Dilla University

College of Engineering and Technology

School of Computing and Informatics

Department of Computer Science

TITLE OF PROJECT

Dilla University Legal Aid Service Management System

A project submitted in partial fulfillment of the requirements for the

degree of B.Sc. in Computer Science (Regular) - Part 1

Submitted by

1.AMANUEL TIZAZU…………………RNS-5648/21

2.CHERINET AFEWORK………………RNS-9619/21

3.BISRAT FIKRE …..…………………RNS-4903/21

4.DEGINET SOLOMON………………...RNS-0260/21

5.ASTAREKEGN DEGEFU……………...RNS-4250/21

Advisor Dinkie G.(Lecturer)

Sub. Data 27/01/2025 GC

Dilla, Ethiopia

CERTIFICATE

The project titled

Dilla University Legal Aid Service Management System

Compiled by

1. Amanuel Tizazu……………………RNS-5648/21

2. Cherinet Afewrk…………….………RNS-9619/21

3. Bisrat Fikre …………………………RNS-4903/21

4. Deginet Solomon…….…………….…RNS-0260/21

5. Astarekegn Degefu……………………RNS-4250/21

As partial fulfilment of the requirements for degree of

Bachelor of Science

In

Computer Science

From

Dilla University

Head of the Department   **Examiners**

<name>

………………………………..

                                                        ………………………………….

                                                        ………………………………….

                                                        ………………………………….

DECLARATION

This is to declare that the project entitled "Dilla University Legal Aid Service Management System" is an original work done by us, undersigned students in the Department of Computer Science, School of Computing and Informatics, College of Engineering and Technology, Dilla University. The reports are based on the project work done entirely by us and not copied from any other source.

Advisor

Dinkie G.(Lecturer)

Signature   _____

Department of Computer Science

Amanuel Tizazu	RNS-5648/21	<signature>
Cherinet Afewrk	RNS-9619/21	<signature>
Bisrat Fikre	RNS-4903/21	<signature>
Deginet Solomon	RNS-0260/21	<signature>
Astarekegn Degefu	RNS-4250/21	<signature>

ACKNOWLEDGEMENT

Acronyms

API ………………………Application Programming Interface

CSS ………………………. Cascading Style Sheets

DU ………………………Dilla University

DULAS…………………Dilla University Legal Aid Service

GUI ………………………Graphical User Interface

LACMS…………………Legal Aid Case Management System

UML………………………Unified Modeling Language

OOP ………………………Object-Oriented Programming

PHP …………………………Hypertext Preprocessor

RAM ……………………...Random Access Memory

SDLC ………………………. Software Development Life Cycle

SSD ………………………Solid-State Drive

UAT ………………………. User Acceptance Testing

Abstract

The proposed Dilla University Legal Aid Service Management System (DULASMS) is a web-based platform designed to streamline and modernize the management of legal aid services provided by Dilla University. The system aims to replace the existing manual processes, which are prone to inefficiencies such as data redundancy, inconsistency, and delays in case handling. DULASMS provides an integrated solution to manage clients, cases, appointments, documents, and requests while ensuring data accuracy, reliability, and security. Key features include client registration, case creation, appointment scheduling, and real-time status updates. Coordinators can assess and manage legal aid requests effectively, and faculty members can collaborate on complex cases. The system supports multi-language functionality, including Amharic, to enhance accessibility and inclusivity. Additionally, robust reporting tools empower administrators to monitor and evaluate the overall performance of the legal aid service for customers. By automating workflows, reducing manual errors, and fostering better coordination, DULASMS improves service delivery and ensures that legal aid reaches those in need efficiently and equitably. This system not only addresses the limitations of the existing framework but also supports Dilla University's commitment to community service and social justice.

Table of Contents

List of Figures

List of Tables

CHAPTER ONE

INTRODUCTION

1. Overview

Technology is playing a significant role in simplifying and enhancing human activities across various sectors. In today's world, where organizations and institutions strive to provide fast, efficient, and accurate services, embracing technology has become a necessity to meet user demands and improve operational efficiency. To remain competitive and effective, organizations must adopt computerized systems that streamline their activities and eliminate the inefficiencies associated with manual processes.

Dilla University Legal Aid Service (DULAS) is a critical initiative designed to address the legal needs of financially disadvantaged and marginalized members of the community. It operates under the Dilla University School of Law in collaboration with the United Nations, aiming to bridge the justice gap by providing free legal assistance. Despite its noble mission, the service currently relies on a manual system for case registration, tracking, and assignment, which leads to inefficiencies such as delays, errors in data handling, and challenges in communication.

Currently, cases are managed manually by coordinators, and legal professionals are assigned without an automated workflow, creating additional challenges in tracking case progress and managing workloads effectively. Reports are prepared manually, making it time-consuming and inefficient. Furthermore, there is no centralized database, leading to disorganized data storage and difficulties in retrieval, which impacts service delivery.

To address these challenges, this project aims to design and develop an automated, web-based system for DULAS. The system will enable efficient case management, centralized data storage, and streamlined communication between coordinators, legal professionals, and clients. This computerized system will not only enhance the speed and accuracy of operations but will also provide transparency and accessibility to legal services, ensuring that DULAS fulfills its mission of promoting justice and equity in a more effective manner.

1.1 Background of the organization

Dilla University is a prominent public higher education institution. Established in 1996 as the Dilla College of Teachers' Education and Health Sciences, it initially focused on training educators and health professionals. In 2001, it became part of Debub University. By 2006, the Council of Ministers granted Dilla its own certification as an independent university, leading to its current status as Dilla University.

The university has experienced significant growth, now operating across multiple campuses with a diverse range of academic programs. It comprises several colleges, including Technology and Engineering, Business and Economics, Health and Medical Science, Social Science and Humanities, Agriculture and Natural Resources, and Natural and Computational Science. Additionally, it houses institutes such as Indigenous Studies and Education and Behavioural Science, along with specialized schools like the School of Law.

Dilla University offers 62 undergraduate, 45 master's, and 4 Ph.D. programs across various disciplines, including Agriculture, Business, Education, Engineering, Health Sciences, and Natural and Social Sciences. The university also hosts specialized research centers focused on energy and the environment, education, food and nutrition, and child, women, and youth studies.

The university's mission is to prepare knowledgeable, skilled, and mature graduates; design and provide training and community services pertinent to the country's and the region's priority needs; and promote and enhance research and consultancy services that cater to the development needs and policy imperatives of the country. Its vision is to strive to be among the top ten universities in East Africa by excelling in teaching, learning, research, and community services.

1.2 Introduction about the project

The DU Legal Aid Service currently operates through a manual system that handles legal aid requests from clients in need. This service is available across multiple offices, including in Dilla and four additional branch offices. In its current state, the process of managing, reporting, and assigning cases is highly unstructured and manual system.

Each office coordinator is responsible for collecting client cases by registering their information, filling out forms, and reporting to the DU School of Law. Coordinators assess whether a case requires advanced legal aid and then DU School of Law decide whether to escalate it to specialized staff or legal experts. Once escalated, a lawyer or faculty member is assigned to the case based on their availability and specialization. However, this manual workflow introduces delays, increases workload inefficiencies, and poses challenges in maintaining organized records.

Moreover, the coordinators must generate daily reports manually for the DU School of Law. This includes information on the number of clients served, the cases assigned, and the status of ongoing cases. The absence of system leads to inconsistencies in data recording, reporting, and case follow ups. Additionally, assigning cases to legal professionals involves assessing their workload manually, which often results in overburdening or underutilization. Notifications regarding assignments are communicated verbally or through informal channels, which further hampers efficiency.

To address these challenges, our project develops an automated web-based system that streamlines the processes of case registration, coordination, reporting, lawyer assignment, and data handling. This system allows coordinators to efficiently register clients and escalate cases by filling out predefined forms. The system checks the availability and specialization of legal staff or faculty, ensuring that cases are assigned to the right professional.

Once a lawyer or faculty member is assigned, they receive a real-time notification through the system, confirming their case assignment and enabling them to track the case status. Additionally, coordinators have access to a centralized platform for daily reporting, reducing manual errors and ensuring consistency. This system also maintains a comprehensive database of case histories, lawyer workloads, and client details, making data management seamless and accessible. By implementing this system as part of our final-year project, we modernize the DU

Legal Aid Service, improve efficiency, and ensure timely and equitable access to legal assistance for the community.

1.3 Statement of problem

The current manual system employed by the DU Legal Aid Service is presenting several key challenges. First, case management is inefficient, as manual processes for registering cases and assigning legal professionals hinder timely service delivery. Additionally, requires coordinators to track and report case progress manually, which is outdated Infrastructure Furthermore, data management poses significant challenges; paper-based storage is disorganized, difficult to access, and at risk of loss or damage. Limited Accessibility also a problem in paper-based system restricts coordinators, legal advisors, and administrators from accessing case records remotely. Communication gaps also exist, as notifications regarding case assignments depend on informal methods, leading to delays and misunderstandings. This project seeks to address these issues by implementing digitalize, centralized and efficient web-based system for the DU Legal Aid Service.

1.4 Objective of the project

1.4.1 General Objective

The main objective of this project is to design web based Legal Aid Case Management System (LACMS) for the Dilla University

1.4.2 Specific objective

To achieve the above general objective the proposed system has the following specific objectives.

- Develop a digital platform where clients register their legal cases through coordinators.
- Replace the manual, paper-based registration and reporting process with an efficient digital system that captures all necessary client and case detail.
- Design a user-friendly interface that ensures accessibility for coordinators, faculty members and other with varying levels of digital literacy

- Streamline Case Assignment: Develop a mechanism to assign complex cases to faculty members based on workload, specialization, and availability. Include a notification feature to inform faculty members about their assignments in real time.
- Data Management: Create a secure, centralized database for storing client information, case details, case progress updates and ensure easy data retrieval.
- Improve Service Accessibility: can access remotely

The realization of these objectives enables DU Legal Aid to modernize its operations, address existing inefficiencies, and significantly improve its capacity to provide timely, equitable, and effective legal assistance to the community.

1.5 Scope and limitation

1.5.1 Scope

The scope of this project outlines the boundaries and features designed to enhance the efficiency and reliability of the DU Legal Aid Service. It includes several key areas, starting with client case management, where coordinators register new cases by collecting client details and uploading relevant documentation. The system features a user-friendly interface that allows easy access to the status of ongoing cases. Additionally, case assignment is streamlined, assigning cases to faculty members or legal professionals based on their specialization and current workload, ensuring a fair distribution of tasks and preventing any staff member from becoming overburdened.

Moreover, the project implements a notification system that sends real-time alerts to assigned lawyers or faculty members regarding their case assignments, while also notifying coordinators and administrators about overdue cases or pending actions. A reporting system will enable coordinators to generate reports. providing administrators with statistical insights on case handling, resolution times, and individual staff performance. The system features database management for storing client details, case histories, and reports, allowing for easy data retrieval and backup. Role-based access control ensures data privacy and integrity, with users accessing only the features relevant to their roles. Finally, performance metrics are established to monitor lawyer performance and generate workload reports, helping administrators evaluate resource efficiency.

1.5.2 Limitation

Limitations are matters and occurrences that arise in a study which are out of the researcher's control. They limit the extension to which a study can go, and sometimes affect the end result and conclusions that can be drawn. The few limitations in this web application are as follows:

- Since, it is a web-based system the service given by the system is not accessed by users as long as there is no connection.

1.6 Methodology

The methodology section outlines the structured approach used to design, develop, and implement the proposed system. It provides a clear roadmap for achieving project goals while ensuring efficiency and accuracy in system development.

1.6.1. Data Gathering Methodology

To gather the necessary information for system development, the following techniques will be employed:

1. Interview

- Conduct interviews with DU Legal Aid coordinators, faculty, and administrators
- Focus on understanding the challenges of the current manual system and the specific features users expect from the new system.

2. Document Analysis

- Analyze existing forms, case records, and reports used in the manual system to understand data flow and content requirements.
- Identify bottlenecks and inefficiencies to address in the system.

3. Observation

Observe current processes such as case registration, report generation, and lawyer assignment. Identify bottlenecks and inefficiencies to address in the system.

1.6.2. Design Methodology

As the methodology, the model we prefer to employ for the design, planning, implementation, and achievement of our project objectives is the Agile Model in the Software Development Life Cycle (SDLC) because of the following advantages:

- Easily adaptable to changing needs: Agile allows for flexibility and quick adaptation to changing project requirements or client needs.
- Cost-effective changes: Adjustments to the scope or requirements are more cost-effective and manageable.
- Encourages collaboration: Agile fosters communication and collaboration between team members and stakeholders, ensuring that the project aligns with user needs.
- Promotes iterative development: Development is divided into short sprints or iterations, allowing incremental progress and quick delivery of functional components.
- Early delivery of results: Operational and functional components of the software are delivered early and periodically, providing continuous value to stakeholders.
- Supports changing requirements: Agile embraces changing priorities and ensures the system evolves as needs arise.
- Continuous testing and feedback: Agile ensures continuous integration, testing, and feedback, reducing risks and improving the system's quality over time.
- Focus on working software: Agile emphasizes delivering a working product over extensive documentation, ensuring more time is spent on development and meeting project goals.
- Improved customer satisfaction: Frequent deliveries and feedback cycles ensure the product meets user expectations and improves satisfaction.

By using the Agile model, we aim to efficiently manage our project, ensuring high-quality outcomes while meeting the dynamic requirements of our stakeholders.

We used Unified modeling language (UML) for the following purpose:

- ➢ To develop diagrams and provide users with ready-to-use expressive modelling examples.
- ➢ It can be used for modelling a system independent of a platform language

- ➢ To present proposed designs and communicate with stakeholders.
- ➢ To understand requirements and etc.(Carter, L., and Belanger, (2005)) .

Generally, we used requirement collection, analysis, designs and implementation of the proposed system as follows:-
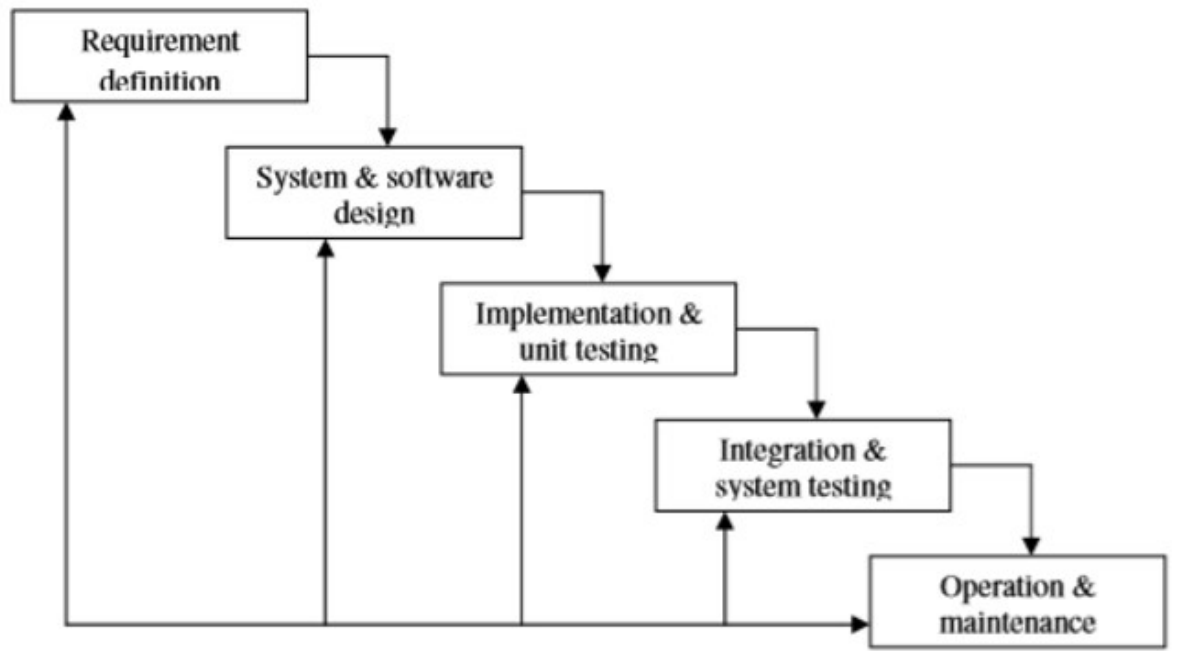
Figure 1. 1 simplified model for designing and implementation

1.6.3 Implementation Methodology

To develop our project, we select appropriate coding method called OOP which is a programming paradigm based on the concept of "objects", which may contain data, in the form of fields, often known as attributes; and code, in the form of procedures, often known as methods.

We select OOP coding approach because of the following its advantage:

- Code reuse and Recycling: objects created for object-oriented programs can easily be reused in other programs.
- Encapsulation: once an object is created, knowledge of its implementation is not necessary for its use.

- Design benefit: Since our project may have large programs are very difficult to write. So, using OOP force us to go through an extensive planning phase, this makes for better designs with less flaws. In addition, once a program reaches a certain size, object-oriented programs are actually easier to program than non-object-oriented ones.
- Software maintenance: - object oriented program is much easier to modify and maintain than a non- object oriented program.

The Implementation Methodology outlines the step-by-step process for developing and deploying the Legal Aid Case Management System (LACMS). This phase translates the design and requirements into a functional system by focusing on efficient coding, configuration, integration, and deployment. The implementation methodology emphasizes modular development, iterative testing, and collaboration to ensure the system is reliable, scalable, and user-friendly

Modular Development

The system will be developed in a modular fashion, dividing the project into smaller, manageable components to streamline the implementation process. key phases of implementation.

> Development: each module will be developed independently to ensure modularity and reusability.

Core functionalities include

Case Registration: Digitalize client and case data input.

