

**ONLINE RESULT PROCESSING SYSTEM
(Case Study of College of Health Technology, Michika)**

**BY
LUKA ISHAYA
(ST/CS/HND/23/026)**

**DEPARTMENT OF COMPUTER SCIENCE,
SCHOOL OF SCIENCE AND TECHNOLOGY,
FEDERAL POLYTECHNIC, MUBI, ADAMAWA STATE.**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF
HIGHER NATIONAL DIPLOMA (HND) IN COMPUTER SCIENCE.**

JULY, 2025

DECLARATION

I hereby declare that the work in this project titled “**Online Result Processing System (Case study of College of Health Technology, Michika)**” was performed by me under the supervision of Mrs. Aaron Catherine. The information derived from literatures has been duly acknowledged in the text and a list of references provided. The work embodied in this project is original and had not been submitted in part or in full for any other diploma or certificate of this or any other institution.

Luka Ishaya

(ST/CS/HND/23/026)

Signature

Date

CERTIFICATION

This project titled “**Online Result Processing System (Case study of College of Health Technology, Michika)**” meets the regulations governing the award of Higher National Diploma (HND) in Computer Science, Federal Polytechnic Mubi, Adamawa State

Mrs. Aaron Catherine
(Project Supervisor)

Sign/Date

Mr. Kassim Mustapha
(Head of Department)

Sign/Date

Mal. Abdulrahman Saidu
(External Examiner)

Sign/Date

DEDICATION

This project is dedicated to my father Late Mr. Vandi Luka and my beloved mother Mrs. Hajara Luka for her advice, encouragement and financial support towards my academic pursuit.

ACKNOWLEDGEMENTS

I want to acknowledge Almighty God for his infinite mercy and protection throughout my academic activities. And for the understanding in achieving our academic success.

I appreciate my Supervisor Mrs. Aaron Catherine who took time despite her busy schedule to direct and guide me throughout this research work.

I also acknowledge the Head of Department Computer Science Mal. Kassim Mustapha for his moral encouragement throughout my period of study.

I wish to acknowledge Mrs. Lucy Bulus and all Staff of Computer Science Department for their support and encouragement and the knowledge they have impacted on me throughout my studies.

My appreciation goes to my lovely parent Mrs. Hajara Luka for her love and care and for giving me the opportunity to be trained and achieve my dreams.

Finally, I appreciate the efforts of my Uncles and aunties, for their encouragement and support throughout the course of my study and also my friends and relatives, course mates and all well-wishers. I love you all, may the Almighty God bless you abundantly, Amen.

TABLE OF CONTENTS

TITLE PAGE	i
DECLARATION	ii
CERTIFICATION.....	iii
DEDICATION	iv
ACKNOWLEDGEMENTS	v
TABLE OF CONTENTS	vi
LIST OF FIGURES.....	viii
LIST OF TABLES.....	ix
ABSTRACT.....	x
CHAPTER ONE	1
INTRODUCTION	1
1.1 Background to the Study	1
1.2 Problem Statement	2
1.3 Aim and Objectives	3
1.4 Significance of the Study	3
1.5 Scope of the Study.....	4
1.6 Definition of Some Operational Terms	4
CHAPTER TWO	5
LITERATURE REVIEW	5
2.1 Introduction	5
2.2 Concept of Cloud Computing in Academic Institutions	5
2.2.4 Cloud Computing and Secure Academic Result Processing.....	6
2.3 Theoretical Framework	7
2.3.1 Diffusion of Innovation (DOI) Theory.....	7
2.3.2 Technology Acceptance Model (TAM)	8
2.4 Related Works on Secure Result Processing in Cloud Computing.....	8
2.5 Summary of Related Works	10
CHAPTER THREE.....	11
SYSTEM ANALYSIS AND DESIGN.....	11
3.1 Introduction	11
3.2 Disadvantages of the Existing System	11
3.3 Advantages of the Proposed System	12
3.4 Software Development Model.....	12

3.5	Method of Data Collection	14
3.6	System Design	14
3.6.1	Algorithm diagram	15
3.6.2	System Architecture	15
3.6.3	Database Tables/Queries Structures	16
3.6.4	Database Entity Relationship Diagram	17
3.6.5	Input and Output Design	18
3.7	System Requirement Specification	20
3.7.1	Hardware Requirements	20
3.7.2	Software Requirements	20
3.7.3	Personnel Requirement.....	20
CHAPTER FOUR.....		21
RESULTS AND DISCUSSION		21
4.1	Introduction	21
4.2	Results	21
4.2.6	Dashboard Interface	24
4.2.7	Result Processing Interface	24
4.3	Discussion	25
4.4	User manual.....	26
4.4.1	System Installation	26
4.4.2	System Operational Guide.....	27
CHAPTER FIVE		28
SUMMARY, CONCLUSION AND RECOMMENDATIONS		28
5.1	Summary	28
5.2	Conclusion.....	28
5.3	Recommendations	29
REFERENCES.....		30
APPENDIX A.....		32
APPENDIX B.....		35

