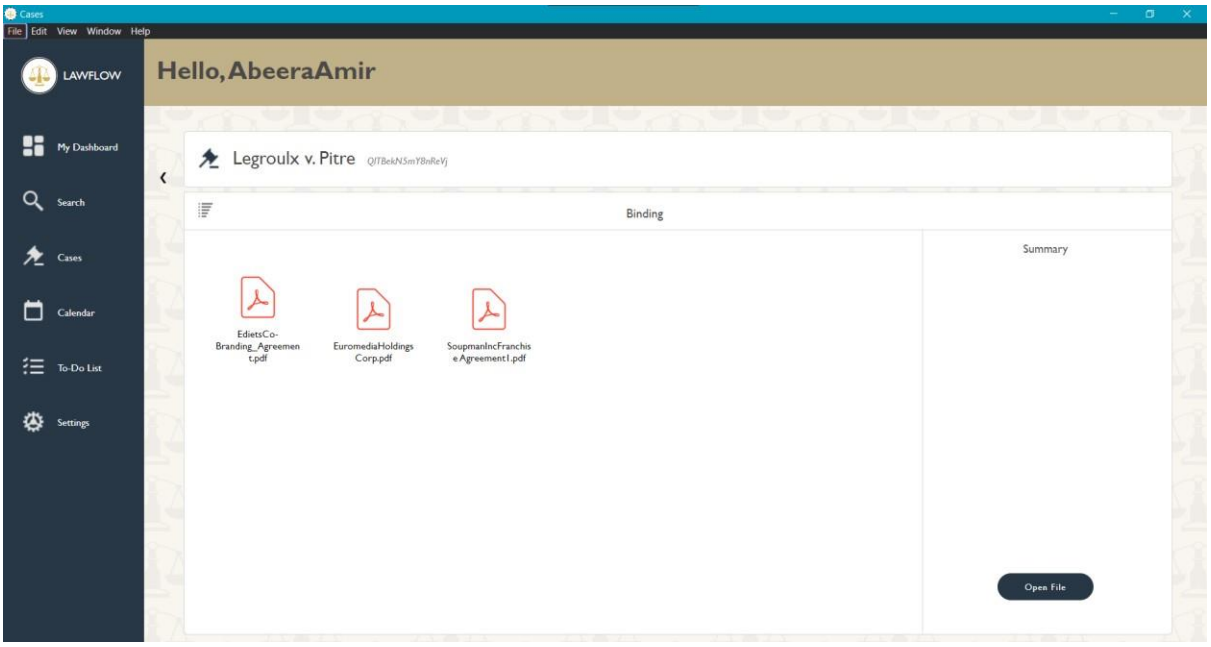




User can choose to either pick and drop a file or browse through the different files present within the system.

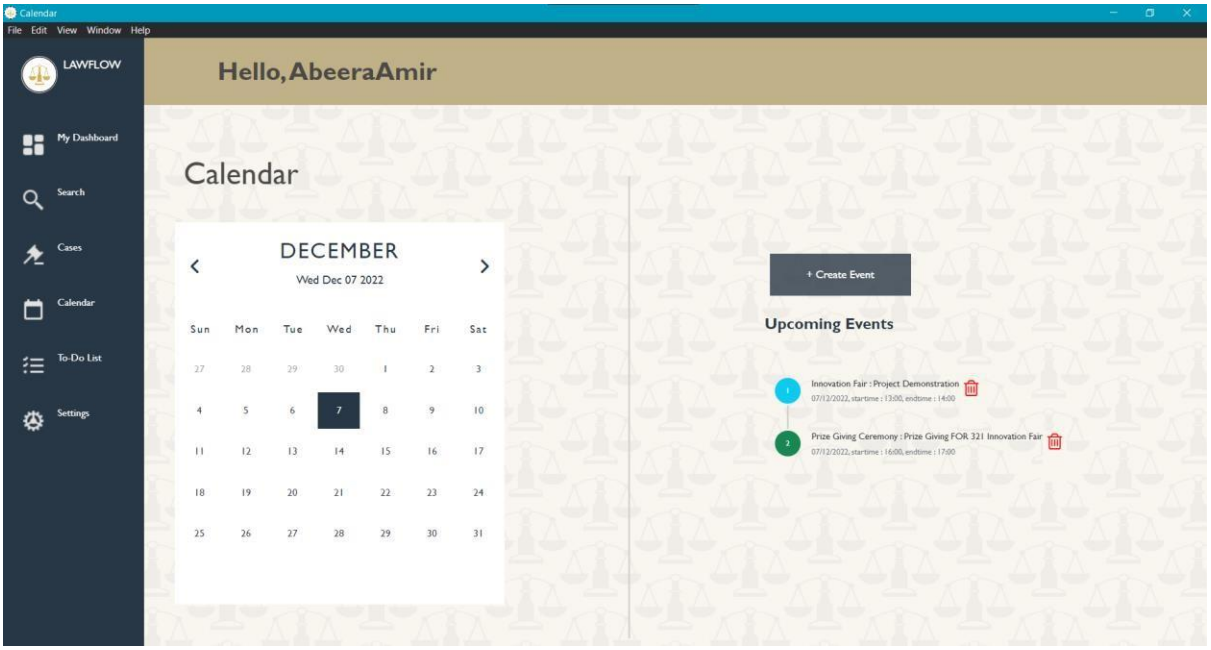
Selecting Category identified from OCR:



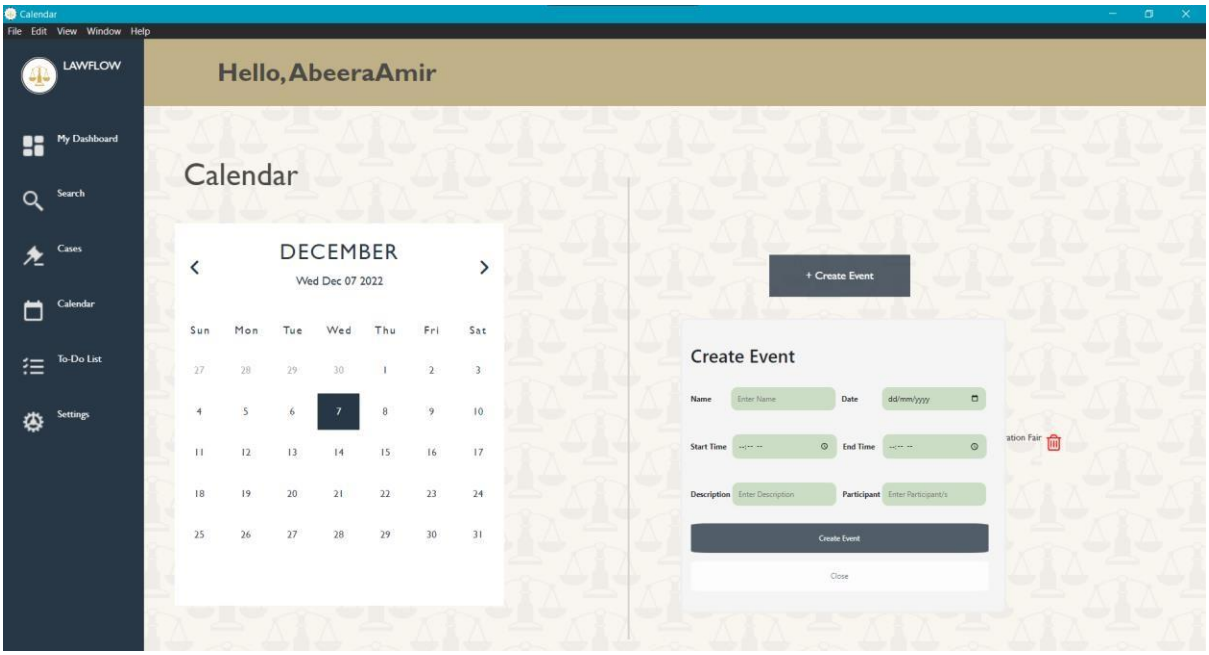


The OCR identifies if the document aligns with commonly used directories within the law world. However, it always returns an "Other" folder giving user the freedom to choose where to store the documents. Here the user chooses others and gives a name as he prefers.