Case Assignment: Allocating cases based on faculty availability and specialization.

Workload Management: Balancing and tracking case distribution.

Reporting System: Generating detailed reports on case progress and faculty workload.

Data management

> Integration: Connecting individual modules (backend and frontend).

Protecting sensitive data and secure access control mechanisms.

> ➢ **Iterative Development**

It involves: Incremental Development, Collaboration, and Continuous Testing

1.6.4. Testing Methodology

Testing ensures that the Legal Aid Case Management System meets functional, performance, and security requirements. A combination of manual and automated testing approaches is applied to achieve robust system validation

Types of Testing

1. Unit Testing: Testing individual components, such as case registration, assignment, reporting features and database in isolation. Unit testing is a fundamental practice in software development that enhances code quality, simplifies debugging, and supports agile methodologies by allowing for continuous integration and delivery. It is an essential part of a robust testing strategy (Duy, 2019).

2. Integration Testing: Verifying communication between modules, such as the backend API and frontend interface. Ensuring data flows seamlessly from user input to database storage and reporting

3. Functional Testing: focus on validating the functional requirements mainly ensures that the system performs as expected and meets the users' needs

- Validating the system's features against the requirements:
- Proper handling of case assignment.
- Accurate workload distribution.
- Notification delivery to faculty

4. User Acceptance Testing (UAT):
Coordinators and faculty members test the system in a real-world environment. feedback guide final adjustments before deployment.

1.6.5. Development Environment and Programming Tool

The development environment and tools have been carefully selected to ensure smooth system creation, scalability, and maintainability.

✦ Development Environment

1.Hardware Requirements

- ➤ Development Machines
- ➤ Processor: Quad-Core (Intel i5 or higher).
- ➤ RAM: 8GB minimum.
- ➤ Storage: SSD with at least 256GB free.

2.Software Requirements:

- ➤ Operating System: Windows 10/11

Programming Tools and Technologies

We utilize Lucidchart and Microsoft Visio to design a variety of professional UML diagrams, including sequence diagrams, use case diagrams, class diagrams, and other specialized representations. These tools enable us to accurately model system interactions, define functional requirements, and outline structural designs with precision and clarity, supporting effective communication and collaboration throughout the software development process.

Programming Languages

1. Primary Programming Languages & Frameworks:

TypeScript (.ts and .tsx files):

TypeScript is chosen as the primary programming language because it offers static typing, which helps catch errors during development rather than runtime (Fenton, 2018). This improves code quality and maintainability, especially for our large-scale project the Dilla University Legal Aid Service Management System. TypeScript's compatibility with JavaScript ensures seamless integration with the ecosystem, while its robust tooling, such as IntelliSense and error detection, accelerates development and reduces debugging time.

React.js (with Next.js framework):

React.js, combined with Next.js, is selected for its efficiency in building dynamic user interfaces and server-side rendering (SSR). React's component-based architecture allows for the reuse of UI components, making the codebase more modular and maintainable. Next.js enhances React by providing SSR and static site generation (SSG), which improves performance. Additionally, Next.js simplifies routing, API integration, and project scalability, making it an ideal choice for this web-based system.

Prisma (for database ORM):

Prisma is use as the database ORM because it abstracts database interactions, providing a type-safe and developer-friendly experience. It automates query generation and schema migrations, reducing the risk of errors and speeding up development. Its integration with TypeScript further ensures type safety, and its support for various databases, including MongoDB, makes it flexible and adaptable to the project's requirements.

2. Frontend Technologies:

TailwindCSS (for styling):

TailwindCSS is use for its utility-first approach to styling, which eliminates the need for writing extensive custom CSS (Kodali, 2024). It speeds up the development process by offering pre-defined classes for styling elements directly in the markup. TailwindCSS also ensures consistency across the application while maintaining flexibility to customize designs. Its small footprint and optimized CSS output enhance performance.

Framer Motion (for animations):

Framer Motion is chosen for animations because of its simplicity and powerful API for creating smooth, responsive, and interactive animations.

React Icons (HeroIcons):

React Icons use to incorporate scalable, consistent, and visually appealing icons into the interface. It simplifies the process of adding icons by offering a collection of pre-designed vector

icons, including HeroIcons, which align well with modern design trends. React Icons ensures responsiveness and integration without compromising performance.

React Hot Toast (for notifications):

React Hot Toast is for notifications because of its lightweight nature and ease of implementation. It provides a customizable and responsive notification system, allowing real-time feedback to users. Its compatibility with React ensures smooth integration and minimal overhead in the application

3. Backend Technologies:

Next.js API Routes:

Next.js API Routes use to handle backend logic and API endpoints within the same framework, reducing the need for separate backend infrastructure. This feature simplifies deployment and ensures faster communication between the client and server. It also provides built-in support for middleware and authentication, enhancing security and scalability

MongoDB (as indicated by Prisma schema):

MongoDB chosen for its flexibility in handling unstructured or semi-structured data. As a NoSQL database, it allows for dynamic schema updates, which is essential for evolving projects. Its JSON-like document structure makes it intuitive for developers and well-suited for modern web applications. MongoDB's integration with Prisma ensures seamless schema management and efficient database queries.

NextAuth.js (for authentication):

NextAuth.js use for authentication due to its easy integration with Next.js and support for multiple providers, such as Google and GitHub. It simplifies session management and ensures secure handling of user credentials. NextAuth.js is highly customizable, enabling role-based access control and protecting sensitive user data with minimal configuration.

4. Development Environment:

Node.js environment:

Node.js chosen for its asynchronous, event-driven architecture, which is well-suited for handling real-time data and API requests. Its vast ecosystem of libraries and tools accelerates development, and its performance ensures scalability. Using Node.js allows for a unified JavaScript environment for both the frontend and backend, reducing complexity.

Next.js 13+ (using App Router):

Next.js 13+ with the App Router chosen for its advanced routing capabilities and improved developer experience. The App Router introduces features like server components, nested layouts, and parallel data fetching, enhancing performance and scalability. This version simplifies complex application structures while maintaining flexibility, making it ideal for building a robust and modern web-based system.

1.7 Feasibility Study

1.7.1 Technical Feasibility

The entire group members are expected the system to be technically feasible. The system is going to be developed by following the Agile development technique. The team has the ability to develop this system without any difficulty since the team has studied the required methodologies and tools. So the system will be technically feasible. The project member also have knowledge about PHP, Java script, CSS, MySQL database and enterprise architect to design the system. Besides, the group members have enough capability to develop the project. So the system is technically feasible. In general using the different techniques, we can perform the work up to the desire of the organization, even though fulfilling human being's need in all direction may be difficulty.

1.7.2 Operational Feasibility

Operational feasibility is a measure of how well a new system solve the problems, how the system operates with existing resource and how much our system easy to operate by different users.

The system has definitely a positive impact on users as they involved in the development of this project by providing their feedback and adapt easily how it will support their day to day business operation.

We follow each step how the existing system is working and our technical skill transform the existing system into a web based one. The new system is also compatible to any web browser. Based on these points our new system is operational feasible.

1.7.3 Financial Feasibility

This is to determine the benefit and savings that are expected from a new system and compare them with costs of time. If benefits outweigh costs, then the decision is made to design and implement the system.

The new system is economically feasible because the new system uses software and hardware tools that are available by low cost or even for free (opensource software), so compared to the benefits it is going to get by implementing the system that eliminates the time and cost consuming process of traditional Legal aid service management system by providing a fully automated way of web based management system.

1.7.4 Time Feasibility

Schedule feasibility determines whether the proposed system will be completed on the given time or not. Whatever the scarcity of time given for the project by the internal motivation and potential of the team member of the project, we surely expect the project will be completed on time.

1.8 Benefit of the project

The proposed system for DU Legal Aid Service is designed to address inefficiencies in manual processes, improve service delivery, and enhance operational transparency. below are the key benefits of the project.

1. Improved Efficiency in Case Management: Automates the process of registering, assigning, data management and tracking legal aid cases

- Reduces delays caused by manual documentation and paperwork.

- Provides coordinators and faculty members with an organized system to manage workloads effectively

2. Real-Time Notifications and Communication

- Provides notifications to assigned coordinators and legal professionals, ensuring timely follow-ups.
- Facilitates communication between clients, coordinators, and faculty members through a centralized platform.

3. Accurate Reporting and Data Management

- Maintains a centralized database for all case records, making it easier to generate detailed reports.
- Ensures accuracy and security of data, reducing the risk of errors and data loss associated with manual systems.

4. Long-Term Impact: Strengthens the capacity of DU Legal Aid Service to provide high-quality, reliable legal assistance.

CHAPTER TWO

EXISTING SYSTEM

2.1 Introduction

The analysis for this project involved a detailed study of the current manual system used by the Dilla University Legal Aid Service (DULAS). The existing system is entirely non-computerized, with all operations, including case registration, tracking, reporting, and storing handled manually.

Analyzing the existing system ensures the development of a new, automated software system that delivers efficiency, effectiveness, accuracy, and reliability in managing legal aid services. This system aims to address the challenges posed by manual processes, such as delays, data mismanagement, and communication gaps.

The goal of conducting a comprehensive and thorough analysis of the current manual operations is to gather critical data and insights. These findings will assist significantly in designing and implementing a modern, web-based system for DULAS. The new system will streamline case management, improve workload distribution, enhance reporting accuracy, and ensure equitable access to justice for marginalized communities.

2.1 Description of the Existing System

The Dilla University Legal Aid Service (DULAS) is a transformative initiative designed to bridge the justice gap for financially disadvantaged and vulnerable members of the community. Established under the auspices of the Dilla University School of Law and supported by the United Nations, DULAS embodies the principles of access to justice and equity, offering free legal services to those who cannot afford representation.

Currently, DULAS operates across five centers, including its primary hub in Dilla. Each center is overseen by a dedicated coordinator who plays a crucial role in assessing cases and forwarding them to the Dilla University School of Law. The law school then assigns cases to specialized faculty members based on their expertise, workload, and availability, ensuring efficient and professional handling of each legal issue.

By providing meticulous and personalized attention to each case, DULAS delivers meaningful support to underserved communities while fostering an inclusive and accessible justice system. This initiative not only addresses immediate legal needs but also contributes to the broader mission of social equity by empowering individuals and strengthening community trust in the legal process. DULAS stands as a beacon of hope and justice, exemplifying the values of compassion, professionalism, and a steadfast commitment to serving society. Through its structured operations and impactful services, DULAS continues to create lasting change in the lives of many, while upholding the highest standards of legal advocacy and support.

2.3 Limitation of existing system

It is crucial to modernize operations in today's fast-paced world to address the challenges associated with manual processes. The current manual system at Dilla University Legal Aid Service (DULAS) presents several challenges and inefficiencies, including:

- **Task Delay:** Registering cases, assigning legal professionals, and tracking progress are time-consuming, leading to delays in service delivery.
- **Data Inconsistency:** Data Inconsistency in the existing system refers to the lack of uniformity and reliability in maintaining records, which can lead to the following issues: Duplicate Records, Conflicting Information, Missing Data, Delayed Access to Information, Errors in Reporting
- **Data Redundancy:** Data Redundancy occurs when duplicate entries of the same data are found across different locations, such as both in branch offices and the main office. This redundancy is common in manual systems where data is recorded separately at each location without a centralized system to synchronize and manage it: Duplicate Records, Missing Data, Delayed Access to Information, Errors in Reporting. To handle data redundancy, implement a centralized database, ensuring a single source of truth. Real-

time data synchronization will be enabled to maintain consistent and up-to-date information across all locations.

- **Inaccurate Information:** Errors in manually handled data reduce the accuracy of case records and reports.

- **Inability to Analyzing Historical Data:** Tracking case outcomes, faculty performance, and workload history is difficult, hindering decision-making and strategic planning.

- **High Operational Costs:** The day-to-day activities of the current manual system are resource-intensive, requiring significant time, paper, and effort.

- **Accessibility Challenges:** Retrieving case files or relevant information is time-consuming and cumbersome due to disorganized, paper-based storage.

- **Workload Imbalance:** Manual assignment of cases to faculty members often results in unequal distribution of work, creating inefficiencies.

- **Communication Gaps:** Notifications and updates rely on informal methods, causing delays and errors in case handling and communication with clients.

An automated system for managing legal aid services is a much more efficient, reliable, and cost-effective alternative to the current manual system. It would eliminate many of the underlying problems, and improve service delivery

2.4 Model of Existing system

The existing DULAS (Dilla University Legal Aid System) operates manually. The system involves actors performing repetitive tasks such as client registration, case assignment, document handling, and decision recording. These tasks lack automation, leading to inefficiencies, delays, and human errors.

2.4.1 Actors

In the existing system, actors rely on manual operations to perform their tasks, which involve handling physical records, communicating in person, and using paper-based documentation. This process is often time-consuming, prone to human error, and inefficient, as it lacks automation and centralized data management. Without the support of a digital system, these actors face

challenges in maintaining consistency, accuracy, and seamless collaboration. The actors are listed below:

Client/Beneficiary:

refers to individuals seeking legal aid from the Dilla University Legal Aid Service (DULAS). These individuals typically require assistance in navigating legal challenges, often because they lack the financial means to hire private legal services. Clients approach the system to receive advice, representation, or guidance on legal matters. Their role involves submitting requests for legal services and providing necessary details about their cases. Clients also collaborate with coordinators or faculty members during the process, answering questions or attending meetings as needed. They rely on the system to address their legal issues professionally and promptly.

Coordinator:

acts as a mediator between clients and the legal system. They are stationed at various centers and are responsible for receiving and reviewing legal aid requests. Coordinators assess the validity and complexity of the cases submitted by clients. In cases where the issues are manageable, the coordinator provides legal advice or services directly. However, for more complex cases, they forward them to the School of Law for further handling by faculty members. Coordinators also maintain records of cases and ensure that clients are informed about the progress of their requests. Their role is critical in managing communication and ensuring smooth case handling within the system.

Faculty Member (Lawyer):

consists of legal professionals from the Dilla University School of Law who provide specialized expertise for complex legal cases. Their responsibilities include reviewing forwarded cases, offering legal consultations, drafting legal documents, and representing clients when necessary. Faculty members rely on the details and background information provided by the coordinator to address cases efficiently. They work closely with coordinators to resolve cases and may collaborate on follow-up actions, ensuring that legal aid is delivered professionally and effectively.

Administrator/Manager:

is responsible for overseeing the entire operation of the legal aid service. This individual ensures that coordinators and faculty members fulfill their roles efficiently and that resources are allocated appropriately. The administrator supervises the workflow of legal aid requests, monitors the performance of different centers, and resolves any challenges encountered during case handling. Additionally, they view reports, track the outcomes of legal aid efforts, and provide strategic guidance to improve the services offered. The administrator serves as the central figure in ensuring that the legal aid system functions cohesively and aligns with its mission to serve clients effectively.

2.4.2 Essential use-case

Use case diagrams are used to show graphically for the interaction between the system and users. In other words, they graphically describe who will use the system and in what ways the user expects to textually describe the   sequence of steps of each interaction.

The UML notation for a use case diagram is shown on, in which

- An oval represents a use case:- Represents a specific functionality or service provided by the system. Use cases describe the interactions between actors and the system.

- A line between an action and a use case represents that the actor initiates and/or participates in the process

- A stick figure represents an actor: - Represents a user or another system that interacts with the system being modeled. Actors can be primary (initiating an interaction) or secondary (receiving a response).
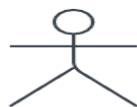
Extend: Indicates that a use case can extend the behavior of another use case under certain conditions.

<<Extend>>------------->

Figure 2. 1 Use-case Diagram For Existing System

This use case diagram for Existing Dilla University Legal Aid Service system as in the above Figure 2.1 illustrates the interactions between different actors. The diagram effectively showcases how tasks are distributed among the actors and how the system facilitates coordination and service delivery. The Client is responsible for requesting services, providing information, contacting the coordinator, and tracking the progress of their case. The Coordinator handles multiple responsibilities, including registering clients, approving or rejecting applications, notifying appointments, providing legal aid services, generating reports, and requesting assistance from faculty members. Faculty Members contribute by collaborating with coordinators and providing legal aid services as needed. The Admin has a more centralized role, which includes assigning coordinators (either permanently or on a project basis), viewing and managing reports, users, coordinators, centers, and faculty members, as well as tracking the overall progress of the system's operations.

2.5 Business rule

The organization has the following principles in the existing system which includes:

BR1: Only authorized legal officers and advocates can access confidential case files. Unauthorized personnel or clients cannot access sensitive documents directly.

BR2: Advocates are responsible for guiding clients in all cases. Clients must provide complete and accurate documentation before their case can proceed.

BR3: Only dis advantageous clients can access legal aid services. Clients must present a valid registration proof.

BR4: Legal officers must ensure that cases are assigned to advocates based on their area of specialization (e.g., civil law, criminal law, etc.) to ensure efficient case handling.

BR5: Advocates and legal officers must adhere to ethical guidelines, such as providing unbiased advice and maintaining confidentiality about clients' cases.

BR6: Clients seeking legal aid doesn't pay any associated fees in advance if they qualify based on pre-approved criteria, such as financial difficulty.

BR7: Clients who fail to attend scheduled appointments without prior notice may have their cases delayed or canceled, requiring re-registration to continue.

BR8: Feedback on the service quality must be documented and periodically reviewed by administrators to improve the overall service delivery.

BR9: Any dispute resolution handled by DULAS must comply with Ethiopian legal frameworks and the ethical standards of Dilla University.

