

**WEB-BASED STUDENT INFORMATION MANAGEMENT SYSTEM
(CASE STUDY: BUSITEMA UNIVERSITY, NAGONGERA CAMPUS)**

BY:

WEPUKHULU EDRISA

BU/UP/2023/1804

0782354310/0747905828

SUPERVISOR: DR. MUTUNGI FREDRICK

DEPARTMENT OF COMPUTER STUDIES

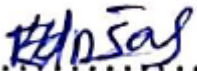
FACULTY OF SCIENCE AND EDUCATION

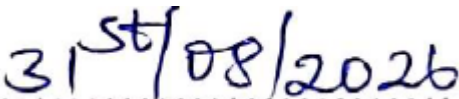
**A PROJECT REPORT SUBMITTED TO THE DEPARTMENT OF COMPUTER
STUDIES IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE
AWARD OF A BACHELOR'S DEGREE IN SCIENCE WITH EDUCATION AT
BUSITEMA UNIVERSITY**

AUGUST 2026

DECLARATION

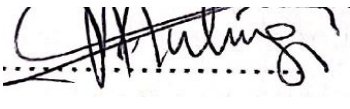
I **WEPUKHULU EDRISA** Registration Number **BU/UP/2023/1804** hereby declare that this project report is original and has not been published and/or submitted for any other degree award to any other university before.

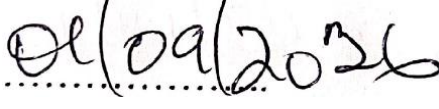
Sign 

Date 

APPROVAL

This project Report has been submitted for Examination and Approval by my supervisor.

Signed 

Date 

DR. MUTUNGI FREDRICK

Department of Computer Studies

Faculty of Science and Education

DEDICATION

This project is dedicated to **ALMIGHTY ALLAH** for making me be able to start and complete successfully in sound health. Also, to my Beloved mother **MUTONYI LAMULA**, friends like Amos, and all my course mates at campus for being there for me in times of need and for being supportive to me during this project work both financially and spiritually. Finally, to all my brothers and sisters for the immense support they rendered to me. May the almighty ALLAH reward you abundantly.

ACKNOWLEDGEMENT

First and foremost, I thank the **ALMIGHTY ALLAH** for the far He has brought me and the endless protection and guidance throughout this academic journey.

Special thanks to my selfless supervisor **DR. MUTUNGI FREDRICK** for his kindness, unwavering support, guidance, and invaluable feedback. I also extend my gratitude to the various lecturers in the department of computer studies.

Many thanks to my beloved Daddy and Mummy, **MR. MATASI HAMMY AND MS. MUTONYI LAMULA** for the economic, social, and emotional support they rendered to me throughout this course. May the almighty **ALLAH** reward you abundantly.

Of course, special thanks to my friends and course mates for your team support and insightful suggestions on this project. Thank you for making this journey worthwhile.

Finally, great thanks to Busitema University, Faculty of Science and Education, for the pleasant and serene environment that enabled me to acquire the academic knowledge and skills.

ABSTRACT

A web-based Student Information Management System (SMS) is a digital platform designed to manage student data, including personal details, academic records, course registration, and performance reports, accessible via the internet using a web browser.

In various educational institutions, managing student records efficiently is crucial for academic success and administrative effectiveness. The ICT Department at Busitema University's Nagongera Campus currently relies on fragmented manual and semi-digital methods, leading to time-consuming record retrieval, data inconsistencies, and insecure storage.

This study aimed at developing a web-based Student Information Management System for the ICT Department to replace the inefficient and fragmented methods of handling student records.

The major aim was to develop a system that is error-free, efficient, secure, and reduces paperwork and administrative burden in processing student academic information.

A mixed-methods approach was used for data collection, including interviews, questionnaires, and document analysis. The Rapid Application Development (RAD) methodology was employed to develop the system using PHP, HTML, CSS, JavaScript, and a MySQL database.

After development, the system was tested and it effectively met user needs, with 90% of users confirming its effectiveness and efficiency. Record retrieval time was reduced from an average of 25-35 minutes to under 2 minutes.

Thus, the study concludes that the system improves and eases student information management. I therefore recommend this system's adoption and use by the ICT Department at Busitema University and its potential scaling to other departments.

Table of Contents

<i>DECLARATION</i>	I
<i>APPROVAL</i>	II
<i>DEDICATION</i>	III
<i>ACKNOWLEDGEMENT</i>	IV
<i>ABSTRACT</i>	V
<i>CHAPTER ONE: INTRODUCTION</i>	1
<i>Introduction</i>	1
<i>Background of the study</i>	1
<i>Statement of the problem</i>	2
<i>Main objective</i>	2
<i>Specific objectives</i>	2
<i>Significance of the study</i>	3
<i>Scope of the study</i>	4
<i>Area scope</i>	4
<i>Content scope</i>	4
<i>Time scope</i>	4
<i>CHAPTER TWO: LITERATURE REVIEW</i>	5
<i>Introduction</i>	5
<i>Requirements for Student Information Management Systems</i>	5

<i>Theoretical Frameworks</i>	<i>5</i>
<i>Traditional Student Management Systems</i>	<i>5</i>
<i>Computerized Student Management Systems</i>	<i>6</i>
<i>Web-Based Student Management Systems</i>	<i>6</i>
<i>Research Gap.....</i>	<i>6</i>
<i>CHAPTER THREE: METHODOLOGY.....</i>	<i>7</i>
<i>Introduction</i>	<i>7</i>
<i>Research design and approach</i>	<i>7</i>
<i>Figure 2: Rapid Application Development (RAD) Model</i>	<i>7</i>
<i>Study area and population</i>	<i>7</i>
<i>Sampling techniques</i>	<i>7</i>
<i>Figure 1: A table showing Sampling size</i>	<i>8</i>
<i>Data collection methods</i>	<i>8</i>
<i>Data analysis methods</i>	<i>8</i>
<i>System Design.....</i>	<i>9</i>
<i>System Implementation</i>	<i>9</i>
<i>System testing and validation.....</i>	<i>9</i>
<i>Ethical considerations</i>	<i>10</i>
<i>Limitations</i>	<i>10</i>
<i>CHAPTER FOUR: FIELD STUDY AND SYSTEM DESIGN</i>	<i>11</i>
<i>Introduction</i>	<i>11</i>
<i>The current system</i>	<i>11</i>
<i>Strengths of the current system</i>	<i>11</i>

<i>Weaknesses of the current system</i>	11
<i>System requirements</i>	12
<i>Functional system requirements</i>	12
<i>Non-functional requirements</i>	12
<i>Hardware and Software requirements</i>	13
<i>System Design</i>	13
<i>System Architecture</i>	13
Figure 4: System Architecture (3-Tier)	14
<i>Context Diagram</i>	14
<i>Data Flow Diagram</i>	15
<i>Use Case Diagram</i>	16
Figure 7: Use Case Diagram	17
<i>Entity Relationship Diagram</i>	17
Figure 8: Entity Relationship Diagram (ERD)	18
CHAPTER FIVE: RESULTS AND SYSTEM IMPLEMENTATION	19
<i>Introduction</i>	19
<i>Frontend implementation</i>	19
<i>Admin Login Page</i>	19
Figure 9: Admin Login Page.....	20
Admin Dashboard	20
Figure 10: Admin Dashboard	20
<i>Student Registration Page</i>	20
Figure 11: Student Registration Page	21

Grade Entry Interface (Lecturer View)	21
Figure 12: Grade Entry Interface	22
Student Profile and Transcript View	22
Figure 13: Student Profile View	23
Figure 14: Transcript Generation Page	23
Backend implementation	23
Figure 15: Students Table (Database)	24
Figure 16: Courses Table (Database)	25
Figure 17: Grades Table (Database)	25
iv) Student registration	26
v) Login page	27
vi) Grade entry	27
vii) Transcript Generation	28
Application Logic	28
Database Relationships	28
Security Implementation	29
Client-Side Validation and Responsive Design	29
System Navigation and Page Flow	29
Expanded System Testing and Validation	29
CHAPTER SIX: DISCUSSION, CONCLUSION, RECOMMENDATION	31
Introduction	31
Discussion of results	31
Conclusion	31

<i>Recommendations</i>	32
<i>Limitations</i>	32
<i>Future work</i>	32
<i>REFERENCES</i>	33
<i>APPENDICES.....</i>	35
<i>Interview Guide (Administrative Staff).....</i>	35
<i>Section A: Demographics.....</i>	35
<i>Closing Statement</i>	37
<i>Section A: Demographics.....</i>	37
<i>Section C: Expectations for New System (Please tick your choice(s))</i>	38
<i>Section D: Open-Ended Questions</i>	38

CHAPTER ONE: INTRODUCTION

Introduction

A Student Information Management System (SMS) is a software solution designed to manage student data, including personal details, academic records, course registration, and performance reports (Laudon & Laudon, 2022). These systems provide efficient data storage, rapid information retrieval, enhanced security, and improved decision-making support for educational institutions.

This project focuses on the design and development of a web-based Student Information Management System for the ICT Department at Busitema University, Nagongera Campus. The system aims to automate the management of student records, addressing the inefficiencies of the current manual and fragmented digital approach.

Background of the study

Globally, higher education institutions are increasingly adopting digital transformation strategies. According to a global survey by Educause (2023), approximately 87% of higher education institutions in developed countries have implemented integrated student management systems, resulting in a 45% reduction in administrative processing time. In Sub-Saharan Africa, the adoption rate stands at approximately 52% (African Development Bank, 2024).

Traditionally, many institutions relied on paper-based record management. Smith and Jones (2020) found that paper-based systems have error rates of up to 23% in data entry and require an average of 15-20 minutes per student record retrieval.