Calendar:

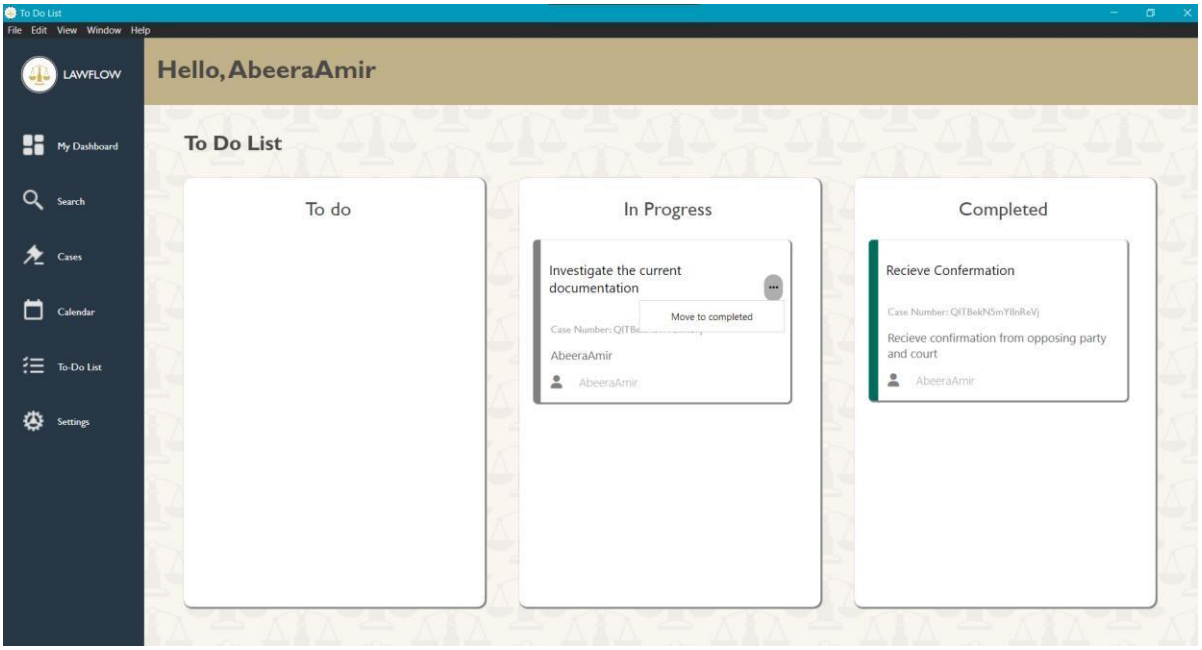


Calendar – Create Event:



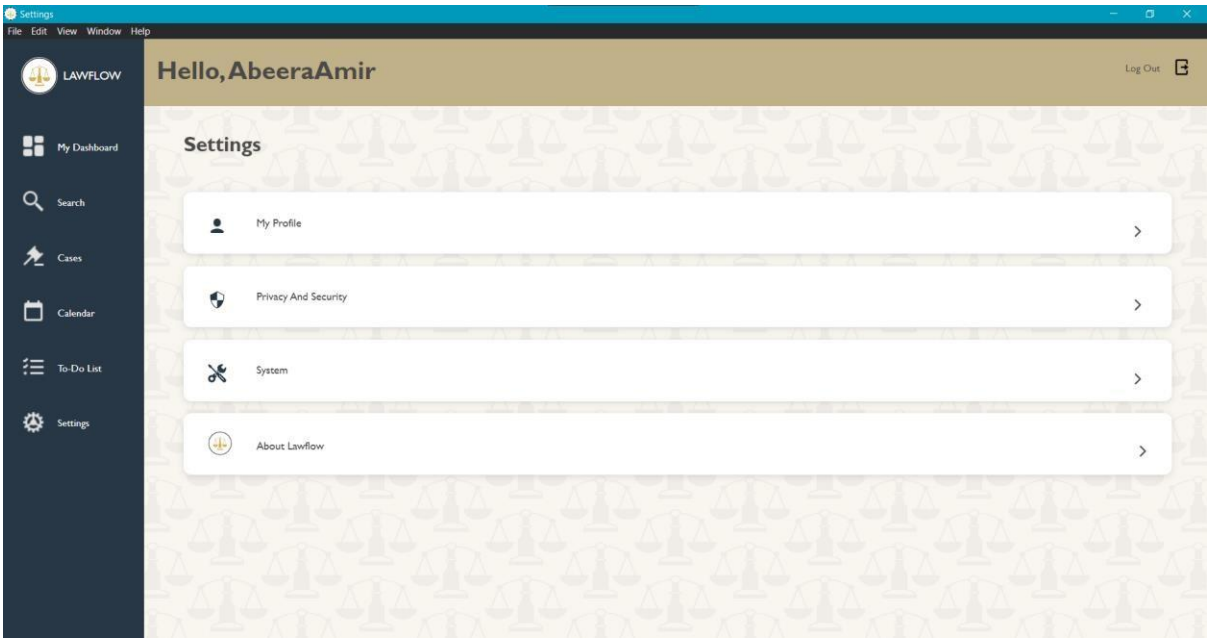
User can access the appointments and can access the appointments from the side bar, user can add, delete, and edit an appointment.