LIST OF FIGURES

Figure 3.1: Waterfall Model	-	-	-	-	-	-	-	14
Figure 3.2: Use Case Diagram	-	-	-	-	-	-	-	15
Figure 3.3: System Architecture	-	-	-	-	-	-	-	15
Figure 3.4: Entity Relationship Model	-	-	-	-	-	-	-	17
Figure 3.5: Add Class	-	-	-	-	-	-	-	18
Figure 3.6: Login form	-	-	-	-	-	-	-	18
Figure 3.7: Add Course	-	-	-	-	-	-	-	18
Figure 3.8: Add Result	-	-	-	-	-	-	-	19
Figure 3.9: Student Result	-	-	-	-	--	-	-	19
Figure 4.1: Welcome Interface	-	-	-	-	-	-	-	21
Figure 4.2: Login page interface	-	-	-	-	-	-	-	22
Figure 4.3: Add Student interface	-	-	-	-	-	-	-	22
Figure 4.4: Add Department	-	-	-	-	-	-	-	23
Figure 4.5: Registered Students Interface	-	-	-	-	-	-	-	23
Figure 4.6: Dashboard interface	-	-	-	-	-	-	-	24
Figure 4.7: Result Processing interface			-	-	-	-	-	24

LIST OF TABLES

Table 1: Admin Table	-	-	-	-	-	-	-	-	16
Table 3.2: Class -	-	-	-	-	-	-	-	-	16
Table 3.3: Course	-	-	-	-	-	-	-	-	16
Table 3.4: Result	-	-	-	-	-	-	-	-	16
Table 3.5: Student Records	-	-	-	-	-	-	-	-	17

ABSTRACT

This project focuses on the design and implementation of a secure online result processing system, using the College of Health Technology, Michika as a case study. The system leverages the power of cloud computing to address the persistent challenges associated with traditional result processing methods, such as manual errors, data loss, limited accessibility, and inadequate security. With increasing threats to academic data integrity, the project integrates robust security measures including encryption, multi-factor authentication, and role-based access control to ensure confidentiality and protect student records from unauthorized access. The study adopts the Waterfall software development model, systematically progressing through requirement analysis, system design, implementation, testing, and deployment. Key components of the system include a web-based interface for both students and staff, a cloud-based database for storing and managing results, and automated result computation and report generation modules. The solution provides remote accessibility, real-time updates, and audit trails, thereby enhancing efficiency, transparency, and accountability in result management. Primary and secondary data were gathered through interviews, observation, academic journals, and library sources to guide the design and validate the system's relevance. The implementation of this system is expected to significantly improve result processing efficiency, enhance user experience, and promote digital transformation within academic institutions.

CHAPTER ONE

INTRODUCTION

1.1 Background to the Study

The advancement of cloud computing has significantly transformed data processing and storage, making it an essential technology for various sectors, including education. Secure results processing is a critical concern for academic institutions that handle sensitive student information, such as examination results, grades, and academic records. Traditional result processing methods often involve manual entry, paper-based documentation, and locally hosted software systems, which can be prone to security vulnerabilities, inefficiencies, and errors (Smith & Brown, 2022).

Cloud computing provides a scalable, cost-effective, and secure environment for processing and storing examination results. Institutions can leverage cloud-based solutions to ensure data integrity, confidentiality, and availability, reducing risks associated with unauthorized access, data breaches, and loss of records (Johnson *et al.*, 2021). Implementing a secure results processing system in cloud computing enhances automation, facilitates real-time access, and provides a centralized platform for academic institutions to manage student results efficiently.

Security in cloud computing remains a significant concern due to potential cyber threats such as hacking, data leakage, and unauthorized modifications. Therefore, adopting robust security measures such as encryption, multi-factor authentication, and role-based access control is necessary to ensure the confidentiality and integrity of academic records (Anderson & White, 2023). By integrating these security features, institutions can safeguard student data while leveraging the benefits of cloud computing.

Furthermore, the adoption of cloud-based result processing systems enhances accessibility and efficiency for both students and academic staff. Unlike traditional systems that require physical presence or on-premises infrastructure, cloud solutions enable students to access their results from anywhere, at any time, using secure login credentials. This level of convenience reduces administrative workload, minimizes delays in result dissemination, and ensures a seamless academic experience. Additionally, automated result processing eliminates human errors associated with manual data entry, leading to more accurate and reliable academic records (Williams *et al.*, 2021). Cloud-based result management systems also support interoperability, allowing seamless integration with existing institutional software, such as learning management systems and student information portals (Kumar & Lee, 2020).

In addition to improving efficiency, cloud-based result processing supports compliance with educational data protection regulations and institutional policies. Academic institutions must adhere to strict data governance frameworks to prevent unauthorized access and misuse of student records. By leveraging cloud computing, schools and universities can implement standardized security protocols, regular data backups, and real-time monitoring to detect and mitigate potential threats (El-Gazzar *et al.*, 2020). The implementation of cloud security frameworks ensures that student academic records are not only protected but also preserved in a structured and retrievable format for future reference.

Moreover, cloud computing promotes cost-effectiveness by reducing the need for expensive hardware infrastructure and minimizing the burden of maintaining on-premises data centers (Miller & Evans, 2022). With cloud-based systems, institutions can adopt a pay-as-you-go model, which allows them to scale their storage and processing capabilities based on demand. This financial flexibility enables educational institutions to allocate resources efficiently, ensuring that funds are directed toward improving academic programs and student support services (Hughes, 2019).

As cloud technology continues to evolve, its integration into academic result processing will play a crucial role in advancing digital transformation in education, ensuring transparency, security, and efficiency in academic record management. The growing adoption of artificial intelligence (AI) and machine learning (ML) in cloud-based systems further enhances result processing by enabling predictive analytics, automated grading, and personalized student feedback (Rahman *et al.*, 2023). Therefore, implementing a secure results processing system in cloud computing is not only a step toward improving academic administration but also a pathway to fostering a more technology-driven and efficient education system.

