

**Ambo University Hachalu Hundessa Campus School of Informatics and Electrical Engineering Computer Science Department**

**Final Year Project I**

**Project Title: Web based Internship and Career Management System (ICMS)**

**Group members’ name and Id no.**

**Abdi Biru------------------------------------------UGR/61129/14**

**Naol Koti-------------------------------------------UGR/61129/14**

**Ephrem Niguse-----------------------------------UGR/61129/14**

**Mihret Bayilu ------------------------------------UGR/61129/14**

**Bontu Mohammed--------------------------------UGR/61129/14**

**Advisor**: Mr. Bedasa.W

Submitted to Department of Computer Science, Ambo University Hachalu Hundessa Campus,

Ambo, Ethiopia

September 24, 2024

**Candidate Declaration**

We, Abdi, Naol, Ephrem, Bontu, and Mihret, BSc in Computer Science (4th year) of the Computer Science Department, Institute of Technology, Ambo University, hereby declare that the Practical Report entitled "Web-Based Internship and Career Management System (ICMS)" is an original work, and the data provided in the study is authentic to the best of our knowledge. This report has not been submitted to any other institute for the award of any other degree.

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Project Members:**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

The project has been submitted for examination with my approval as university advisor.

**Advisor**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_ Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Approval of the Examining Board**

Project Coordinator: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Examiner: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Examiner: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Signature: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Date: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Contents

[List of Figures IV](#_Toc198882366)

[**List of Tables** V](#_Toc198882367)

[Acronyms 0](#_Toc198882368)

[CHAPTER ONE 1](#_Toc198882369)

[Introduction 1](#_Toc198882370)

[1.1. Background of the Organization 1](#_Toc198882371)

[1.2. Statement of the Problem and Justification 3](#_Toc198882372)

[1.3. Objective of the Project 3](#_Toc198882373)

[1.3.1. General Objective 3](#_Toc198882374)

[1.3.2. Specific Objectives 4](#_Toc198882375)

[1.4 Methodology 4](#_Toc198882376)

[1.4.1 Requirement Gathering method 4](#_Toc198882377)

[1.4.2 Requirement Modelling 5](#_Toc198882378)

[1.5. Tools 5](#_Toc198882379)

[1.6. Scope and Limitation 6](#_Toc198882380)

[1.6.1 Scope of the project 6](#_Toc198882381)

[1.6.2 Limitation of the project 7](#_Toc198882382)

[1.7 Significance of the Project 7](#_Toc198882383)

[1.8 Feasibility 9](#_Toc198882384)

[1.9. Risk Assessment 11](#_Toc198882385)

[1.10 Work Breakdown Structure (WBS) 12](#_Toc198882386)

[CHAPTER TWO 14](#_Toc198882387)

[System Requirement Specification 14](#_Toc198882388)

[2.1 Current system 14](#_Toc198882389)

[2.2 Business Rules 14](#_Toc198882390)

[2.3 Proposed system 15](#_Toc198882391)

[2.3.1 Overall Description 15](#_Toc198882392)

[2.3.2 Functional requirements 16](#_Toc198882393)

[2.3.3 Nonfunctional requirements 17](#_Toc198882394)

[2.3.4 System models 18](#_Toc198882395)

[2.3.4.1 Use case model 19](#_Toc198882396)

[2.3.4.2 Sequence Diagram. 32](#_Toc198882397)

[2.3.4.3 State chart diagram 45](#_Toc198882398)

[2.3.4.4 Activity Diagram 50](#_Toc198882399)

[2.3.4.5 Class Diagram 61](#_Toc198882400)

[2.3.4.6 User interface Prototyping 61](#_Toc198882401)

[CHAPTER THREE 63](#_Toc198882402)

[System Design 63](#_Toc198882403)

[3.1 Introduction 63](#_Toc198882404)

[3.2 Purpose of the System 63](#_Toc198882405)

[3.3 Design Goals 64](#_Toc198882406)

[3.4 Current Software Architecture 64](#_Toc198882407)

[3.5 Proposed Software Architecture 64](#_Toc198882408)

[3.5.1 Subsystem decomposition 66](#_Toc198882409)

[3.5.2 Component Diagram 67](#_Toc198882410)

[3.5.3 Deployment Diagram 68](#_Toc198882411)

[3.5.4 Persistent Data Management 68](#_Toc198882412)

[3.5.5 Access Control and Security 73](#_Toc198882413)

[3.5.6 Global Software Control 75](#_Toc198882414)

[3.5.7 Boundary Conditions 75](#_Toc198882415)

# List of Figures

[Figure 1 : Work breakdown structure 13](#_Toc198882416)

[Figure 2:ICMS Use Case Diagram 20](#_Toc198882417)

[Figure 3: Sequence diagram for login 33](#_Toc198882418)

[Figure 4: Sequence for Apply career 34](#_Toc198882419)

[Figure 5: Sequence Apply post 35](#_Toc198882420)

[Figure 6: Sequence diagram for attach files 36](#_Toc198882421)

[Figure 7: Sequence diagram for reset password 37](#_Toc198882422)

[Figure 8: Sequence diagram for Manage Account 38](#_Toc198882423)

[Figure 9: Sequence diagram for Manage profile 39](#_Toc198882424)

[Figure 10: Sequence diagram for delete account 40](#_Toc198882425)

[Figure 11: Sequence diagram for view notification 41](#_Toc198882426)

[Figure 12: Sequence diagram for apply internship 42](#_Toc198882427)

[Figure 13: Sequence diagram for registration 43](#_Toc198882428)

[Figure 14: Sequence View user 44](#_Toc198882429)

[Figure 15: State chart diagram for log in 45](#_Toc198882430)

[Figure 16: State chart diagram for log in 46](#_Toc198882431)

[Figure 17 : State machine diagram for Apply internship 47](#_Toc198882432)

[Figure 18: State machine diagram for Attach file 48](#_Toc198882433)

[Figure 19: State machine diagram for apply carer 49](#_Toc198882434)

[Figure 20 : Activity diagram for apply internship 50](#_Toc198882435)

[Figure 21 : Activity diagram for apply career 51](#_Toc198882436)

[Figure 22 : Activity diagram for attach file 52](#_Toc198882437)

[Figure 23 : Activity diagram for delete 53](#_Toc198882438)

[Figure 24 : Activity diagram for edit profile 54](#_Toc198882439)

[Figure 25 : Activity diagram for login 55](#_Toc198882440)

[Figure 26 : Activity diagram for notification 56](#_Toc198882441)

[Figure 27 : Activity diagram for post 57](#_Toc198882442)

[Figure 28 : Activity diagram for Account admin 58](#_Toc198882443)

[Figure 29 : Activity diagram for Change Password 59](#_Toc198882444)

[Figure 30 : Activity diagram for registration 60](#_Toc198882445)

[Figure 32 : Class Diagram 61](#_Toc198882446)

[Figure 33 : User interface prototype 62](#_Toc198882447)

[Figure 34 : ICMS Proposed software architecture 65](#_Toc198882448)

[Figure 35 : Component Diagram 67](#_Toc198882449)

[Figure 37 : Deployment Diagram of ICMS 68](#_Toc198882450)

[Figure 39 : ICMS Persistent Diagram 73](#_Toc198882451)

**List of Tables**

[Table 1: List of Tools 5](#_Toc198882452)

[Table 2:Business rules 15](#_Toc198882453)

[Table 5:Actors 19](#_Toc198882454)

[Table 6:Login Use Case Description Login Use Case Description 21](#_Toc198882455)

[Table 7: Signup (Register) Use Case description 22](#_Toc198882456)

[Table 8: Apply internship Use Case Description 23](#_Toc198882457)

[Table 9: Apply job Use Case Description 24](#_Toc198882458)

[Table 10: Attach Files Use Case Description 25](#_Toc198882459)

[Table 11: View Notification Use Case Description 26](#_Toc198882460)

[Table 12 : Post news Use Case description 27](#_Toc198882461)

[Table 13: View user Use Case description 28](#_Toc198882462)

[Table 14: Manage profile use case description 29](#_Toc198882463)

[Table 15: Manage account use case description 30](#_Toc198882464)

[Table 16: Reset Password Use Case description 30](#_Toc198882465)

[Table 17: Logout Use Case Description 31](#_Toc198882466)

[Table 18 : Student Account Table 69](#_Toc198882467)

[Table 19 : Faculty Account Table 69](#_Toc198882468)

[Table 20 : Company Account Table 70](#_Toc198882469)

[Table 21 : Profile Table 70](#_Toc198882470)

[Table 22 : Message Table 71](#_Toc198882471)

[Table 23 : Vacancy table 71](#_Toc198882472)

[Table 24 : Applicants Table 71](#_Toc198882473)

[Table 25 : Admin Account table 72](#_Toc198882474)

[Table 27 : Access Control and Security 74](#_Toc198882475)

# Acronyms

|  |
| --- |
| API Application Program Interface |
| CSS Cascading Style Sheet |
| GUI Graphical User Interface |
| HTML Hypertext Markup Language |
| ICMS Internship Career Management System |
| IDE Integrated Developmental Environment |
| PHP Hypertext Preprocessor |
| SQL Structured Query Language |
| REQ Requirement |
| UML Unified Modelling Language |

# CHAPTER ONE

# Introduction

## 1.1. Background of the Organization

Ambo is a dynamic city and the capital of West Shewa Zone of Oromia Regional State in Ethiopia. 112 km away from Addis Ababa towards the west, It is a pleasant city with a friendly climate where it is excellent to reside and study. Ambo University is the nucleus of this booming population, which was established in 1947 (1939 E.C.). Over the decades, the university has developed into a thriving institution with several colleges and departments, offering a pillar of higher education in Ethiopia.

Like with most of the Ethiopian universities, Ambo University also mandates internships as an essential component of its academic curriculum, with students gaining practical experience in their field of study. However, given the growing need for internship placements both locally and nationwide, the current system of connecting students with meaningful placements is trailing. It is difficult for students to find quality internships easily, and paper-based, time-consuming processes cause enormous hassle for all parties concerned.

Under the existing system, managing internships is a logistical nightmare. Advisors must physically travel to students, who are often assigned to near their hometowns, and the transportation cost is excessive. Additionally, relying on paperwork for applications, evaluations, and reports is time-consuming and dangerous loss or damaged documents can put a student's internship on the spot. These inefficiencies led us to develop the Internship and Career Management System (ICMS), an online system for automating and streamlining the internship and career management process into a more modernized and efficient form.