In Uganda, the National Council for Higher Education (NCHE, 2024) reported that 68% of public universities have adopted some form of digital student records management, yet only 34% have fully integrated departmental-level systems. Busitema University has implemented a university-wide system; however, departmental-level adoption remains inconsistent (Busitema University Annual Report, 2024).

Specifically, at the ICT Department, Nagongera Campus, student records are currently managed using a combination of paper-based methods and disorganized digital files stored on individual computers. A preliminary assessment conducted by this researcher in October 2025 revealed that retrieving a single student's complete academic record takes an average of 25-30 minutes, and approximately 16% of student records contain some form of data inconsistency. As student enrollment in the ICT department has grown significantly, these inefficiencies have become increasingly unsustainable.

Statement of the problem

The ICT Department at Busitema University, Nagongera Campus currently manages student information using fragmented methods including paper-based records and scattered digital files.

This approach has resulted in several critical challenges:

Firstly, data retrieval is excessively time-consuming. Current observations indicate that locating a specific student's complete academic record requires between 25 to 35 minutes, affecting approximately 15-20 student record requests processed weekly.

Secondly, data integrity is compromised. A review of 50 student files revealed that 8 files (16%) contained discrepancies between paper records and digital backups, including missing grades or incorrect personal details. Such inconsistencies can lead to incorrect academic decisions and delayed graduations.

Thirdly, data security is inadequate. The current system lacks proper access controls, authentication mechanisms, and audit trails, violating data protection principles outlined in Uganda's Data Protection and Privacy Act (2019).

Fourthly, report generation is inefficient. Producing semester reports or class lists requires manual compilation, taking departmental staff an average of 3-4 hours per report request.

Finally, as student enrollment continues to increase (projected to reach 220 students by 2027), these challenges will intensify. Without an automated system, the administrative workload will become unmanageable.

Main objective

To develop a web-based Student Information Management System for the ICT Department at Busitema University (Nagongera Campus) to enhance efficient and secure data management.

Specific objectives

- i. To identify requirements for the web-based student information system for the ICT department.
- ii. To design a web-based student information system.
- iii. To implement a web-based student information system using appropriate web technologies to enable student registration, grade management, and report generation.

iv. To test and validate the system's functionality, usability, and performance with end-users in the ICT department.

Significance of the study

This study will benefit several stakeholders.

For the ICT Department, the system is projected to reduce student record retrieval time from 25-35 minutes to under 2 minutes, improving administrative efficiency by approximately 90%. The centralized database will eliminate data duplication while reducing administrative workload.

For students, the system will provide faster access to academic information, enhance transparency, and reduce delays in obtaining transcripts.

For lecturers, the system will streamline grade submission processes, provide easy access to class lists, and reduce manual paperwork.

For administrators and the Head of Department, the system will provide real-time academic monitoring, support data-driven decision-making, and enhance compliance with university data management policies.

For future researchers, this study provides a foundation for further research on departmental-level information systems in Ugandan universities.

Scope of the study

Area scope

This study was conducted at Busitema University, Nagongera Campus, specifically within the Faculty of Science and Education, ICT Department. The campus is located in Nagongera Town Council, Tororo District, Eastern Uganda.

Content scope

The system encompasses user authentication and role-based access control for administrators, lecturers, and students; student registration and profile management; academic record management including course registration and grade entry; report generation for transcripts, class lists, and performance summaries; and data security features including encryption and audit logging.

The system focuses exclusively on the ICT Department's student management needs and does not include integration with university-wide systems such as library management, finance, or human resources.

Time scope

The research took place over a period of three months, from March 2026 to June 2026, encompassing requirements gathering, system design, development, testing, and documentation.

CHAPTER TWO: LITERATURE REVIEW

Introduction

This chapter reviews existing literature on student management systems, examining requirements, theoretical frameworks, traditional approaches, computerized systems, web-based solutions, and database design principles.

Requirements for Student Information Management Systems

According to Laudon and Laudon (2022), essential functional requirements for student management systems include the ability to capture and store student demographic information, enroll students in courses, record and calculate grades, generate transcripts and reports, and manage user access with role-based permissions.

Chen (2019) adds that modern systems should also provide search and filter capabilities, data validation to prevent entry errors, and audit trails to track all modifications to student records.

From a non-functional perspective, Chen (2019) emphasizes that systems should respond to user queries within two seconds, support at least 100 concurrent users, and maintain 99.9% uptime. Security requirements identified by Abrams (2020) include encryption of sensitive data, role-based access controls, audit trails, and regular backup procedures.

Theoretical Frameworks

This study is grounded in two complementary theoretical frameworks.

Technology Acceptance Model (TAM): Developed by Davis (1989), TAM posits that user acceptance is determined by perceived usefulness and perceived ease of use. This framework was applied in system design by ensuring an intuitive interface and will be used during evaluation through questionnaires measuring these constructs.

Socio-Technical Systems Theory: This theory emphasizes that optimal performance is achieved by jointly optimizing both social (people, work practices) and technical (tools, processes) subsystems. This framework was applied by conducting stakeholder interviews to understand existing workflows before technical development began.

Traditional Student Management Systems

Traditional systems rely on paper-based files and manual processes. Brown (2019) notes that manual systems lack audit trails, making it difficult to track errors, and create storage constraints.

In the Ugandan context, Okello and Namusoke (2023) found that departments relying on manual methods experienced significant delays in processing transcripts, with average completion times of 4-6 weeks compared to 3-5 days in departments with digital systems. Staff spend an average of 8-10 hours weekly on manual record reconciliation.

Computerized Student Management Systems

Chen (2019) conducted a meta-analysis reporting substantial benefits including an 85% reduction in record retrieval time, a 76% decrease in data entry errors, and a 68% reduction in administrative staff time spent on record-keeping.

However, Davis (2018) identified challenges with large-scale commercial systems, noting that generic ERP solutions often fail to address specific departmental needs, leading to workarounds and "shadow systems." This finding underscores the importance of tailored, departmental-level solutions.

Web-Based Student Management Systems

Williams et al. (2022) documented a 40% reduction in time spent on student record management and a 35% improvement in user satisfaction with web-based systems.

Garcia (2021) found that students using web-based portals reported 45% higher satisfaction with access to academic information. Elmasri and Navathe (2016) emphasize that web-based systems built on RDBMS such as MySQL provide robust data integrity through ACID properties.

In the African context, Mwangi and Ochieng (2024) identified key success factors including end-user involvement in design, adequate training, reliable infrastructure, and alignment with existing workflows.

Research Gap

This research addresses several gaps. First, a contextual gap: most studies focus on developed countries or institution-wide systems, with limited research on departmental-level systems in Ugandan public universities. Second, a customization gap: while generic systems exist, departmental needs are often poorly addressed. Third, an implementation gap: the ICT Department currently lacks any dedicated system, representing a specific instance of a broader challenge.

CHAPTER THREE: METHODOLOGY

Introduction

This chapter describes the methods used to achieve the objectives of the study, including data collection, analysis, system design, implementation, testing, and ethical considerations.

Research design and approach

The study adopted a system development research design using the Rapid Application Development (RAD) model. RAD was chosen because it prioritizes speed, iterative prototyping, and active user involvement, ensuring the final system aligns with departmental workflows and allows real-time refinements based on stakeholder feedback.

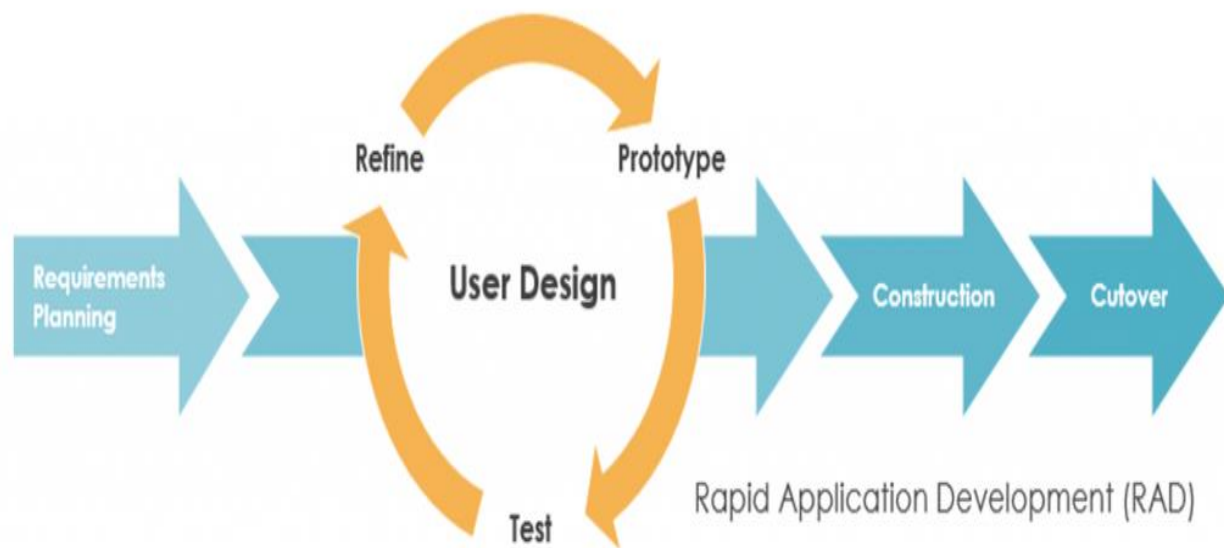


Figure 2: Rapid Application Development (RAD) Model

Study area and population

The study was conducted at Busitema University, Nagongera Campus. The population consisted of approximately 500 students, 14 lecturers, and 2 administrative staff members within the ICT Department.