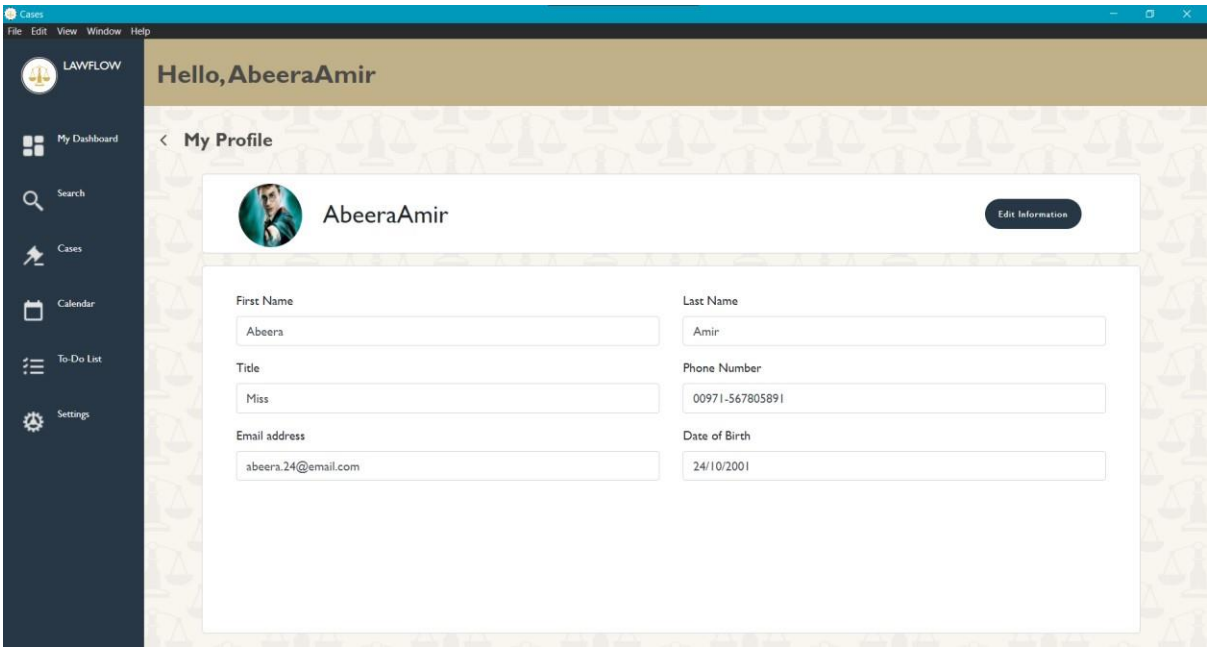
To-do List



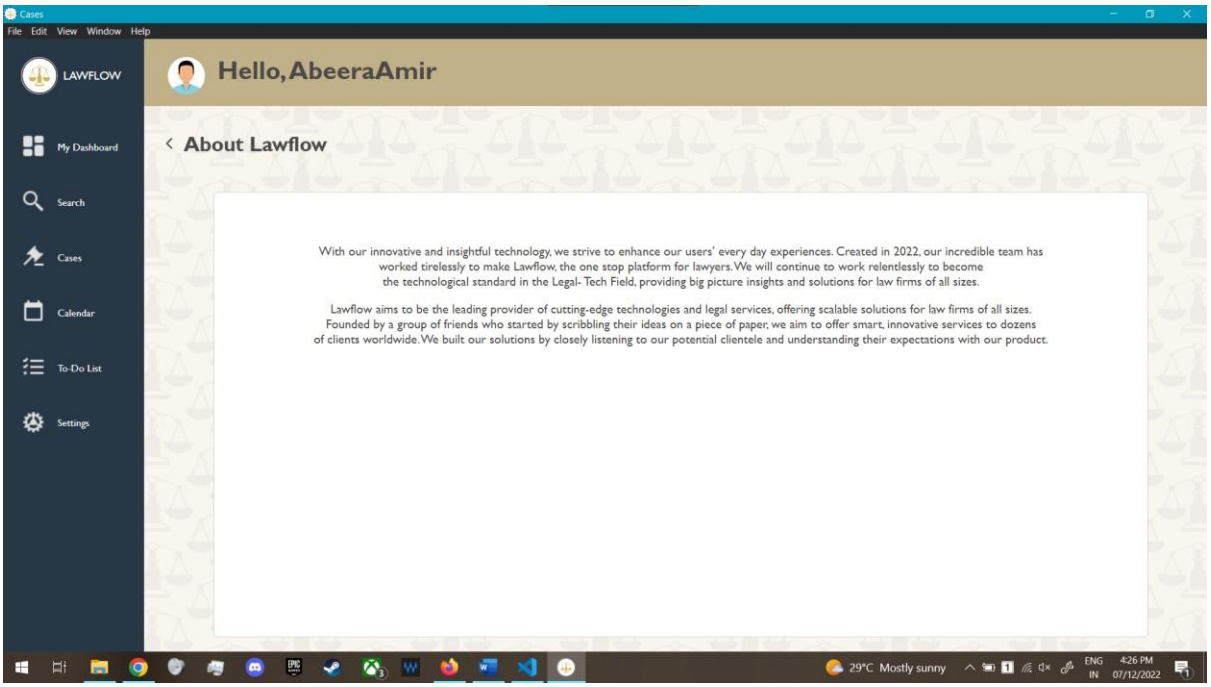
(Mulla, et al.)

Settings:





(Mulla, et al.)


Cases
File Edit View Window Help
LAWFLOW
Hello, AbeeraAmir
My Dashboard
Search
Cases
Calendar
To-Do List
Settings
About Lawflow
With our innovative and insightful technology, we strive to enhance our users' every day experiences. Created in 2022, our incredible team has worked tirelessly to make Lawflow, the one stop platform for lawyers. We will continue to work relentlessly to become the technological standard in the Legal- Tech Field, providing big picture insights and solutions for law firms of all sizes.
Lawflow aims to be the leading provider of cutting-edge technologies and legal services, offering scalable solutions for law firms of all sizes. Founded by a group of friends who started by scribbling their ideas on a piece of paper, we aim to offer smart, innovative services to dozens of clients worldwide. We built our solutions by closely listening to our potential clientele and understanding their expectations with our product.
29°C Mostly sunny
ENG IN
4:26 PM 07/12/2022

(Mulla, et al.)

V. IMPLEMENTATION

(Mulla, et al.)

OCR:

1. Document Categorization

Implementation of automated document categorization

The document categorization feature would allow users to upload files to the system and the uploaded file will automatically assigned to the appropriate folders and users can also manually assigned uploaded files by creating their own folders.

Pre-requisites

To implement the document categorization feature, the following libraries and APIs are used:

- For uploaded images : Google Vision
- For PDF Format Documents: pypdf2 library
- Remove Common words: Nltk.stopwords
- To save categories: FireStore

Testing

After running multiple tests, the uploaded documents/images/files were sorted into suggested categories with high accuracy.

Database

The document categorization was coded with python itself, and a FireStore was used to store the data of the user.

Rationale

The uploaded file gets directed to the Google Vision API to get the OCR, and selected keywords are used to compare the categories with the existing categories from the database, if a matching category is found the file gets categorized. Into that folder, in case of no matches found, the user gets the option to create a new category of their own choice.

2. Summarization

Implementation of automated document categorization

The summarization feature would allow users to get summary of their cases based on the case files uploaded to the system.

Pre-requisites

To implement the summarization feature, the following libraries and APIs are used:

- NumPy
- Nltk.cluster
- Nltk.cosine_distance
- For PDF Format Documents: pypdf2 library
- Remove Common words: Nltk.stopwords

- To save files: FireStore

Testing
After running multiple tests, every case created had a summary that describes the key elements for the lawyers.

Database
The summarization was coded with python itself, and a FireStore was used to store the data of the user.

Rationale

Translation

Implementation of the translation feature
The translation feature would allow users to get Arabic to English translation for the files uploaded by them onto the system.PDF Format files are translated.

Pre-requisites
To implement the translation feature, the following libraries and APIs are used:

- For uploaded files : Google Cloud Translation
- For PDF Format Documents: pypdf2 library
- To save files: FireStore

Database
The document categorization was coded with python itself, and a FireStore was used to store the data of the user.

Testing
After running multiple tests, every file uploaded was translated with high precision.

Rationale
The uploaded file gets directed to the Google Cloud Translation Library and then to the bucket, after translation from Arabic to English, the file is added to the new bucket and the translated version can be accessed by the user on the front end of the application.

Workflow

Implementation of the Workflow Engine
The workflow engine feature is one of the key features of the application. While creating a new case, the user will be directed to create a workflow, which consists of main tasks and sub tasks, the lawyer will assign the tasks to the lawyers and the progress of the task is set by the user that assigns the tasks including the task description and responsibilities.