3. CHAPTER THREE

PROPOSED SYSTEM

3.1 Overview of The Proposed System

The proposed Dilla University Legal Aid Service (DULAS) Management System is designed to address the shortcomings of the current manual system by introducing a web-based platform that ensures efficiency, accuracy, and accessibility for all stakeholders involved. The new system aims to replace the manual processes with an automated solution to meet the increasing demand for streamlined legal aid services. The new system will provide the following functionalities and features:

- Client Registration and Case Management

Coordinators can register new client information efficiently and store case details in the system's secure database. This ensures easy access to records and minimizes the risk of losing or mismanaging case files.

- Case Assignment

The system will allow the School of Law to assign cases to lawyers based on their expertise and availability. Notifications will be sent to the assigned lawyer with all the case details.

- Real-Time Notifications

Clients, coordinators, and lawyers will receive updates about case progress, appointment dates, and deadlines through the system's notification feature, ensuring clear communication.

- Efficient Reporting Mechanisms

Coordinators can generate daily, weekly, or monthly reports about client activities, court schedules, and case statuses. These reports will assist in organizational decision-making and resource management.

- Secure Data Storage and Accessibility

Utilizing Google Cloud for data storage ensures scalability, reliability, and robust security. Only authorized users with appropriate access rights will be able to interact with sensitive data.

🞣 Enhanced User Experience

The system will be user-friendly, providing a simple interface for coordinators, lawyers, and clients to interact with. It will support efficient searching, updating, and deleting of records to ensure accuracy and easy data modification.

🞣 Legal Case Tracking

Clients will be able to track their cases in real-time, and increasing transparency in the legal process.

🞣 Support for Organizational Growth

By automating repetitive tasks, the system will save time and reduce costs, allowing the organization to focus more on providing legal aid services and expanding its outreach.

3.2 Requirement specification

In requirement specification there are two main ideas defined or explained functional requirement (what things are performed by the system that we developing) and non-functional (what are system aspects). So, it consists two parts: - Functional and Nonfunctional Requirements.

3.2.1 Functional Requirement

Functional requirements are intended behaviors of the system. These behaviors include statements of services the system should provide, how the system should react to particular inputs and how the system should behave in particular situations. It specifies the software functionality that the developers must build into the product to enable users to accomplish their tasks.

The following are some functionalities of the system.

🞣 Data Management

The system will properly handle and store data in a secure database to ensure that documents do not fade or get lost. All client and case details will be preserved for easy retrieval.

- Efficient Search Mechanism

The system will enable coordinators and administrators to quickly search for client or case information

- Case Assignment

The system facilitates assigning cases to lawyers. The School of Law will review case requirements and assign them to available lawyers based on expertise and workload

- Appointment Scheduling

Coordinators will be able to set up appointments for clients and notify them about appointment schedules through the system.

- Feedback Mechanism

Clients will be able to provide feedback

- Report Generation

This will generate detailed reports on case statuses, client activities, and overall service delivery.

- Record Keeping

The system will allow authorized users to record decisions, notes, and outcomes of cases securely. This includes uploading and managing evidence files.

- Client Information Registration

The system enable coordinators to register details about clients, such as full name, sex, age, case type, and case description, and other related information.

- User Account Management

The system allow administrator to add, modify, and delete user accounts. This includes updating client, coordinator, or lawyer details when necessary.

➕ Access Control

The system enforces strict access control measures. Only authorized users with the appropriate privileges can access sensitive information or perform specific tasks.

➕ Password Management

Users able to change their passwords securely through the system.

➕ Real-Time Notifications

Notifications about case updates, appointments, and assignments will be sent to coordinators, lawyers, and clients.

➕ Web Access

Clients able to access general case-related information and updates through the DULAS website, ensuring transparency and ease of use

3.2.2 Non-functional requirement

Non-functional requirements, as the name suggests, are requirements that are not directly concerned with the specific services delivered by the system to its users. They may relate to evolving system properties such as reliability, response time and performance. Alternatively, they may define constraints on the system implementation such as the capabilities of input output devices or the data representations used in interfaces with other systems. Non-functional requirements specify characteristics of the system as a whole. It describes aspects of the system that are concerned with how the system provides the functional requirements.

1. Security

The system ensures only authorized users can log in using unique credentials (username and password) assigned based on roles (clients, coordinators, administrators, and lawyers).

2. Reusability

The system components, such as modules for client registration, case assignment, and report generation, are designed to be reusable for future expansions or similar applications.

3. Performance

The system will operate 24/7, ensuring users can access services at any time. And the response time for retrieving or updating case information will not exceed a few seconds under normal operation.

4. Reliability

Proper data storage and robust error handling mechanisms will enhance the system's reliability. Users can confidently depend on the system for accurate and uninterrupted service.

5. No Redundancy

Data redundancy is eliminated through a normalized database structure, ensuring that no duplicate information exists in the system.

6. Availability

The system will ensure all records, including client details, case histories, and reports, are readily available to authorized users whenever needed.

7. Efficiency

Resources such as storage, memory, and processing power are optimized to ensure efficient system operations. the system minimizes the time and effort required to process client requests or update case information.

8. User-Friendly Interface

The interface will be intuitive and simple to navigate, enabling users to easily input or retrieve case-related information. And Accessibility features, such as clear labels and prompts, will be included to accommodate diverse users.

3.2.3 System Requirement

System design is the transformation of an analysis model into a system design model. During system design, developers define the design goals of the project and decompose the system into smaller subsystems that can be realized by teams.

we have identified the functional and non-functional requirements of the system and produced the analysis model. The following are discussed design goals, system architecture, system decomposition, and deployment view

Design Goal

The design goals are derived from the nonfunctional requirements. It guides the decisions to be made by the developers when trade-off is needed. They describe the qualities of the system that developers should optimize. Design goals are grouped into four categories. These are

- Maintainability
- Performance
- Dependability
- End User

1.Maintenance

The system should be easily extensible to add new functionalities such us fatal death vital event, at a later stage. It should also be easily modifiable to make changes to the features and functionalities.

2.Performance

Response time of the system will be high with maximum throughput. It will support multiple users at a time. It will also save storage space; the system shouldn't take much space in memory.

3.Dependability

The vital event registration offices need the system to be highly dependable. The system should be robust and fault tolerant. Additionally, the system is handling sensitive data of vital event registration offices, high emphasis should be given with regards to security, as there are subsystems to be accessed through web.

4.Security

 For security issue, only authenticated or authorized user can visit the system. An authorized user can create, update, delete, maintain the system and do other work. But customers can give their comment without login to the system. Knowing website is enough to give comment.

5.Usability

is the extent to which a product can be used by any user to achieve specified goal with effectiveness, efficiency and satisfaction in specified context of use from the end user perspective

Architectural Design

The proposed software has three- tier architecture.

The presentation tier: is the top most level of the application. It is the one the clients directly interacted. It provides GUI to allow the client gaining access of the system.

Logical tier/ middle tier: It accepts inputs from the client and performs detailed processing. It is a bridge between data access tier and presentation tier.

Data access tier: provides data persistence mechanism and storage to the data. It consists of a mechanism to access the database without installing data base dependent drivers and libraries on the client device
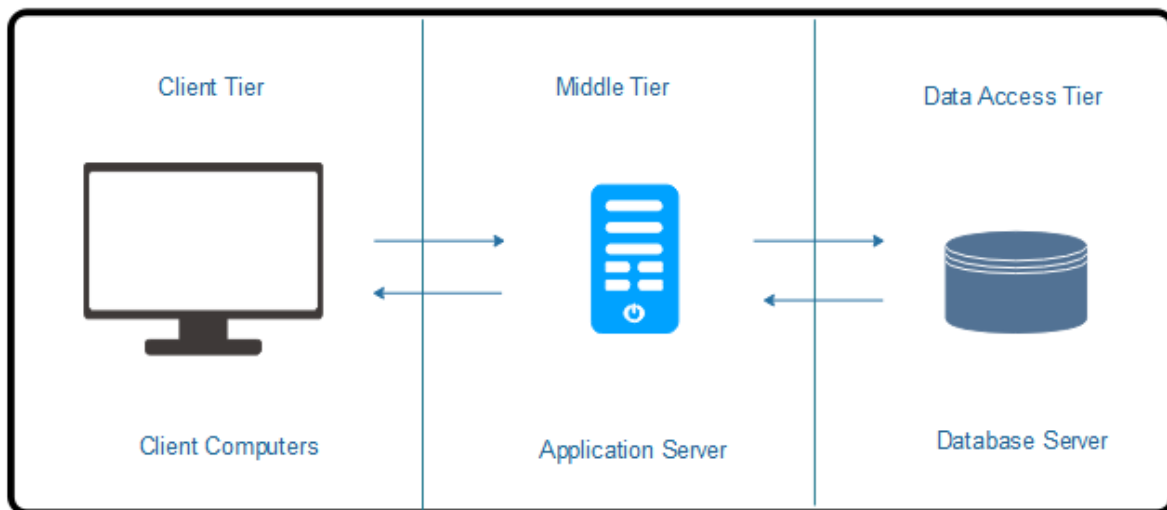

Figure 3. 1 Architectural style

CHAPTER FOUR

SYSTEM MODELING

4.1 Introduction

Under this topic we use different kinds of UML diagrams to model the functionality, structure and Sequence of activities of the system. In this chapter the major activities performed or identified are: modeling the function of the use case, identifying actors, identifying use case, constructing use case model and use case scenarios and finally designing of user interface.

System modeling involves the evaluation of system components in relationship with one another to determine their requirements and how to satisfy them. Some system modeling tools are employed during the course of this project that support development tasks, from analysis to design, then to implementation. This is represented with the use of the sequence diagram, use case diagram, activity diagram, state chart diagram, and class diagram for the Dilla University Legal Aid Service Management System.

4.2 Use case diagram

Use Case

A use case is used during requirement elicitation and analysis to represent the functionality of the system. Use case focus on the behavior of the system from an external point of view. A use case describes a sequence of actions that provide a measurable value to an actor. In other words, it shows a way in which a real world actor interacts with the system. An essential use case is a simplified, abstract, generalized use case that captures the intentions of the user in a technology and implementation independent manner. In order to identify use cases the team examined the needs of users, the main tasks of users, the information user needs to examine, create or change, what users inform the system and what the system informs users.

An Actor is something that accomplish use cases upon a system. It is just an entity, meaning it can be a Human or other system that directly play an external role in the system.

Name: System admin

Description: which are responsible for controlling and auditing system

Role: Manage the system and control users

Name: Client

Description: Individuals seeking legal aid services, including consultation and representation for their cases.

Role: Register their cases with the system, receive consultations, follow the progress of their cases.

Name: Coordinator

Description: Oversees and manages client cases in their designated center.

Role: Handle client registrations, assess cases, provide initial legal assistance, escalate complex cases to the law school, and collaborate with specialized faculty members for advanced case resolution.

Name: Faculty Member

Description: Legal experts from the Dilla University School of Law assigned to handle complex cases in collaboration with coordinators.

Role: Provide specialized legal support for escalated cases, collaborate with coordinators for case resolution, and ensure professional and accurate handling of complex legal issues based on their expertise and workload.

A use case diagram illustrates a set of use cases for a system, the actors of these use cases, the relations between the actors and these use cases, and the relations among the use cases.

Name: Kebele Member

Description: Responsible for determining if a client is genuinely economically disadvantaged and notifying the coordinator.

Role: Verify clients' economic status and notify the coordinator about their eligibility.

Description for Use case

The Use Diagram represents for the Dilla University Legal Aid Service Management System, showcasing the interactions between different actors and the system along with its functionalities. The main actors involved are clients, coordinators, faculty members, kebele staff, and administrators.

Clients interact with the system by registering, submitting service requests, adding necessary information, uploading documents, and viewing their appointments and notifications. They also receive updates and notifications from coordinators or faculty members regarding their cases. Coordinators are central to the system's functionality as they manage service requests by categorizing cases, approving or rejecting them, and notifying relevant parties. They provide legal aid, communicate with faculty members, collaborate with kebele staff, and generate reports to track progress within the system.

Faculty members contribute to the system by collaborating with coordinators to provide legal services and maintaining communication with other stakeholders to ensure efficient service delivery. Kebele staff, on the other hand, have a more specific role. They assess whether a client is genuinely economically disadvantaged and, based on their findings, notify the coordinator. They do not handle broader client registration or system management tasks.

Administrators manage the overall system, handling user accounts by adding, updating, or deactivating users, assigning coordinators and faculty members to cases, tracking progress, viewing reports, and maintaining system backups. They also oversee login functionalities, including verifying accounts and managing password changes.
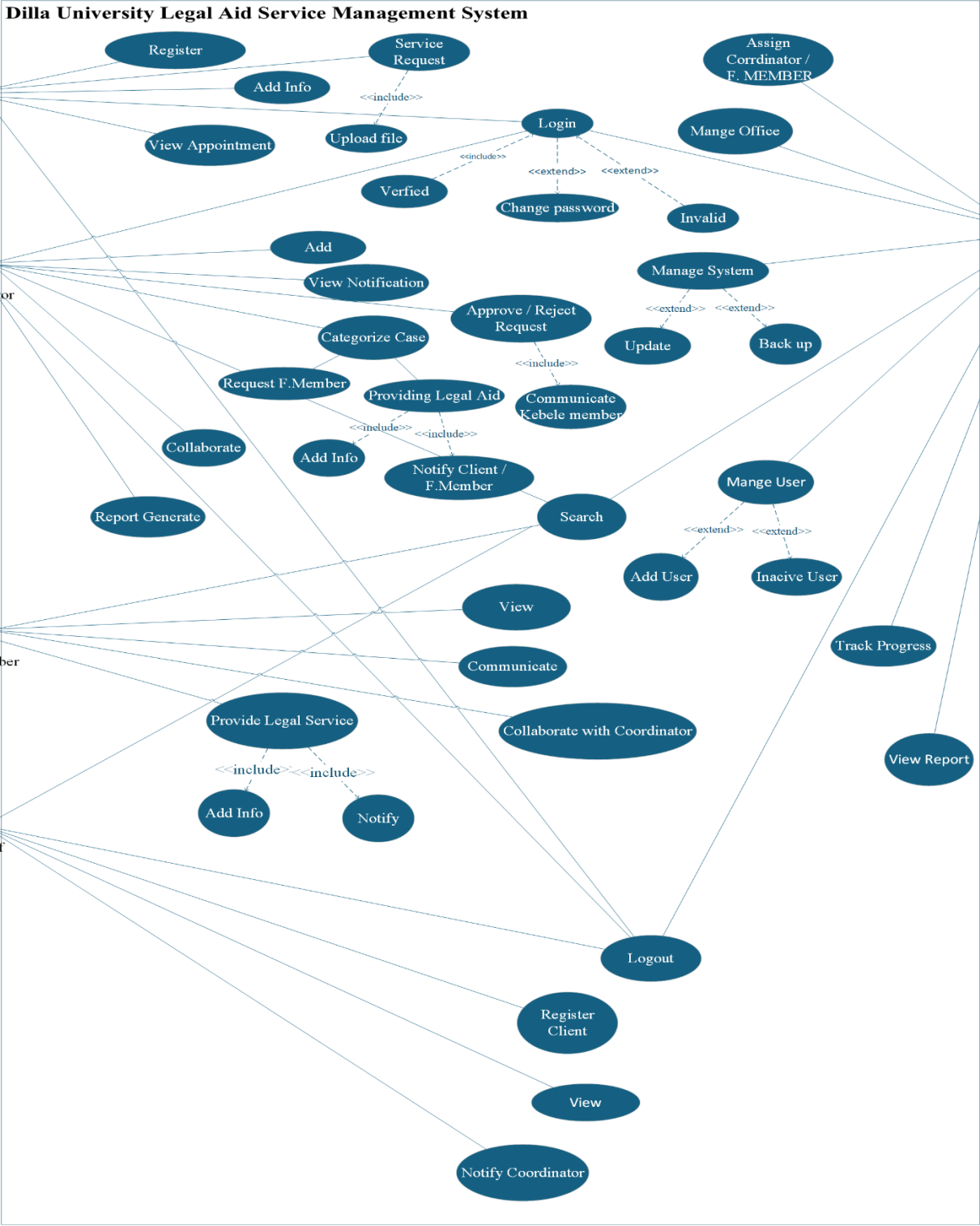
Figure 4. 1 Use case diagram

As shown in the above Figure 4.1 use case diagram the description of each actors name, roles who involved on the system are represented in detail on the following tables.

Table 1: Register

Description 1	
Use case name	Register
Use number	1
Participating actor	Client
Flow of event	1. The system displays the registration form including username and password form. 2. Fill the form and click register button. 3. User register in to the system
Post condition	The system saves the entered data into database.
Alternative flow-of events	If he/she fills miss forms it will display error message.

Table 2: Login

Description 2	
Use Case Name	Login
Use Case Number	2
Use Case Description	Authentication
Participating Actor	Client, coordinator and Staff Member
Flow of Event	1. The system displays the login form including username and password form. 2. Fill the form and click login button. 3. User login in to the system
Post Condition	The system saves the entered data into database.
Alternative Flow-of-events	If he/she fills the username or the password it will display error message.

Table 3: view

Description 3	
Use Case Name	view
Use Case Number	3
Use Case Description	Enables users to view data or files.
Participating Actor	All user
Flow of Event	1. User logs in.
	2. User selects the item to view.
	3. Data is displayed.
Post Condition	Data is displayed correctly.
Alternative Flow-of-events	If data is unavailable, an error shown

Table 4: Add Information

Description 4	
Use Case Name	Add Information
Use Case Number	4
Use Case Description	Enables users to add necessary information.
Participating Actor	Client, Coordinator and faculty member
Flow of Event	. User logs in.
	2. User selects "Add Information."
	3. User enters data and submits.
Post Condition	Information is stored in the database.
Alternative Flow-of-events	If required fields are missing, an error is displayed.