**What is an internship?**

An internship is a structured, professional learning experience that puts students in the middle of actual work related to their academic and career aspirations. It provides the opportunity to explore career choices, learn new skills, and gain practical work experience. For employers, internships bring new ideas, foster talent development, and offer a pipeline for future talent. Internships typically require a minimum of 120 hours of work, usually on a weekly basis of 10 hours during the academic year or full-time in the summer.

The Internship Management System (IMS) module in ICMS is designed to substitute seamless communication between students, faculty advisors, and industry sponsors. Students can use a user-friendly interface to hunt for internship postings, electronically submit proposals, and track progress. Advisors can update company information, review student submissions, and provide feedback without on-site visit costs. Real-time notification keeps advisors informed of new proposals, allowing them to approve or reject submissions effortlessly.

The Career Management System (CMS) is an expansion of the ICMS features to career opportunities where students can submit applications for career vacancies by uploading required documents and receiving selection notices from employers. The two systems are integrated under one easy-to-use interface whereby users can view available internship and job postings, apply with ease, and keep their profiles up to date, including updates of personal details such as passwords and contact information.

Administrators are accountable for the maintenance of ICMS, undertaking activities such as database backups, restoring old information, and adding staff to the system. ICMS eliminates wasted time and resources and ensures significant documents are preserved and easily accessible by automating these manual activities.

Our aim with ICMS is to enrich Ambo University students, educators, and employers through the utilization of a world-class, effective platform that bridges academic education with professional experience. This project not only addresses short-term internship and career management needs but also paves the way for an increasingly connected and opportunity-driven future for all.

## 1.2. Statement of the Problem and Justification

Since its founding in 1939 E.C. (1947 G.C.), Ambo University has grown into a leading institution, delivering diverse academic programs to a growing student body. With internships now central to many programs, mirroring a nationwide rise in experiential learning, the university faces mounting pressure to manage these opportunities efficiently, supporting students eager to build their futures.

Yet, the university’s manual processes for internship and career management create significant inefficiencies. Students struggle to track applications and submit documents, slowed by paper-based systems that lead to delays and errors. Faculty advisors and company supervisors lack a centralized platform to review applications or share feedback, causing communication gaps and misplaced records. Administrators spend excessive time managing tasks like posting internship opportunities or verifying student data, as information remains scattered in physical files. These challenges frustrate stakeholders and hinder timely access to internship opportunities.

The Internship and Career Management System (ICMS) provides a web-based solution to streamline these processes. Through a centralized platform, ICMS enables students to submit applications, upload documents, and receive real-time updates. Faculty and supervisors can review submissions and provide feedback efficiently, while administrators manage accounts and post opportunities with ease. By automating workflows, ICMS saves time, reduces errors, and fosters collaboration, empowering stakeholders to meet the rising demand for internship opportunities at Ambo University

## 1.3. Objective of the Project

### 1.3.1. General Objective

The general objective of this project is to develop an automated web-based system for managing internships, career opportunities, and communication between students, faculty, and employers

### 1.3.2. Specific Objectives

* To build a secure database for storing student records, company info, and application data.
* To develop a user-friendly web system for student registration and internship/job applications.
* To simplify the application process by enabling companies to post positions and manage candidates.
* To enable file sharing and feedback between students, faculty, and company supervisors.
* To support communication through attachments, notifications, and status updates.
* To provide admin tools for managing accounts and posting announcements.

## 1.4 Methodology

### 1.4.1 Requirement Gathering method

The following primary and secondary methods of data collection are used.

**Interview:** - Interviews of stakeholders and users are critical to creating the great software. Without understanding the goals and expectations of the users and stakeholders, we are very unlikely to satisfy them. We also must recognize the perspective of each interviewee, so that we can properly weigh and address their inputs. Listening is the skill that helps a great analyst to get more value from an interview than an average analyst.

Firstly, we interviewed the Heads of different departments of Ambo university. And, the Coordinator of the industrial linkage office gave us some information about students, Rules of internship like, duration of intern program, how students apply for intern and how they are assigned.

**Observation:** we observed the total process of how students assigned for intern program as we are the parts of the program last year.

**Document analysis:** - we also collected information about the way students assigned to intern. Also, we review guidelines for internship programs.

### 1.4.2 Requirement Modelling

We use Object-Oriented methodology because of the reasons that it implements the concept of Object-Oriented Programming (OOP), inheritance, encapsulation and polymorphism; the ability to challenging the problem domains; to make simple communication among users, analysts, designers and programmers.

The reasons that we use the object-oriented approach are:

* These techniques have a reusability feature.
* These techniques provide greater opportunities for users to participate in the development process.
* This increases flexibility.
* This also improved quality.
* We can inherit properties of the class that are defined in the super class.
* We can reuse methods to avoid redundancy.
* The data and functions are encapsulated in objects that help us with easily debugging purpose.
* Modification of the object implementation is easy.
* Understanding the structure is easy because object-oriented modelling represents real world entities.
* Direct manipulation of architectural components is possible because several object-oriented programming languages exist.

## 1.5. Tools

Table 1: List of Tools

|  |  |
| --- | --- |
| **Hardware Tools** | **Software Tools** |
| - Server (for data storage, request handling, and response generation) | Application Requirements: - Web browsers supporting Groovy/JavaScript (Maxthon, Firefox, Chrome) - OS compatibility: UNIX, Linux, Windows, Mac |
| - User devices: - Android smartphones - Computers (Windows/Mac/Linux) | External APIs: - MySQL API (for database communication) |
|  | **Development Environment:** - OS: Windows - IDEs: Visual Studio Code, Notepad++, Sublime Text - **Design tool**: Wonder share EdrawMax, Draw IO **- Programming language:**  Frontend development (HTML, CSS, PHP, JAVASCRIPT). Backend (MySQL). |

## 1.6. Scope and Limitation

### 1.6.1 Scope of the project

This project focuses on designing and implementing web-based internship management

system for Ambo University to provide a university with a friendly set of webs pages that are easy to navigate and at the same time provides sufficient depth and information about the system and how it works. By using this system, students will perform online registration process to get the application letter from the coordinator. Since the process can be done anytime and anywhere, students can save their time. They can focus on their lessons instead Spending huge amount of time with internship application matters.

The project will cover the following activities:

✓ Allows students to apply online for intern

✓ Allows Students only to apply career.

✓ Allows companies to post their intern positions

✓ Manage all users of the system.

✓ Allows Students to report of every activity during training.

✓ Sending the notification system via e-mail to both students, company supervisor and

Administrator

### 1.6.2 Limitation of the project

The project cannot provide the following functionality:

* Our system can’t provide online interviews and payment process
* Our system can’t provide other languages (i.e., our system will be developed by English language).
* Our system does not help visually impaired people
* Our system does not work only for Ambo university students only

## 1.7 Significance of the Project

Now a days we are living in an information age, so everything has changed from the manual

system to automated system, which makes everything simple, interactive, time saving and

requires less storage space for allocating resources. The manual system has so many drawbacks

like as in above expressed. So, automating the system has significance like: -

✓ To save time and resources needed.

✓ To easily manage and control the system.

✓ Reduce the problem facing the student when taking an internship and requires job.

✓ This automated system reduces the load from the advisor and enables access to information easily about the host organization and the student taking internship.

✓ To provide immediate and updated information for the users.

✓ To store individual information and manage information permanently.

✓ To have an effective and efficient operations for the organization to the user.

**Target Beneficiaries & Their Benefits**

1. Ambo University

Tangible Benefits:

* Cost Savings: Reduces expenses on paper, printing, and manual documentation
* Operational Efficiency: Automates internship processes to reduce administrative workload

Intangible Benefits:

* Enhanced Reputation: Projects modern image through digital internship platform
* Improved Productivity: Streamlines workflows for staff and faculty advisors

2. Companies (Internship Providers)

Tangible Benefits:

* Lower Recruitment Expenses: Saves time and resources through digital intern selection

Intangible Benefits:

* Better Talent Access: Easily connects with qualified Ambo University students
* Improved Employer Branding: Enhances perception as tech-savvy organization

3. Faculty Advisors

Tangible Benefits:

* Time Savings: Automates student allocation/deallocation and performance tracking
* Reduced Paperwork: Digital submissions replace manual report handling

Intangible Benefits:

* Improved Mentorship: Enables communication with students/companies
* Job Satisfaction: Simplifies supervision tasks

4. Students

Tangible Benefits:

* Cost Savings: Eliminates transportation and printing expenses
* Convenience: Apply for internships anytime without physical visits

Intangible Benefits:

* Career Readiness: Improves employability through timely opportunities
* Confidence & Trust: Secure system enhances user satisfaction
* Skill Development: Prepares for modern workplaces through digital experience

5. Project Developers

Tangible Benefits:

* Financial Gain: Potential earnings from implementation/maintenance

Intangible Benefits:

* Skill Enhancement: Gains technical and project management expertise
* Reputation Boost: Recognition for developing an impactful university system in the morale of our team.

## 1.8 Feasibility

The aim of our project is to objectively and really uncover the strength and weaknesses of the existing business or proposed opportunities and threats as presented by the environment

resources required or materials and any information or experience to the management of

internship.

As much as the university communicates by English language, this project will work in English language only. So, we don’t have to use Cross Natural language and Speech Synthesizing (sound) machine. Therefore, it is feasible.

**Economic Feasibility**

The system to be developed is economically feasible and the benefit outweighs the

cost. Since this project already computerizes the existing system, by now the reduction of

cost for materials used in manual operation becomes beneficiary to the organization.

**Political Feasibility**

Political feasibility is a measure of how well a solution to a policy problem will be accepted by a set of decision makers and the public. For a policy to be enacted and implemented, it must be politically acceptable, or feasible. So, the developed system is protected by law not duplicated without the permission of the developed team and not re-med it.

**Technical Feasibility:**

We system developers understand the scope, objectives including specific objectives and

limitations of the proposed system well and the users have technical capability to use this system.

As a result, we develop the website for internship and career successfully within proposed

resources (budget, time, etc.), so the project is technically feasible.