Pre-requisites

To implement the workflow engine feature, the following libraries and APIs are used:

- For uploaded files : Google Cloud Translation
- For PDF Format Documents: pypdf2 library
- To access and track tasks : FireStore

Database

The document categorization was coded with python itself, and a FireStore was used to store the data of the user.

Implementation of application (Back-end)

Overview

The backend of the application is completely done in Python and flask commands are placed on the server so that when a flask command is issued from the front-end, the server can provide the appropriate features.

Pre-requisites

- Our backend server was built primarily with python with flask implementation
- For uploading files to the database, node js was used
- Python libraries
- Flask
- Nodejs upload file
- Python libraries and flask

Implementation of application (Front-end)

The frontend of the application is done with electronJS.

To use Electron, we need to install https://nodejs.org/en/download/ Nodejs.

- To check that Node.js was installed correctly, we use the following commands
  node -v
  npm -v
- We create a folder and initialize an npm package.
  mkdir my-electron-app && cd my-electron-app
  npm init
- The interactive init command will prompt you to set some fields in your config. There are a few rules to follow for the purposes of this tutorial:
  - entry point should be main.js.
  - author and description can be any value, but are necessary for app packaging.
  - install the electron package into your app's devDependencies.
  - npm install --save-dev electron

The main app function is in main.exe which is the exe file that has to be run to start the application. The starting page includes the screen shown before the user has successfully logged in or signed up, depending on the user.

VI. TESTING

(Mulla, et al.)

1. Functional Testing Goals

The functional testing goals process comprises of testing all requirements. All functionalities of LawFlow will be tested including Signup, login, OCR, creation of appointments and contacts, workflow, and various search functionalities (by keyword, image, and filename). All the functionalities will be tested under different scenarios to ensure that they would work accurately in all conditions.

Different lawyer profiles were created and assigned different case to test whether the profiles could access data that was assigned to them.

All testing will be completed within the frame of 1 week. In the unfortunate event that a functionality does not work according to its expected behaviour, the development team will be sent feedback and all functionalities will go through another testing process.

2. Functional Test Plan Scope

Functions to be tested:

1. Login
2. Creating account
3. Ability to add, delete, edit and read appointments
4. Ability to create a case, add files to a case
5. Able to get the automatic categorization from each file
6. User can search with filename, keyword present within each file
7. User can also search with image
8. User can view the summary for each file
9. On opening a case, user can get an overview of the case, with the actors and the relation each actor has with the other actors present in the case

3. Functional Test Plan Assumptions and Constraints

Assumptions

- The device/laptop must have a continuous access to the internet as the application is based on the client server architecture.
- The user has the desktop application installed on their system.
- For the speech-to-text feature, the recordings must be done in a quiet environment and the user should be within 1m from the device.

Constraints

- Document Recognition

- The document that is sent to the system needs to clear and any smudged ink on the document can decipher results.

4. Functional Test Entry Criteria, Approach and Tools

The test entry criteria for our system are as follows:

- Requirements for the system have been defined and approved.
- A ready set of test cases have been created.
- Tools, devices and data have been set up for test environment
-

For our testing purposes we will be using the Alpha and Beta testing method.

Alpha testing will be the testing done internally among the system developers. Its main aim is to figure out all the expected inputs and actions an actual user might take without application and apply appropriate fixes for them.

For our beta testing we will be asking lawyers to use our system and provide feedback on it. This will provide the team with an insight on how the system will be used when it out in the market and the user will be interacting with it. It will also highlight someone the bugs that were overlooked during the alpha phase.

5. Functional Test Cases

The main functionalities of LawFlow are as follows:

1. Workflow Engine

Tracks the status and progress of each case and allows lawyers to assign the particular task to another lawyer within the same case.

2. OCR and Automatic Category Assignment

Scans the document and grasps keywords that will later be used for search purpose assigns documents their related categories.

3. Encryption

Ensures all the case files are encrypted and inaccessible by external parties. Only the case creator and assigned lawyers will be able to view the case by a shared pin.

4. Advanced Search

- Search by keywords – searches for files from a given keywords and retrieves matching files
- Search by filename – searches for file by a given filename
- Search by image – searches for files when an image is given.

5. Appointments, Contacts and To-Do Lists

Allows user to view, create and modify appointments in the calendar and create to-do lists. Users can see their upcoming meetings in the dashboard and add tasks to the to-do lists.

(Mulla, et al.)

6. Testing forms and test results

- **Workflow**

Tested Requirement	**Workflow** **Create Workflow– Positive Case**
Quick Description	This test check if the lawyer is able to create a workflow by providing proper information.
Prerequisite	The user is logged in, the case exists for which the workflow is to be made for, lawyer has access to the case
Input	Tasks: ”Send file to Lawyer A”,” Get client information documents” etc. Task Description: “File x needs to be send to Lawyer A for review”, “Need to get client’s identification documents” Parties Involved: “Lawyer A” “Lawyer B” “Client X”
Expected Output	Workflow created and first task is started
Observed Output	Workflow created
Verdict	Pass
Comments	No error, works as intended.

Table 30:Testing -create workfow positive

Tested Requirement	**Workflow**

	Create Workflow– Negative Case
Quick Description	This test check if the lawyer is able to create a workflow by providing proper information.
Prerequisite	The user is logged in, the case exists for which the workflow is to be made for, lawyer has access to the case
Input	Tasks: ” ” Task Description: “File x needs to be send to Lawyer A for review”, “Need to get client’s identification documents” Parties Involved: “Lawyer A” “Lawyer B” “Client X”
Expected Output	Workflow not created, No tasks provided in field.
Observed Output	Workflow no created, Tasks need at least 1 value
Verdict	Pass
Comments	Error, no information given.

Table 31:Testing Create workflow Negative

Tested Requirement	**Workflow** **Create Workflow Subtasks– Positive Case**
Quick Description	This test check if the lawyer is able to create subtasks in the current workflow.

Prerequisite	The user is logged in, the case exists for which the workflow is to be made for, lawyer has access to the case, Workflow exists for case.
Input	Subtask Name: “Wait for document review”, Subtask Description: “ Wait from document x from lawyer A” Subtask Parties Involved:” Lawyer a, lawyer B ” Main Task: “ Send file to lawyer A”
Expected Output	Subtask created for mentioned main task
Observed Output	Subtask for workflow created
Verdict	Pass
Comments	No error, works as intended.

Table 32:Testing - Create workflo subtask positive case

Tested Requirement	**Workflow** **Create Workflow Subtasks– Negative Case**
Quick Description	This test check if the lawyer can create subtasks in the current workflow.
Prerequisite	The user is logged in, the case exists for which the workflow is to be made for, lawyer has access to the case, Workflow exists for case.

Input	Subtask Name: “Wait for document review”, Subtask Description: “ Wait from document x from lawyer A” Subtask Parties Involved:” Lawyer a, lawyer B ” Main Task: “ “
Expected Output	No subtask created as no main task was mentioned
Observed Output	No subtask was created
Verdict	Pass
Comments	Error, missing main task

Table 33:Testing - Create Workflow subtask - Negative Case

Tested Requirement	**Workflow** **Update status– Positive Case**
Quick Description	This test check if the lawyer is able to update task status for workflow
Prerequisite	The user is logged in, the case exists for which the workflow is to be made for, lawyer has access to the case, and Workflow exists for case with up to 2 tasks.
Input	Task ID: “Task 1”

Expected Output	Task 1 is marked as “Finished”
Observed Output	Task 1 is marked as “Finished”
Verdict	Pass
Comments	No error, works as intended.