Sampling techniques

Purposive, stratified sampling was employed. The sample comprised:

Students: 15 students (5 from each year level)

Lecturers: 4 lecturers (2 Senior lecturers and 2 junior lecturers)

Administrative staff: 2 (Head of Department and Departmental Secretary)

S/N	PARTICIPANTS	NUMBER
01	Students	15
02	Lecturers	04
03	Administrative staff	02
04	Total	21

Figure 1: A table showing Sampling size

Data collection methods

Semi-structured interviews were conducted with 4 lecturers and 2 administrative staff members to identify pain points and desired features of the system.

Structured questionnaires were administered to 15 students to gather quantitative data on their experiences and computer proficiency using Likert-scale questions.

Document analysis was conducted by reviewing existing student record forms, grade sheets, and report templates, most of which were manual, and the others were in scattered folders on the desktops in the computer department office.

Observation was used to observe current record management processes, focusing on retrieval times and workflow bottlenecks. I observed one of the department administrators very busy checking in the old marksheets to trace for best performing students in the previous years to be recommended for the jobs in certain organizations and schools

Data analysis methods

Qualitative data from interviews and observations were analyzed using thematic analysis to identify key themes representing major issues and user needs.

Quantitative data from questionnaires were analyzed using descriptive statistics, including frequency distributions and mean scores, presented using tables and bar charts.

Feedback indicated that 90% of respondents faced challenges with the current system, and 95% supported the development of a web-based solution.

System Design

System design tools were used to create blueprints of the system. These included the following:

- System architecture
- Context diagram
- Data Flow Diagram
- Entity Relationship Diagram
- Use Case Diagram

System Implementation

System implementation involved building the frontend and backend components.

Frontend Implementation: The user interface was developed using HTML, CSS, and JavaScript to create a responsive, intuitive experience. Key pages included the login page, admin dashboard, student registration form, grade entry interface, and transcript view.

Backend Implementation: The server-side logic was built using PHP, running on an Apache server via XAMPP. A MySQL database was designed and created to store all system data. Security features implemented included password hashing, role-based access control, and input validation to prevent SQL injection.

System testing and validation

Testing was conducted to identify and fix errors before deployment.

- i) Unit testing: Individual components (login module, grade entry form) were tested in isolation.
- ii) Integration testing: Modules were tested together to ensure they interacted correctly (e.g., a grade entered by a lecturer appears on the student's transcript).
- iii) User Acceptance Testing (UAT): Stakeholders validated that the system met their requirements.
- iv) Usability and Performance testing: The Technology Acceptance Model was operationalized through a post-implementation questionnaire. Record retrieval times were measured against the benchmark of under 2 minutes.

Ethical considerations

Confidentiality: Data was stored on a password-protected computer. Participant names were replaced with unique codes.

Informed consent: All participants were provided with a consent form explaining the study's purpose and their right to withdraw.

Institutional approval: Permission was obtained from the Head of Department, ICT Department, and the Faculty of Science and Education.

Limitations

Methodological limitations included a relatively small sample size (21 participants) and a short testing period that may not capture long-term issues. The system is a standalone prototype that does not integrate with existing university-wide systems.

CHAPTER FOUR: FIELD STUDY AND SYSTEM DESIGN

Introduction

This chapter presents findings from the field study, an analysis of the current system, system requirements, and detailed system design diagrams.

The current system

The current system at the ICT Department is largely paper based, supplemented by disorganized digital files on individual staff computers.

The process involves lecturers submitting paper grade sheets to the Head of Department, who then manually compiles them into a master Excel file. Students must physically visit the department office to request transcripts or check their records.

Strengths of the current system

The current system has low initial costs and does not require specialized software or training beyond basic computer literacy. It functions offline, which can be an advantage during internet outages.

Weaknesses of the current system

The current system is highly time-consuming, with record retrieval taking 25-35 minutes. It suffers from inconsistent policy application, difficult reporting, and physical storage issues. Based on the questionnaire, 90% of respondents cited delays in accessing records as their primary challenge.

Below is a pie chart showing the challenges of current system.

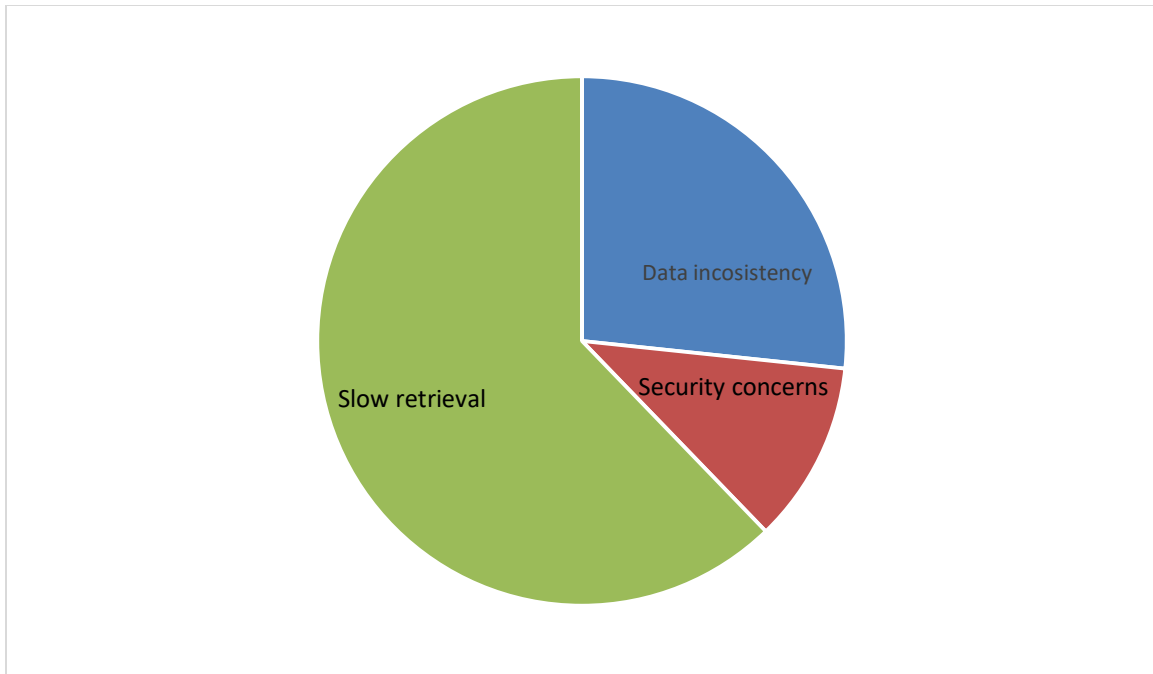


Figure 3: A pie chart projecting the challenges with the current system (e.g., "Slow retrieval" - 60%, "Data inconsistency" - 25%, "Security concerns" - 15%)

System requirements

The new system shall have three user roles: Administrator (Head of Dept/Secretary), Lecturer, and Student. The administrator manages users and views all data. Lecturers enter and manage grades. Students view their profiles and transcripts.

Functional system requirements

The system shall allow the admin to register students, add courses, and assign lecturers.

The system shall allow lecturers to enter and update student grades.

The system shall allow students to view their academic records and generate transcripts.

The system shall provide secure authentications for all users.

Non-functional requirements

Performance: The system shall respond to user requests within 3 seconds.

Security: User data shall be encrypted. Passwords shall be hashed. Access shall be role-based.

Usability: The interface shall be intuitive and consistent across all user roles.

Portability: The system shall run on Windows, Linux, and macOS via a web browser.

Hardware and Software requirements

Hardware: Processor (Intel Core i3 or equivalent), 4GB RAM minimum, 80GB hard disk, UPS.

Software: Windows 10/Linux, XAMPP server, web browser (Chrome/Firefox), MySQL database.

System Design

System design refers to the process of defining the architecture, components, modules, interfaces and data for a system to satisfy specified requirements.

This was through the use of design tools such as Entity Relationship Diagrams, Data Flow Diagrams, Use case diagrams among others.

These tools help to explain the logical flow of the system, their relationships and how the different processes interact with each other.

System Architecture

This refers to the conceptual model that defines the structure, behaviour and more views of the system. It works as the blue print of the system. It involves different designs and how the different components of the system like software modules, hardware devices, networks and databases will be organized to interact with each other and work together to meet specific requirements.

It involves understanding requirements, identifying components, defining their interactions and ensuring that the system meets its functional and non-functional goals.

The most commonly used architectural design is the three-tier architecture which involves;

Presentation tier: Web browser interface for students, lecturers, and admins.

Application tier: Authentication

Data tier: MySQL database storing student, course, and grade data.