Table 5: Manage Case

Description 5	
Use Case Name	Manage Case
Use Case Number	5
Use Case Description	Manages case records in the system.
Participating Actor	Coordinator, Faculty Member
Flow of Event	1.user logs in 2.user selects a case 3.user update or close the case
Post Condition	Case details are updated in the database.
Alternative Flow-of-events	If case is invalid, an error message is shown.

Table 6: Manage Items

Description 6	
Use Case Name	Manage Items
Use Case Number	6
Use Case Description	The system can lead to manage things
Participating Actor	coordinator
Flow of Event	1. User logs in. 2. User selects a item.
Post Condition	The system saves the entered data into database.
Alternative Flow-of-events	If case is invalid, an error message is shown

Table 7: upload

Description 7	
Use Case Name	upload
Use Case Number	7
Use Case Description	Allows users to upload files to the system.
Participating Actor	Client, Coordinator, Faculty Member
Flow of Event	1. User logs in. 2. User navigates to the "Upload File" section. 3. User selects a file from their device. 4. User clicks the "Upload" button. 5. System saves the file to the server.
Post Condition	The file is successfully stored in the system and linked to the appropriate record.
Alternative Flow-of-events	If the file format or size is invalid, an error message is displayed.

Table 8: Manage user

Description 8	
Use Case Name	Manage user
Use Case Number	8
Use Case Description	Admin manages users (add, block, delete).
Participating Actor	Admin
Flow of Event	1. Admin logs in. 2. Admin selects "Manage Users." 3. Admin performs actions.
Post Condition	User actions are updated in the database.
Alternative Flow-of-events	If action fails, an error message is displayed.

Table 9: Update Client

Description 9	
Use Case Name	Update Client
Use Case Number	9
Use Case Description	Allows the coordinator to update client information
Participating Actor	coordinator
Flow of Event	1. Coordinator logs in. 2. Coordinator selects a client from the list. 3. Coordinator updates the client's information (e.g., name, contact, case details). 4. Coordinator saves the changes.
Post Condition	The updated client information is saved in the database.
Alternative Flow-of-events	If the client record does not exist or required fields are missing, an error message is displayed.

Table 10: Report

Description 10	
Use Case Name	Report
Use Case Number	10
Use Case Description	Generates reports for system data.
Participating Actor	Admin, Coordinator
Flow of Event	1. User logs in. 2. User selects "Generate Report." 3. Report is displayed.
Post Condition	Report is generated and available for download
Alternative Flow-of-events	If no data exists, a massage is displayed

Table 11: Logout

Description 11	
Use Case Name	Logout
Use Case Number	11
Use Case Description	Ends the session for the user.
Participating Actor	All users
Flow of Event	1.user clicks logout 2.session ends
Post Condition	User is logged out of the system.
Alternative Flow-of-events	None

4.3 Sequence Diagram

- A sequence diagram is an interaction diagram that shows how objects operate with one another and in what order. It is a construct of a message sequence chart.

- A sequence diagram shows object interactions arranged in time sequence. It depicts the objects and classes involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the Logical View of the system under development.
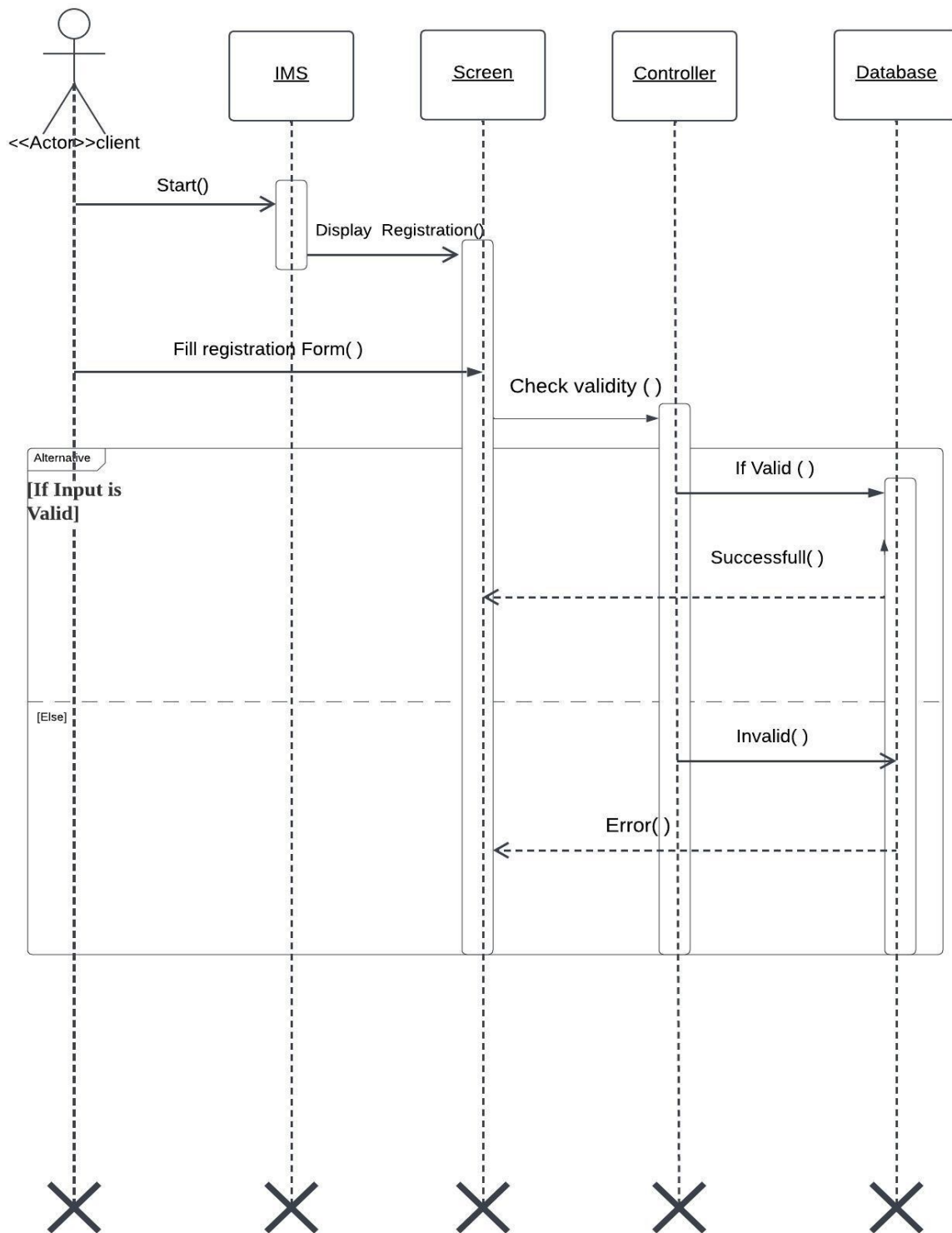
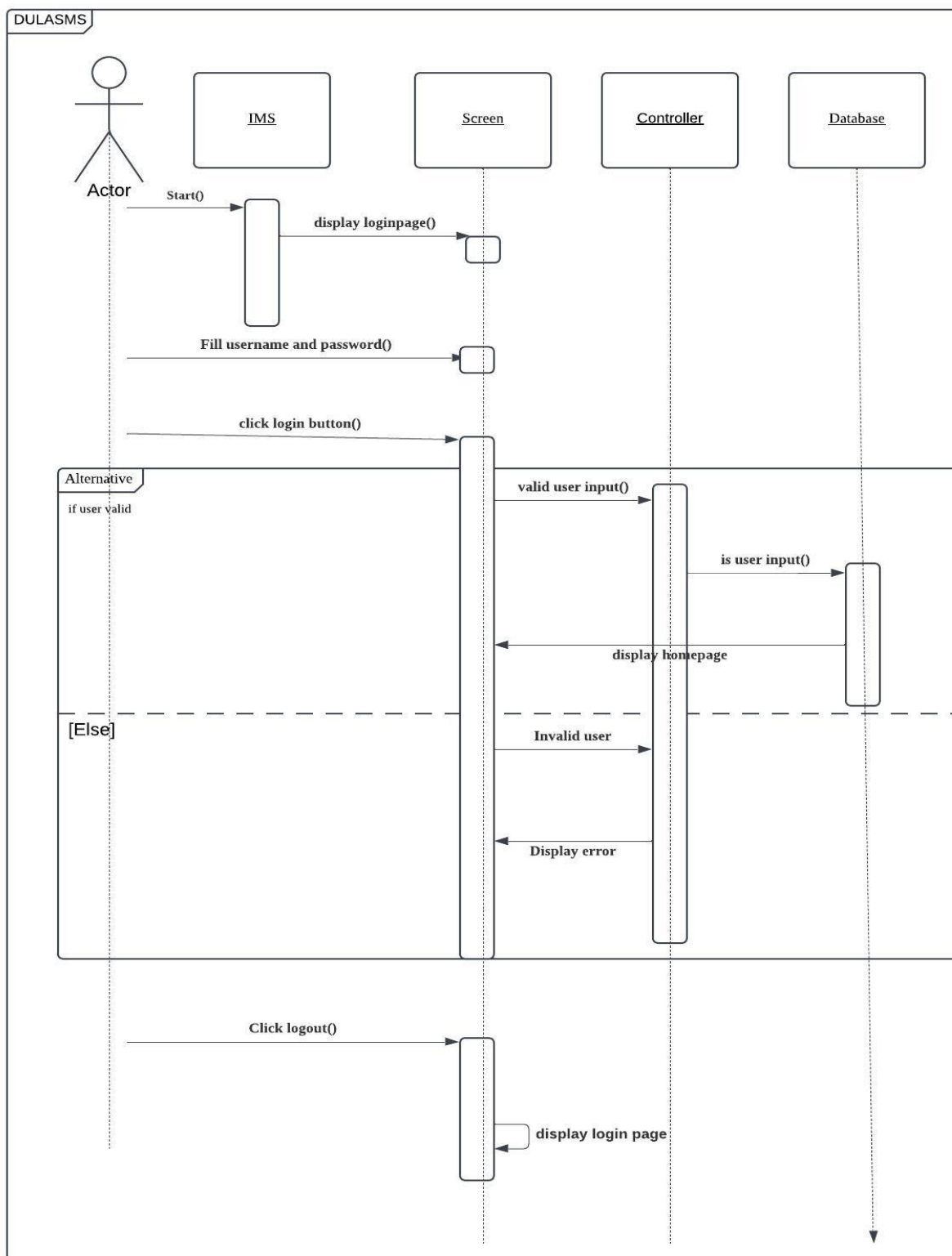Figure 4. 2 Sequence diagram for registration
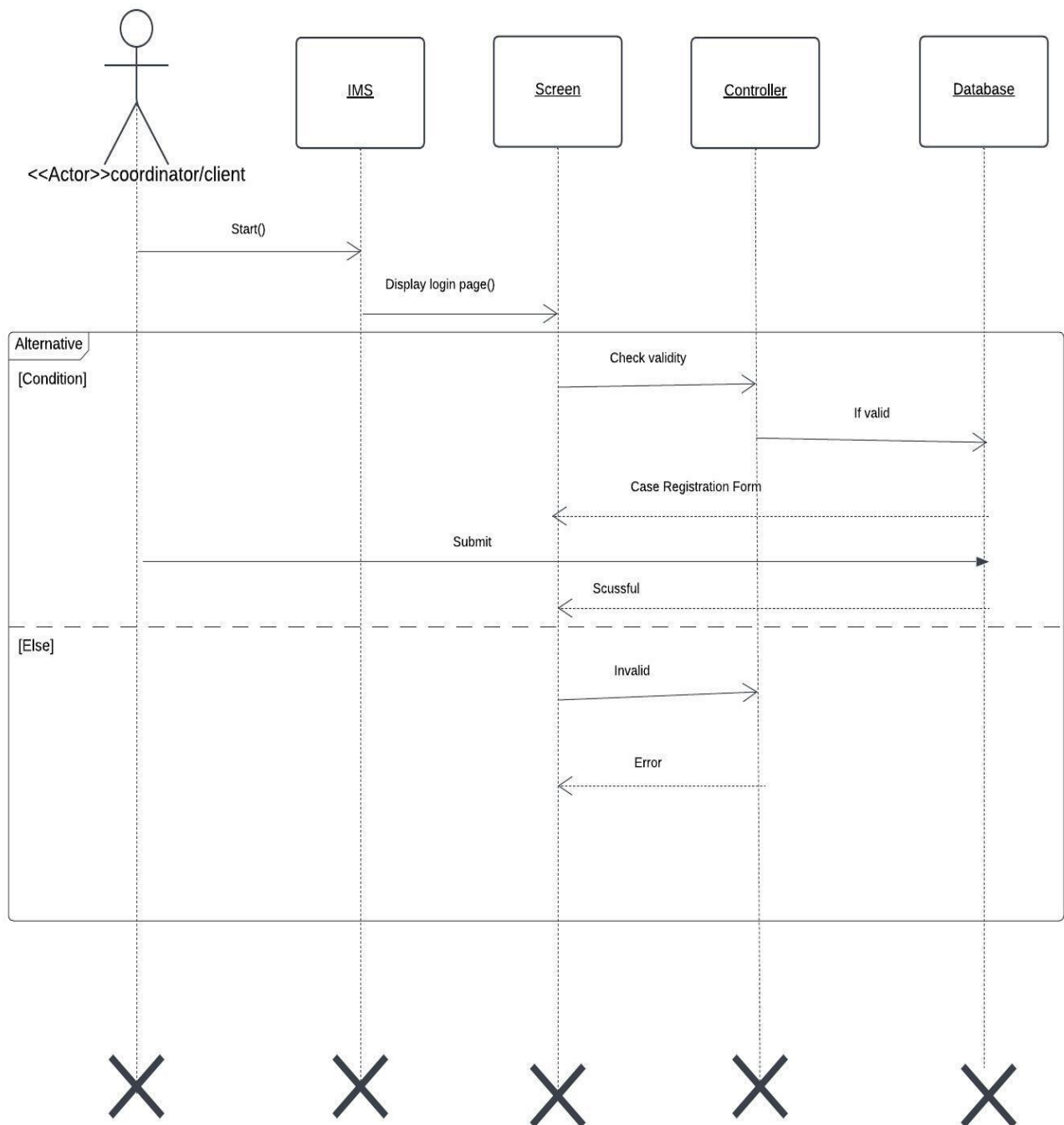
Figure 4. 3 Sequence diagram for login