**Operational feasibility:**

The new system can provide sufficient service for the students, there was bulky process in

giving service in manual processing like, the student or intern's candidate need to fill a paper form and need to submit it by hand to the office. During the training, candidates need to keep updating in the training by writing it and paste any attachment in the schedule book. Finally, progress of the training in the schedule book will be examined by the supervisor in the company every week, and by the university supervisor at the end of the training by writing it in the schedule book. This implies that the students cannot be satisfied with the service of finishing the internship process. But this proposed system is automated consequently the student can get sufficient service meaning that facilitates the process that saves student’s time. The system is operationally feasible as it quite easy for the End users to operate it. It only needs basic information about Windows platform.

## 1.9. Risk Assessment

Risk assessment is a critical part of the project planning process. It involves identifying potential risks that could impact the successful completion of the project and developing strategies to mitigate or manage those risks. Below is a detailed risk assessment for the Web-Based Internship and Career Management System (ICMS):

Technical Risks:

Risk: System failure due to inadequate server capacity or software compatibility.

Mitigation: Conduct rigorous load testing and compatibility with multiple browsers and operating systems.

Security Risks:

Risk: Insecure access to sensitive student or company data.

Mitigation: Implement encryption, secure authentication mechanisms, and security audits conducted on a regular basis.

User Adoption Risks

Risk: User resistance (students, instructors, businesses) because of unfamiliarity with the system.

Mitigation: Provide thorough training sessions and user-friendly documentation.

Budget and Timeline Risks:

Risk: Delays in development or unexpected expenses due to scope creep.

Mitigation: Remain committed to Agile methodologies, prioritize core functionalities, and maintain a close budget oversight.

Internet Dependency:

Risk: System non-availability in low-internet areas.

Mitigation: Provide offline functionality for form submission and synchronize data upon connectivity return.

## 1.10 Work Breakdown Structure (WBS)

|  |  |  |
| --- | --- | --- |
| Phase | Tasks | Responsible Team Members |
| 1. Project Initiation | - Define project scope and objectives. - Conduct feasibility analysis. | Abdi, |
| 2.Requirement Gathering | - Interview stakeholders (students, faculty, companies). - Document functional and non-functional requirements. | Naol, Ephrem |
| 3. System Design | - Create UML diagrams (use case, sequence,activity). - Design database schema and ER diagrams. | Abdi, Mihret, Bontu |
| 4. Development | - Frontend development (HTML, CSS, PHP). - Backend integration (MySQL, APIs). - Implement security features (authentication, encryption). | Abdi, Naol, Ephrem, Bontu, Mihret |
| 5. Testing | - Perform unit testing for modules. - Conduct integration and user acceptance testing (UAT). | Ephrem, Bontu |
| 6. Deployment | - Host the system on a server. - Configure domain, SSL, and firewall settings. | Mihret, Abdi |
| 7. Training | - Prepare user manuals with screenshots. - Conduct workshops for students, faculty, and company supervisors. | Naol, Abdi |
| 8. Maintenance | - Monitor system performance post-deployment. - Address bug reports and roll out updates. | Bontu, Mihret |

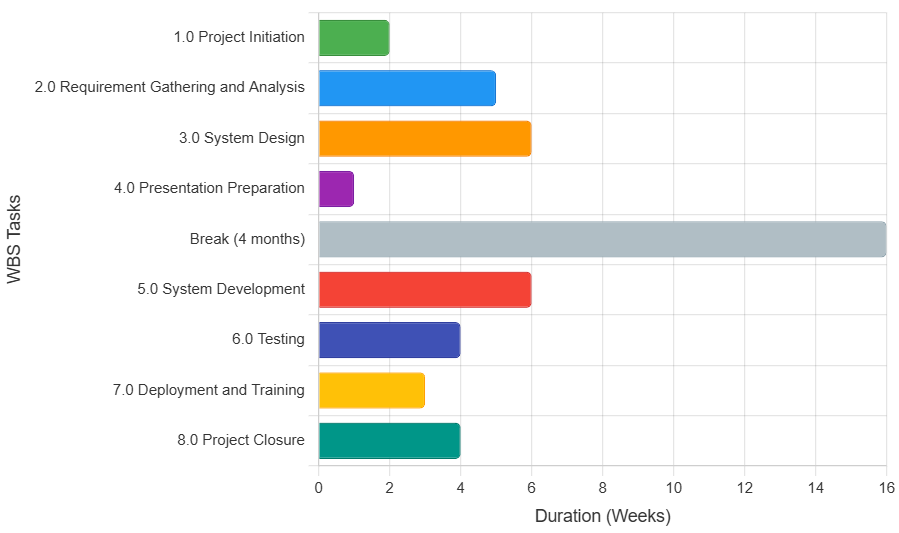
****

Figure 1 : Work breakdown structure

# CHAPTER TWO

# System Requirement Specification

## 2.1 Current system

The Existing system refers to the system that is being followed still now. The existing system is the manual system that uses a paper-based documentation system for performing various tasks that are related to internship management system. The advisor gets company or host organization information by phone calling and visiting. The student also accesses company/host organization information after attending the internship and when taking internship, the student must provide the paper given from university to host organization or company.

**Problems of the Existing System**

The problems of the existing system are the following:

* There is a huge communication gap between the advisors and students during the internship program.
* It takes huge budget and human resources by advisor to visit student attend internship.
* Individual students are not able to give feedback since the existing system does not  
  Provide such facility.
* The students are not able reached to respective organization.
* The existing system takes a lot of time to get information about host organization or company

## 2.2 Business Rules

A business rule is effectively operating principle or polices that must be fulfilled and obligated in order the system to function properly and effectively. It often pertains to access control issues, business calculations, or operating polices and principles of the organization. Some of the rules are the following: -

Table 2:Business rules

|  |  |  |
| --- | --- | --- |
| **ID NO** | **NAMES** | **DESCRIPTION** |
| 01 | Valid User | Anyone who uses the system should be an Ambo University student or members of Ambo university who take internship program |
| 02 | Registration to system | Supervisors of campus and students must register into the system to interact with each other |
| 03 | Apply Internship | Student must apply the internship before the allocation |
| 04 | Select company | The students select company based on the field of study area and select one company |
| 05 | Assign advisor | Supervisor can assign advisors for students |
| 06 | Attach file | The students, advisors, company supervisors can attach file to improve daily activity done by the students |

## 2.3 Proposed system

### 2.3.1 Overall Description

Internship and Career Management System consider the main problem of students and organization to find each other, meaning that when students searches for internship and job the organization requires candidates for internship and available job position. So, this project will try to bring both students and organization in the way that students are apply online internship or career opportunities and organization to find candidates online.

The system is to provide an easy-to-use interface for students and faculty members to interact with each other and in a better way how students can apply online internship or career at anytime and anywhere regardless of whether present to organization or not. It also works by allowing the university to deploy a web portal, to collect applicants’ data, review that data, decide, and then continue to interact with applicants and reviewers.

By using this system, students will perform online registration process to get the application letter from the coordinator. Since the process can be done anytime and anywhere, students can save their time. They can focus on their studies instead of spending time on internship application matters.

### 2.3.2 Functional requirements

The system shall provide the following functional capabilities:

1. **User Profile Management**
   * REQ1: The system shall allow users to update their background information.
2. **Administrative Account Control**
   * REQ2: The system shall allow administrators to view and delete inactive or unnecessary user accounts.
   * REQ3: The system shall allow administrators to post news announcements.
3. **User Registration**
   * REQ4: The system shall allow new users to register accounts.
4. **Document Management**
   * REQ5: The system shall enable students to submit files to both company and faculty supervisors.
   * REQ6: The system shall allow faculty supervisors to view files submitted by students or company supervisors.
   * REQ7: The system shall allow company supervisors to view files submitted by students or faculty supervisors.
5. **Internship/Job Application Processing**
   * REQ8: The system shall enable students to apply for internship training and job opportunities.
   * REQ9: The system shall allow company supervisors to approve or reject internship/job applications.
6. **Communication Features**
   * REQ10: The system shall display notifications to users in real-time.
   * REQ11: The system shall allow users to provide comments and feedback.
   * REQ12: The system should enable company supervisors to send application status notifications to students.

### 2.3.3 Nonfunctional requirements

Non-functional requirements are additional quality of system in terms of the following issues.

* **User Interface**
* The system GUI should be consistent with all other programs.
* The caption and text of GUI should be clear to understand and clear descriptive one.
* The GUI should be accompanied with the help of files that describe the usage of each GUI.
* The GUI should be easily customizable.
* **Economy**
* The system will reduce costs.
* The system should give maximum benefit with minimum time and minimum resources.
* **Reliability**
* The system provides the user with correct information.
* When the user entered wrong inputs, it notifies to correct the input data
* **Performance**
* Our proposed system should provide 7x24 hours’ access for the users.
* The proposed system runs on any kind of operating system
* After the user information is submitted the response time must be fast enough
* The system shall accommodate number of users at a time.
* **Security and Access permissions**
* Our system is secure, which means unauthorized body cannot damage system and system resource
* The system is secured due to the username and password for administrative activity.
* **Usability:**
* The system is easy to use and understandable.
* The system is free to use.
* **Availability:**
* The system is available to the user as a web design application.
* Authorized users can access the system easily.
* The system has backup if there is a failure.
* **Backup and Recovery:**
* It is easy to recover data stored on the server in case of unexpected server failure.
* The first option to backup data is using distributed database.
* **Integrity:**
* The system should be secure and must use encryption to protect the databases.A. User requirements

The proposed system has one side i.e., website:

* The GUI of the website must be responsive to have ease of access
* Should contain attractive user interface throughout the page
* The website should have a special login interface for the user of the website
* The website should have an attractive and easily manageable interface after login
* The website should display warning messages in order prevent users from unnecessary stress and loss of data
* The website shall protect any wrong entry of the user
* The website shall interact with the server as fast as possible
* The color used in the GUI should provide user with safe vision and should not be sharp to the eye.