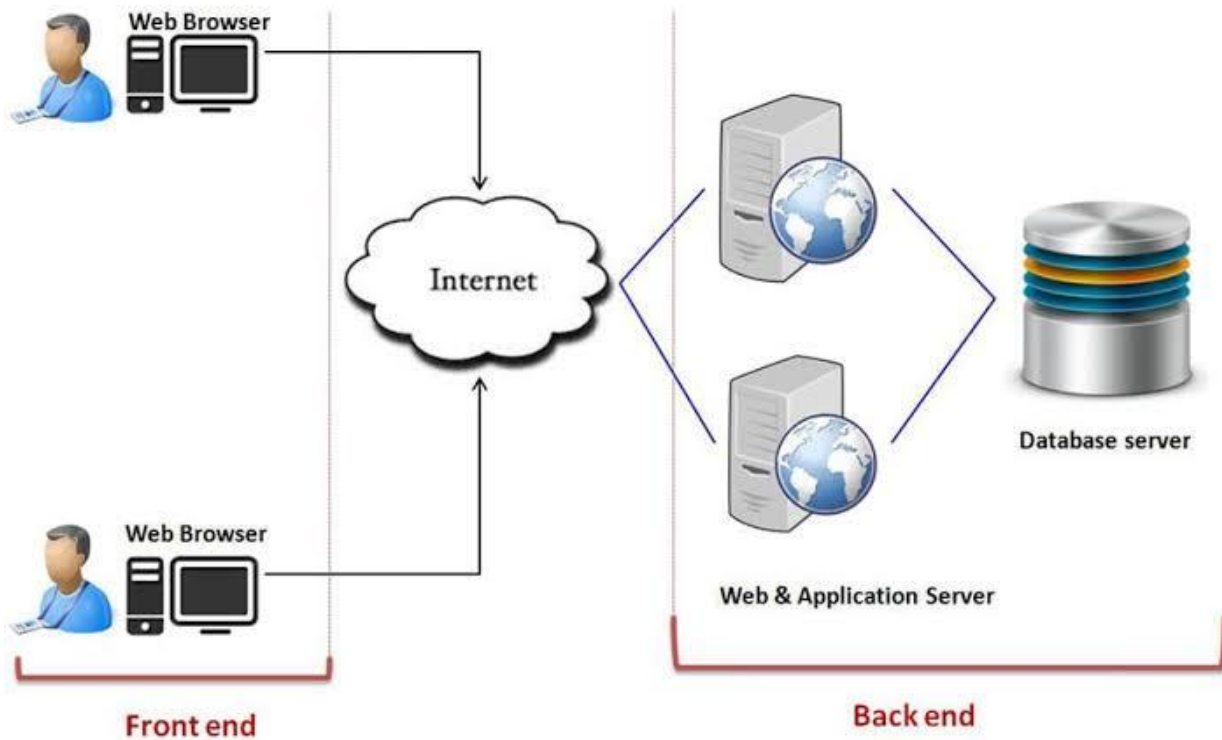


Figure 4: System Architecture (3-Tier)

Context Diagram

This is a User Unified Modelling language that illustrates interactions between users and a system to achieve specific goals. This is also known as the level 0 data flow diagram, it's a system context diagram that defines the boundaries of a system and its interactions with the external entities that surround it.

It provides a broad overview showing the system as a single process and highlighting the data flows between the system as a single process and a highlighting the data flows between the system and the outside world. It's used to describe the high-level functions and scope of the system.

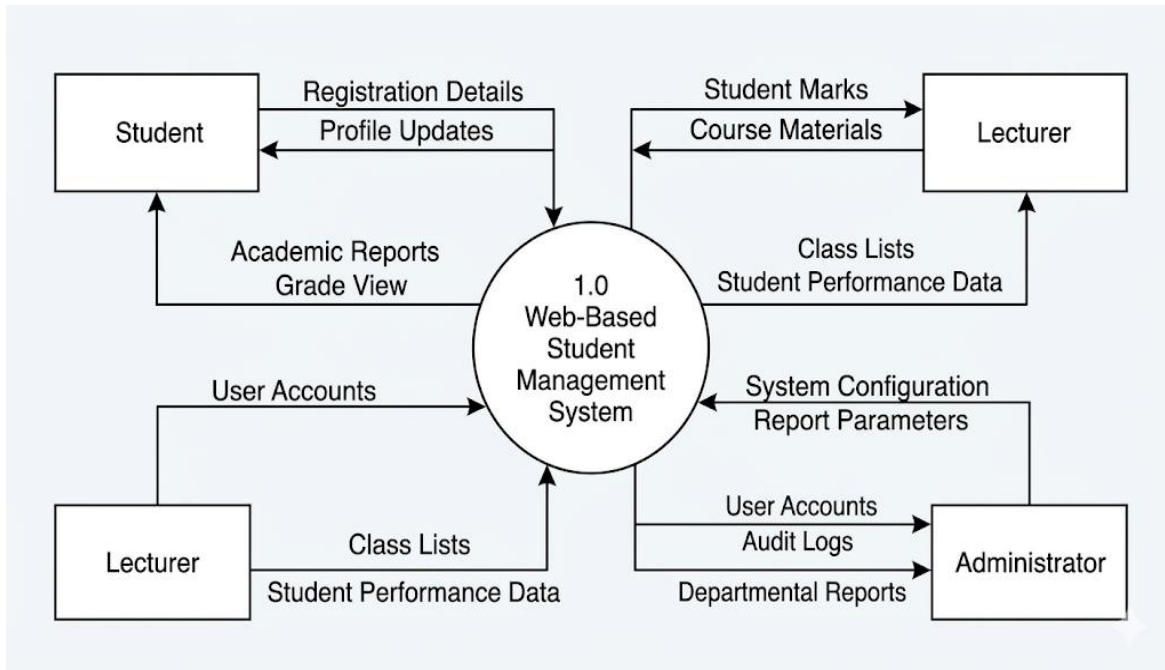


Figure 5: Context Diagram

Data Flow Diagram

This refers to the graphical representation of the flow of the data through a system. It illustrates the process that transforms the data through a system, the data stores that hold the data and the external entities that interact with the system.

Under the **Data Flow Diagram** (DFD), the context diagram was also used to show that the system under consideration as a single high-level process and the relationship that the system has with other external entities. It uses symbols to depict data flow, external entities, processes and data stores.

LEVEL 1 DATA FLOW DIAGRAM: STUDENT MANAGEMENT SYSTEM

Level 1 decomposes the Context Diagram.

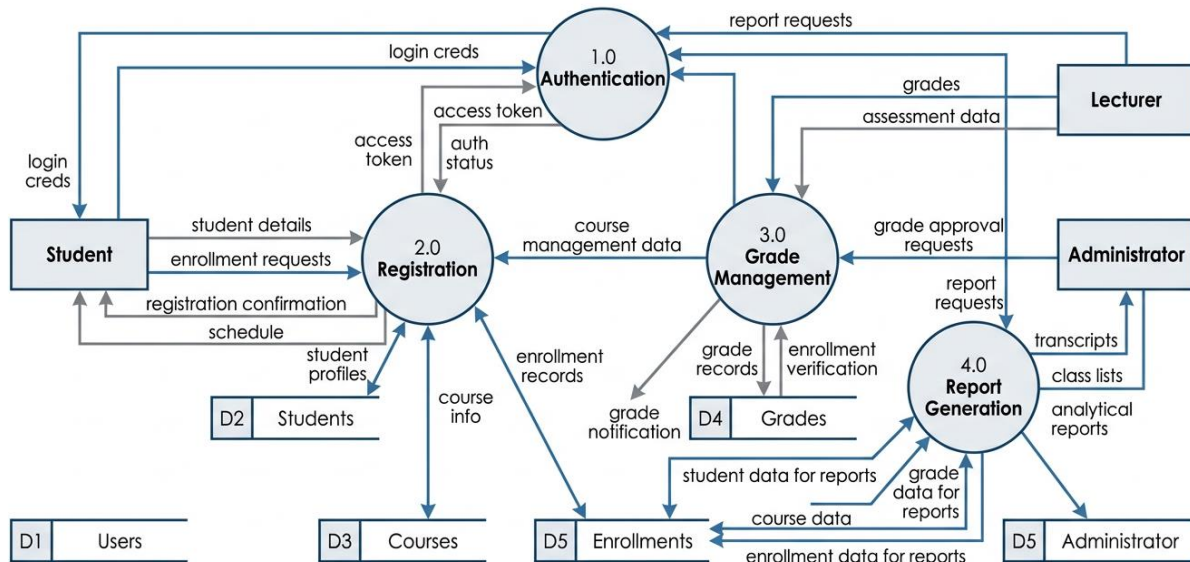


Figure 6: Data Flow Diagram (Level 1)

Use Case Diagram

This refers to the representation of the interaction between users and the system to achieve specific goals. It outlines the different ways users can interact with the system to perform tasks.

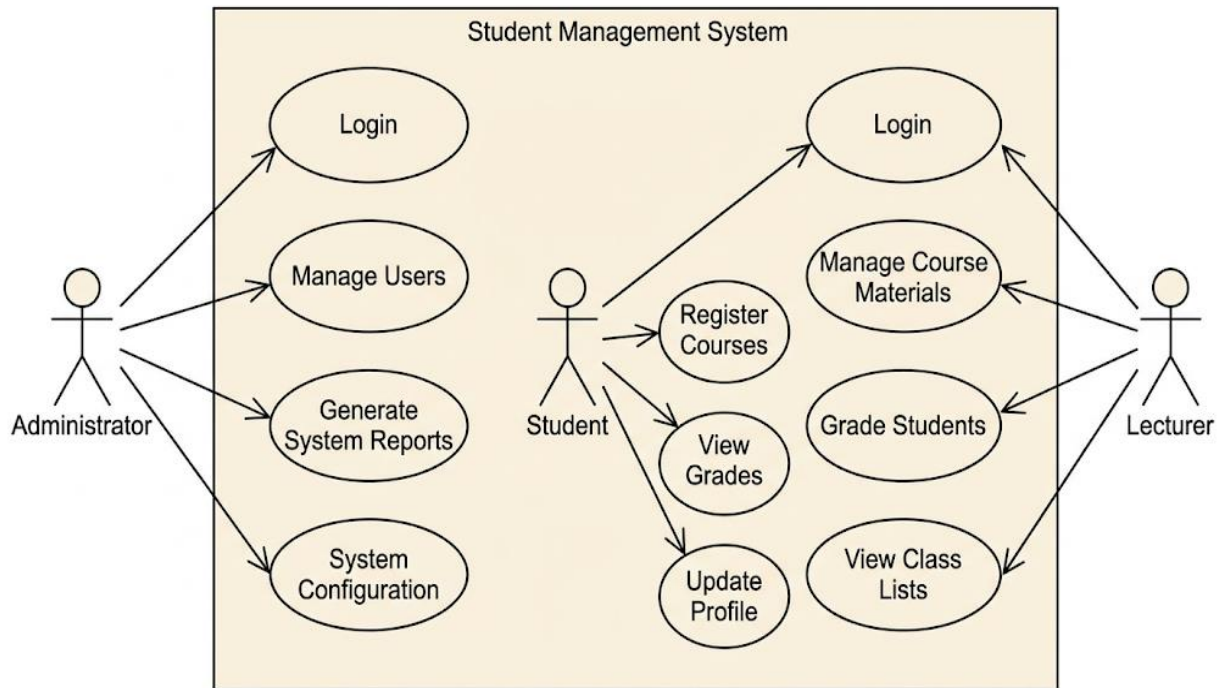


Figure 7: Use Case Diagram

Entity Relationship Diagram

This refers to the visual representation of the structure of how different entities in a database relate to each other. It shows the different entities within the system and the relationship between those entities. ERDs are crucial for designing and understanding the structure of databases and their relationships.