Table 34:Testing Update status Positive

Tested Requirement	**Workflow** **Update status– Negative Case**
Quick Description	This test check if the lawyer is able to update task status for workflow
Prerequisite	The user is logged in, the case exists for which the workflow is to be made for, lawyer has access to the case, and Workflow exists for case with up to 2 tasks.
Input	Task ID: “Task 642672”
Expected Output	Task ID 642672 doesn’t not exist could not mark as finished.
Observed Output	Task ID 642672 doesn’t not exist could not mark as finished.
Verdict	Pass

Comments	Error, wrong Task id given

Table 35:Testing Update Status Negative

Tested Requirement	**Workflow** **Assign task to – Positive Case**
Quick Description	This test check if the lawyer is able to assign task to other lawyers
Prerequisite	The user is logged in, the case exists for which the workflow is to be made for, lawyer has access to the case, and Workflow exists for case with up to 2 tasks.
Input	Lawyer Name: “John Doe” Task id: Task 2
Expected Output	Task 2 was assigned to John Doe
Observed Output	Task 2 was assigned to John Doe
Verdict	Pass
Comments	No error, works as intended.

Table 36: Testing Assign task to Positive

Tested Requirement	**Workflow**

(Mulla, et al.)

	Assign task to – Negative Case
Quick Description	This test check if the lawyer is able to assign task to other lawyers
Prerequisite	The user is logged in, the case exists for which the workflow is to be made for, lawyer has access to the case, and Workflow exists for case with up to 2 tasks.
Input	Lawyer Name: “John Doe” Task id: Task 755
Expected Output	Task 755 doesn’t exist
Observed Output	Task 755 could not be assigned to John Doe, task does not exist
Verdict	Pass
Comments	Error, Wrong task id

Table 37: Testing - Assign task to Negative

- ## OCR and Automatic Category Assignment

Tested Requirement	**OCR** **Pick Category – Positive Case**

Quick Description	This test check if the lawyer is able to select a category for a image
Prerequisite	The user is logged in lawyer has access to the case, 2 categories are suggested for the image: A,B
Input	Image : ‘image.png” Category: “A”
Expected Output	The image is stored under category A
Observed Output	The image is stored under category A
Verdict	Pass
Comments	Works as intended

Table 38:Testing OCR pick Category Positive

Comments	Test Successful

- **Advanced Search**

Search by Keywords

Tested Requirement	**Search** **By Keywords – Positive Case**

Quick Description	This test case checks if the system is able to find a file based on the keywords entered by the user.
Prerequisite	The user is logged in and has access to the particular file being requested for
Input	Keyword: “abandonment”
Expected Output	File is found and the file is also displayed on the search screen
Observed Output	File found
Verdict	Pass
Comments	NIL

Table 39: Testing- Search by keywords Positive

Tested Requirement	**Search** **By Keywords – Negative Case**
Quick Description	This test case checks if the system is able to find a file based on the keywords entered by the user.
Prerequisite	The user is logged in and has access to the particular file being requested for
Input	Keyword: “abruptly”

Expected Output	File is not found and an error message is displayed saying "File with keyword not found"
Observed Output	File not found
Verdict	Pass
Comments	NIL

Table 40:Testing Search by keyword Negative case

Search by Filename

Tested Requirement	**Search** **By Filename – Positive Case**
Quick Description	This test case checks if the system is able to find a file based on the filename entered by the user.
Prerequisite	The user is logged in and has access to the particular file being requested for
Input	Filename : "legaldocument_casex7891"
Expected Output	File is found and the file is also displayed on the search screen
Observed Output	File found
Verdict	Pass

Comments	NIL
Tested Requirement	Search By Filename – Negative Case
Quick Description	This test case checks if the system is able to find a file based on the filename entered by the user.
Prerequisite	The user is logged in and has access to the particular file being requested for
Input	Filename : “ldocument_casex789”
Expected Output	File is not found and error message saying “File doesn’t exist” is displayed on the screen
Observed Output	File found
Verdict	Pass
Comments	NIL

Table 41:Testing search by filename Positive

Search by Image

Tested requirement	**Search** **By Image - Positive**

Quick description	This test case is to check if the system provides the user with appropriate search results related to the image uploaded by the user.
Prerequisite	User uploads an image.
Input	Image path = “Desktop/Image.jpeg”
Expected output	Appropriate results displayed.
Observed output	Appropriate results displayed.
Verdict	Pass
Comments	NIL

Table 42:Testing search by image positive

Tested requirement	**Search** **By Image - Negative**
Quick description	This test case is to check if the system provides the user with appropriate search results related to the image uploaded by the user.
Prerequisite	User uploads an image.

(Mulla, et al.)

Input	Image path = “Desktop/Image.jpeg”
Expected output	No content found.
Observed output	No content found.
Verdict	Pass
Comments	NIL

Table 43:Testing Search by image negative

- ## Appointments and To-Do Lists

Tested Requirement	**Add appointment – Positive Case**
Quick Description	This test case checks if the system is able to add an appointment/meeting for a date on the calendar.
Prerequisite	The user is logged in.
Input	Date:”01/12/2022”, Name: “Meeting for ASGH Fraud Case”, StartTime: “09:00”, EndTime: “10:00”, Description: ”abc .....”
Expected Output	After clicking the ‘create event’ option, the form to create a new event and then it can be seen on the event directory.
Observed Output	Appointment Added.

Verdict	Pass
Comments	NIL

Table 44:Testing add appointment Positive

Tested Requirement	**Add appointment – Negative Case**
Quick Description	This test case checks if the system is able to add an appointment/meeting for a date on the calendar.
Prerequisite	The user is logged in.
Input	Date:"01/12/2022", Name: "Meeting for ASGH Fraud Case", StartTime: "09:00", EndTime: "10:00", Description: "abc ....."
Expected Output	After clicking the 'create event' option, the form to create a new event. Error displayed "Time slot not available"
Observed Output	Error: "Time slot not available".
Verdict	Pass
Comments	NIL

Table 45:Testing add appointment Negative

Tested Requirement	**Delete appointment – Positive Case**

Quick Description	This test case checks if the system is able to delete an appointment/meeting from the calendar.
Prerequisite	The user is logged in.
Input	Select event in the event directory
Expected Output	After selecting the event from the event directory, click the delete icon and contact will be deleted.
Observed Output	Appointment deleted.
Verdict	Pass
Comments	NIL

Tested Requirement	**Update appointment – Positive Case**
Quick Description	This test case checks if the system is able to update an appointment/meeting from the calendar.
Prerequisite	The user is logged in.
Input	New Date: “04/11/2022”, newStartTime:”13:00”. newEndTime:” 15:00”
Expected Output	After selecting the event from the event directory, click on the edit icon to update the appointment details.