### 2.3.4 System models

System modeling is the process of developing abstract models of a system, with each model representing a different view or perspective of that system. Meaning that, representing a system using some kind of graphical notation based on unified modeling language (UML) which helps to understand the functionality of the system and models are used to communicate with customer. These system models are

* Use Case diagrams
* Sequence diagrams
* Activity diagrams
* Class diagrams
* State diagrams

**Actors**

Table 5:Actors

|  |  |
| --- | --- |
| **User class** | **Description** |
| System Administrator | Admin is the one who controls the entire system.  Admin has the right to control the system database and able to alert the records, view profiles and approves any application of the students can deny or permit any services for users of the system. |
| Student | Student is anyone who is Students of university who is applying for interns or career.  Can anybody who is looking for intern and jobs like employees within other organizations. |
| Faculty advisor | Lecturers who advise students during their internship. |
| Company supervisor | People who supervise students during their internship |

#### 2.3.4.1 Use case model

These diagrams show the interaction between a system and its environment. Each use case represents a discrete task that involves external interaction with a system. Diagrammatically represented to provide an overview of the use case and in more detailed textual form. Actors in a use case may be people or other system. In our system we have the following actors

* System Administrator
* Company supervisor
* Faculty advisor and
* Student

Each use case diagram describes a behaviorally related sequence of transaction in a dialogue between the system Administrator, company supervisor, faculty advisor and Student

Figure 2:ICMS Use Case Diagram

ICMS

Manage Account

Reset Password

view user

Sign up

Login

Post News

Attach file

View notification

Manage

profile

Logout

Apply Career

Apply internship



Admin

Student

Company Supervisor

Faculty

Advisor

<<include>>

<<include>>

<<include>>

<<include>>

<<include>>

<<include>>

<<include>>

<<include>>

<<include>>

<<include>>

<<include>>

Table 6:Login Use Case Description Login Use Case Description

|  |  |  |
| --- | --- | --- |
| Name | Login | |
| ID | UC1 | |
| Actors | Admin, Student, Faculty advisor and Company supervisor | |
| Description | To permit the user to access the system | |
| Priority | High | |
| Pre-condition | The system must be loaded successfully | |
| Post condition | Home screen will be displayed to the user | |
| Basic Course of Action | **Actor Action**  1 Open login interface  3 fill username and password  4 click login button  8 Use Case end | **System Response**  2 loads login interface  5 validate information  6 verify users  7 display home screen pages |
| Alternate course of action | 5.1 If the user inserts invalid username and password the system must display error message and return them to step 3  6.1 If the user inserts wrong password for available user, the user will be notified that he/she entered wrong password and returned to step 3 | |
| Include | Login | |
| Note/Issues | The system must display appropriate errors and warning messages for the user. | |

Table 7: Signup (Register) Use Case description

|  |  |  |
| --- | --- | --- |
| Name | Signup | |
| ID | UC2 | |
| Actors | Student, Faculty advisor and Company supervisor | |
| Description | To permit the user to register into the system | |
| Priority | High | |
| Pre-condition | The system must be loaded successfully | |
| Post condition | Home screen will be displayed to the user | |
| Basic Course of Action | Actor Action  1 Open URL on web browser  3 Click on signup button  5 Fill required information from page  7 Click save button  9 Use Case end | System Response  2 Login screens with Signup button will be displayed.  4 Display signup screens  6 check validity of information  8 Successfully registered message display |
| Alternate Course of Action | 6.1 If the user unfilled required information the system must display error message and return them to step 5 | |
| Note/Issues | The system must display appropriate errors and warning messages for the user. | |

Table 8: Apply internship Use Case Description

|  |  |  |
| --- | --- | --- |
| Name | Apply internship | |
| ID | UC3 | |
| Actors | Students | |
| Description | To allow the students to apply for internship (sent application form) to company | |
| Requirement Traceability | The student should be registered on the system | |
| Priority | High | |
| Pre-condition | The user must be logged in successfully to the system | |
| Post condition | Accepted or rejected notification will be displayed according to the company click either of the accept or reject button and company supervisor request sent notification will be displayed for the student by the system | |
| Basic Course of Action | Actor Action  1 Click on internship program menu from home page  3 View and select program to attend  4 Flicks apply buttons  6 Fills and click send button  7 Use Case End | System Response  2 Display available programs  5 Apply forms display |
| Alternate Course of Action | 4.1 If there is no position in company and apply duration is expires notify and return to step 3 | |
| Include | Login | |

Table 9: Apply job Use Case Description

|  |  |  |
| --- | --- | --- |
| Name | Apply job | |
| ID | UC4 | |
| Actors | Students | |
| Description | To allow the student to send a job application form to a company or organization for available job vacancies. | |
| Requirement Traceability | The student should be logged into the system. Registration is not mandatory if job postings are publicly accessible. | |
| Priority | High | |
| Pre-condition | The user must be logged in successfully to the system or access public job postings. The system must display available job vacancies. | |
| Post condition | Notification of application sent displayed to students. Company supervisor receives application and sends accept or reject notification. | |
| Basic Course of Action | Actor Action  1. Click on career opportunities menu from home page.  3. View and select job vacancies to apply.  5. Click apply button.  7. Fill application form and attach documents.  9. Click the send button.  11. Use Case End | System Response  2. Display available job vacancies.  4. Show selected vacancy details.  6. Display application form.  8. Validate form and documents.  10. Confirm submission and notify company supervisor. |
| Alternate Course of Action | 2.1 If no job vacancies available, display “No vacancies” and end use case.  6.1 If registration is required and user not logged in, redirect it to the signup page.  8.1 If form is incomplete or documents are invalid, notify and return to step 7. | |
| Include | |  |  | | --- | --- | |  | Login (optional) | | |

Table 10: Attach Files Use Case Description

|  |  |  |
| --- | --- | --- |
| Name | Attach Files | |
| ID | UC5 | |
| Actors | Student, faculty supervisor and company supervisor | |
| Description | To allow the user to send and receive files of any format and size to or from another user of the system respectively | |
| Requirement Traceability | The user must have permission to which he or she wants to either send or receive a file. | |
| Priority | High | |
| Pre-condition | The user must be logged in successfully to the system | |
| Post condition | Successfully sent messages will be displayed to the sender. | |
| Basic Course of Action | Actor Action  1 Click on attach file menu from the user account page  3 Click send files button  5 Select and click upload button  7 Click the attached button  8 Use Case end | System Response  2 Display text boxes with send files button  4 Navigate through local files with upload and cancel button  6 Display uploaded files |
| Alternate course of action | 6.1 if file does not upload successfully or incorrect format notify and return them to step 3 | |

Table 11: View Notification Use Case Description

|  |  |  |
| --- | --- | --- |
| Name | Notification | |
| ID | UC6 | |
| Actors | Admin, Student, Faculty advisor and Company supervisor | |
| Description | To allow the user to view some information and file from other users of the system. | |
| Priority | High | |
| Pre-condition | The user must be logged in successfully to the system. | |
| Post condition | View notification | |
| Basic Course of Action | Actor Action  1 Click on notification menu from user account screen  3 click on notification want to view  5 use case ends | System Response  2 displays received information and data in ascending order of received date  4 display detail parts of clicked notification |
| Alternate course of action | 2.1 Nothing is displayed if there is no received information  3.1 Display no new notification | |

Table 12 : Post news Use Case description

|  |  |  |
| --- | --- | --- |
| Name | Apply posts | |
| ID | UC7 | |
| Actors | Company supervisor | |
| Description | To allow the company supervisor to post apply form and other information for the students | |
| Priority | High | |
| Pre-condition | The user must be logged in successfully to the system. | |
| Post condition | Posts announcement | |
| Basic Course of Action | Actor Action  1 Click on post apply menu from company supervisor screen  3 Select posts  5 End of use case | System Response  2 Display post apply form  4 Display posts |
| Alternate course of action | 2.1 If there’s no posts the wall displayed empty | |

Table 13: View user Use Case description

|  |  |  |
| --- | --- | --- |
| Name | Apply | |
| ID | UC8 | |
| Actors | System Administrator | |
| Description | To allow the System Administrator to view the details of registered users (students, faculty advisors, company supervisors) in the system. | |
| Requirement Traceability | The System Administrator must be logged into the system with appropriate privileges to access user information. | |
| Priority | High | |
| Pre-condition | The System Administrator must be logged in successfully to the system. The system must have registered users whose details can be viewed. | |
| Post condition | The System Administrator presents with the selected user’s details, including profile information (e.g., name, email, role) and account status | |
| Basic Course of Action | Actor Action  1. Click on the "Manage Accounts"  3. Select a user from the list of registered users.  5. Click the "View Details" button.  7. Review the user’s information.  8. Use Case End. | System Response  2. Display a list of all registered users with basic details  4. Highlight the selected user.  6. Display the user’s detailed information. |
| Alternate Course of Action | 2.1 If no users are registered, display a message: "No users available" and end the use case.  4.1 If the selected user’s data is inaccessible (e.g., due to database error), display an error message and return to step 3. | |
| Include | Login | |

Table 14: Manage profile use case description

|  |  |  |
| --- | --- | --- |
| Name | Manage Profile | |
| ID | UC9 | |
| Actors | Admin, Student, Faculty advisor and Company supervisor | |
| Description | To allow the user to view and edit his/her profile information | |
| Priority | High | |
| Pre-condition | The user must be logged in to the system | |
| Post condition | “Successfully changed” message will be displayed if user changes any information of his/her profile | |
| Basic Course of Action | **Actors Action**  1 Click on profile menu from his/her account page  3 Click update info button to change password  5 Insert new password  7 Click change button  9 Use case ends | **System Response**  2 Display registered information with update info button  4 Ask old password to update it.  6 Checks if it’s valid  8 Store new password |
| Alternate course of action | 4.1 If the old password does not match ask user to re-enter password return them to step 3  6.1 If the new password is poor ask the user to make it strong and return them step 5 | |

Table 15: Manage account use case description

|  |  |  |
| --- | --- | --- |
| Name | Manage Account | |
| ID | UC10 | |
| Actors | Admin | |
| Description | To allow admin to view and delete unneeded information and block unnecessary account or unblock necessary account. | |
| Requirement Traceability | The user must be only admin | |
| Priority | High | |
| Pre-condition | The admin must be logged in to the system | |
| Post condition | “Account block or unblock” and “successfully delete” message will be displayed if Admin delete(remove) any information of his/her page | |
| Basic Course of Action | Actor action  1 Click on manage account menu from admin page  3 Select account to manage (delete, modify)  4 Click on delete or update button  6 Click “yes” button  8 Use case ends | System response  2 Display the user account  5 Display warning messages (i.e. are you sure to delete or modify accounts)  7 Display successfully message |
| Alternate course of action | 4.1 Notifies admin if it select the right account | |