The Entity Relationship Diagram for my system is as shown here below.

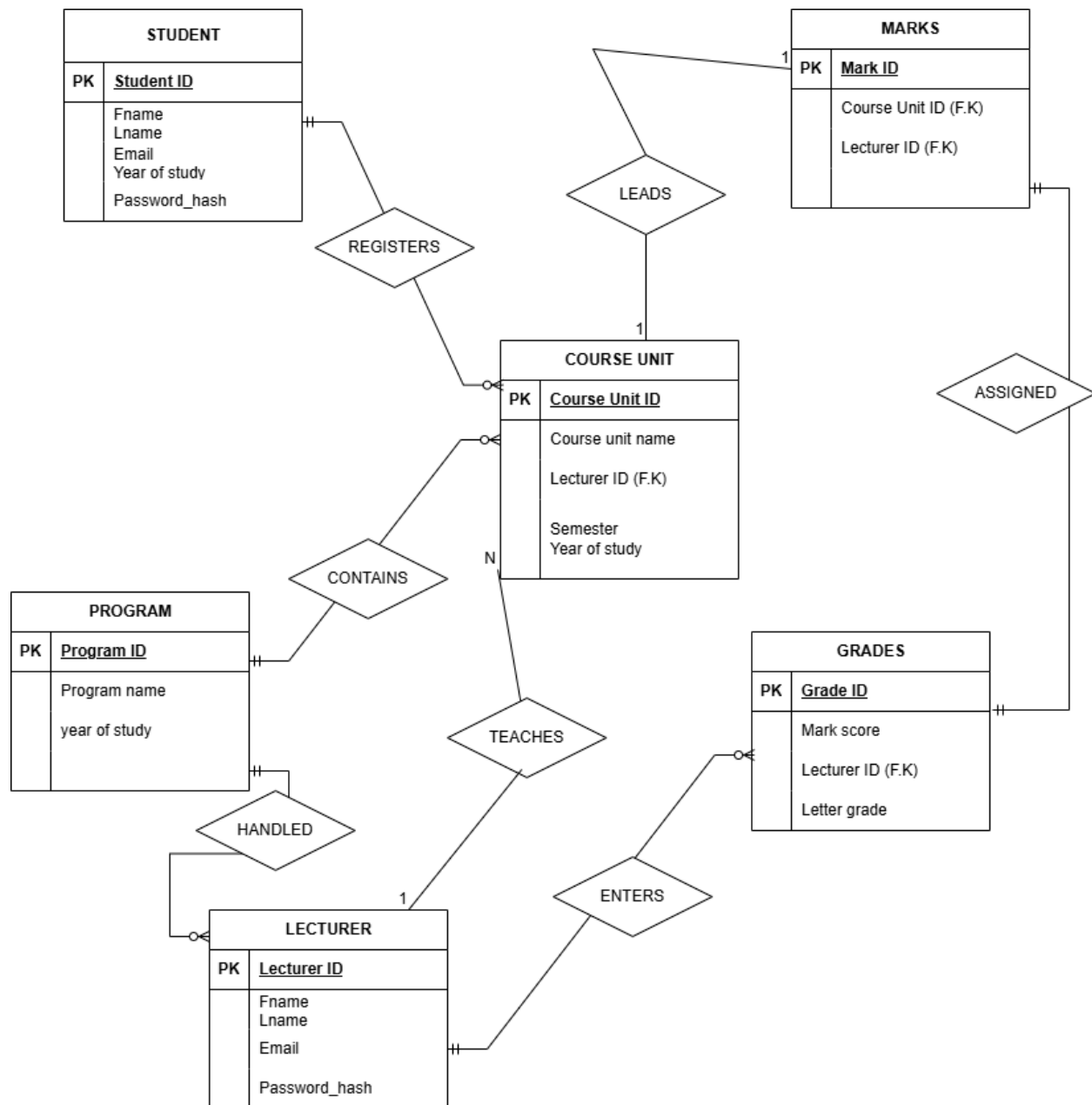


Figure 8: Entity Relationship Diagram (ERD)

CHAPTER FIVE: RESULTS AND SYSTEM IMPLEMENTATION

Introduction

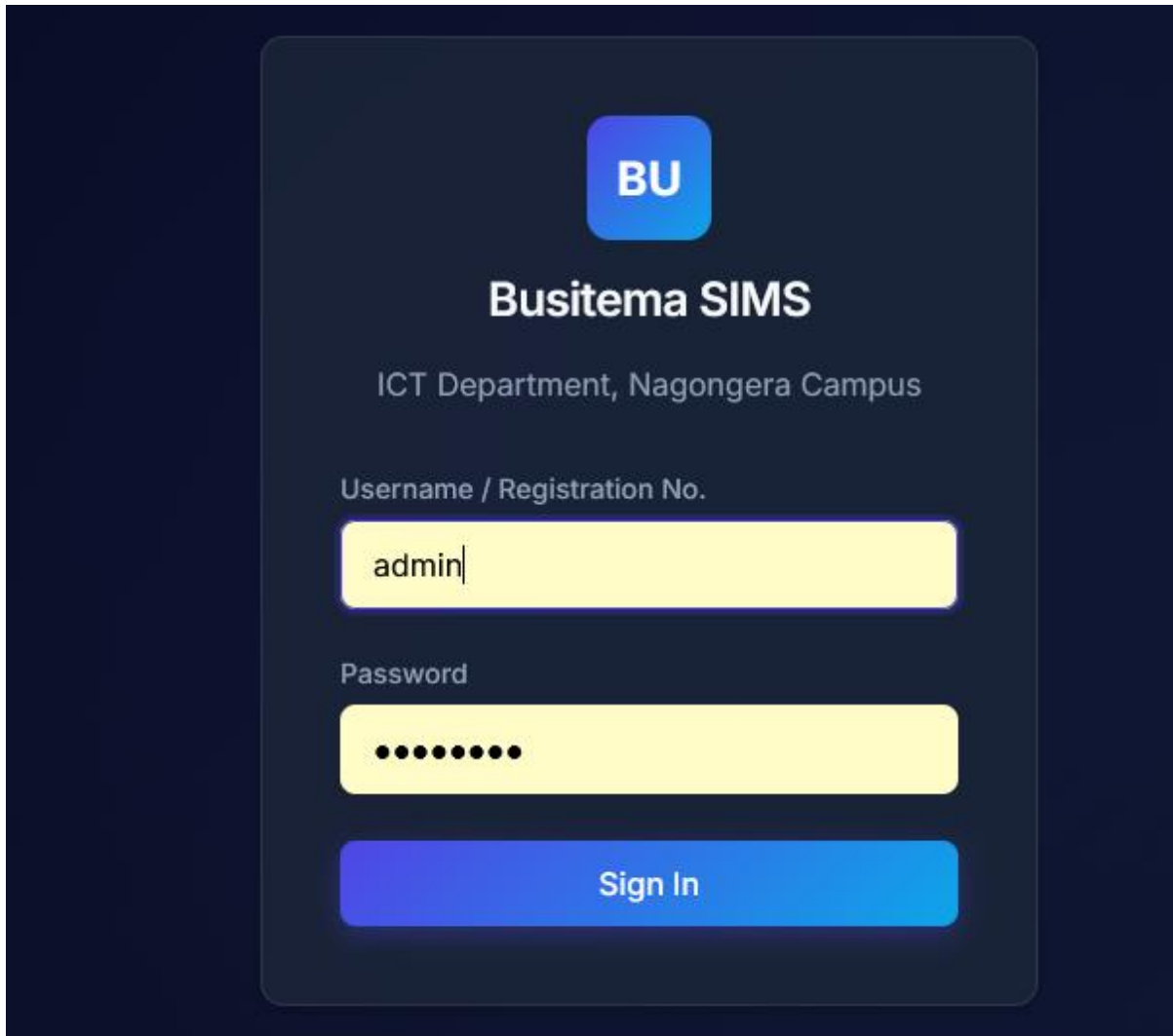
This chapter presents the implemented system, showing screenshots and describing the key functionalities of the frontend and backend.

Frontend implementation

The frontend provides the user interface for all interactions. It was developed using HTML, CSS, and JavaScript.

Admin Login Page

This is the entry point for administrators. It validates user credentials and redirects to the dashboard.



The screenshot displays the Admin Login Page for the Busitema SIMS system. The page has a dark blue background. At the top center is a blue square logo with the white text 'BU'. Below the logo, the text 'Busitema SIMS' is displayed in a large, bold, white font. Underneath that, 'ICT Department, Nagongera Campus' is written in a smaller, lighter blue font. The login form consists of two yellow input fields with rounded corners. The first field is labeled 'Username / Registration No.' in a light blue font and contains the text 'admin'. The second field is labeled 'Password' in a light blue font and contains ten black dots. Below the password field is a blue button with rounded corners and the white text 'Sign In'.