Observed Output	Appointment update.
Verdict	Pass
Comments	NIL

Table 46:Testing update appointment positive

Tested Requirement	**To-do Lists – Positive Case**
Quick Description	This test case checks if the system is able to add to to-do lists
Prerequisite	The user is logged in
Input	New list item
Expected Output	Item added to to-do lists
Observed Output	Item added to to-do lists
Verdict	Pass
Comments	Test Successful
Tested Requirement	To-do Lists – Positive Case
Quick Description	This test case checks if the system is able to modify to-do lists

Prerequisite	The user is logged in and a list item exists
Input	Add changes to list time
Expected Output	Item updated
Observed Output	Item updated
Verdict	Pass
Comments	Test Successful

Table 47:Testing todo List Positive

- **Login**

Tested requirement	**Login - Positive**
Quick description	This test case is to check if the system allows the user to login to the system when the user provides valid credentials.
Prerequisite	User record exists in database.
Input	Username=”Admin007”, password=”pass445”
Expected output	Successful login.
Observed output	User logged in successfully.

Verdict	Pass
Comments	NIL

Table 48:Testing login Positve

Tested requirement	**Login - Negative**
Quick description	This test case is to check if the system allows the user to login to the system when the user provides credentials.
Prerequisite	User record exists in database.
Input	Username=”Admin223”, password=”testing000”
Expected output	Username not found.
Observed output	Invalid credentials
Verdict	Pass
Comments	NIL

Table 49:Testing Login Negative

Tested requirement	**Login - Negative**
Quick description	This test case is to check if the system allows the user to login to the system when the user provides credentials.

(Mulla, et al.)

Prerequisite	User record exists in database.
Input	Username=”Admin007”, password=”testing000”
Expected output	Incorrect Password.
Observed output	Invalid credentials
Verdict	Pass
Comments	NIL

Table 50:Testing Login Negative

- ## Signup

Tested requirement	**Lawyer Signup - Positive**
Quick description	This test case is to check if the system allows the user to sign up when the user provides the required information.
Prerequisite	User is aged 18 and above, user provides a valid email, password provided by user contains a special character, a capital letter and a number.
Input	Username=”Lawyer44”, password=”tester#1T”, email = “tester@email.com”, age = 21

Expected output	Registered successfully.
Observed output	Registered successfully.
Verdict	Pass
Comments	NIL

Table 51:Testing Lawyer Signup Positive

Tested requirement	**Lawyer Signup - Negative**
Quick description	This test case is to check if the system allows the user to sign up when the user provides the required information.
Prerequisite	User is aged 18 and above, user provides a valid email, password provided by user contains a special character, a capital letter and a number.
Input	Username=”Lawyer44”, password=”tester#1”, email = “tester@email.com”, age = 21
Expected output	Password needs to contain a special character, a number and a capital letter.
Observed output	Password needs to contain a special character, a number and a capital letter.

Verdict	Pass
Comments	NIL

Table 52:Lawyer Signup Negative

Tested requirement	**Lawyer Signup - Negative**
Quick description	This test case is to check if the system allows the user to sign up when the user provides the required information.
Prerequisite	User is aged 18 and above, user provides a valid email, password provided by user contains a special character, a capital letter and a number.
Input	Username=”Lawyer44”, password=”tester#1”, email = “tester@email.com”, age = 16
Expected output	Age needs to be greater than or equal to 18.
Observed output	Age needs to be greater than or equal to 18.
Verdict	Pass
Comments	NIL

Table 53:Lawyer Signup Negative

(Mulla, et al.)

VII. MAINTENANCE

(Mulla, et al.)

In this section, we will discuss on how we will be maintaining the system in the long run. We will be focusing on 4 methods to follow:

• Adaptive – We have developed a maintenance plan that is put into action to make sure the application will be extensible and versatile. Through the different stages of development, the features are subject to change or be upgraded, the addition of new features as well based on the market demand is taken into consideration.

• Perfective – To maintain our set standards we will perform different tests on our application as well as researching on enhancements that can be implemented by observing our competitors and current market demands.

• Corrective – An open forum will be put into action for the application to take in user reviews and complaints that will be used for bug fixing and mishaps that could occur in the system.

• Preventive – We will use multiple layers of testing to prevent errors and all initial security, as well as protection methods, are already implemented. Once a bug is found, we will have a recovery team who will act on immediately to mitigate it.

Six software maintenance processes as:

1. The implementation process contains software preparation and transition activities, such as the conception and creation of the maintenance plan; the preparation for handling problems identified during development; and the follow-up on product configuration management.
2. The problem and modification analysis process, which is executed once the application has become the responsibility of the maintenance group. The maintenance programmer must analyze each request, confirm it (by reproducing the situation) and check its validity, investigate it and propose a solution, document the request and the solution proposal, and finally, obtain all the required authorizations to apply the modifications.
3. The process considering the implementation of the modification itself.
4. The process acceptance of the modification, by confirming the modified work with the individual who submitted the request in order to make sure the modification provided a solution.
5. The migration process (platform migration, for example) is exceptional, and is not part of daily maintenance tasks. If the software must be ported to another platform without any change in functionality, this process will be used and a maintenance project team is likely to be assigned to this task.
6. Finally, the last maintenance process, also an event which does not occur on a daily basis, is the retirement of a piece of software.

Appendices

Glossary

Multifactor Authentication - Multi-factor Authentication (MFA) is an authentication method that requires the user to provide two or more verification factors to gain access to a resource such as an application, online account, or a VPN.

Document categorization - Document classification or document categorization is a problem in library science, information science and computer science. The task is to assign a document to one or more classes or categories. This may be done "manually" (or "intellectually") or algorithmically.

Document recognition - Document recognition is an OCR-powered technology that can classify and process different types of unstructured / semi-structured information from documents. DR uses optical character recognition (OCR) to process data in the fastest and most accurate way possible.

ElectronJs - Electron is a free and open-source software framework developed and maintained by GitHub. The framework is designed to create desktop applications using web technologies which are rendered using a flavour of the Chromium browser engine, and a backend using the Node.js runtime environment.

FireStore - Cloud Firestore is a NoSQL document database that helps easily store, sync, and query data for your mobile and web apps, at global scale.

Flask - Flask is a micro web framework written in Python. It is classified as a microframework because it does not require particular tools or libraries.

Google cloud translation – Google Cloud Translation helps dynamically translate between languages using pre-trained or custom ML models.

Google Vision API - The Google Cloud Vision API allows developers to easily integrate vision detection features within applications, including image labelling, face and landmark detection, optical character recognition (OCR), and tagging of explicit content.

High level design - High-level design (HLD) explains the architecture that would be used to develop a system. The architecture diagram provides an overview of an entire system, identifying the main components that would be developed for the product and their interfaces.

Integration testing - Integration testing (sometimes called integration and testing, abbreviated I&T) is the phase in software testing in which individual software modules are combined and tested as a group.

Nltk.cluster - Clustering describes the task of discovering groups of similar items with a large collection. It is also describe as unsupervised machine learning, as the data from which it learns is unannotated with class information, as is the case for supervised learning.

Nltk.stopwords - Stop words are words that are so common they are basically ignored by typical tokenizers. By default, NLTK (Natural Language Toolkit) includes a list of 40 stop words, including: “a”, “an”, “the”, “of”, “in”, etc. The stopwords in nltk are the most common words in data.

NodeJS - Node.js is an open-source server environment. Node.js is cross-platform and runs on Windows, Linux, Unix, and macOS. Node.js is a back-end JavaScript runtime environment. Node.js runs on the V8 JavaScript Engine and executes JavaScript code outside a web browser.

Non-functional requirements - Non-functional requirements or NFRs are a set of specifications that describe the system's operation capabilities and constraints and attempt to improve its functionality.