Table 16: Reset Password Use Case description

|  |  |  |
| --- | --- | --- |
| Name | Reset Password | |
| ID | UC11 | |
| Actor | Admin, Student, Faculty advisor and Company supervisor | |
| Description | Enables both the administrator and the users to reset their password when forgotten | |
| Priority | Medium | |
| Pre-condition | Signup required | |
| Post condition | Enabling the users and administrators to use the system | |
| Basic Course of Action | Actor Action (user)  1 Click reset password link  3 Answer the security questions  4 Fill the new password and confirm password  5 click reset button  8 end use case | System response  2 display security question  6 validate questions and passwords  7 stores updated information |
| Alternate course of action | 6.1 if the right information is filled up return users to step 3 and 4 | |

Table 17: Logout Use Case Description

|  |  |
| --- | --- |
| Name | Logout |
| ID | UC12 |
| Actors | Admin, Student, Faculty advisor and Company supervisor |
| Purpose | To disable the permission of accessing the account or the system |
| Requirement Traceability | Logged in users |
| Priority | High |
| Pre-condition | The users must be successfully logged in to the system |
| Post condition | The permission of accessing that account must be blocked until he/she logged in back again |
| Extends | Login |
| Basic course of action | 1.Click on logout button from his/her account |
| Alternate course of action | 1.1 warning if sure to leave pages |

#### 2.3.4.2 Sequence Diagram.

A part of UML used to model the interaction between the actors and the objects within a system. Also, shows sequences of interactions that takes place during particular use case. At the top of diagram objects and actors listed with a dotted line drawn vertically from these. Interaction between objects is indicated by annotated arrows.



**ICMS**

**I**

Figure 3: Sequence diagram for login



Figure 4: Sequence for Apply career



**ICMS**

Figure 5: Sequence Apply post



Figure 6: Sequence diagram for attach files



Figure 7: Sequence diagram for reset password

****

Figure 8: Sequence diagram for Manage Account



**ICMS**

Figure 9: Sequence diagram for Manage profile

**ICMS**



Figure 10: Sequence diagram for delete account

**ICMS**



Figure 11: Sequence diagram for view notification

**ICMS**

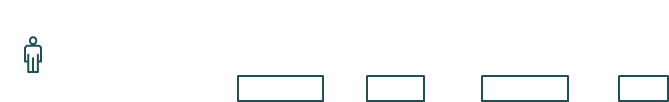


Figure 12: Sequence diagram for apply internship

**ICMS**



Figure 13: Sequence diagram for registration



**admin**

**Icms Admin**

**login form**

**Home**

**page**

**view**

**users list**

**controll**



start()

Display()

insert user name and password()

click login button()

**Database**



verify

Valid [check usename and password()]

incorrect syntax()

response()

valid or display()

invalid or display()

display()



**ICMS**

Figure 14: Sequence View user

#### 2.3.4.3 State chart diagram

A state chart diagram is a view of a state machine that models the changing behavior of a state. State chart diagrams show the various states that an object goes through, as well as the events that cause a transition from one state to another.

The common model elements that state chart diagrams contain are:

* States
* Start and end states
* Transitions

A state represents a condition during the life of an object during which it satisfies some condition or waits for some event. Start and end states represent the beginning or ending of a process.