The Federal Polytechnic Mubi Staff School, like many other academic institutions, faces challenges in securely processing and managing examination results. This study aims to design and implement a secure cloud-based results processing system that mitigates security risks, improves efficiency, and ensures compliance with data protection regulations.

1.2 Problem Statement

Despite the increasing adoption of digital solutions in education, many institutions still rely on outdated and insecure methods for processing student results. The key challenges include:

- i. The reliance on manual result processing methods increases the risk of errors, data loss, and inefficiencies.

- ii. Traditional systems lack robust security mechanisms, making them vulnerable to cyber threats such as unauthorized access and data breaches.
- iii. There is no centralized system for result processing, leading to delays and inconsistencies in the dissemination of academic records.
- iv. Students and staff face difficulties in accessing results remotely due to the limitations of on-premise storage systems.
- v. Compliance with data protection policies is often inadequate, exposing institutions to legal and ethical risks.

Implementing a secure cloud-based results processing system can address these challenges by providing a scalable, efficient, and secure platform for managing student academic records.

1.3 Aim and Objectives

The aim of this study is to implement an Online Result Processing System using (case study of College of Health Technology, Michika). The specific objectives include:

- i. To design a cloud-based system that ensures the secure processing and storage of student results.
- ii. To create robust security measures, including encryption and multi-factor authentication, to prevent unauthorized access.
- iii. To develop a user-friendly interface for students and staff to access results securely.
- iv. To evaluate the effectiveness of the system in enhancing data security, accessibility, and processing efficiency.

1.4 Significance of the Study

The implementation of a secure result processing system in cloud computing provides a reliable and efficient alternative to traditional methods, significantly reducing errors and security risks. By leveraging encryption, authentication mechanisms, and secure access controls, the system ensures that student academic records remain protected from unauthorized modifications or breaches. This enhances the credibility of the grading process and eliminates common issues associated with manual result processing, such as data loss, duplication, and unauthorized access.

Moreover, the system improves accessibility by allowing students and academic staff to retrieve results remotely via a secure cloud platform. Unlike conventional methods that require physical presence or paper-based documentation, a cloud-based solution ensures that results can be accessed anytime and anywhere, enhancing convenience and efficiency. Additionally, the system enforces compliance with data protection regulations, ensuring that sensitive student records are handled securely and in accordance with institutional policies and legal standards.

Beyond security and accessibility, the proposed system contributes to the advancement of cloud computing applications in academic institutions, promoting digital transformation in education. Automating result processing tasks reduces administrative burdens, allowing academic staff to focus on more critical responsibilities such as student engagement and curriculum development. By integrating cloud technology into academic record management, institutions can streamline operations, improve efficiency, and enhance overall educational service delivery.

1.5 Scope of the Study

This study focuses on the design, development, and implementation of a secure cloud-based results processing system for College of Health Technology, Michika. The system includes result entry, verification, storage, and retrieval functionalities. Security measures such as encryption, authentication, and access control will be integrated to ensure data protection. The study does not cover non-academic data processing or unrelated cloud computing applications.

1.6 Definition of Some Operational Terms

Access Control: A security feature that restricts access to data based on user roles and permissions (Williams *et al.*, 2021).

Authentication: A process that verifies the identity of users before granting access to a system (Patil *et al.*, 2024).

Cloud Computing: A technology that enables remote storage, processing, and management of data over the internet instead of local servers or personal computers (Miller & Evans, 2022).

Cyber Threats: Malicious activities aimed at compromising digital systems and data (Smith, 2023).

Data Security: Measures taken to protect digital data from unauthorized access, corruption, or theft (Hughes, 2019).

Encryption: A security technique that converts data into a coded format to prevent unauthorized access (Kim & Lee, 2021).

Results Processing System: A digital platform used to compute, store, and manage students' academic results (Johnson & Brown, 2022).

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter reviews existing literature on secure result processing systems in cloud computing. It explores fundamental concepts, theoretical frameworks, related works, advantages, challenges, and security measures in cloud-based academic result processing. The review provides insights into how cloud computing enhances data security, accessibility, and efficiency while addressing potential threats and implementation challenges.

2.2 Concept of Online Result Processing

Cloud computing is a technology that enables on-demand access to computing resources such as storage, servers, and applications over the internet (Mell & Grance, 2011). It has revolutionized data management in various sectors, including education, by providing institutions with scalable and cost-effective solutions for handling vast amounts of academic data (Patel *et al.*, 2022). In academic institutions, cloud computing is utilized for managing student records, e-learning platforms, and result processing, ensuring seamless access to critical information for both students and faculty members (Kumar & Lee, 2020).

Cloud-based result processing systems provide institutions with an efficient means of managing and disseminating student examination results securely. Unlike traditional manual result processing methods, which are prone to errors, delays, and data loss, cloud-based systems enhance automation, accuracy, and real-time access (Williams *et al.*, 2021). These systems allow students to access their academic results from any location using secure credentials, reducing the administrative burden on staff and eliminating the need for physical result sheets (Rahman *et al.*, 2023). As cloud technology continues to evolve, its role in academic institutions is expected to expand significantly. The adoption of Artificial Intelligence (AI) and Machine Learning (ML) in cloud-based educational platforms will enhance personalized learning experiences, automate administrative processes, and improve decision-making for academic staff (Patel *et al.*, 2022). AI-powered cloud solutions can provide intelligent tutoring systems, predictive analytics for student performance, and automated grading systems, reducing the workload on educators (Rahman *et al.*, 2023).