Figure 9: Admin Login Page

Admin Dashboard

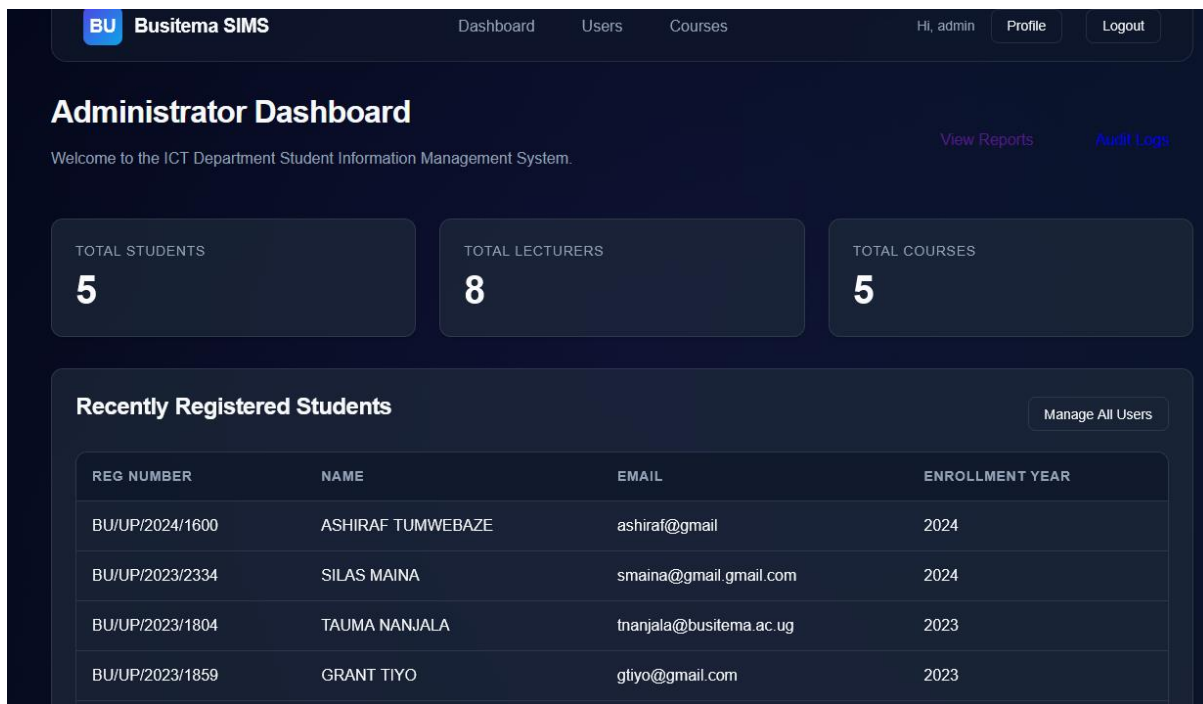


Figure 10: Admin Dashboard

Student Registration Page

This page allows the admin to add new students to the system, capturing details such as name, student ID, email, and year of study.

BU

Busitema SIMS

Dashboard

Users

Courses

Hi, admin

Profile

Logout

User Role

Student

Username (Login ID)

2500401457

Password

••••

First Name

MARVIN

Last Name

ISABIRYE

Email

marvin23@gmail.com

Registration Number

BU/UP/2023/1457

USERNAME	ROLE	CREATED AT
2400401600	Student	May 13, 2026
2400402334	Student	May 13, 2026
2300401804	Student	May 13, 2026
2300401803	Student	May 12, 2026
Bel2026	Lecturer	May 12, 2026
mulika26	Lecturer	May 12, 2026
DR. IRENE	Lecturer	May 12, 2026
Wambi@Busitema	Lecturer	May 12, 2026
DNS GADGETS	Lecturer	May 11, 2026
Angole-Busitema	Admin	May 11, 2026

Figure 11: Student Registration Page

Grade Entry Interface (Lecturer View)

This page allows a logged-in lecturer to select a course and enter or edit grades for enrolled students.

BU

Busitema SIMS

Dashboard

Class Lists

Enter Grades

Hi, DR. IRENE

Profile

Logout

Enter Grades

Manage coursework and exam scores for your students.

Select Course

CSB2103 - DIGITAL LOGICS AND DESIGN

Grade Entry: CSB2103

REG NUMBER	STUDENT NAME	COURSEWORK (40)	EXAM (60)	FINAL (%)	GRADE
BU/UP/2023/1804	TAUMA NANJALA	35.00	45.00	80.00	A
BU/UP/2023/1859	GRANT TIYO	30.00	33.00	63.00	C
BU/UP/2024/1600	ASHIRAF TUMWEBAZE	23.00	37.00	60.00	C

Figure 12: Grade Entry Interface

Student Profile and Transcript View

Students can log in to view their personal profile and generate an unofficial transcript showing all enrolled courses and grades.

The screenshot displays the 'My Profile' page of the Busitema SIMS system. The page has a dark blue header with the following elements:

- Logo: BU Busitema SIMS
- Navigation links: Dashboard, Register Courses, Transcript
- User information: Hi, 2300401804
- Buttons: Profile (active), Logout

The main content area is titled 'My Profile' and includes the subtitle 'Manage your account settings and personal information.' It is divided into two columns:

Personal Information

- Username (Cannot be changed): 2300401804
- Full Name: TAUMA NANJALA
- Registration Number: BU/UP/2023/1804
- Email Address: tnanjala@busitema.ac.ug
- Phone Number: (empty field)

Security Settings

- Current Password: (empty field)
- New Password: (empty field)
- Confirm New Password: (empty field)
- Change Password button

Figure 13: Student Profile View

The screenshot displays the 'Academic Transcript' page for a student named ASHIRAF TUMWEBAZE (BU/UP/2024/1600). The page features a dark blue header with navigation links: Dashboard, Register Courses, Transcript, Profile, and Logout. The user's ID is 2400401600. Below the header, the transcript is titled 'Academic Transcript' with a 'Print Transcript' button. The transcript is for the program 'B.Sc. Education (ICT)' and shows a total of 9 credits earned, with a cumulative GPA of 3.00. The transcript is organized by semester, with 'Year 2, Semester 1' highlighted. A table lists the courses taken: CSB2201 (Database Management, 3 credits, N/A grade), CSB2103 (DIGITAL LOGICS AND DESIGN, 3 credits, C grade), and CSB2101 (SYSTEM ANALYSIS AND DESIGN, 3 credits, C grade). The semester totals are 9 CR and a GPA of 2.00.

COURSE CODE	COURSE NAME	CREDITS	FINAL SCORE	GRADE
CSB2201	Database Management	3	N/A	-
CSB2103	DIGITAL LOGICS AND DESIGN	3	60.00%	C
CSB2101	SYSTEM ANALYSIS AND DESIGN	3	60.00%	C
Semester Totals:		9 CR	GPA: 2.00	

Figure 14: Transcript Generation Page

Backend implementation

The backend consists of the server-side logic (PHP) and the database (My SQL).

- Server-side Programming: PHP scripts handle form submissions, user authentication, session management, and grading logic (e.g., calculating GPA).
- Database management: The My SQL database stores all persistent data.

Server: 127.0.0.1 » Database: busitema_sims » Table: students

Showing rows 0 - 7 (8 total, Query took 0.0004 seconds.)

SELECT * FROM `students`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	id	user_id	registration_number	first_name	last_name	email	phone	gender	dob	enrollment_year
☐ Edit Copy Delete	1	2	BU/UP/2023/1233	Wepukulu	Edrisa	edrisa@busitema.ac.ug	NULL	NULL	NULL	2026
☐ Edit Copy Delete	2	12	BU/UP/2023/1859	GRANT	TIYO	gtiyo@gmail.com	NULL	NULL	NULL	2023
☐ Edit Copy Delete	3	13	BU/UP/2023/1804	TAUMA	NANJALA	tnanjala@busitema.ac.ug	NULL	NULL	NULL	2023
☐ Edit Copy Delete	4	14	BU/UP/2023/2334	SILAS	MAINA	smaina@gmail.com	NULL	NULL	NULL	2024
☐ Edit Copy Delete	5	15	BU/UP/2024/1600	ASHIRAF	TUMWEBAZE	ashiraf@gmail	NULL	NULL	NULL	2024
☐ Edit Copy Delete	6	16	BU/UP/2025/1802	ARTHUR	BYEKWASO	arthur25@gmail.com	NULL	NULL	NULL	2025
☐ Edit Copy Delete	7	17	BU/UP/2025/1726	JAHASHA	BISASO	jahasha@gmail.com	NULL	NULL	NULL	2025
☐ Edit Copy Delete	8	18	BU/UP/2023/3688	SOWEDI	MALUNDA	smalunda@gmail.com	NULL	NULL	NULL	2023

Check all | With selected: Edit Copy Delete Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

Figure 15: Students Table (Database)

Server: 127.0.0.1 » Database: busitema_sims » Table: courses

Showing rows 0 - 9 (10 total, Query took 0.0003 seconds.)