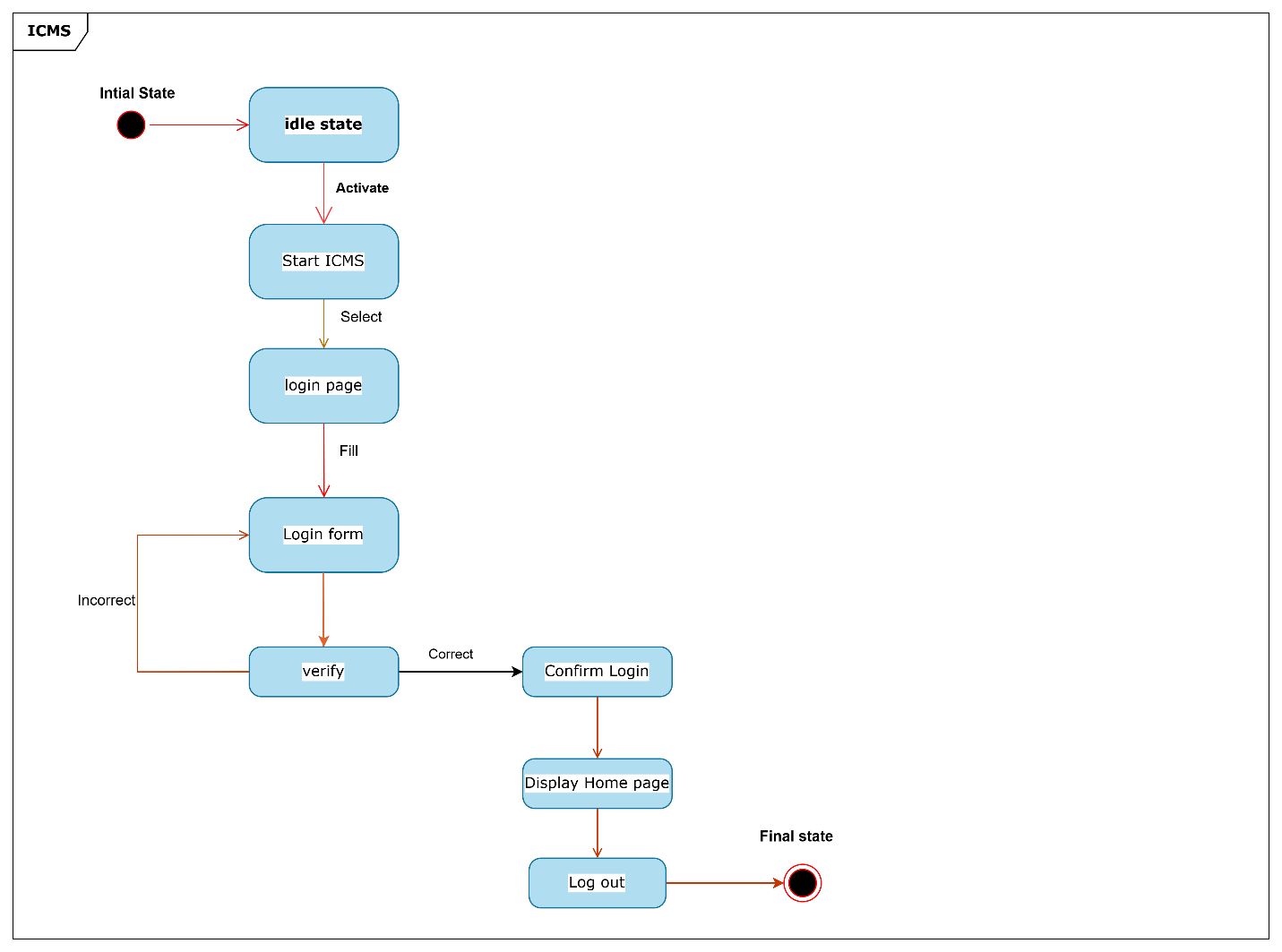


Figure 15: State chart diagram for log in

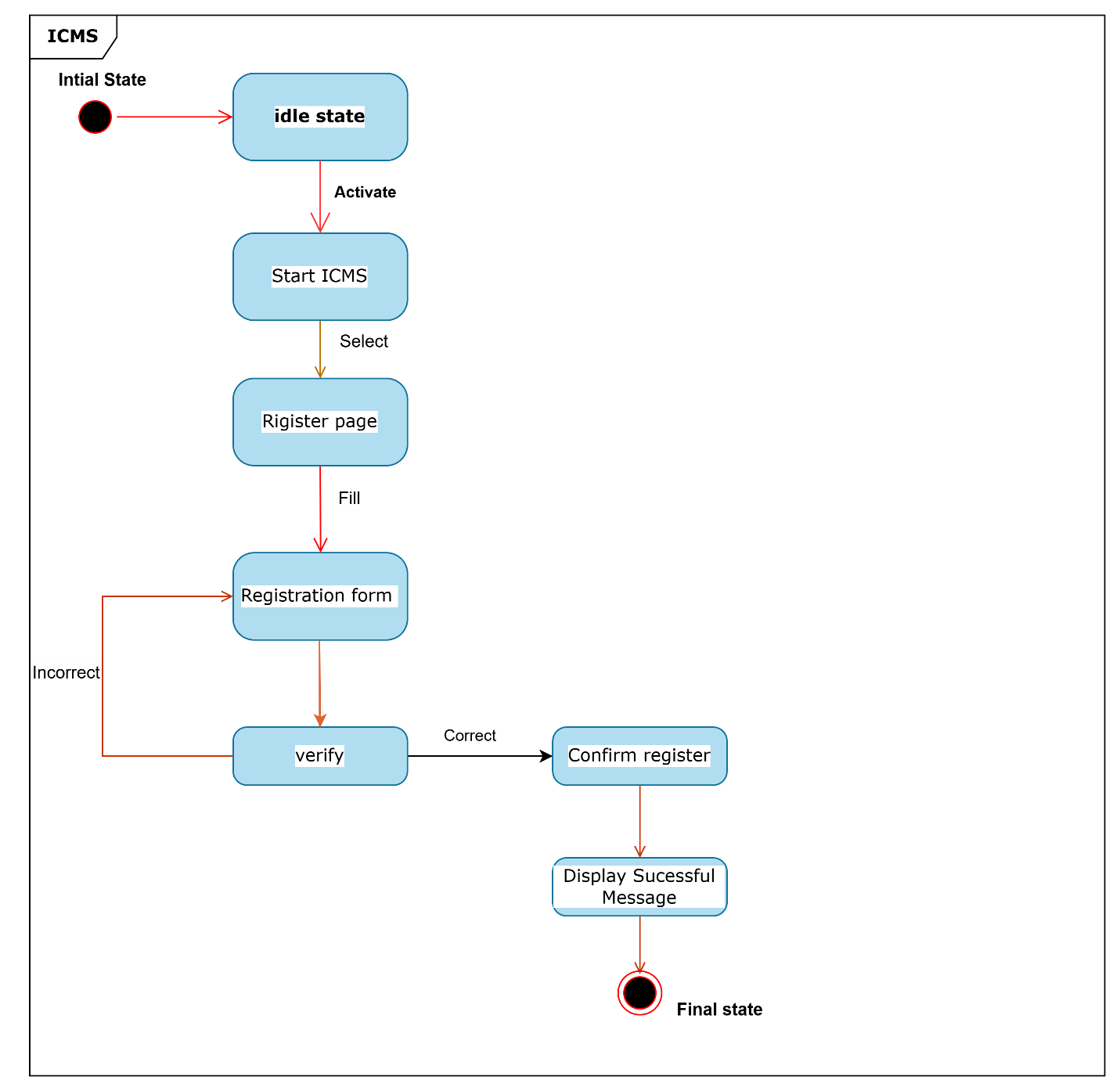


Figure 16: State chart diagram for log in

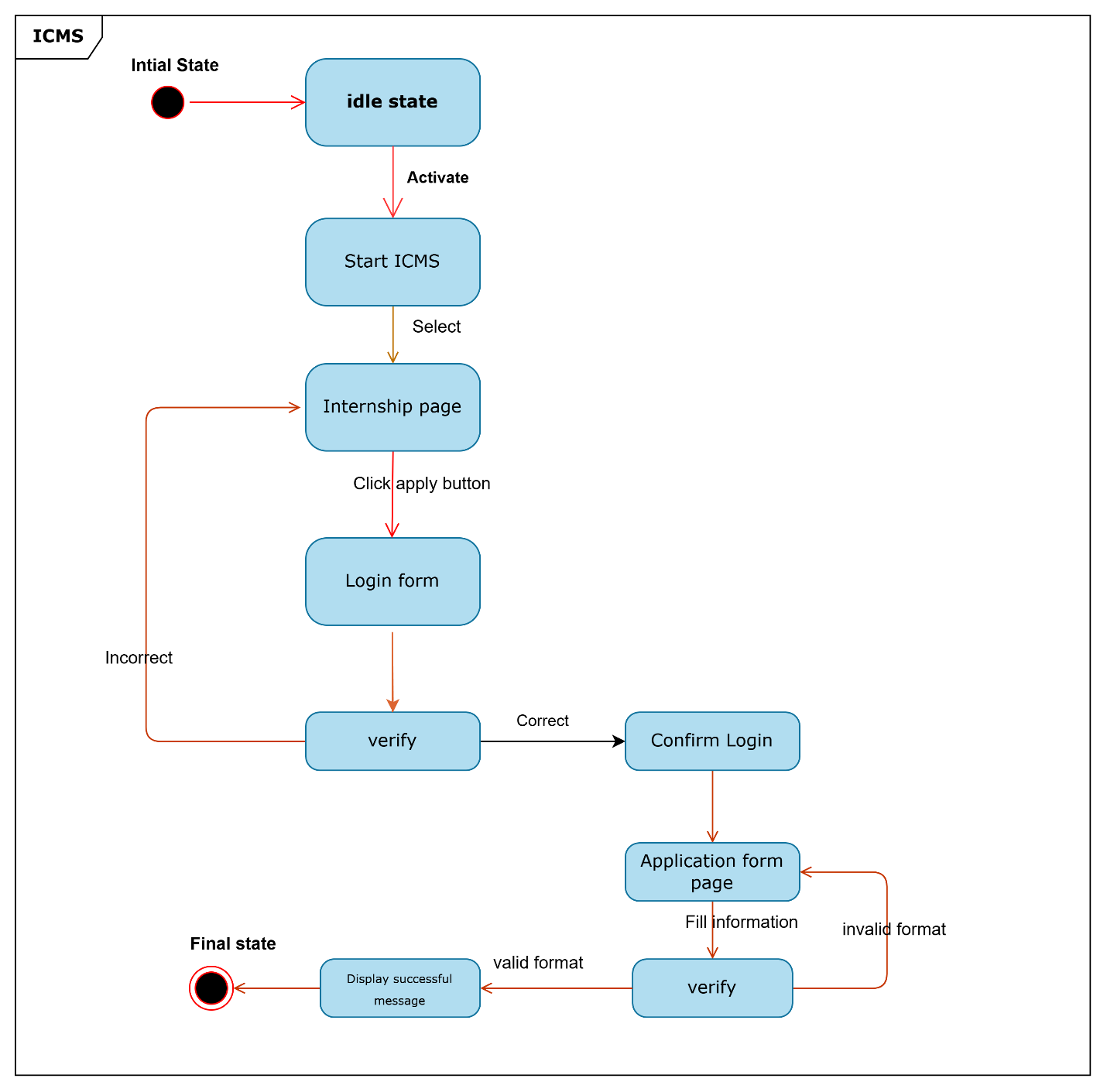


Figure 17 : State machine diagram for Apply internship

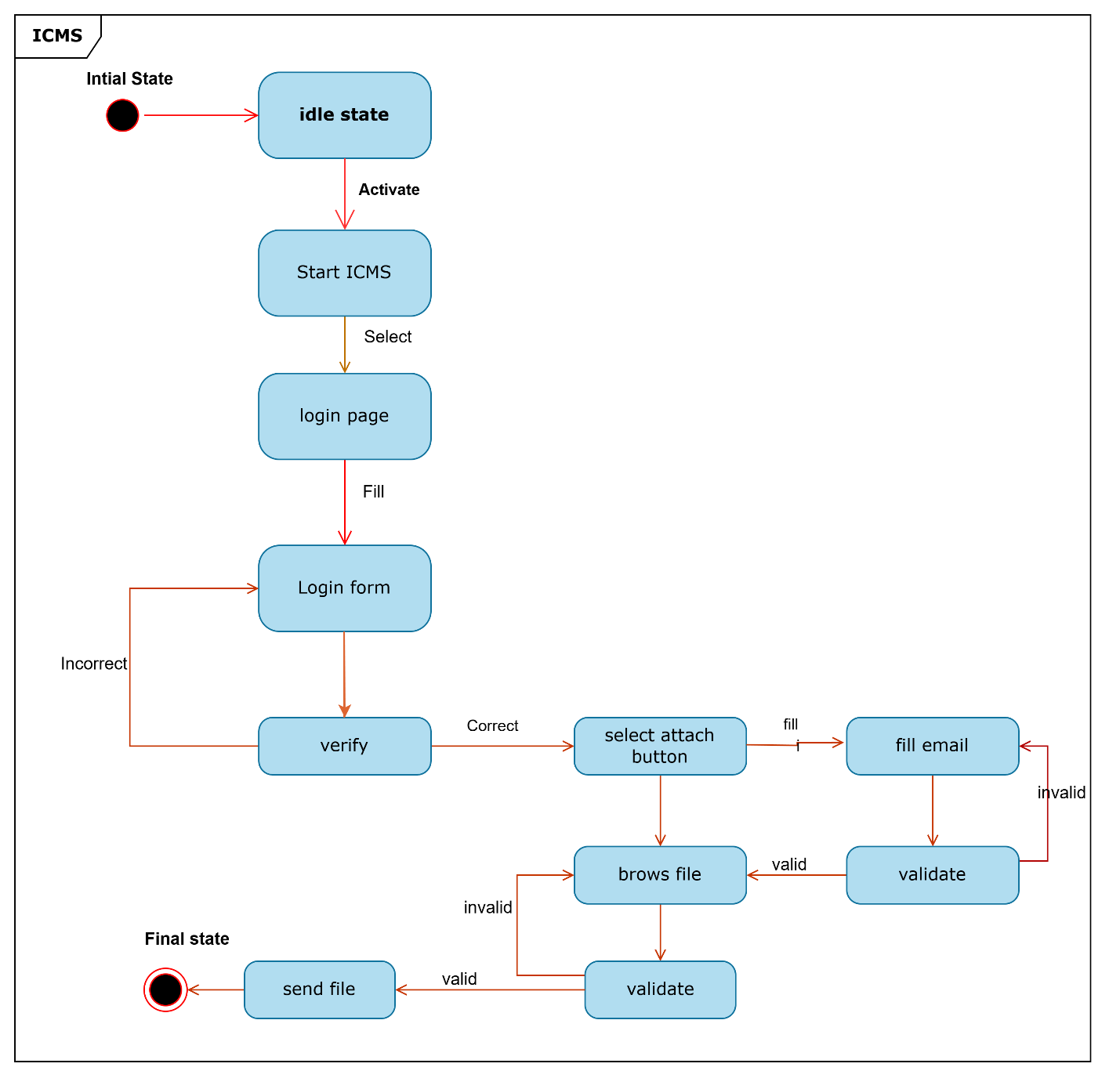


Figure 18: State machine diagram for Attach file

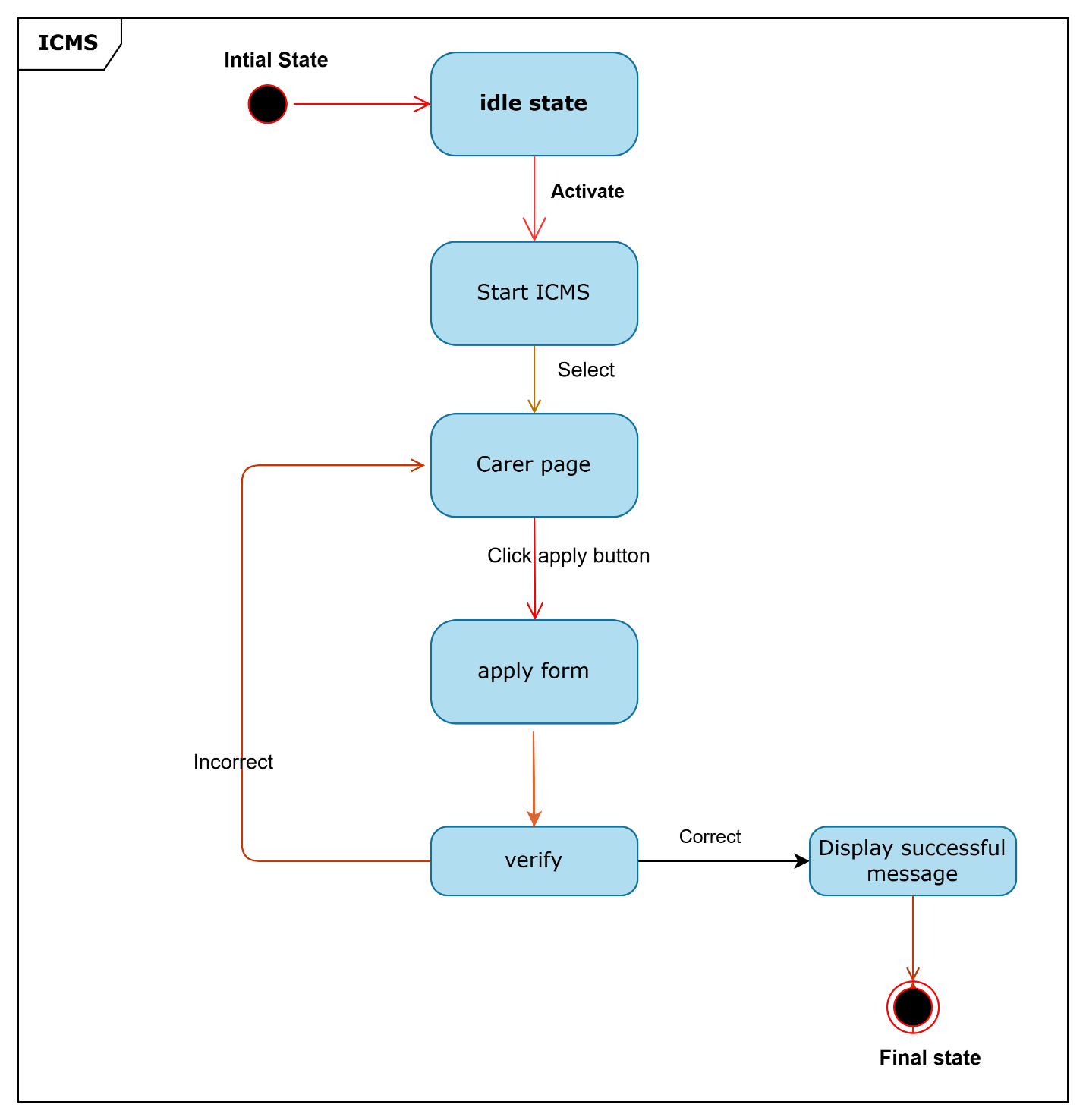


Figure 19: State machine diagram for apply carer

#### 2.3.4.4 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

Figure 20 : Activity diagram for apply internship



Figure 21 : Activity diagram for apply career



Figure 22 : Activity diagram for attach file

****

Figure 23 : Activity diagram for delete



Figure 24 : Activity diagram for edit profile



Figure 25 : Activity diagram for login



Figure 26 : Activity diagram for notification



Figure 27 : Activity diagram for post



Figure 28 : Activity diagram for Account admin



Figure 29 : Activity diagram for Change Password

****

Figure 30 : Activity diagram for registration

#### 2.3.4.5 Class Diagram

It represents the properties of entities, their operations and relationships. Also, it drives use case diagrams from use case. The class diagram is the main building block in our project modeling. It is used both for general conceptual modeling of the systematic of the application and for detailed modeling translating the models into programming code. Generally, the project includes the following class in the class diagram and the overview of the class diagram is: -



Figure 32 : Class Diagram

#### 2.3.4.6 User interface Prototyping

The GUI of the system can be shown in the desktop version. GUI play a great role in the success of the application. GUI is easy to understand and interact with GUI components, this application should provide an interactive and responsive interface for making it early accessed by the user. For this reason, we tried to make the interface

* The buttons and icons on every page shall be labeled with descriptive verbs.
* Menu items if present shall consist of elements (dropdown list) that are clearly associated with it.
* The color used in the GUI shall provide user with a safe version and shall not be sharp to eye.
* The interface should protect any wrong entry before they send their attachment or apply or reach to database.
* An application shall have a means to get back to login form by means of logout from any page.
* We used an interactive GUI component as much as possible for better understanding of the operation, a customized theme is applied for the system.



Figure 33 : User interface prototype

# CHAPTER THREE

# System Design

## 3.1 Introduction

The Internship and Career Management System that has been described in this document is a new version for the way that students are apply online internship or career opportunities.

The system is to provide an easy-to-use interface for students and faculty members to interact with each other and in a better way how students can apply online internship or career at anytime and anywhere regardless of whether present to organization or not. It also works by allowing the university to deploy a web portal, to collect applicants’ data, review that data, decide and then continue to interact with applicants and reviewers.

By using this system, students will perform online registration process to get the application letter from the coordinator. Since the process can be done anytime and anywhere, students can save their time. They can focus on their lessons instead of spending huge amounts of time with internship application matters.

## 3.2 Purpose of the System

Design Describes the important system qualities and defines the values against which options are evaluated (i.e., it describes the qualities of the system that developers should optimize). The objectives of design are to model the system with high quality. The implementation of high-quality system depend on the nature of design created by the designer. If one wants to change to the system after it has been put into operation it depends on the quality of the system design. So, if the system is designed effectively, it will be easy to make changes to it. The design goals are derived from the nonfunctional requirements. The main goal of the system design is to manage complexity by dividing the system into smaller manageable pieces. Some of the goal is listed below this Design goals describe the qualities of the system that the developers should consider.

## 3.3 Design Goals

The proposed system has one side i.e., website:

* The GUI of the website must be responsive to have ease of access
* Should contain attractive user interface throughout the page
* The website should have a special login interface for the user of the website
* The website should have an attractive and easily manageable interface after login
* The website should display warning messages in order prevent users from unnecessary stress and loss of data
* The website shall protect any wrong entry of the user
* The website shall interact with the server as fast as possible
* The color used in the GUI should provide user with safe vision and should not be sharp to the eye.

## 3.4 Current Software Architecture

Currently Ambo University internship system is not using software for which we can draw its architecture, because almost all transactions are performed manually