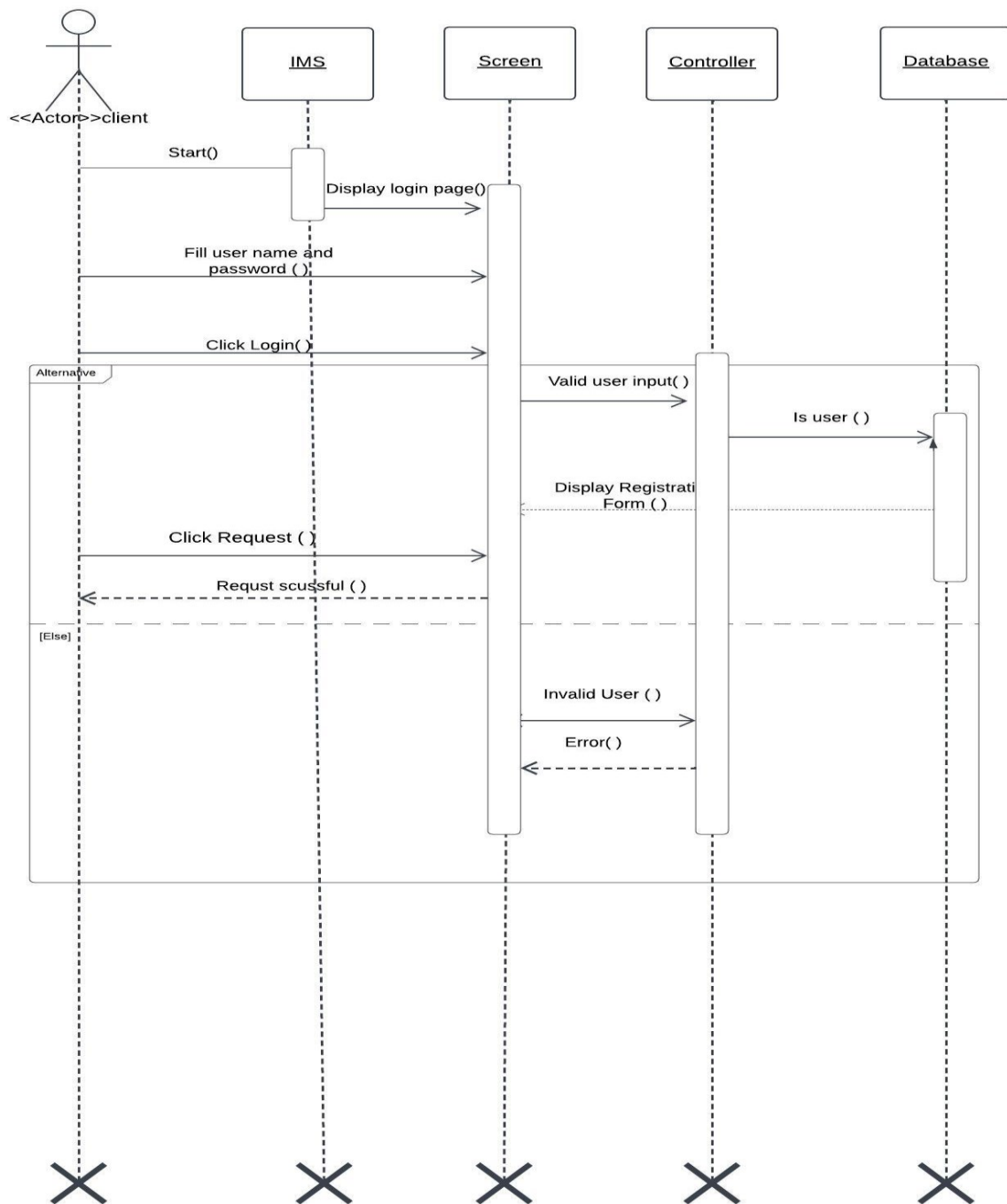Figure 4. 4 Sequence diagram for case registration
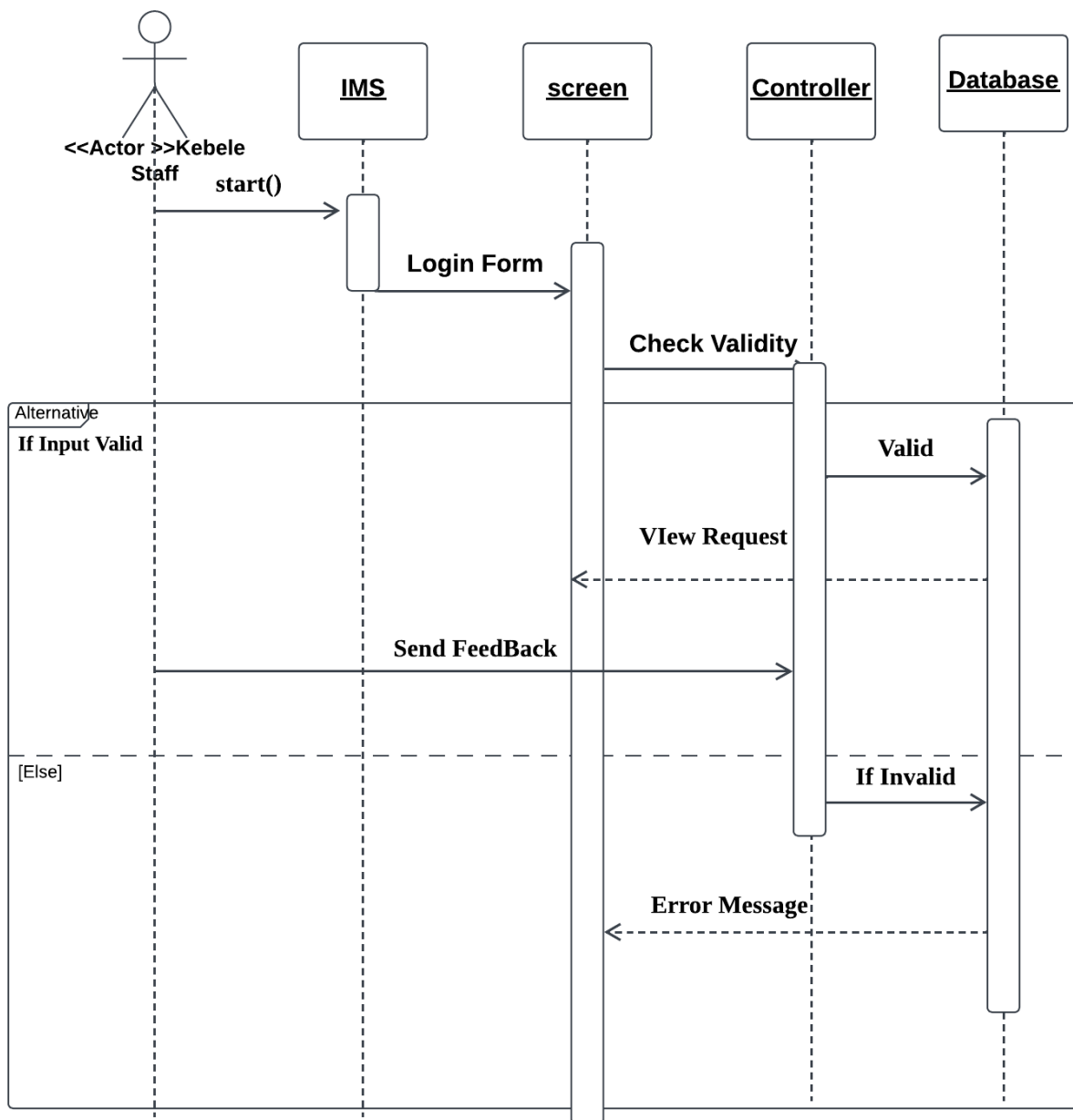
'

Figure 4. 5 Sequence diagram for Request

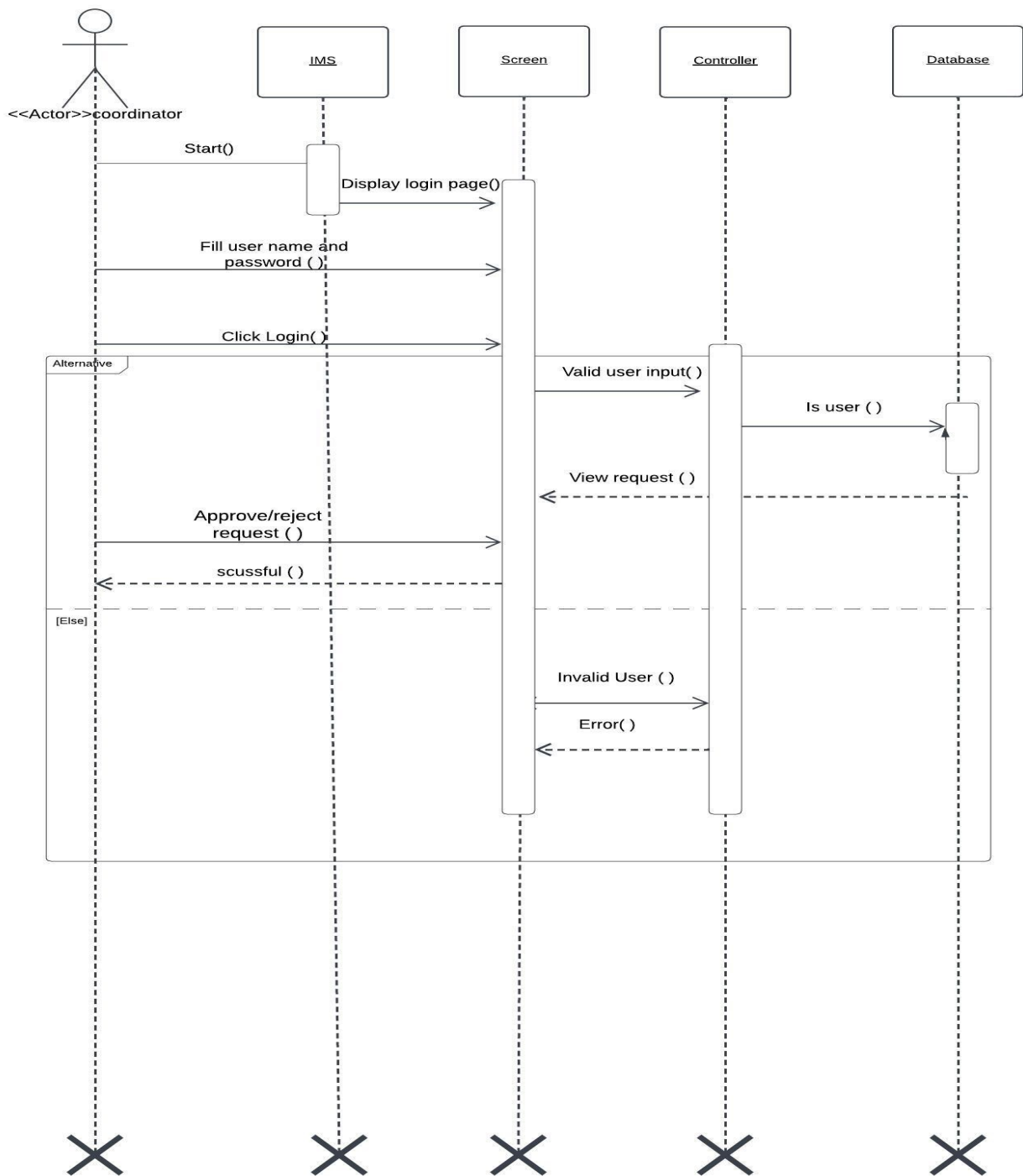Figure 4. 6 Sequence diagram for checking eligibility of legal aid service
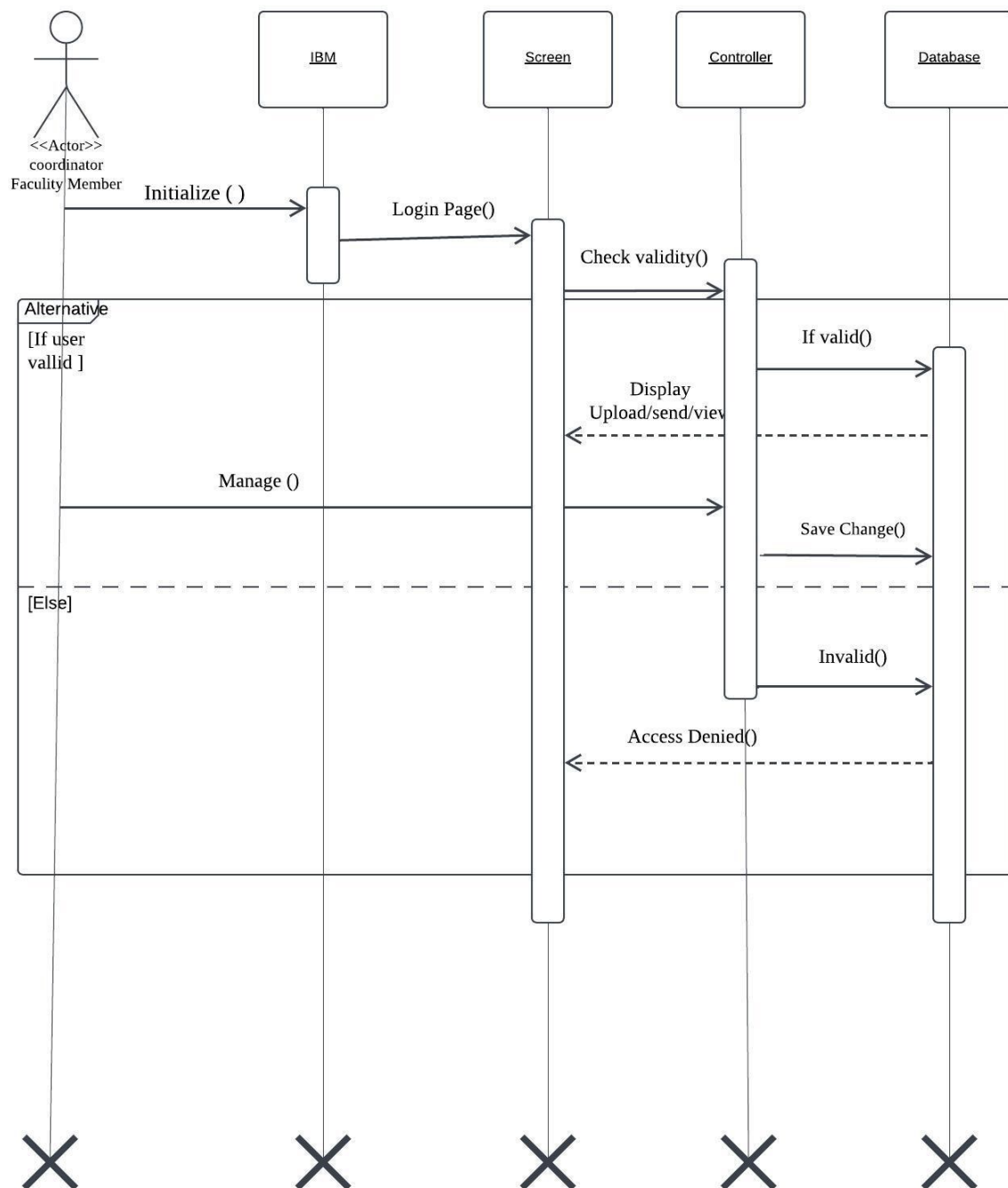
Figure 4. 7 Sequence diagram for Approve Request

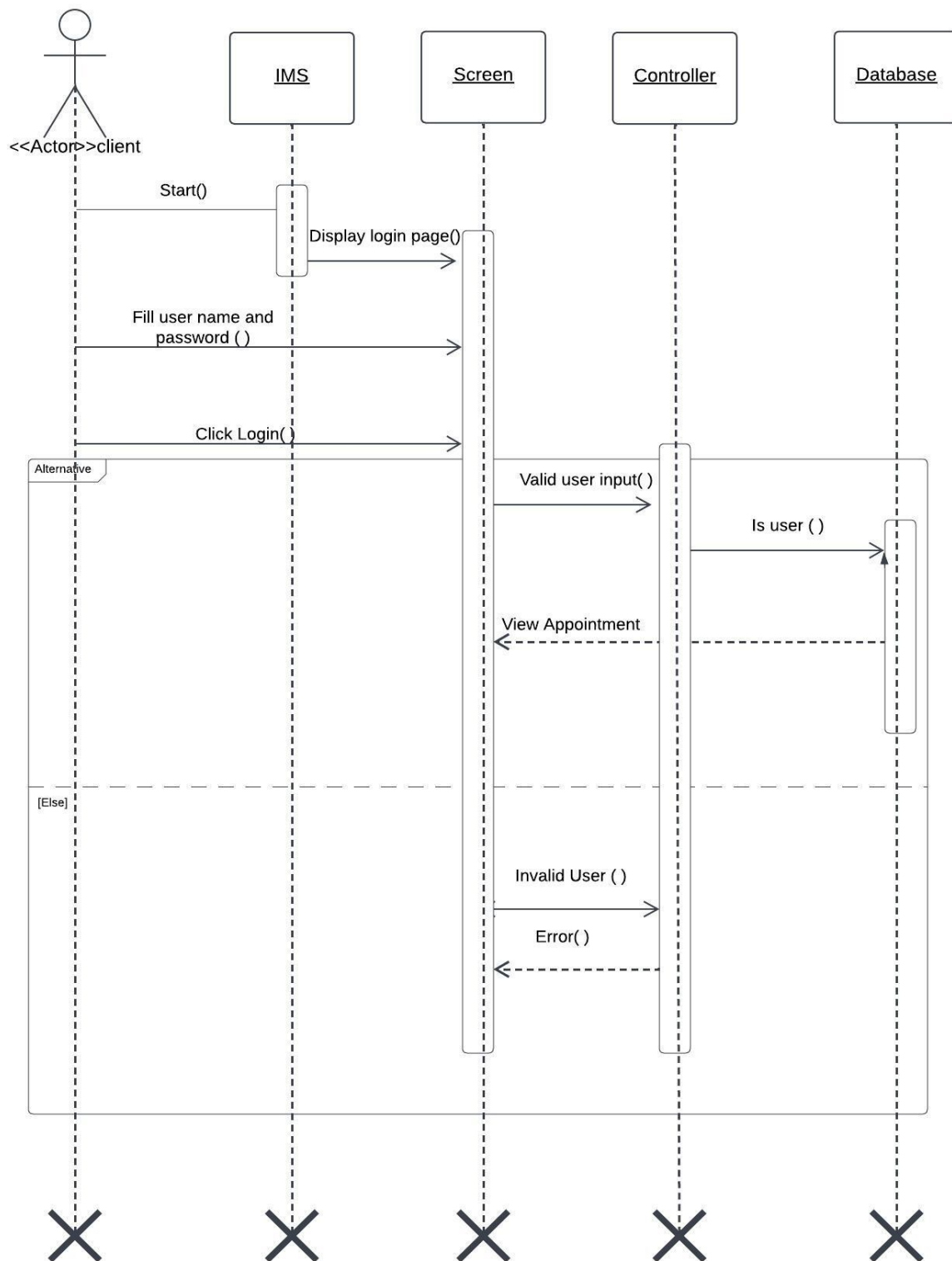Figure 4. 8 Sequence diagram for Manage case