NumPy - NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

OCR - Optical character recognition or optical character reader is the electronic or mechanical conversion of images of typed, handwritten or printed text into machine-encoded text, whether from a scanned document, a photo of a document, a scene-photo or from subtitle text superimposed on an image.

pypdf2 - PyPDF2 is an open-source pure-python PDF library capable of splitting, merging, cropping, and transforming the pages of PDF files.

Workflow engine - A workflow engine is a software application that manages business processes. It is a key component in workflow technology and typically makes use of a database server.

Version Index

Date	Version#	Description
7/12/2022	1.0	Final Report Document

Table 54:Version Index

References

Batchu, V., 2019. 5 navigation tips to improve the UX of your product/service. [online]
Medium. Available at: <https://uxdesign.cc/5-navigation-tips-to-improve-ux-of-yourproduct-service-f64840ef2f4c>

Crowd Favorite. 2013. Agile design: what we’ve learned. [online] Available at: https://crowdfavorite.com/agile-design-what-weve-learned/

H, D., 2018. A day in the life of a lawyer. [online] https://www.simplylawjobs.com. Available at: <https://www.simplylawjobs.com/career-hub/articles/insights/day-in-the-life-of/a-day-in-the-life-of-alawyer>

Hamilton, T., 2021, 'Alpha Testing Vs Beta Testing: What's the Difference?', Guru 99,
Available at: https://www.guru99.com/alpha-beta-testing-demystified.html

Minhas, S., 2019. Agile design process. [online] UX Collective. Available at: https://uxdesign.cc/agile-design-process-24be92018ad2

n.d. [online] Available at: https://www.cloudflare.com/learning/privacy/what-is-data privacy/

Nielsen, J., 1994. 10 Usability Heuristics for User Interface Design. [online] Nielsen Norman
Group. Available at: <https://www.nngroup.com/articles/ten-usability-heuristics/>

Zalavadia, S., 2022, 'Complete Functional Testing Guide With Its Types And Example',
Software Testing Help, Available at: https://www.softwaretestinghelp.com/guide-tofunctional-testing/