## 3.5 Proposed Software Architecture

Three-tier client-server architecture was chosen for ICMS due to its advantages compared to one-tier or two-tier architecture:

* + **Scalability:** Scalability is possible by the number of increasing users, by scaling a specific tier at a time such as server capacity increasing.
  + **High Performance:** The presentation tier buffers requests and, thus, reduces usage of the network and lessens the loads on the data and application tiers.
  + **Maintainability**: Each tier can be modified individually, without affecting other layers while updating the user interface, business logic, or database.
  + **Improved Data Integrity:** The data tier provides consistent and accurate data through structured storage.
  + **Enhanced Security:** The clients are not allowed to access the database directly, reducing unauthorized exposure of data.
  + **Ease of Modification:** Change in one tier does not affect others, making system updating simple.
* **Comparison with Other Architectures**
  + **One-Tier Architecture**: All layers (presentation, application, data) on one computer with a built-in database. Not appropriate for ICMS due to low scalability and the inability to cater to an enormous user base.
  + **Two-Tier Architecture:** Combinations of presentation and application layers or application and data layers onto one or two computers. Limits scalability and flexibility, thus not completely appropriate for ICMS's expected population.

Three-tier architecture is most appropriate for ICMS because it needs to support a large number of users and deliver good performance.



Figure 34 : ICMS Proposed software architecture

### 3.5.1 Subsystem decomposition

The Internship and Career Management System (ICMS) is organized into four core subsystems that form its three-tier client-server architecture, enabling efficient support for students, faculty advisors, company supervisors, and administrators at Ambo University. These subsystems—Web Browser, Access Control, Persistent Data Management, and MySQLi Database—collaborate to provide a secure, scalable, and intuitive platform for managing internship and career processes. Each subsystem handles specific tasks, from user interactions to data storage, as depicted in the Component Diagram (see Figure 35), ensuring modularity and seamless communication between clients and servers.

1. **Web Browser Subsystem:** As the presentation tier, this subsystem delivers the user interface, allowing users to interact with ICMS through modern web browsers such as Mozilla Firefox, Google Chrome, and Microsoft Edge. It presents dynamic web pages, built with HTML, CSS, and PHP, for tasks like registering accounts, updating profiles, submitting internship applications, and uploading documents. By sending HTTP requests to the application tier and displaying responses, the subsystem ensures an accessible and responsive experience for all users.
2. **Access Control Subsystem:** Operating within the application tier, this subsystem secures the system by managing user authentication and access. It assigns unique usernames and passwords to students, faculty, company supervisors, and administrators, verifying their identities before granting access to authorized features, such as viewing applications or managing accounts. Using PHP for authentication logic, it checks credentials against data stored in the MySQLi Database Subsystem, safeguarding user privacy and system integrity.
3. **Persistent Data Management Subsystem:** Also, part of the application tier, this subsystem processes and organizes data for ICMS’s core functionalities. It handles tasks like processing internship applications, managing vacancy postings, uploading and sharing files, delivering notifications, and posting news announcements. Implemented with PHP for business logic, it prepares data for storage and retrieves information as needed, communicating with the MySQLi Database Subsystem to ensure accurate and timely operations across the platform.
4. **MySQLi Database Subsystem:** Forming the data tier, this subsystem manages the storage and retrieval of all ICMS data using MySQLi, a robust and scalable database technology. It securely stores user profiles, application details, vacancy listings, uploaded files, notification messages, and administrative records. By processing SQL queries from the Persistent Data Management Subsystem, it efficiently writes, updates, and retrieves data, supporting the system’s reliability and performance for Ambo University’s diverse user base.

### 3.5.2 Component Diagram

The component diagram illustrates the software components within the ICMS application tier and their interactions with the presentation and data tiers. It provides a detailed view of how the subsystems identified in Section



Figure 35 : Component Diagram

### 3.5.3 Deployment Diagram

Deployment diagrams are used for describing the hardware components where software components are deployed. It would show what hardware components or node (Example web server, Application server, Data base server) exists and describes what software components (Example HTTP or HTTPS) run on each node

The deployment view of ICMS is used to describe which components are deployed on which device or hardware while deployment of the software is taking place. The components of ICMS are mainly placed on different devices like on computers.