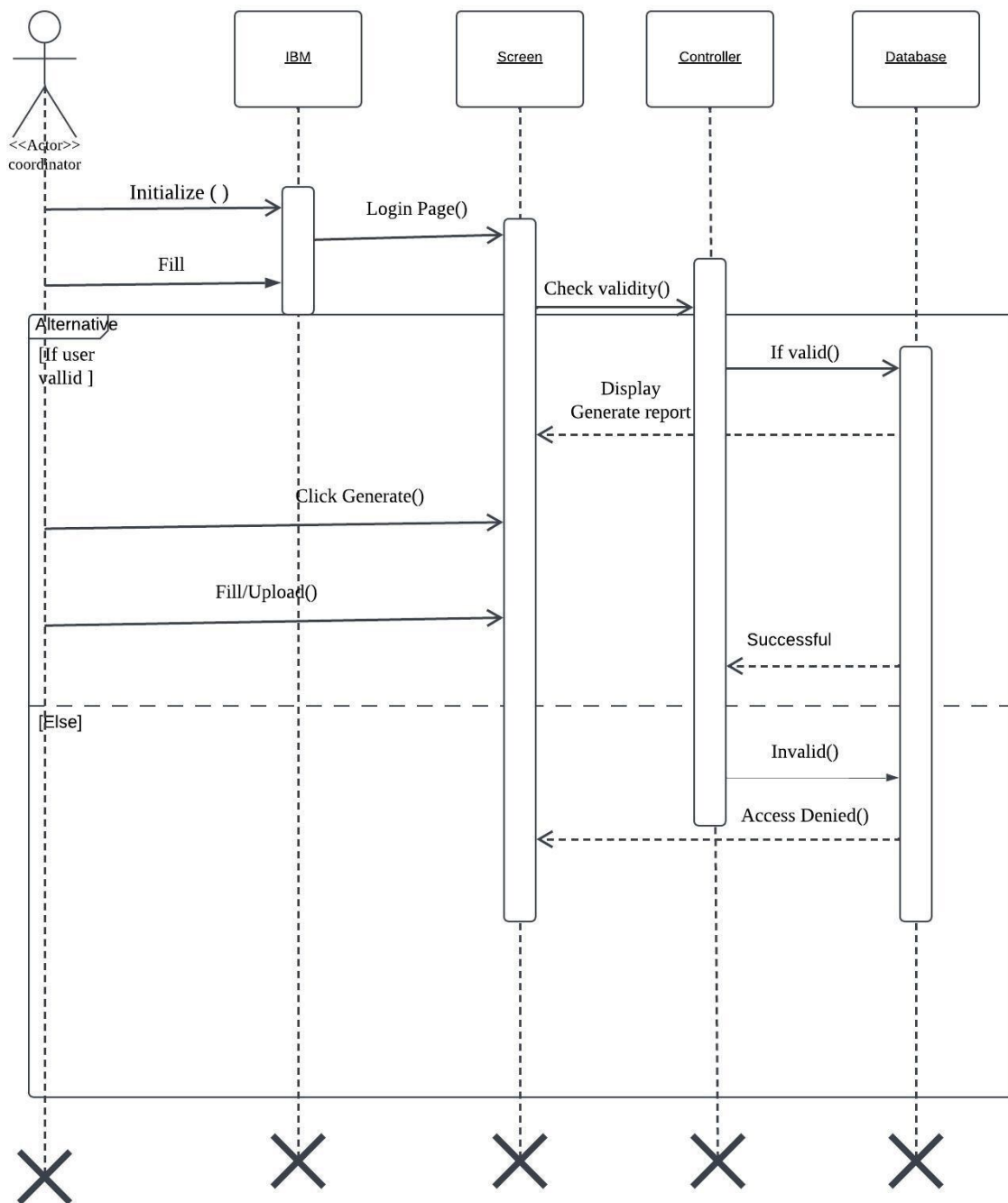Figure 4. 9 Sequence diagram for View Appointment

Figure 4. 10 Sequence diagram for Generate report
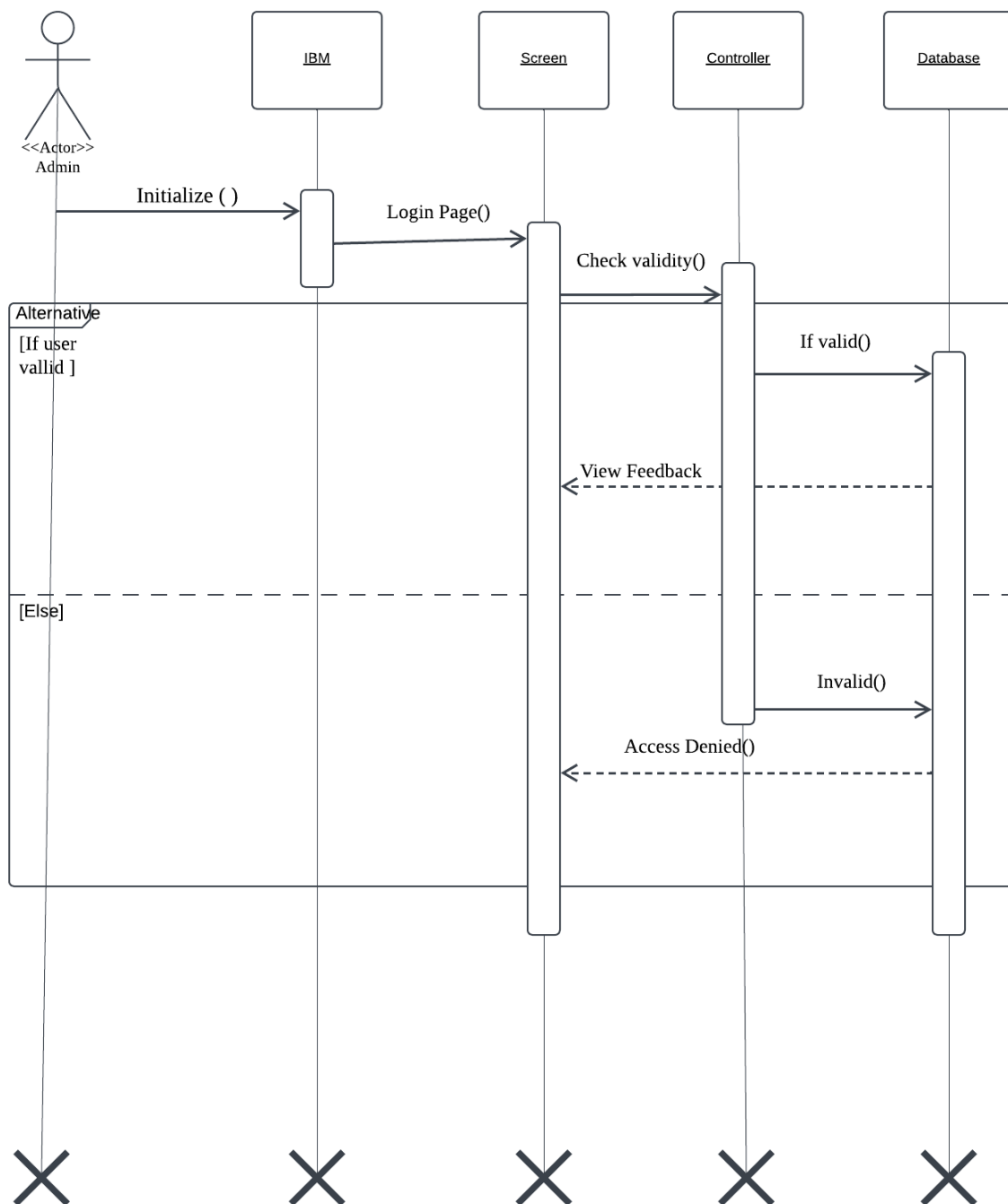
Figure 4. 11 Sequence diagram for View report

4.4 Activity diagram

An activity diagram describes a system in terms of activities. Activities are states that represent the execution of a set of operations. The completion of these operations triggers a transition to another activity. Activity diagrams are similar to flowchart diagrams in that they can be used to represent control flow (i.e., the order in which operations occur) and data flow (i.e., the objects that are exchanged among operations).

The UML notation for activity diagram

Oval (Start symbol)

Rectangle (Process/state)

Diamond(Decision)

End

Arrow(Flow)
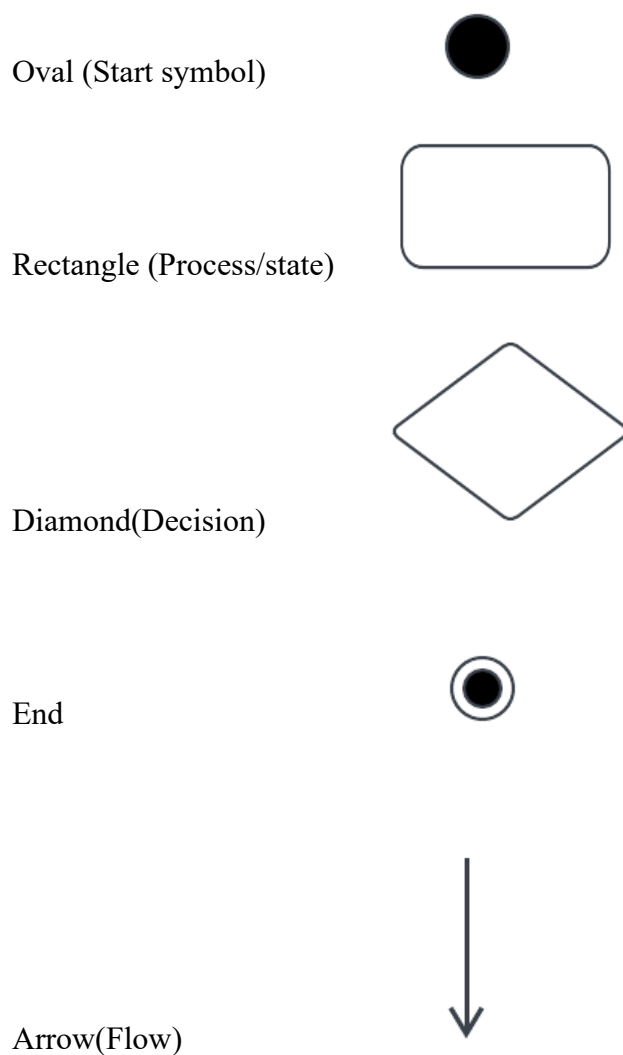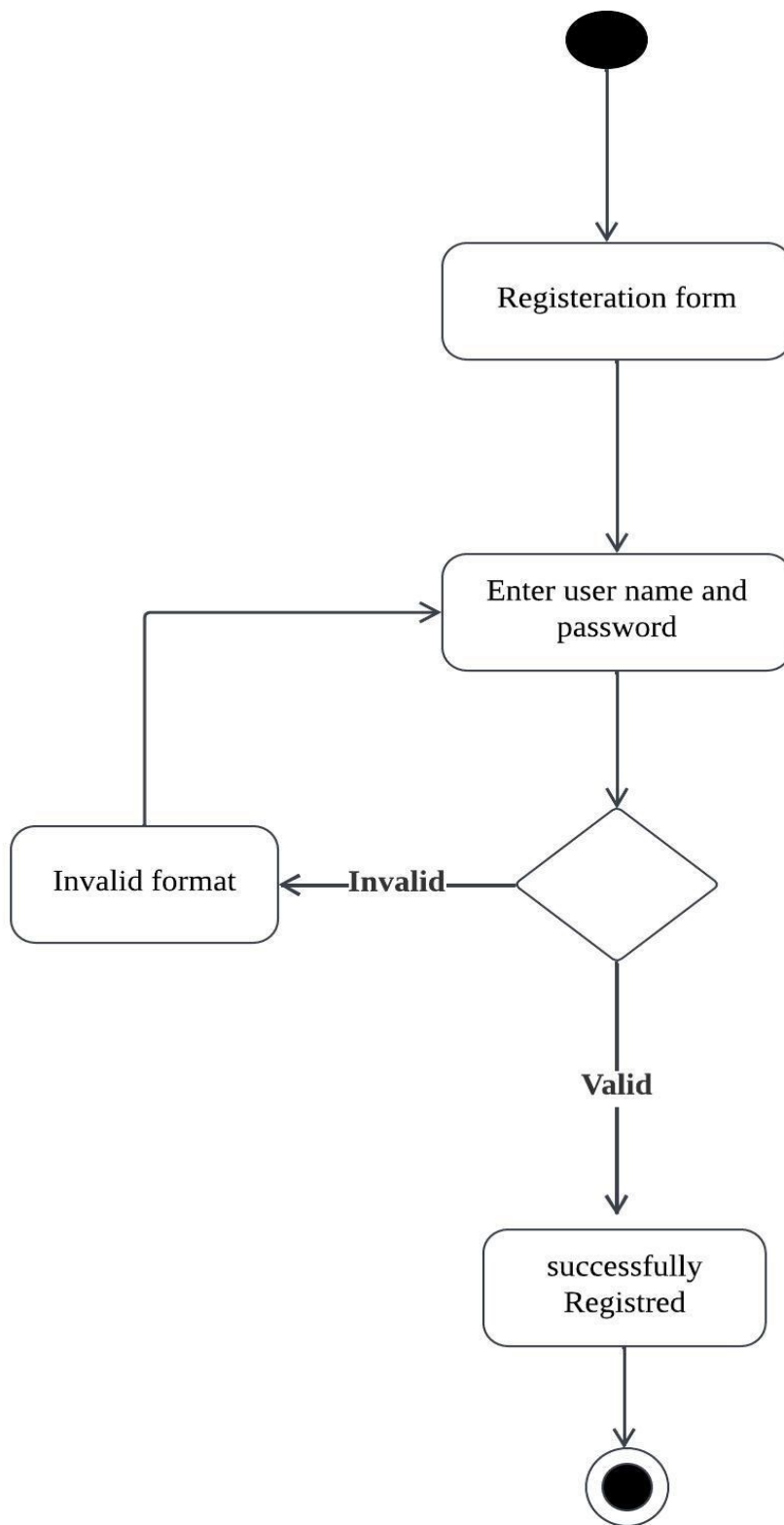
Figure 4. 12 Activity diagram for register
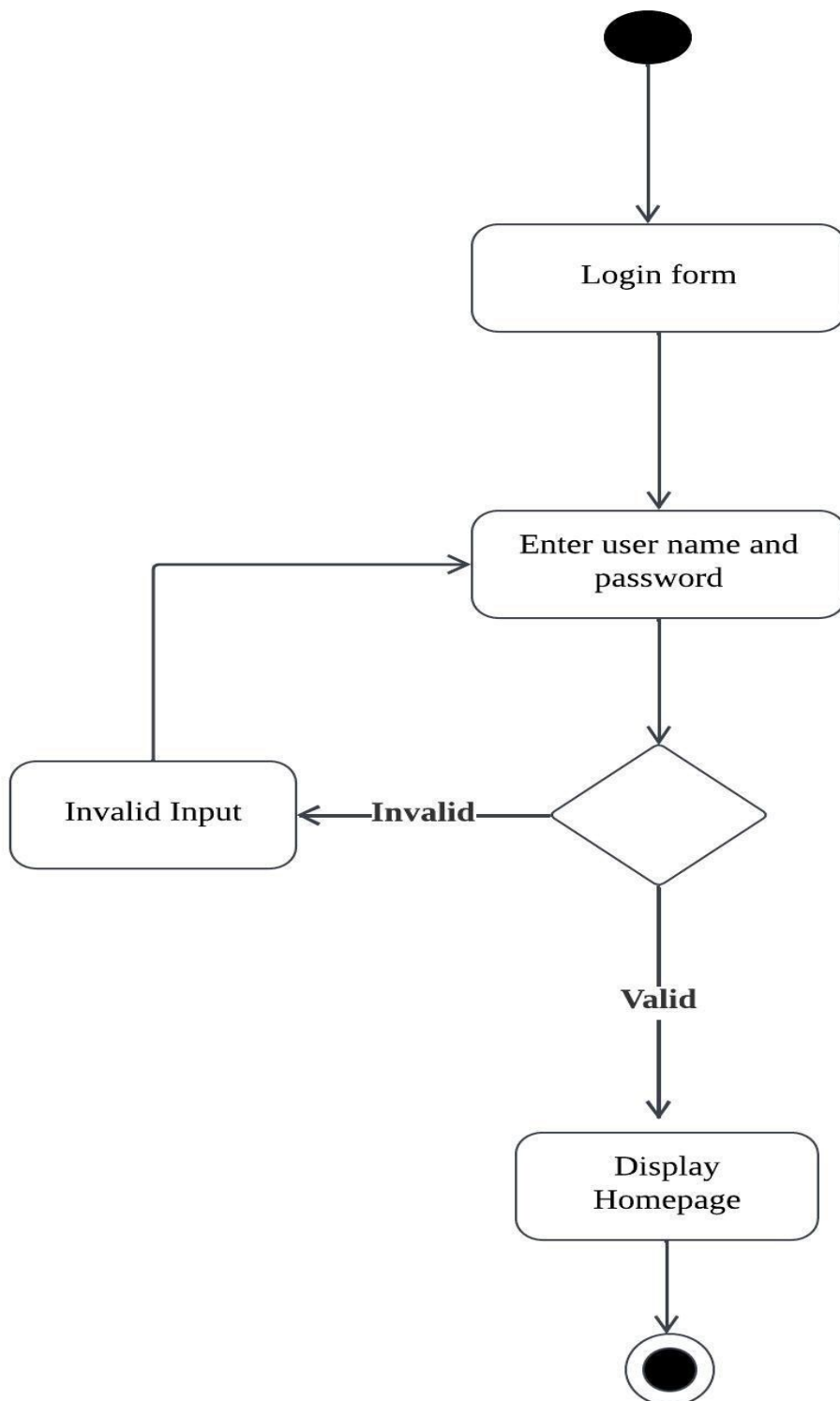
Figure 4. 13 Activity diagram for Login
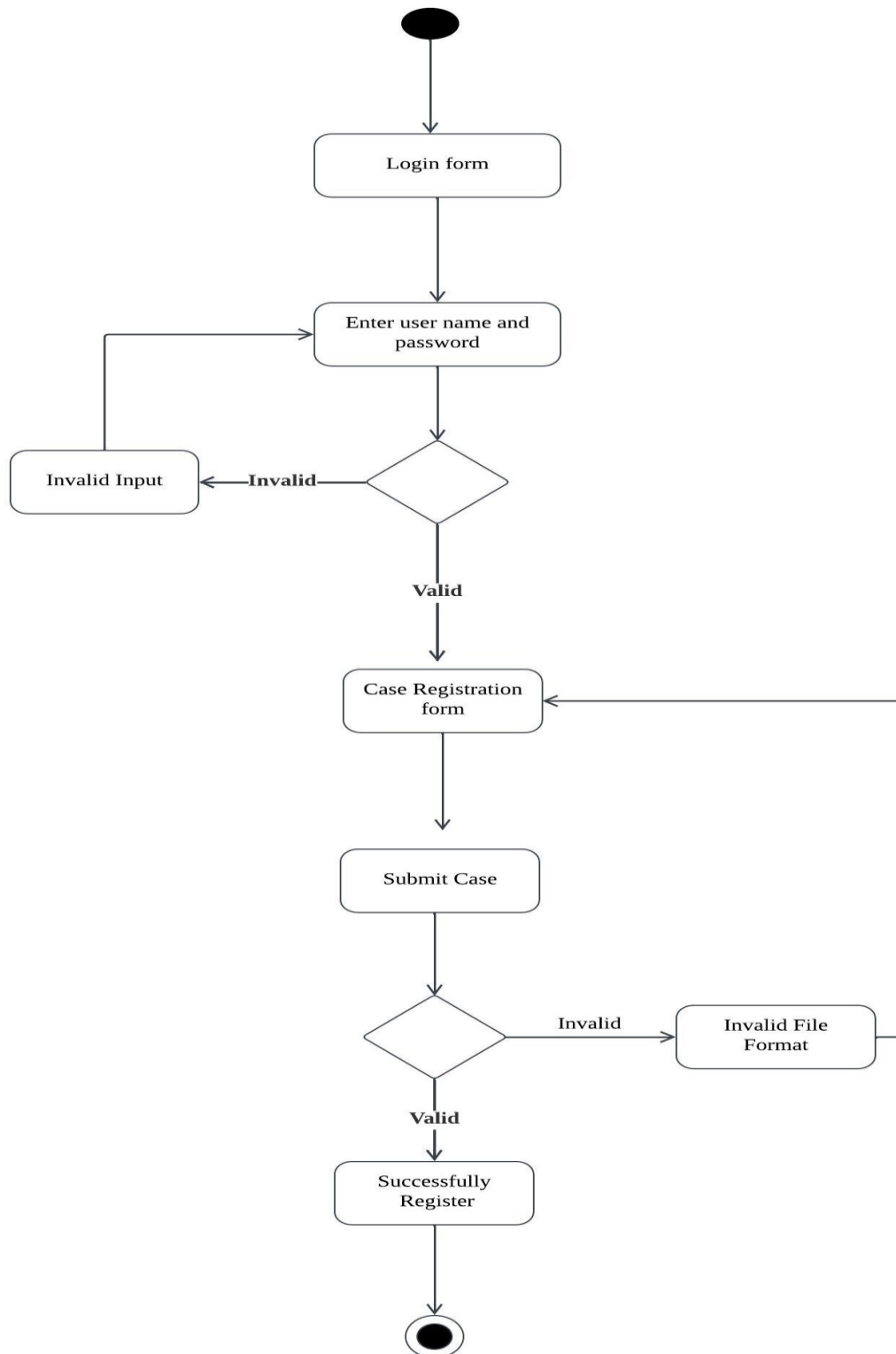
Figure 4. 14 Activity diagram for Case registration
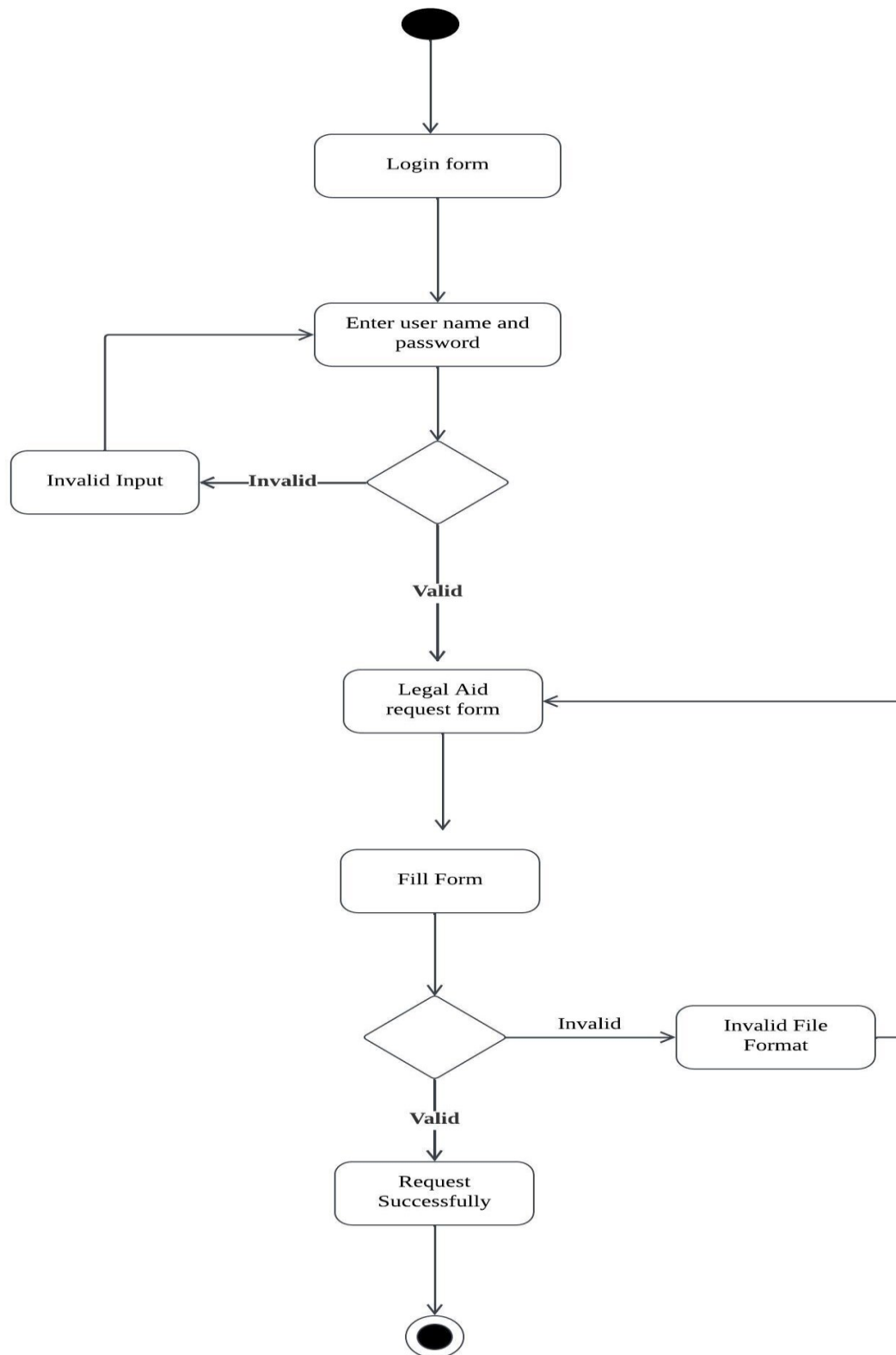
Figure 4. 15 Activity diagram for Service Request