The integration of cloud computing with the Internet of Things (IoT) is another emerging trend in academia. IoT-enabled cloud systems can support smart classrooms, real-time student attendance tracking, and remote laboratory access, improving the overall learning experience (Hashem *et al.*, 2015). These technologies will further enhance remote learning and digital education, enabling

students to access high-quality educational content from anywhere in the world. Moreover, academic institutions will increasingly adopt blockchain technology alongside cloud computing to enhance data security and transparency in result processing and credential verification (Johnson *et al.*, 2021). Blockchain-powered cloud platforms can provide tamper-proof digital certificates and transcripts, reducing fraud and improving the credibility of academic records (El-Gazzar *et al.*, 2020). As institutions continue to embrace cloud computing, the focus will shift towards sustainable and green cloud solutions that reduce energy consumption and carbon footprints. Cloud providers are investing in eco-friendly data centers powered by renewable energy, ensuring that educational institutions can benefit from environmentally responsible computing solutions (Williams *et al.*, 2021).

2.2.4 Cloud Computing and Secure Academic Result Processing

Academic institutions generate large volumes of student data, including examination results, transcripts, and attendance records. Traditional result processing methods, which rely on manual data entry and paper-based documentation, are prone to errors, inefficiencies, and security risks (Smith & Brown, 2022). Cloud-based result processing systems offer a secure and efficient alternative by automating result computation, storage, and retrieval while ensuring data integrity (Johnson *et al.*, 2021). Moreover, cloud computing enables institutions to implement advanced security measures such as role-based access control (RBAC), data encryption, and audit logging to prevent unauthorized access and ensure compliance with data protection regulations (El-Gazzar *et al.*, 2020). By adopting cloud computing, academic institutions can enhance transparency, streamline administrative processes, and provide students with real-time access to their academic records.

Cloud-based result processing also facilitates scalability and flexibility, allowing institutions to efficiently manage large datasets without the constraints of physical storage infrastructure (Hashem *et al.*, 2015). As student enrollment numbers grow, cloud platforms can dynamically scale storage and processing power to accommodate increasing data volumes. Unlike traditional systems that require periodic upgrades and maintenance, cloud solutions offer automatic updates and seamless integration with other academic management systems, reducing administrative burdens (Zhang *et al.*, 2020). Another key advantage is real-time access and remote availability. Cloud-based result processing platforms allow students, faculty, and administrative staff to securely access examination results and academic records from anywhere, eliminating the need for physical visits to administrative offices (Rahman *et al.*, 2023). This level of accessibility is particularly beneficial for distance learning institutions and universities with multiple campuses, as it promotes inclusivity and enhances the overall student experience. Additionally, automated

notifications can be implemented to alert students of newly released results, minimizing delays in result dissemination (Williams *et al.*, 2021).

Disaster recovery and data redundancy are also significant benefits of using cloud computing for result processing. Unlike traditional storage methods that are vulnerable to physical damage, cyberattacks, or accidental data loss, cloud-based systems employ automated backups and redundancy mechanisms to ensure data availability even in cases of unexpected failures (El-Gazzar *et al.*, 2020). Institutions can rely on cloud providers' data recovery solutions to restore lost or corrupted records, thereby enhancing data security and operational continuity (Anderson & White, 2023). Additionally, compliance with data protection regulations is a crucial factor in academic result processing. Many cloud service providers comply with international standards such as the General Data Protection Regulation (GDPR) and the Family Educational Rights and Privacy Act (FERPA), ensuring that student data is handled securely and transparently (Almorsy *et al.*, 2016). By leveraging cloud-based solutions, institutions can establish access control mechanisms, audit trails, and user authentication protocols to prevent unauthorized modifications and breaches, safeguarding the credibility of academic records (Johnson *et al.*, 2021).

Looking ahead, advancements in Artificial Intelligence (AI) and Blockchain Technology are expected to further enhance secure academic result processing in the cloud. AI-driven analytics can help detect anomalies in result processing, flagging potential errors or inconsistencies before they are finalized (Patel *et al.*, 2022). Meanwhile, blockchain technology can introduce tamper-proof ledgers for academic records, ensuring that grades and transcripts cannot be altered without authorization, thereby enhancing trust in result authenticity (Williams *et al.*, 2021).

2.3 Theoretical Framework

Several theories support the adoption of cloud computing for academic result processing, including the Diffusion of Innovation (DOI) Theory and the Technology Acceptance Model (TAM).

2.3.1 Diffusion of Innovation (DOI) Theory

Rogers' (1995) Diffusion of Innovation (DOI) Theory explains how new technologies spread within organizations and societies. It suggests that innovations are adopted through five stages: knowledge, persuasion, decision, implementation, and confirmation. Cloud-based result processing systems align with this theory as institutions transition from traditional manual methods to modern digital solutions. The increasing need for secure and efficient result processing has driven many educational institutions to adopt cloud technology (El-Gazzar *et al.*, 2020).

2.3.2 Technology Acceptance Model (TAM)

The Technology Acceptance Model (TAM), developed by Davis (1989), explains how users accept and use new technologies. The model identifies two key factors influencing adoption: perceived usefulness and perceived ease of use. Academic institutions adopt cloud-based result processing systems due to their efficiency, accessibility, and ability to reduce administrative workload. Studies have shown that institutions with well-structured cloud adoption policies experience improved data security, reduced errors, and enhanced service delivery (Johnson *et al.*, 2021).