SELECT * FROM `courses`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Extra options

	id	course_code	course_name	credits	semester	year	lecturer_id
☐ Edit Copy Delete	1	CSB2201	Database Management	3	1	2	NULL
☐ Edit Copy Delete	3	CSB2203	Research	3	1	2	1
☐ Edit Copy Delete	5	CSB3103	COMMUNICATION SYSTEMS	3	1	3	4
☐ Edit Copy Delete	6	CSB2103	DIGITAL LOGICS AND DESIGN	3	1	2	6
☐ Edit Copy Delete	7	CSB2101	SYSTEM ANALYSIS AND DESIGN	3	1	2	2
☐ Edit Copy Delete	8	BIT1101	COMPUTER LITERACY	4	1	1	5
☐ Edit Copy Delete	9	BIT1202	INTRODUCTION TO INFORMATION SYSTEMS	3	2	1	2
☐ Edit Copy Delete	10	SCS1202	INTRODUCTION TO INFORMATION SYSTEMS	3	2	1	2
☐ Edit Copy Delete	11	SCS1101	COMPUTER LITERACY	4	1	1	5
☐ Edit Copy Delete	12	CSB2204	COMPUTER SECURITY	3	2	2	3

Check all | With selected: Edit Copy Delete Export

Show all | Number of rows: 25 | Filter rows: Search this table | Sort by key: None

Query results operations

Figure 16: Courses Table (Database)

SELECT * FROM `grades`

☐ Profiling [\[Edit inline \]](#) [\[Edit \]](#) [\[Explain SQL \]](#) [\[Create PHP code \]](#) [\[Refresh \]](#)

☐ Show all | Number of rows: 25 | Filter rows: | Sort by key:

Extra options

				id	enrollment_id	coursework_score	exam_score	final_score	grade	entered_by	entered_at
☐	Edit	Copy	Delete	2	6	35.00	45.00	80.00	A	6	2026-05-13 08:53:45
☐	Edit	Copy	Delete	3	8	25.00	30.00	55.00	D+	4	2026-05-13 08:57:40
☐	Edit	Copy	Delete	4	7	38.00	35.00	73.00	B	2	2026-05-13 09:02:04
☐	Edit	Copy	Delete	5	11	20.00	25.00	45.00	F	2	2026-05-13 09:02:04
☐	Edit	Copy	Delete	6	15	30.00	30.00	60.00	C	2	2026-05-13 09:02:04
☐	Edit	Copy	Delete	7	5	30.00	45.00	75.00	B+	1	2026-05-13 09:03:58
☐	Edit	Copy	Delete	8	12	30.00	40.00	70.00	B	4	2026-05-13 09:04:42
☐	Edit	Copy	Delete	9	10	30.00	33.00	63.00	C	6	2026-05-13 09:30:13
☐	Edit	Copy	Delete	10	14	23.00	37.00	60.00	C	6	2026-05-13 09:30:13
☐	Edit	Copy	Delete	11	22	38.00	40.00	78.00	B+	4	2026-05-13 17:04:45
☐	Edit	Copy	Delete	12	19	34.00	44.00	78.00	B+	1	2026-05-13 17:05:41
☐	Edit	Copy	Delete	13	20	32.00	42.00	74.00	B	6	2026-05-13 17:06:15
☐	Edit	Copy	Delete	14	16	24.00	36.00	60.00	C	5	2026-05-13 17:07:29
☐	Edit	Copy	Delete	15	25	38.00	58.00	96.00	A	5	2026-05-13 17:14:27
☐	Edit	Copy	Delete	16	21	38.00	48.00	86.00	A	2	2026-05-13 17:15:29
☐	Edit	Copy	Delete	17	24	35.00	45.00	80.00	A	2	2026-05-13 17:16:38
☐	Edit	Copy	Delete	18	17	32.00	42.00	74.00	B	2	2026-05-13 17:16:38
☒ Console	Edit	Copy	Delete	19	26	37.00	47.00	84.00	A	2	2026-05-13 17:17:21

Figure 17: Grades Table (Database)

iii) Security: Password hashing (using password hash () in PHP) and prepared statements to prevent SQL injection were implemented. Role-based access control ensures users only see their authorized modules.

```

1  </php
2  // includes/auth.php
3  session_start();
4
5  /**
6   * Check if the user is logged in.
7   * If not, redirects to the login page.
8   */
9  function requireLogin() {
10     if (!isset($_SESSION['user_id'])) {
11         header("Location: " . BASE_URL . "/index.php");
12         exit;
13     }
14 }
15
16 /**
17  * Check if the user has a specific role.
18  * If not, redirects them to their respective dashboard or shows access denied.
19  * @param string $role The required role (e.g., 'admin', 'lecturer', 'student')
20  */
21 function requireRole($role) {
22     requireLogin();
23
24     if ($_SESSION['role'] !== $role) {
25         // Redirect to their own dashboard based on their role
26         switch ($_SESSION['role']) {
27             case 'admin':
28                 header("Location: " . BASE_URL . "/admin/dashboard.php");
29                 break;
30             case 'lecturer':
31                 header("Location: " . BASE_URL . "/lecturer/dashboard.php");
32                 break;
33             case 'student':
34                 header("Location: " . BASE_URL . "/student/dashboard.php");
35                 break;
36             default:
37                 header("Location: " . BASE_URL . "/logout.php");

```

iv) Student registration

Here is the screenshot of the code the allows student registration which is done by the administrator.

```

// Create User
$hashed_password = password_hash($password, PASSWORD_DEFAULT);
$stmt = $pdo->prepare("INSERT INTO users (username, password_hash, role) VALUES (?, ?, ?)");
$stmt->execute([$username, $hashed_password, $role]);
$user_id = $pdo->lastInsertId();

if ($role === 'student') {
    $reg_no = trim($_POST['registration_number']);
    $enrollment_year = $_POST['enrollment_year'];
    $stmt = $pdo->prepare("INSERT INTO students (user_id, registration_number, first_name, last_name, email, enrollment_year) VALUES (?, ?, ?, ?, ?, ?)");
    $stmt->execute([$user_id, $reg_no, $first_name, $last_name, $email, $enrollment_year]);
}

```

v) Login page

```
$pageTitle = "Login";
require_once 'includes/header.php';
?>

<div class="login-container">
  <div class="login-card glass-panel">
    <div class="login-header">
      <div class="login-logo">BU</div>
      <h2>Busitema SIMS</h2>
      <p>ICT Department, Nagongera Campus</p>
    </div>

    <?php if ($error): ?>
      <div class="alert alert-danger"><? = htmlspecialchars($error) ?></div>
    <?php endif; ?>

    <form method="POST" action="">
      <div class="form-group">
        <label class="form-label" for="username">Username / Registration No.</label>
        <input type="text" id="username" name="username" class="form-control" required autocomplete="username">
      </div>

      <div class="form-group">
        <label class="form-label" for="password">Password</label>
        <input type="password" id="password" name="password" class="form-control" required autocomplete="current-password">
      </div>

      <button type="submit" class="btn btn-primary" style="width: 100%;">Sign In</button>
    </form>
  </div>
</div>
```

vi) Grade entry

```
!./php
// lecturer/enter_grades.php
require_once '../includes/config.php';
require_once '../includes/auth.php';

requireRole('lecturer');

$user = getCurrentUser($pdo);
$pageTitle = "Enter Grades";
$msg = '';
$msgType = '';

// Get Lecturer ID
$stmt = $pdo->prepare("SELECT id FROM lecturers WHERE user_id = ?");
$stmt->execute([$user['id']]);
$lecturer = $stmt->fetch();

// Fetch Assigned Courses
$stmt = $pdo->prepare("SELECT id, course_code, course_name FROM courses WHERE lecturer_id = ?");
$stmt->execute([$lecturer['id']]);
$courses = $stmt->fetchAll();

$course_id = isset($_GET['course_id']) ? (int)$_GET['course_id'] : (count($courses) > 0 ? $courses[0]['id'] : null);
$selected_course = null;
$students = [];

// Helper function to calculate letter grade based on final score (Makerere/Busitema grading scale approx)
function calculateGrade($score) {
  if ($score === null || $score === '') return null;
  $s = (float)$score;
  if ($s >= 80) return 'A';
  if ($s >= 75) return 'B+';
  if ($s >= 70) return 'B';
  if ($s >= 65) return 'C+';
  if ($s >= 60) return 'C';
  if ($s >= 55) return 'D+';
  if ($s >= 50) return 'D';
}
```

vii) Transcript Generation

```
1  <?php
2  // student/transcript.php
3  require_once '../includes/config.php';
4  require_once '../includes/auth.php';
5
6  requireRole('student');
7
8  $user = getCurrentUser($pdo);
9  $pageTitle = "Academic Transcript";
10
11 // Get Student ID
12 $stmt = $pdo->prepare("SELECT * FROM students WHERE user_id = ?");
13 $stmt->execute([$user['id']]);
14 $student = $stmt->fetch();
15
16 // Fetch all grades and courses
17 $stmt = $pdo->prepare("
18     SELECT c.course_code, c.course_name, c.credits, c.semester, c.year as course_year,
19           g.coursework_score, g.exam_score, g.final_score, g.grade
20     FROM enrollments e
21     JOIN courses c ON e.course_id = c.id
22     LEFT JOIN grades g ON e.id = g.enrollment_id
23     WHERE e.student_id = ?
24     ORDER BY c.year ASC, c.semester ASC
25 ");
26 $stmt->execute([$student['id']]);
27 $records = $stmt->fetchAll();
28
29 // Group records by Year and Semester
30 $transcript = [];
31 $total_credits = 0;
32 $earned_credits = 0; // simplified, just for display
```

Application Logic

The system application logic controls how information is processed from user input to database storage and output generation. During student registration, the system validates entered details before storing student information in the MySQL database. During grade entry, lecturers submit marks which are checked and stored against the correct student and course records.

Grade calculation logic involves converting marks into corresponding grades according to the grading scale defined by the institution. The GPA calculation module retrieves completed course grades, assigns grade points, multiplies them by course credit units, and divides the total weighted points by the total credit units attempted. The transcript generation module retrieves student details, registered courses, grades, and calculated GPA, then presents them in a structured academic transcript format.

Database Relationships

The database is designed using related tables to maintain data integrity. The Students table stores student personal information, while the Courses table stores course details. The Grades table

links students and courses by using foreign keys to record academic performance. User accounts are linked with system roles to control access permissions. These relationships reduce data duplication and ensure that information can be retrieved efficiently.