Figure 4. 16 Activity diagram for Service Request
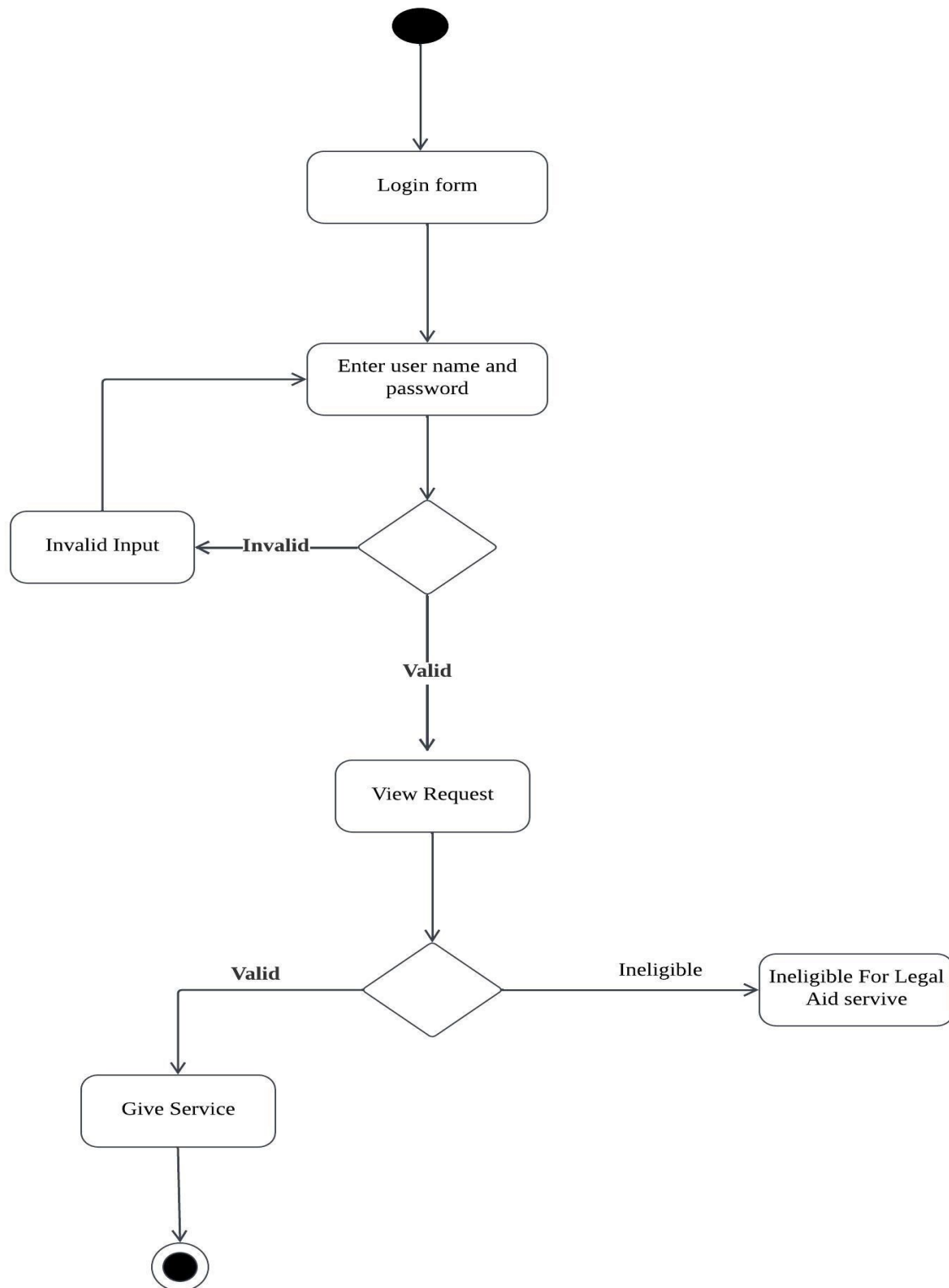
Figure 4. 17 Activity diagram for Approve Request
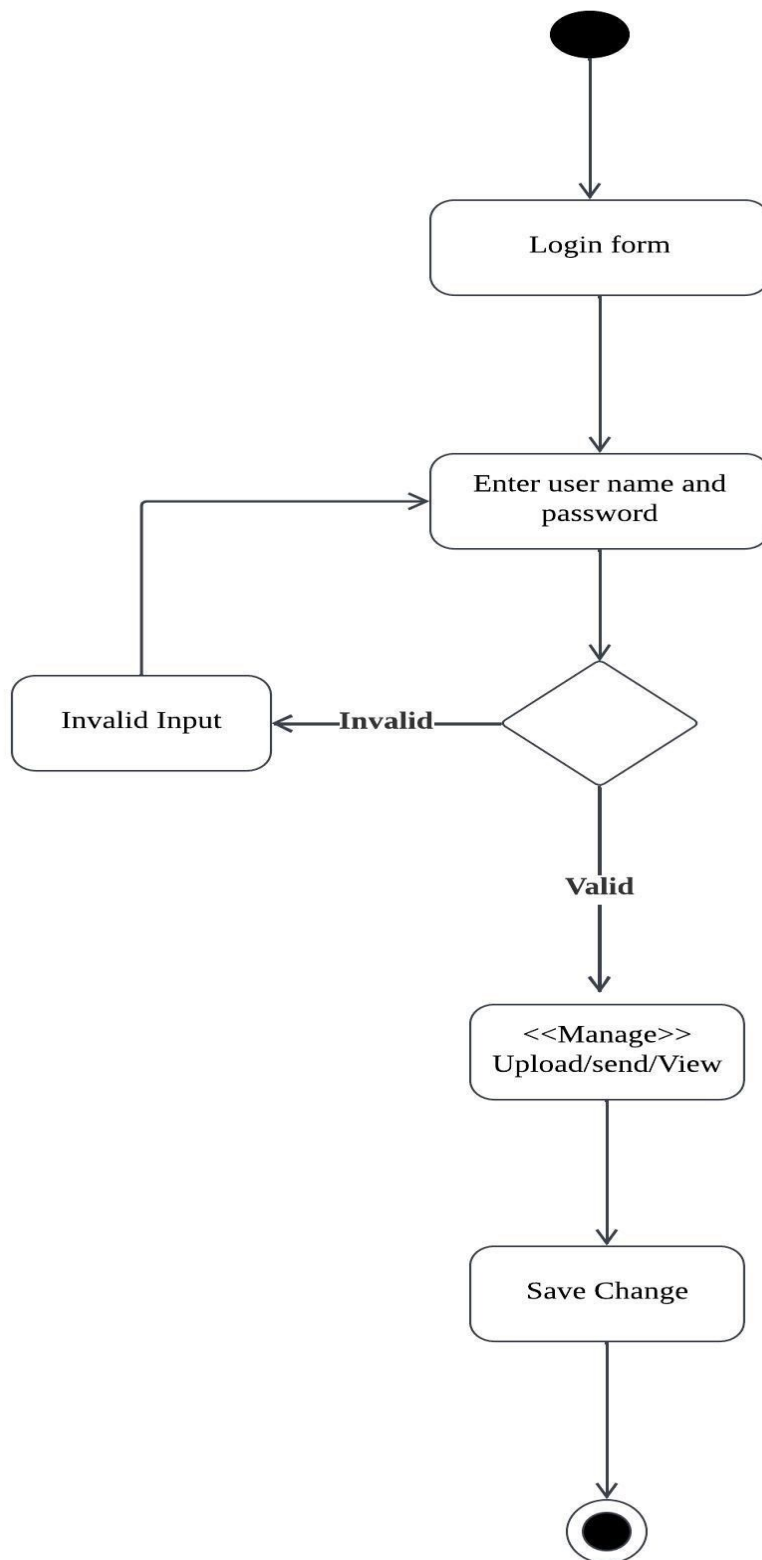
Figure 4. 18 Activity diagram for Mange
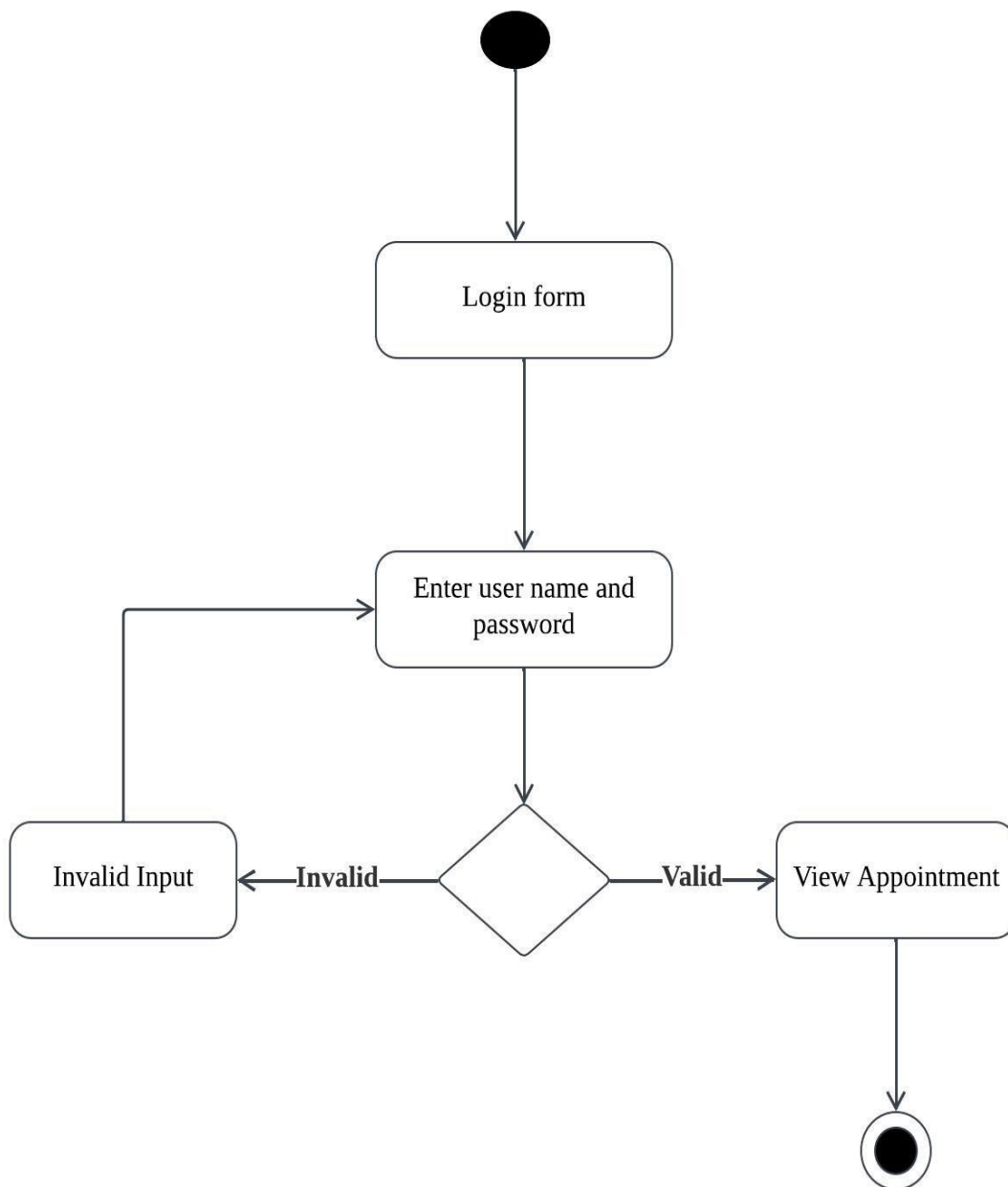
Figure 4. 19 Activity diagram for View Appointment
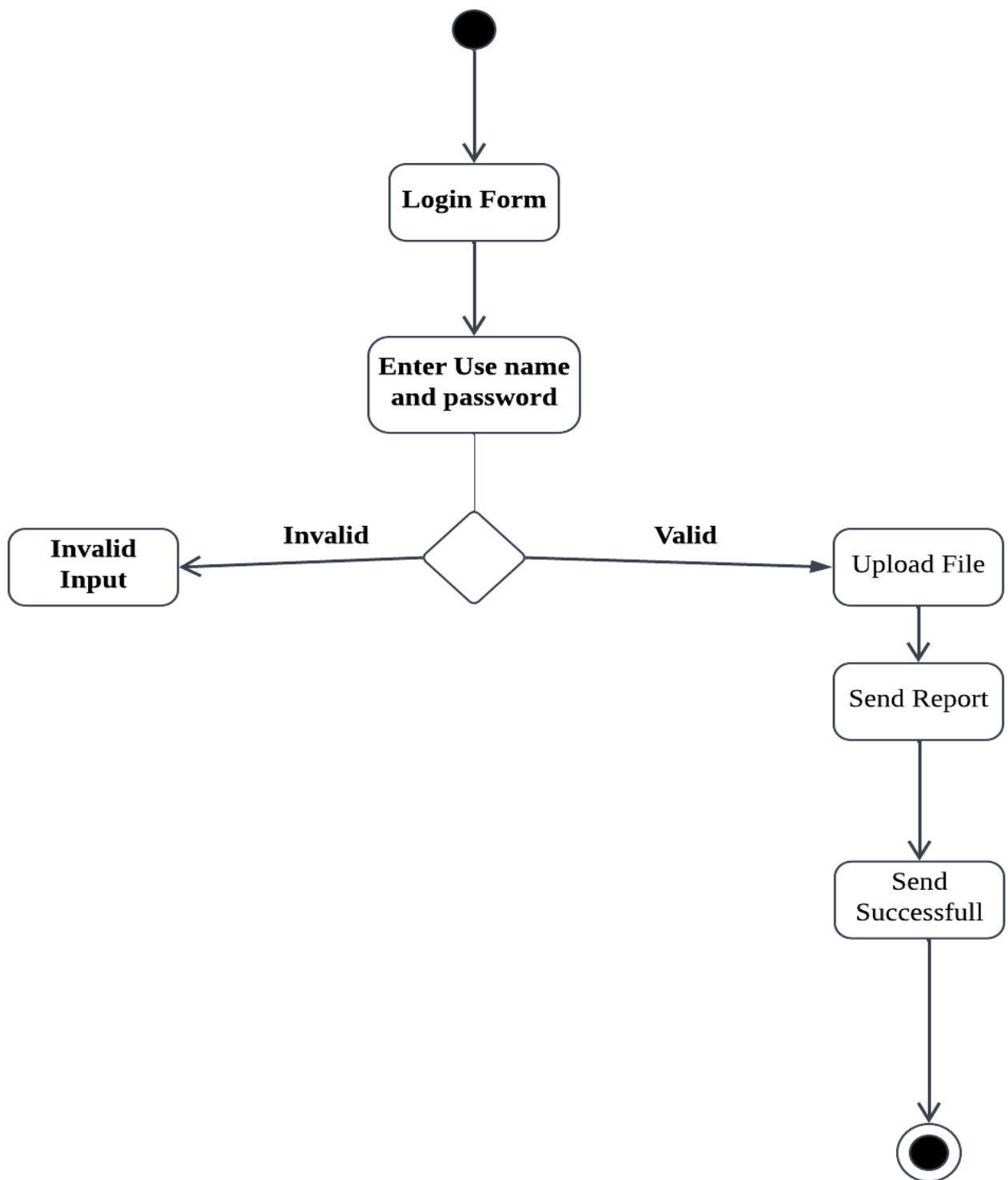
Figure 4. 20 Activity diagram for Send Report
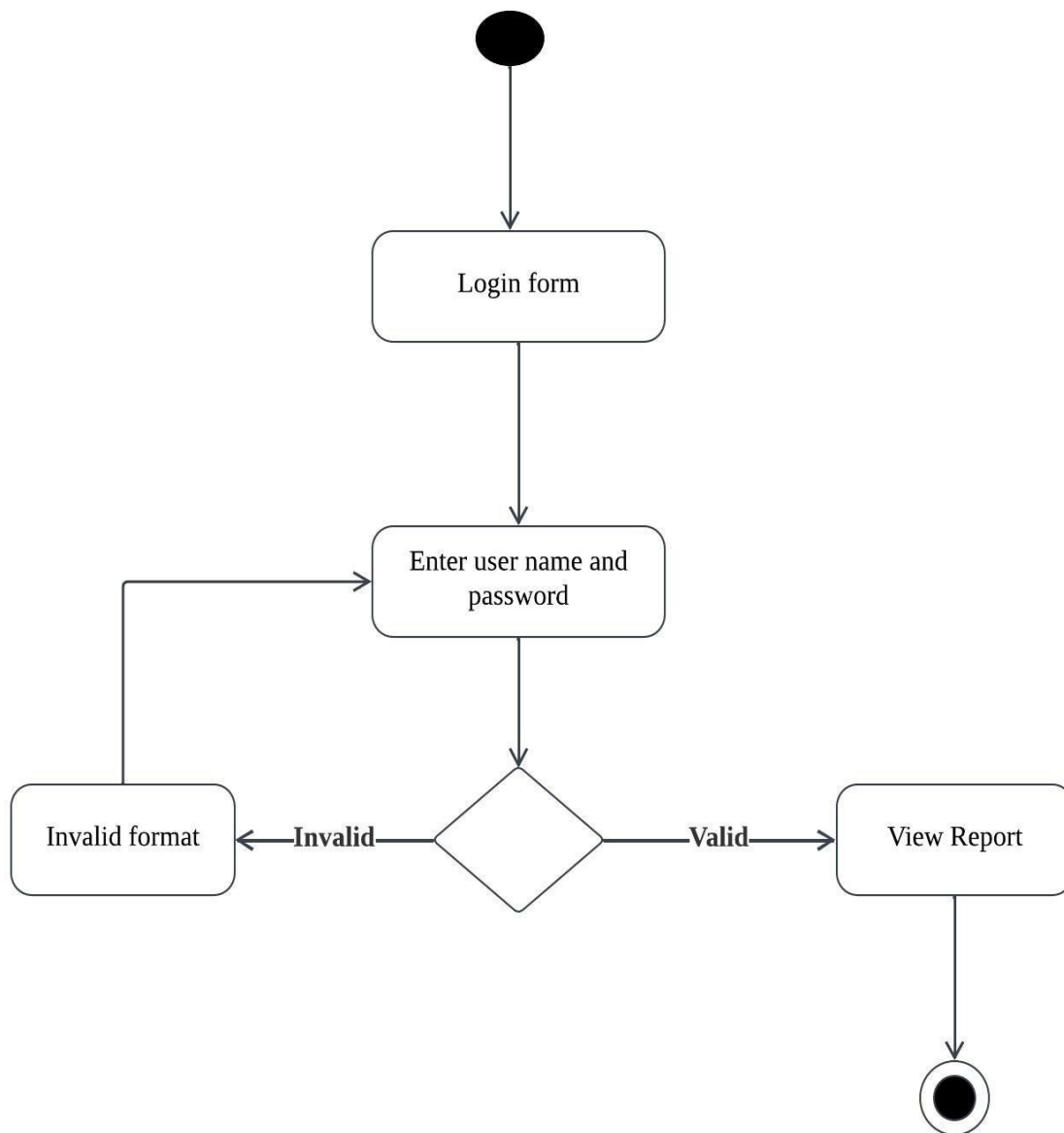
Figure 4. 21 Activity diagram for View Report
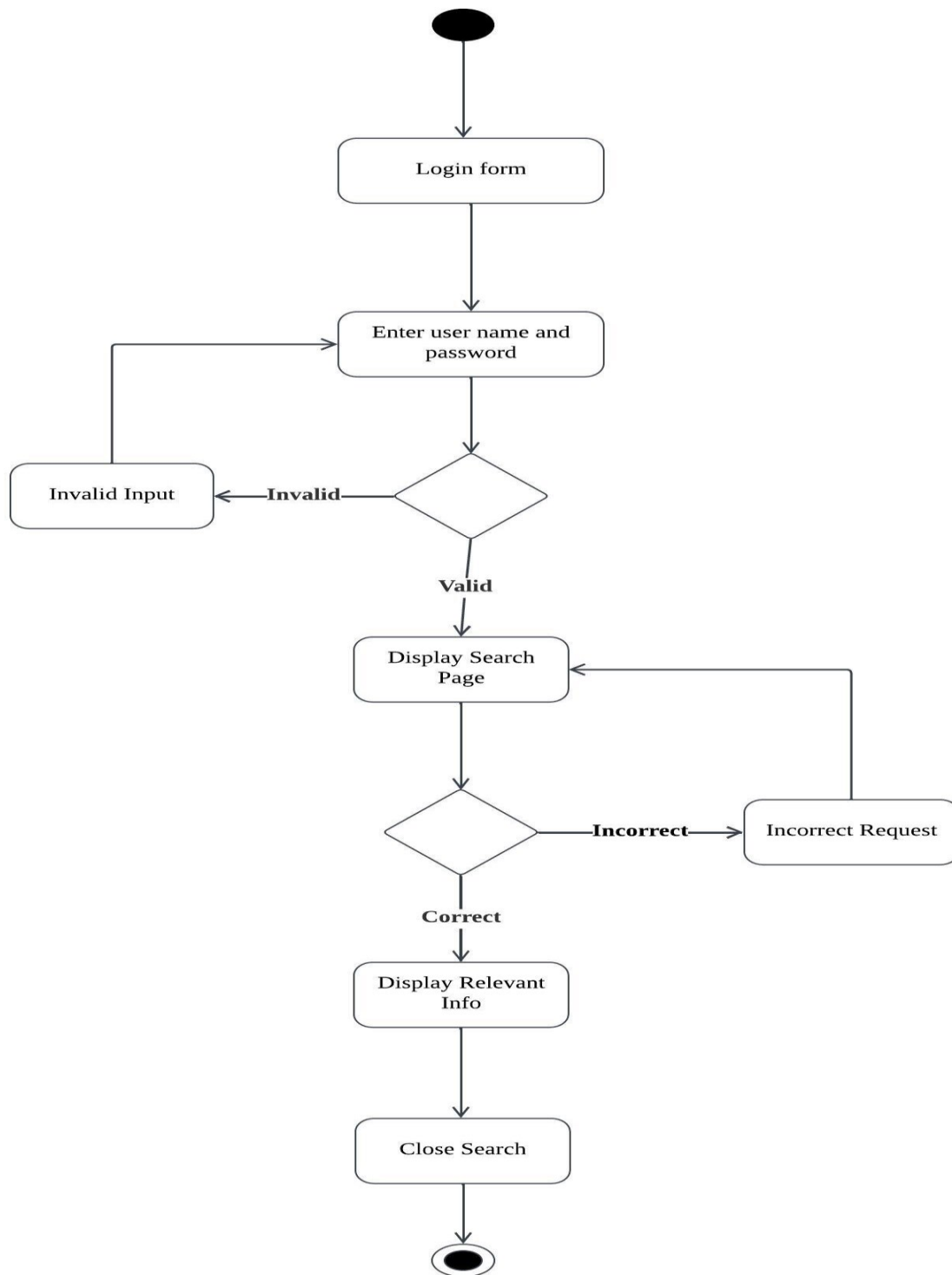
Figure 4. 22 Activity diagram for Search