2.4 Related Works on Secure Result Processing in Cloud Computing

Several studies have explored the application of cloud computing in secure academic result processing. Researchers have investigated various aspects, including security mechanisms, efficiency improvements, data integrity, and accessibility in cloud-based result management systems.

Smith and Brown (2022) examined the implementation of a secure cloud-based examination result processing system to enhance data security and minimize errors associated with manual result computation. Their study highlighted the importance of encryption and access control in preventing unauthorized alterations to student records.

Furthermore, the researchers emphasized that cloud-based result processing reduces administrative workload by automating result entry and retrieval. By implementing multi-layer encryption techniques, academic institutions can ensure that student results remain confidential and protected from cyber threats. The study also suggested that integrating biometric authentication could further enhance security and prevent unauthorized access.

Additionally, Smith and Brown (2022) noted that adopting a cloud-based system improves scalability, allowing institutions to manage increasing student populations without compromising system efficiency. Their findings suggested that real-time access to results via secure portals enables students to retrieve their grades instantly, promoting transparency and reducing disputes related to grade processing delays.

Johnson *et al.* (2021) proposed a cloud-based framework for academic record management, incorporating blockchain technology to ensure data integrity. Their study found that blockchain's immutable ledger significantly reduced the risk of data tampering in cloud environments.

By utilizing a decentralized approach, blockchain technology ensures that all modifications to academic records are permanently logged and verified across multiple nodes. This prevents unauthorized alterations and provides a verifiable audit trail for educational institutions. The study

highlighted that blockchain-integrated result processing improves institutional credibility and fosters trust among students and faculty members.

Additionally, Johnson *et al.* (2021) demonstrated that blockchain technology enables automatic verification of credentials, reducing fraud associated with fake academic certificates. The study recommended further research on integrating artificial intelligence with blockchain to detect anomalies in student records, thereby enhancing accuracy and security in academic result processing.

El-Gazzar *et al.* (2020) developed a role-based access control (RBAC) model for securing academic result processing in cloud computing. The study emphasized that restricting access based on user roles (students, lecturers, administrators) enhances data confidentiality and prevents unauthorized modifications.

The researchers implemented an access control system that assigned specific permissions based on institutional roles. This ensured that lecturers could input grades while students could only view their final results, preventing accidental or intentional data alterations. The study found that RBAC significantly reduced security breaches caused by misconfigured access permissions.

Moreover, the study recommended integrating two-factor authentication (2FA) with RBAC to further strengthen access control mechanisms. El-Gazzar *et al.* (2020) concluded that combining RBAC with encryption protocols provides a multi-layered security approach, making cloud-based result processing systems more resilient against cyber threats.

Zhang *et al.* (2020) investigated the scalability of cloud-based result processing systems in large universities. Their findings indicated that cloud computing provides flexible and scalable infrastructure to handle growing student populations without compromising processing speed or data security.

The study highlighted that cloud-based result processing reduces delays by enabling parallel data processing across multiple servers. This significantly improves system responsiveness during peak result release periods. Zhang *et al.* (2020) demonstrated that using elastic cloud resources allows institutions to scale infrastructure dynamically based on workload demands.

Furthermore, the researchers found that cloud-based result systems reduce infrastructure costs, as institutions no longer require extensive on-premises hardware. The study suggested that combining cloud computing with machine learning algorithms could enhance result prediction models, providing early academic performance insights to educators.

Rahman *et al.* (2023) explored how cloud computing improves result accessibility for students and faculty. The study revealed that cloud-based result systems enable real-time access, eliminating geographical barriers and reducing administrative workload.

The researchers found that cloud-based platforms provide secure result retrieval via web and mobile applications, increasing convenience for students who may be off-campus. By integrating responsive design and user-friendly interfaces, institutions can improve student engagement and ease of access to academic records.

Additionally, Rahman *et al.* (2023) emphasized that cloud computing supports accessibility features such as voice-based navigation and text-to-speech functionality, making result processing systems more inclusive for students with disabilities. Their study recommended continuous improvements in accessibility features to ensure equal access to academic results for all students.

Patel *et al.* (2022) integrated artificial intelligence (AI) with cloud-based result processing to detect anomalies and inconsistencies in academic records. Their model used machine learning algorithms to identify potential errors before result publication, improving accuracy and reliability.

The study highlighted that AI-powered result verification can automatically flag unusual grading patterns, preventing errors caused by manual data entry mistakes. By utilizing predictive analytics, academic institutions can identify students at risk of failing and provide early interventions to support academic success.

Furthermore, Patel *et al.* (2022) demonstrated that AI-driven result processing reduces administrative burden by automating result compilation and generating comprehensive performance reports. Their findings suggested that future research should explore the integration of AI chatbots for automated student inquiries regarding academic results.

2.5 Summary of Related Works

The reviewed studies highlight key advancements in secure result processing in cloud computing. Researchers emphasize the importance of encryption, access control, scalability, AI-driven anomaly detection, and blockchain integration in improving academic result security and efficiency. Future studies may focus on refining cloud security mechanisms to counter evolving cyber threats and enhance trust in cloud-based academic systems.

CHAPTER THREE

SYSTEM ANALYSIS AND DESIGN

3.1 Introduction

This chapter contains the system design, the disadvantages of the existing system, the advantages of the proposed system over the existing system, the system requirements (Hardware and Software), the design and the system architecture.

3.2 Disadvantages of the Existing System

While traditional manual result processing systems have been used for years in educational institutions, they come with several inefficiencies and limitations. These disadvantages are particularly evident when compared to modern cloud-based solutions.