Progress reports

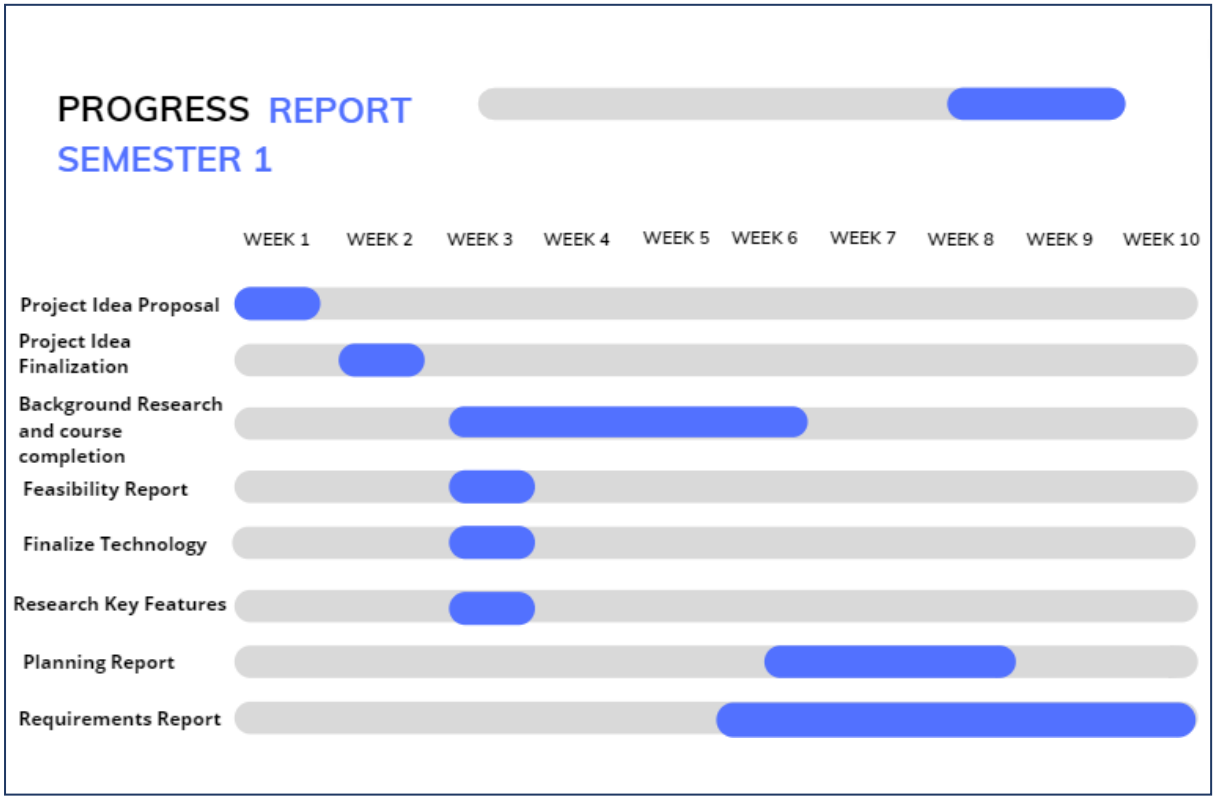


Figure 28:Progress Report 1

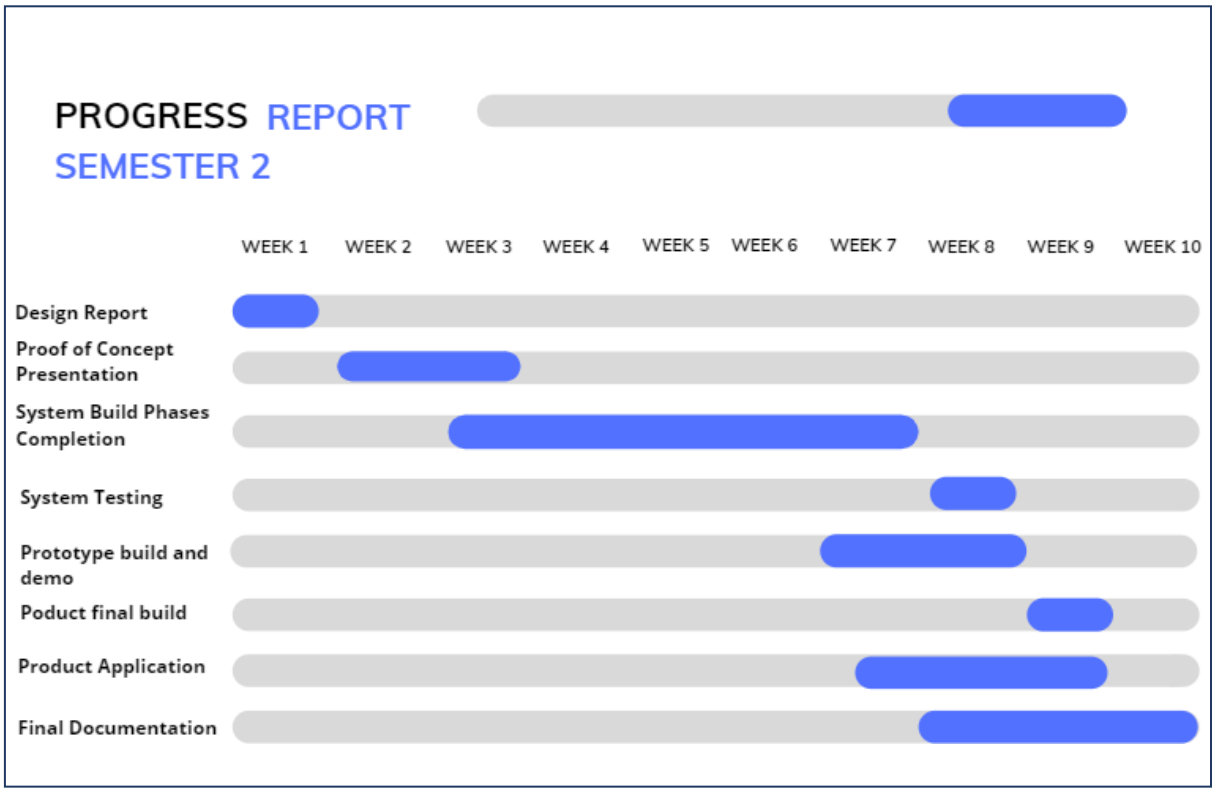


Figure 29:Progress Report 2

Meeting Minutes

Week: 5 Date: 12/05/2022 Time:5PM

Present

1. Harman Singh
2. Eman Mulla
3. Abeera Amir
4. Muhammad Shaad
5. Ayushi Agnihotri

Next meeting date: 14/05/2022

Week: 5 Date: 14/05/2022 Time: 1PM

Present

6. Harman Singh
7. Eman Mulla
8. Abeera Amir
9. Muhammad Shaad
10. Ayushi Agnihotri

SR	Agenda and Issues to be Discussed	Resolution	Date of Resolution
1	Reviewed all our UI screens and ran few test cases for consistency	Few changes are to be made to make the feel more design friendly.	14/05/2022

Next meeting date: 14/05/2022

Proposed agenda:

SR	Agenda and Issues to be Discussed	Raised by
1	Finalize the ui layout and forward it to WISP and faculty mentors for feedback	everyone

Week: 5 Date: 12/05/2022 Time: 5PM

Present

1. Harman Singh
2. Eman Mulla
3. Abeera Amir
4. Muhammad Shaad
5. Ayushi Agnihotri

SR	Agenda and Issues to be Discussed	Resolution	Date of Resolution
1	Discussed the UI and distributed screens to do among us.	Work was equally distributed with everyone getting at least 1 screen to do.	12/05/2022

Next meeting date: 14/05/2022

Proposed agenda:

SR	Agenda and Issues to be Discussed	Raised by
1	Go over everyone's ui and match its consistency and run test cases.	everyone

Week: 5 Date: 14/05/2022 Time: 1PM

Present

6. Harman Singh
7. Eman Mulla
8. Abeera Amir
9. Muhammad Shaad
10. Ayushi Agnihotri

SR	Agenda and Issues to be Discussed	Resolution	Date of Resolution
1	Reviewed all our UI screens and ran few test cases for consistency	Few changes are to be made to make the feel more design friendly.	14/05/2022

Next meeting date: 14/05/2022

Proposed agenda:

SR	Agenda and Issues to be Discussed	Raised by
1	Finalize the ui layout and forward it to WISP and faculty mentors for feedback	everyone

Task Tracking Ledger Week 5

Member 1 Target	Achieved (Tick)	Member 2 Target	Achieved (Tick)	Member 3 Target	Achieved (Tick)	Member 4 Target	Achieved (Tick)	Member 5 Target	Achieved (Tick)	Team Target
Complete the Feasibility report part	(Tick)	Complete the Feasibility report part	(Tick)	Complete the Feasibility report part	(Tick)	Complete the Feasibility report part	(Tick)	Complete the Feasibility report part	(Tick)	Complete the Feasibility report
Complete the Feasibility report part	(Tick)	Complete the Feasibility report part	(Tick)	Complete the Feasibility report part	(Tick)	Complete the Feasibility report part	(Tick)	Complete the Feasibility report part	(Tick)	Complete the Feasibility report
Compile and submit the report	(Tick)	Compile and submit the report	(Tick)	Compile and submit the report	(Tick)	Compile and submit the report	(Tick)	Compile and submit the report	(Tick)	Compile and submit the report
Split up the UI design work	(Tick)	Split up the UI design work	(Tick)	Split up the UI design work	(Tick)	Split up the UI design work	(Tick)	Split up the UI design work	(Tick)	Split up the UI design work
Make UI front page	(Tick)	Make UI Client main page	(Tick)	Make Schedule Page	(Tick)	Make search page ui	(Tick)	Make audio page ui	(Tick)	Make complete ui design.

Week: 6 Date: 18/05/2022 Time: 5PM

Present

11. Harman Singh
12. Eman Mulla
13. Abeera Amir
14. Muhammad Shaad
15. Ayushi Agnihotri

SR	Agenda and Issues to be Discussed	Resolution	Date of Resolution
1	Had meeting with Dr.Farhad where we showed our draft UI screens	Got some good feedback regarding the UI.	18/05/2022
2	Discussed some changes to the UI and some of its functionality.	Got feedback on few pages that need a rework and few components to be changed.	18/05/2022

Next meeting date: 18/05/2022

Proposed agenda:

SR	Agenda and Issues to be Discussed	Raised by
1	Discuss the feedback gives by our mentor within ourselves	everyone
2	Divide up work for the features report	everyone

Week: #6 Date: 18/05/2022 Time: 10PM

Present

16. Harman Singh
17. Eman Mulla
18. Abeera Amir
19. Muhammad Shaad
20. Ayushi Agnihotri

SR	Agenda and Issues to be Discussed	Resolution	Date of Resolution

1	Discussed the UI feedback and made some changes according to them.	Changed made, still need some more improvement with the UI flow before presentation.	18/05/2022
2	Discussed what to write for features report	Split up work among ourselves for the features report	18/05/2022

Next meeting date: 14/05/2022

Proposed agenda:

SR	Agenda and Issues to be Discussed	Raised by
1	Discuss the technology that will be used.	everyone

Week: 6 Date: 19/05/2022 Time: 3PM

Present

21. Harman Singh
22. Eman Mulla
23. Abeera Amir
24. Muhammad Shaad
25. Ayushi Agnihotri

SR	Agenda and Issues to be Discussed	Resolution	Date of Resolution
1	Discussed the main technology that will be used for the project	Figured out the main technology.	19/05/2022
2	Started making the system architecture on paper.	Made the basic architecture design, still more work needed for it.	19/05/2022

Next meeting date: 14/05/2022

Proposed agenda:

SR	Agenda and Issues to be Discussed	Raised by
1	Combine work for feature report.	everyone

2	Start working on the presentation	everyone

Task Tracking Ledger Week 6

	Member 1 Target	Achieved (Tick)	Member 2 Target	Achieved (Tick)	Member 3 Target	Achieved (Tick)	Member 4 Target	Achieved (Tick)	Member 5 Target	Team Target
Day 4	Work on their part of feature report	(Tick)	Work on their part of feature report	(Tick)	Work on their part of feature report	(Tick)	Work on their part of feature report	(Tick)	Work on their part of feature report	Work on feature report individually
Day 5	Find main technology to use for system	(Tick)	Find main technology to use for system	(Tick)	Find main technology to use for system	(Tick)	Find main technology to use for system	(Tick)	Find main technology to use for system	Figure out main technology for project
Day 6	Start working on the PPT	(Tick)	Start working on the PPT	(Tick)	Start working on the PPT	(Tick)	Start working on the PPT	(Tick)	Start working on the PPT	Start working on the PPT