4.5 Class Diagram

Class diagram is a type of static structure diagram that describes the structure of a system by the system's classes, their attributes, operations and the relationship among the objects. Class diagram is the main building block of object oriented modeling.

Its main uses are:

- For general conceptual modeling of the systematics of the application.
- For detailed modeling translating the models into programming code.

A class diagram is typically modeled as rectangle with three sections:

- ➢ The top indicates the name of the class.
- ➢ The middle contains the attributes of the class.
- ➢ The bottom contains the operations the class can execute.

Figure 4. 23 Class diagram

Conclusion

The Dilla University Legal Aid Service Management System has been carefully planned and designed to address the challenges of the existing manual system. The new system is expected to provide improved functionality, enhanced flexibility, and better security compared to the current system. By automating tasks such as client registration, case management, user access control, and data tracking, the system will significantly reduce workload, and increase the efficiency and effectiveness of service delivery.

This project document outlines the system's design, implementation plan, and functionalities, ensuring that all requirements are met to facilitate a more user-friendly and efficient process for all stakeholders. The system not only supports better coordination between clients, coordinators, faculty members, and administrators but also encourages continuity and improvement in the management of legal aid services at Dilla University.

The project has been an invaluable learning opportunity for the team, enhancing both technical and collaborative skills. The experience gained in designing and documenting this system has prepared the team to undertake future projects with greater expertise and confidence. Once implemented, the system is anticipated to revolutionize legal aid service management at Dilla University, ensuring accessibility, transparency, and sustainability.

Recommendation

In today's competitive world, organizations must adopt modern technologies to remain efficient and effective in delivering their services. Information and Communication Technology (ICT) plays a vital role in achieving organizational objectives. We recommend that Dilla University adopt the developed Legal Aid Service Management System to streamline its operations, reduce manual workload, and enhance the accessibility and transparency of legal aid services.

To ensure the effective implementation and use of the system, we make the following recommendations:

➢ The university should ensure the availability of adequate computer infrastructure and network facilities to support the system's deployment and operation.

➢ All stakeholders, including coordinators, faculty members, and administrators, should undergo training to become proficient in using the system and understanding its functionalities.

➢ The system should be integrated with a secure authentication mechanism to ensure that only authorized users can access sensitive information and perform critical tasks.

➢ Regular system updates and maintenance should be scheduled to ensure its continued functionality and to address potential issues or bugs.

➢ The university should consider expanding the system's functionality in the future to incorporate advanced features like real-time case tracking, automated notifications, and improved reporting tools.

➢ Proper usage guidelines and protocols should be established to prevent unauthorized access or misuse of the system.

Finally, this system should serve as a foundation for adopting other innovative technologies within the university, contributing to its mission of providing efficient and effective legal aid services. With careful implementation and regular evaluation, the Dilla University Legal Aid Service Management System can become a vital tool in improving the university's service delivery and operational efficiency.

References

1. Duy, A. N. (2019). *nstant Approach to Software Testing: Principles, Applications, Techniques and practices.* Duy Tan University 620.

2. Fenton, S. (2018). Pro TypeScript: Application-Scale JavaScript Development. *Library of Congress Control Number*, 18-28.

3. Kodali, N. (2024). Tailwind CSS Integration in Angular: A Technical Overview. *international jornal of innovative Research of Science and Technology*, 16652-16659.

Appendix

Project work plan /schedule

Note that the project schedule is designed with a balance of structure and adaptability to ensure successful completion within the given timeframe but it may flexible because of Overlapping tasks, adaptable task prioritization, feedback integration and others.

Table 12 Project schedule

Task	Month	Nov			Dec				Feb				Jan				March				May			
	Week	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4
1. Literature review		█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█				
2. Research proposal writing		█																						
4. Data collection					█	█	█	█	█	█	█	█												
5. Model design					█	█	█	█	█	█	█	█	█	█	█	█	█	█	█	█				
7. model evaluation													█	█	█	█	█	█	█	█	█			

Budget Plane

To finish the proposed project the following budget should be needed.

Table13 project budget plan

Device type	Purpose	Quantity	Unit price in Birr	Total price/cost in Birr
RAM	Increase PC performance	2	2000.00	2000.00
Hard disc	Backup and dataset storage	1 TB	3200.00	3200.00
Photocopy and Printing	To print Related work and document submitted in hard copy, color print, and cover		500	500.00
Pen, notebook	Writing notes draft	5,1	20,100	200
Paper	Draft documentation and related work note	400 pieces	1.5	600.00
Total				13500 birr

ዲላ ዩኒቨርሲቲ
ኢንጂነሪንግ እና ቴክኖሎጂ ኮሌጅ
የኮምፒውተር ሳይንስ ት/ት ክፍል

Dilla University
College of Engineering and Technology
Department of Computer Science

Ref: Comp/Info/Sch/ 39/17
Date: **09/12/2024**

To Law School
Dilla University, Dilla

Subject: Exploring Technical Needs for Student Projects

I hope this message finds you well. I am writing to you on behalf of the computer science department at Dilla University. As part of their academic journey, our students are working on various projects and research initiatives, and we are seeking valuable input from organizations like yours to ensure that their work aligns with industry needs and best practices. Their specific title is **Dilla University Legal Aid Service Management System**.

As a vital part of their coursework, our students are tasked with gathering real-world requirements, data, and insights from professionals and organizations to help shape their projects. These initiatives often involve software development, data analysis, system design, and other key areas of computer science, and we believe that your organization's expertise would be incredibly beneficial to their learning experience. If you are open to further discussion, we would be happy to arrange a brief meeting at your convenience. Please feel free to contact us at our office number or com.scie@du.edu.et to explore this opportunity further.

Thank you for considering this request. We look forward to hearing from you.

With best regards!

List of the Students

1. **Amanuel Tizazu**
2. **Chernet Afework**
3. **Bisrat Fikrie**
4. **Deginet Solomon**
5. **Astarekegn Degefu**

Samuel Gudeta Chufa
Head, Department of Computer Science, CIT

CC:

College of Engineering & Technology

+251-461313135   E-mail
com.scie@du.edu.et   P.O Box 419

Figure 4. 24 Letter