- i. **Inefficiency in Data Management:** The manual processing of student results is time-consuming and labor-intensive. It requires multiple steps, including data collection, entry, verification, and storage, which increases the workload on staff and delays the release of results.
- ii. **High Susceptibility to Errors:** Manual data entry is prone to human errors, such as miscalculations, incorrect data recording, and typographical mistakes. These errors can compromise the accuracy of student results, leading to disputes and potential academic setbacks.
- iii. **Lack of Security and Data Integrity:** The traditional system relies heavily on physical records or local storage, making it vulnerable to unauthorized access, data loss, and manipulation. There is also a risk of records being altered or tampered with, leading to concerns about result authenticity.
- iv. **Limited Accessibility and Scalability:** With a manual system, accessing student records requires physical presence or on-premise infrastructure. This restricts the ability of students, faculty, and administrative staff to access results remotely, making it difficult to manage large volumes of data efficiently.
- v. **Lack of Transparency and Auditability:** Manual processing lacks built-in tracking mechanisms, making it difficult to audit changes or identify errors. This can lead to inconsistencies and a lack of trust in the result management process.

3.3 Advantages of the Proposed System

The introduction of a secure cloud-based result processing system offers significant improvements over traditional manual methods.

- i. **Enhanced Efficiency and Automation:** The cloud-based system automates result processing tasks such as data entry, computation, storage, and retrieval, significantly reducing processing time and administrative workload.
- ii. **Improved Accuracy and Error Reduction:** Automated data entry and validation mechanisms help minimize human errors, ensuring that student results are calculated and stored correctly. Built-in algorithms verify data accuracy before final submission.
- iii. **Strengthened Security and Data Protection:** The system integrates robust security features, including encryption, multi-factor authentication, and access controls, to safeguard student data from unauthorized access, cyber threats, and data breaches.
- iv. **Remote Accessibility and Scalability:** A cloud-based system allows faculty and students to access results securely from any location with an internet connection. This flexibility supports institutions with multiple campuses and growing student populations.
- v. **Real-Time Updates and Modification Handling:** The system provides a streamlined approach for making corrections or modifications. Changes can be tracked and approved electronically, ensuring transparency while maintaining the integrity of records.
- vi. **Auditability and Transparency:** With built-in logging and audit trails, the cloud-based system keeps records of every change made, ensuring accountability and preventing unauthorized modifications. This fosters trust in the result processing system.
- vii. **Cost-Effectiveness and Resource Optimization:** By leveraging cloud computing, institutions can reduce infrastructure costs related to data storage and maintenance while improving overall system performance.

3.4 Software Development Model

To ensure a systematic and structured development process, the Waterfall model is adopted for the implementation of the secure result processing system. The model follows a linear and sequential approach, consisting of the following key phases:

3.4.1 Requirements Gathering and Analysis

In this phase, the project team collaborates with stakeholders, including administrators, faculty members, and IT professionals, to define the system requirements. Key activities include:

- i. Identifying pain points in the current result processing system.
- ii. Establishing security requirements, such as encryption and access control.

- iii. Defining system functionalities, including automated grading, report generation, and cloud integration.
- iv. Understanding compliance requirements for data protection and academic regulations.

3.4.2 System Design

Based on the gathered requirements, the system is designed with the following components:

- i. **User Interface (UI):** A web-based dashboard for students, faculty, and administrators to access results securely.
- ii. **Database Structure:** A cloud-based database architecture designed to store student records securely while enabling efficient data retrieval.
- iii. **Security Framework:** Implementation of encryption protocols, authentication mechanisms, and access control to safeguard sensitive information.
- iv. **Result Processing Engine:** An algorithmic module responsible for automatic result computation, error validation, and report generation.
- v. **Integration with Existing Systems:** The system is designed to integrate with learning management systems (LMS) and student portals for seamless data exchange.

3.4.3 Implementation

This phase involves:

- i. Developing the front-end interface using web technologies (HTML, CSS, JavaScript).
- ii. Implementing back-end logic using secure programming frameworks such as Django, Flask (Python), or Node.js.
- iii. Deploying the cloud-based database on platforms such as AWS, Google Cloud, or Microsoft Azure.
- iv. Implementing security measures, including encryption (AES, SSL/TLS) and authentication (OAuth, Multi-Factor Authentication).

3.4.4 Testing

Before deployment, rigorous testing is conducted to ensure the system meets performance, security, and functionality standards. Testing includes:

- i. **Unit Testing:** Checking individual components, such as login authentication and result computation.
- ii. **Integration Testing:** Verifying data flow between different system modules.
- iii. **Security Testing:** Assessing vulnerabilities and ensuring encryption effectiveness.
- iv. **User Acceptance Testing (UAT):** Allowing faculty and students to test the system and provide feedback before full deployment.

3.4.5 Deployment and Maintenance

Once testing is completed, the system is deployed to the cloud and made accessible to users.