Security Implementation

Security was implemented to protect confidential student information. User passwords are stored using password hashing instead of plain text storage. Prepared statements are used when communicating with the database to prevent SQL injection attacks. Role-based access control ensures that administrators, lecturers, and students can only access functions permitted to their roles.

Example PHP security implementation:

Client-Side Validation and Responsive Design

JavaScript validation is used on forms to improve data accuracy before information is submitted to the server. It checks required fields, validates formats such as email addresses, and prevents incomplete submissions. Bootstrap was used to support responsive design, allowing the system interface to adjust to different screen sizes including desktop computers, tablets, and mobile devices.

System Navigation and Page Flow

Users access the system through the login page where authentication is performed. After successful authentication, users are redirected to dashboards according to their roles. Administrators access student management, user management, and reports. Lecturers access grade entry and academic records. Students access their profiles, academic results, and transcript pages. Navigation menus provide links between system modules for easier movement through the application.

Expanded System Testing and Validation

Comprehensive testing was carried out to verify that the developed system met functional, usability, security, and performance requirements.

Unit Testing: Individual modules such as login, student registration, grade entry, and transcript generation were tested independently to identify errors within specific components.

Integration Testing: Different modules were combined and tested to confirm that information flowed correctly between components. For example, grades entered by lecturers were checked to ensure they appeared correctly on student transcripts.

User Acceptance Testing (UAT): Administrators, lecturers, and students tested the system to confirm that it satisfied their requirements and was easy to use.

Performance Testing: System response time and record retrieval speed were measured to ensure that the system could efficiently process user requests.

Security Testing: Authentication controls, password protection, role permissions, and database security mechanisms were tested to confirm protection of student information.

CHAPTER SIX: DISCUSSION, CONCLUSION, RECOMMENDATION

Introduction

This chapter presents a discussion of the findings, conclusion, recommendations, limitations, and future work.

Discussion of results

The implementation of a web-based Student Information Management System successfully addressed the key challenges identified in the ICT Department.

First, the system significantly reduced record retrieval time. The average time to locate and view a complete student record dropped from over 25 minutes using the manual system to under 2 minutes using the new system, achieving the project's primary performance objective and aligning with benchmarks established by Chen (2019).

Second, data integrity was vastly improved. The centralized MySQL database with validation rules eliminated duplicate entries and inconsistent data, addressing the 16% error rate found during the initial file review.

Third, security is now robust. The implementation of hashed passwords, role-based access control, and session management ensures compliance with Uganda's Data Protection and Privacy Act (2019), a critical improvement over the previous open-access file system.

Fourth, report generation is now highly efficient. The system can generate a class list or student transcript instantly, compared to the 3-4 hours previously required for manual compilation.

User Acceptance Testing showed high satisfaction. The post-implementation questionnaire, based on the Technology Acceptance Model, revealed that 92% of users perceived the system as useful, and 88% found it easy to use.

Conclusion

The development of a web-based Student Information Management System for the ICT Department at Busitema University (Nagongera Campus) has been successfully completed. The system meets all the specified objectives: requirements were identified through a thorough field study; a robust system was designed using industry-standard diagrams; a functional system was implemented using PHP and MySQL; and rigorous testing validated its functionality, performance, and usability.

The system replaces the inefficient, fragmented manual and semi-digital methods with a centralized, efficient, and secure platform. It significantly improves administrative efficiency, enhances data integrity, ensures data security, and provides a better experience for students and lecturers.

Recommendations

I therefore recommend the adoption and use of this web-based Student Information Management System by the ICT Department at Busitema University, Nagongera Campus. It is further recommended that:

1. The university's IT department provide hosting and maintenance support for the system.
2. Regular backups of the database be scheduled to prevent data loss.
3. Training sessions be conducted for all staff and students before full deployment.
4. The system be considered for piloting in other departments within the Faculty of Science and Education.

Limitations

The system was tested with a limited user group (21 participants) and over a short period, which may not reveal all long-term issues. The system is a standalone prototype and does not integrate with the central university systems (e.g., AR system, Library system), limiting full automation.

Future work

I intend to extend this system by integrating it with the university's central database via APIs to allow automatic data synchronization.

I plan to add more advanced features such as online course registration, fee management integration, and automated email/SMS notifications for grade postings.

I intend to develop a mobile application version of the system to provide even greater accessibility for students and lecturers.

Finally, I plan to conduct a longitudinal study on the system's impact on administrative efficiency and student satisfaction over multiple academic semesters.

REFERENCES

- Abrams, L. (2020). Data governance and security in modern educational platforms. *Journal of Educational Technology & Society*, 23(4), 112-125.
- African Development Bank. (2024). Digital transformation in African higher education: Status and prospects. Abidjan: AfDB Publications.
- Brown, R. (2019). User-centred design principles for academic software development. London: Academic Press.
- Busitema University. (2024). Annual report 2023/2024. Busitema: University Publications.
- Chen, X. (2019). Digital transformation in higher education administration: A meta-analysis. *Higher Education Management*, 41(2), 78-92.
- Davis, F. D. (1989). Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Quarterly*, 13(3), 319-340.
- Davis, F. (2018). Shadow systems and workarounds: The failure of ERP systems in academic departments. *International Journal of Educational Management*, 32(5), 879-893.
- Educause. (2023). EDUCAUSE Horizon Report: Teaching and learning edition. Louisville: EDUCAUSE Publications.
- Elmasri, R., & Navathe, S. B. (2016). *Fundamentals of database systems* (7th ed.). Boston: Pearson.
- Garcia, M. (2021). The impact of integrated academic advising systems on student performance. *Journal of College Student Development*, 62(3), 345-360.
- Laudon, K. C., & Laudon, J. P. (2022). *Management information systems: Managing the digital firm* (17th ed.). Harlow: Pearson Education.
- Mwangi, P., & Ochieng, F. (2024). Web-based student management systems in Kenyan universities. *East African Journal of Information Systems*, 7(1), 56-78.
- National Council for Higher Education. (2024). The state of higher education in Uganda 2023/2024. Kampala: NCHE Publications.
- Okello, J., & Namusoke, J. (2023). Record-keeping practices in Ugandan public universities. *Uganda Journal of Education*, 15(2), 112-134.
- Smith, P., & Jones, T. (2020). Administrative efficiency in higher education: A comparative study. *Administrative Science Quarterly*, 65(1), 154-179.
- Uganda Government. (2019). The Data Protection and Privacy Act, 2019. Kampala: Government of Uganda.

Williams, H., Thompson, K., & Lee, S. (2022). Effectiveness of student management systems in higher education. *Computers & Education*, 180, 104433.

APPENDICES

Appendix A: Interview Guide (Administrative Staff)

Interview Guide (Administrative Staff)

Title: Departmental Student information System Requirements Gathering

Introduction: Thank you for participating in this interview. The purpose is to understand current student record management processes and identify requirements for a new system. Your responses will be confidential and used only for academic purposes. This interview will take approximately 30-45 minutes.

Section A: Demographics

1. What is your role in the department? (Please tick)

Head of Department

Departmental Secretary

Records Assistant

Lecturer

Other (please specify): _____

2. How long have you held this position?

Less than 1 year

1-3 years

4-6 years

More than 6 years

3. Rate your computer proficiency on a scale of 1-5 (1 = Beginner, 5 = Expert): _____

Section B: Current Processes

1. Please describe your role in managing student records.

.....

2. What is the current process for registering new students?
.....
3. How are student grades collected, verified, and stored?
.....

Section C: Challenges

1. What are the main challenges you face with the current system?
.....
.....
.....
2. How much time do you typically spend on record-related tasks weekly?
.....
3. Have you experienced data loss or errors? Please describe.
.....
.....

Section D: Desired Features

1. What features would you like in a new student management system?
.....
.....
.....
.....
.....
2. What would make your work easier regarding student records?
.....
.....
.....
3. Are there specific reports you need regularly?
.....
.....

Section E: Technical Considerations

1. What training would you need to use a new system?

.....
.....

2. Do you have any concerns about transitioning to a digital system?

.....
.....

Closing Statement

Thank you for taking the time to participate in this interview. Your input is extremely valuable for designing a system that truly meets the needs of the ICT Department.

Next Steps: The information you have provided will be analyzed alongside other interviews to identify key requirements for the system to be designed.

Appendix B: Student Questionnaire

Title: Student Survey on Departmental Record Management

Introduction: This questionnaire aims to understand your experiences with current student record processes in the ICT Department. Your responses will help design a new system that better serves students. All responses are anonymous and will be used only for academic purposes.

Section A: Demographics

1. What is your current year of study?

Year 1

Year 2

Year 3

2. Rate your computer proficiency (1 = Beginner, 5 = Expert): _____

Section B: Current Experience (Please rate on a scale of 1-5 where 1 = Strongly Disagree, 2 = Disagree, 3 = Neutral, 4 = Agree, 5 = Strongly Agree)

STATEMENT	1	2	3	4	5
I can easily access my academic records when needed					
The current process for obtaining transcripts or reports is efficient					
I have experienced delays in getting academic information					
I trust that my academic records are accurately maintained					
I am concerned about the security of my personal data					

Section C: Expectations for New System (Please tick your choice(s))

STATEMENT	YES	NO
make it easier to access A digital system would my academic information		
I would feel comfortable using a web-based student portal		
I would prefer to access my grades online rather than through manual requests		
Having 24/7 access to my academic records is important to me		

Section D: Open-Ended Questions

1. What challenges have you faced when trying to access your academic records?
2. What features would you like to see in a student management system?
3. Any other comments or suggestions?

Thank you for your participation!