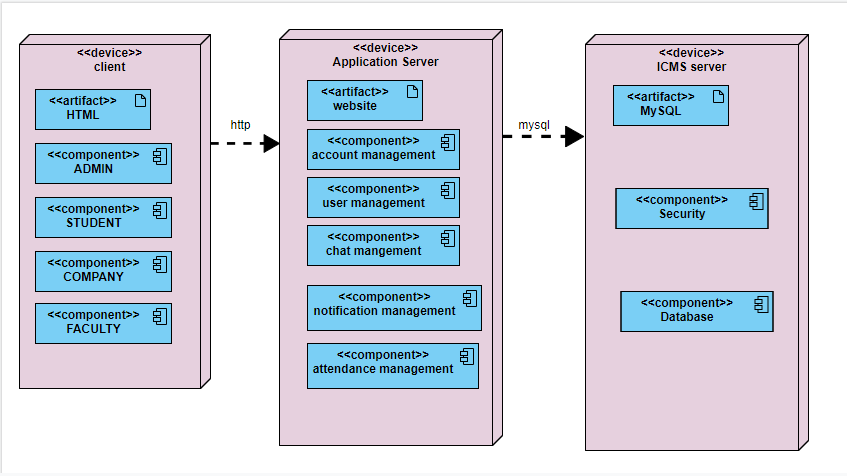


Figure 37 : Deployment Diagram of ICMS

### 3.5.4 Persistent Data Management

Persistent data management deals with how the proposed system is going to handle the actual data needs to be stored on the database of the system. The purpose of persistence modeling is to decide which objects in the system design are required to be stored persistently. Persistence of our object can be achieved by relational database since it is used as machine to make object persistent. It describes the persistent data aspect of software system. Our system includes the basic table that handles the data of system implemented using MySQL server

Table 18 : Student Account Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Size | Data type | Description | Constraint |
| Stu\_ID | 6 | Varchar | Identify row in entity | Primary key |
| User\_Name | 15 | Varchar | Unique name of account | Not-null |
| Stu\_Psw | 12 | Varchar | Password of student | Not-null |
| Email\_add | 25 | Varchar | Email address of student | Primary key |
| Stu\_FN | 20 | Varchar | Students name | Not-null |
| Stu\_LN | 20 | Varchar | Students’ father name | Not-null |

Table 19 : Faculty Account Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Size | Data type | Description | Constraint |
| Fac\_ID | 6 | Varchar | Identify row in entity | Primary key |
| User\_Name | 15 | Varchar | Unique name of account | Not-null |
| Fac\_Psw | 12 | Varchar | Password of faculty advisor | Not-null |
| Fac\_Email | 25 | Varchar | Email address of advisor | Primary key |
| Unv\_Name | 20 | varchar | Name of university | Not-null |
| Fac\_FN | 20 | Varchar | Faculty advisor name | Not-null |
| Fac\_LN | 20 | Varchar | Faculty advisor father name | Not-null |

Table 20 : Company Account Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Size | Data type | Description | Constraint |
| Cmp \_ID | 6 | Varchar | Identify row in entity | Primary key |
| User\_Name | 15 | Varchar | Unique name of account | Not-null |
| Cmp \_Psw | 12 | Varchar | Password of company account | Not-null |
| Cmp\_Email | 25 | Varchar | Official email address of company | Primary key |
| Cmp\_name | 20 | Varchar | Company name | Not-null |
| Cmp\_add | 20 | Varchar | Company addresses | Not-null |

Table 21 : Profile Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Size | Data type | Description | Constraint |
| User\_ID | 6 | Varchar | Identify row in entity | Primary key |
| Sex | 1 | Char | Sex of users | Not-null |
| Age | 2 | INT | Password of users | Not-null |
| Address | 25 | Varchar | Address of user | Primary key |
| Job | 20 | Varchar | Specific job of user | Not-null |
| Position | 20 | Varchar | Level of education | Not-null |
| Pro\_Pic | Max | BLOB | Picture of user | Not-null |
| Phone | 13 | INT | Phone number of users | Not-NULL |

Table 22 : Message Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Size | Data type | Description | Constraint |
| M\_ID | 6 | Varchar | Message\_id | Primary key |
| Sender\_Name | 15 | Varchar | Sender name | Not-null |
| Receiver\_name | 12 | Boolean | Receiver name | Not-null |
| M\_type | 25 | Varchar | Message type | Primary key |
| Sent\_Time | 20 | Varchar | Sent time of message | Not-null |
| Username | 20 | Varchar | Username of users | Foreign key |

Table 23 : Vacancy table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Size | Data type | Description | Constraint |
| Ann\_ID | 6 | Varchar | Identify row in entity | Primary key |
| Title | 15 | Varchar | Title of announcement | Not-null |
| Description | 255 | Varchar | Detail description | Not-null |
| Start date | 25 | Date | Start date of vacancy | Not-null |
| End date | 20 | Date | Expires of the vacancy | Not-null |
| Start month | 20 | Date | Start of intern | Not-null |
| End month | 20 | Date | End of intern | Not-null |

Table 24 : Applicants Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Size | Data type | Description | Constraint |
| App\_ID | 6 | Varchar | Id of letter of application | Primary key |
| App\_fn | 15 | Varchar | Name of applicant | Not-null |
| App\_ln | 15 | Varchar | Last name of applicants | Not-null |
| Email\_add | 25 | Varchar | Email address of student | Not-null |
| Gender | 1 | Char | Gender of applicants | Not-null |
| Age | 2 | INT | Age of applicants | Not-null |
| Address | 20 | Varchar | Address of applicant | Not-null |
| Cmp\_name | 20 | Varchar | Name of company | Not-null |
| Letter | 255 | Varchar | Letter to apply | Not-null |

Table 25 : Admin Account table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Field Name | Size | Data type | Description | Constraint |
| Ad\_ID | 6 | INT | Identify row in entity | Not null |
| Ad\_email | 25 | Varchar | Unique name of account | Primary key |
| UserName | 12 | Varchar | Password of student | Not-null |
| Ad\_psw | 12 | Varchar | Email address of student | Not-null |



Figure 39 : ICMS Persistent Diagram

### 3.5.5 Access Control and Security

Access control and security describes security issues, such as the selection of an authentication mechanism, the use of encryption, and the management of keys. Access controls are ways used to prevent unauthorized access of resources and used to achieve security goals i.e., confidentiality (data need to be hidden from unauthorized access), integrity (protected from unauthorized change), availability (the right person should access the right thing). We use the following access control mechanisms in particular:

**Username:** It is a name we use to be able to use a computer program or system. The user identification is that which is required to access the system.

**Password:** It is a secret word or phrase that you need to know to allow into a system. The password must be immediately preceded by the username. Proper authorization and authentication provisions have been made for the security of the site so that only the registered user can look at the special offer details. Without proper login, no one is allowed to access the special offer list of this site. In this Application/system, different actors have access to different functionality and data. The actors/users are administrator, student, faculty and company. The following table shows Actor and their privilege.

Table 27 : Access Control and Security

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Privilege | Actor | | | |
|  | Admin | Company | Faculty | Student |
| Register (sign up) |  |  |  |  |
| Account control |  |  |  |  |
| Login |  |  |  |  |
| View user |  |  |  |  |
| Mange profile |  |  |  |  |
| Attach file |  |  |  |  |
| View notification |  |  |  |  |
| Post news |  |  |  |  |
| Download file |  |  |  |  |
| Reset password |  |  |  |  |
| Apply career |  |  |  |  |
| Apply intern |  |  |  |  |
| Logout |  |  |  |  |

### 3.5.6 Global Software Control

Due to its nature as a web application, ICMS will be event driven. When a user is presented with a web page, they will have the option of clicking, double clicking and dragging. When a link is clicked, a function in the system management subsystem will be invoked. Functions in a subsystem may call other functions within that same subsystem or they may communicate with other systems using the interface provided in those subsystems.

### 3.5.7 Boundary Conditions

The Internship and Career Management System (ICMS) is built to connect students, faculty, supervisors, and administrators at Ambo University seamlessly, but like any system, it has its limits. Boundary conditions define how ICMS behaves when it starts, stops, or faces challenges like network issues or heavy usage. Below, we outline five key scenarios startup, shutdown, network failure, database overload, and unauthorized access attempts to show how ICMS handles these situations while keeping user experience and data integrity first. These conditions reflect the system’s design, as detailed in the Component Diagram (see Figure 35), ensuring reliability within its three-tier architecture.

1. **Starting the System**: When ICMS boots up, it’s like flipping the switch on a busy office. The web server, hosting the Access Control and Persistent Data Management subsystems, initializes PHP processes to prepare for user requests. The MySQLi Database subsystem connects to the database server, ensuring tables for user profiles, applications, and notifications are ready. The Web Browser subsystem waits for users to access the platform via browsers like Chrome or Firefox. If the database connection fails, ICMS displays an error message on the web interface, prompting administrators to check the server before users can log in or submit applications.
2. **Shutting Down Gracefully**: Shutting down ICMS is like closing the office for the day—everything needs to wrap up neatly. Administrators initiate a shutdown through the web server’s admin panel, signaling the Persistent Data Management subsystem to complete ongoing tasks, like saving an application or uploading a file. The Access Control subsystem logs out active users, and the MySQLi Database subsystem closes open connections to prevent data corruption. Once all processes stop, the web interface becomes unavailable, ensuring no new requests sneak in during shutdown.
3. **Handling Network Failures**: Imagine a student trying to submit an application when the internet drops—it’s frustrating, but ICMS is prepared. If the connection between the Web Browser subsystem on a user’s device and the web server fails, the browser displays a “Connection Lost” message, encouraging the user to try again later. The Persistent Data Management subsystem queues any incomplete requests (like a half-submitted form) and retries them once connectivity is restored. The MySQLi Database remains unaffected, safely storing data until the network is back online.
4. **Managing Database Overload**: During peak times, like when hundreds of students apply for internships at once, the MySQLi Database subsystem can feel the strain. To handle this, the Persistent Data Management subsystem limits simultaneous database queries, prioritizing critical tasks like saving applications over less urgent ones, like retrieving old notifications. If the load is too high, ICMS slows down, and users may see a “System Busy” message on their browsers. Administrators are alerted via the admin panel to monitor server performance and scale resources if needed, keeping the system stable.
5. **Blocking Unauthorized Access**: Security is a priority, so ICMS stands firm against unauthorized access attempts. If someone enters an incorrect username or password, the Access Control subsystem denies entry and displays a “Login Failed” message on the web interface. After multiple failed attempts, it temporarily locks the account, requiring the user to reset their password via an email link. The Persistent Data Management subsystem logs these attempts for admin review, ensuring the MySQLi Database remains protected from potential threats and only authorized users access sensitive data like applications or user profiles.