Ongoing maintenance includes:

- i. Monitoring for security threats and system performance.
- ii. Regular software updates and patches to enhance functionality.
- iii. Collecting user feedback for future improvements

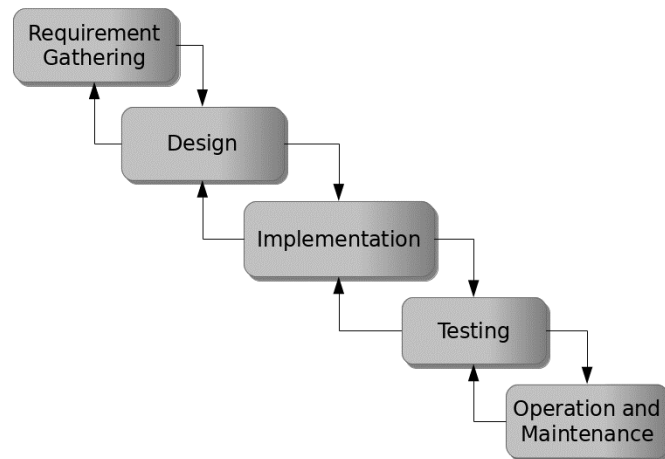


Figure 3.1: Waterfall model

3.5 Method of Data Collection

There are two main sources of data collection in carrying out this study, information was basically obtained from the two sources which are the primary and secondary source.

Primary Source: Primary source of data that will be used in this study will be personal interview and observation.

Secondary Source: The secondary data used in the study will be obtained from magazines, Journal, newspapers, library source and most of the information from the library research has been covered in my literature review in the previous chapter of this project.

3.6 System Design

Systems design is the process of defining the architecture, modules, interfaces, and data for a system to satisfy specified requirements. The system design of the secure results processing system in a cloud computing environment defines the architecture, components, and security measures needed for efficient and tamper-proof academic record management. It follows a three-tier cloud-based architecture, comprising a presentation layer (user interfaces for students and faculty), an application layer (result processing, authentication, and encryption modules), and a data layer (secure cloud storage and databases).

3.6.1 Algorithm diagram

Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system and depicting the specifications of a use case. A use case diagram shows the system and the various ways that they interact with the system as shown in Figure 3.2.

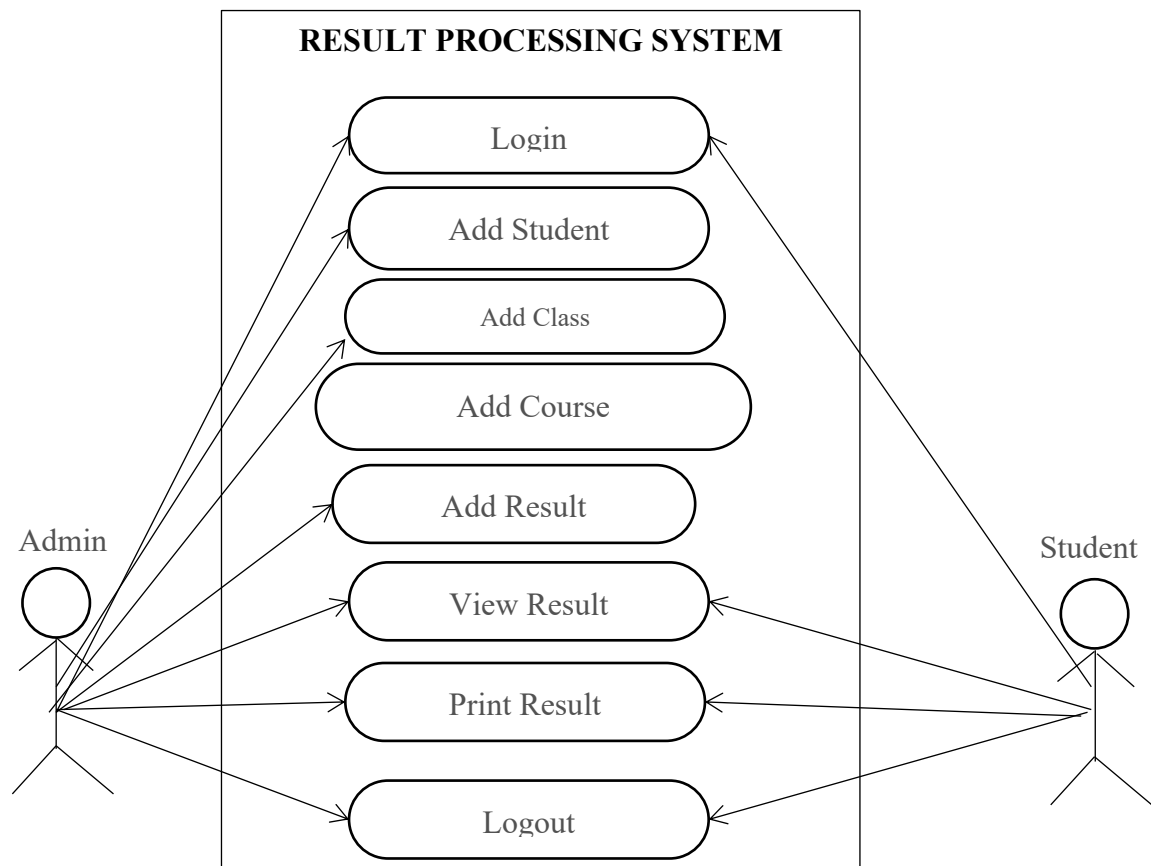


Figure 3.2: Use Case Diagram

3.6.2 System Architecture

System Architecture refers to the structural design of a system, defining how different components interact to achieve functionality, performance, and security. It provides a blueprint for organizing hardware, software, data flow, and user interactions.

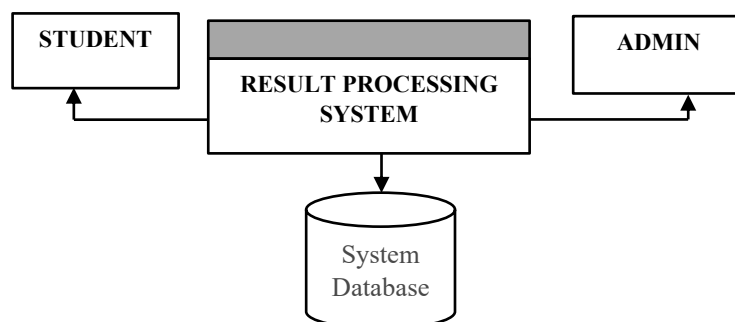


Figure 3.3: System Architecture

3.6.3 Database Tables/Queries Structures

The database is used to store all information that pertain the result processing records. Below are the database table for the new system.

Table 3.1: Admin Details

Name	Type	Extra
id	int(11)	AUTO_INCREMENT
Name	varchar(50)	
Password	varchar(50)	

Table 3.2: Class

Name	Type	Extra
id	int(11)	AUTO_INCREMENT
classname	varchar(250)	
Section	varchar(250)	
Date	timestamp	

Table 3.3: Course

Name	Type	Extra
id	int(11)	AUTO_INCREMENT
Course title	varchar(250)	
Course code	varchar(250)	
Date	timestamp	

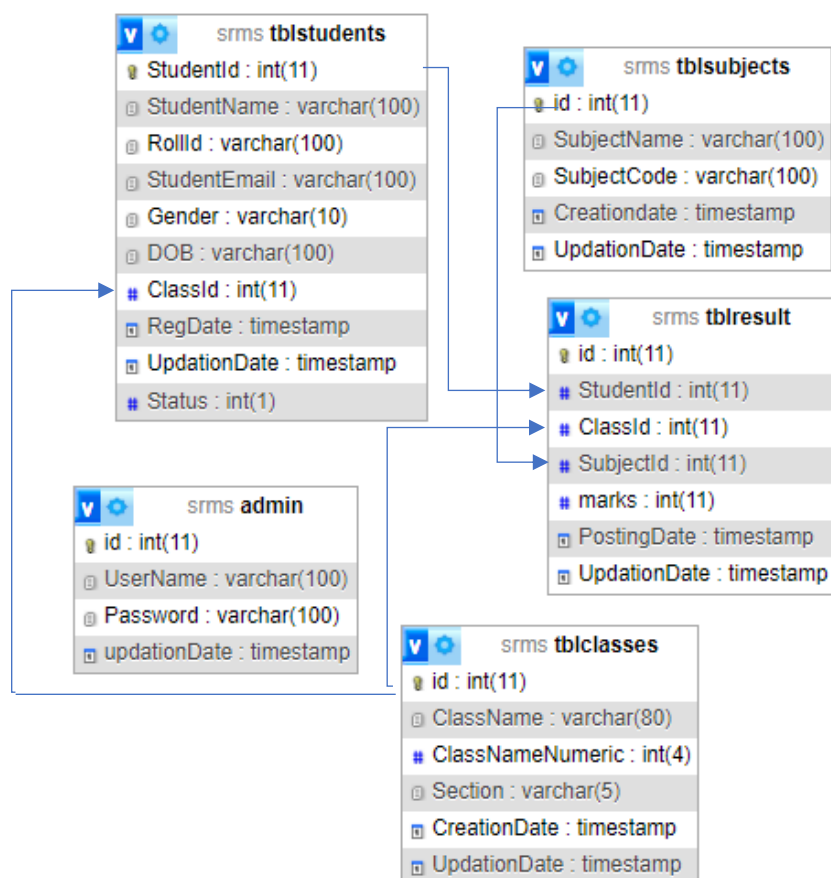
Table 3.4: Result

Name	Type	Extra
id	int(11)	AUTO_INCREMENT
Student_	varchar(250)	
class_id	varchar(250)	
course_id	varchar(250)	
Marks	varchar(250)	
Date	Time stamp	

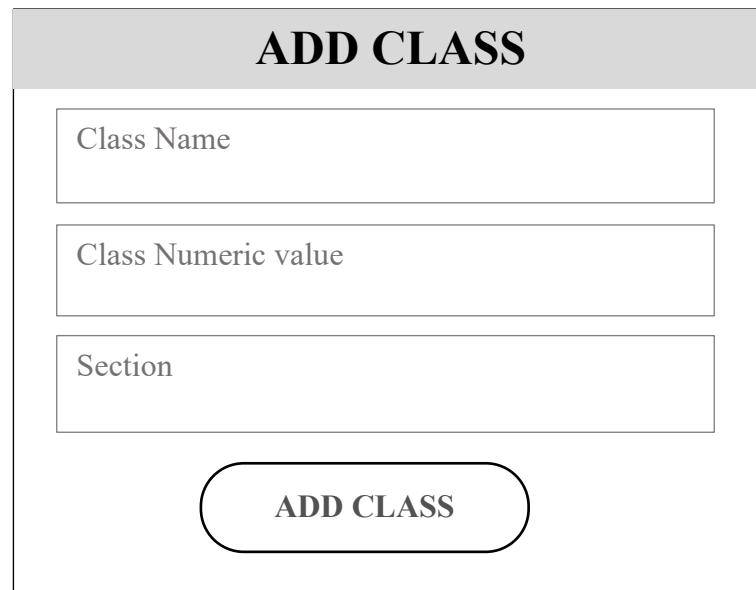
Table 3.5: Student Records

Name	Type	Extra
id	int(11)	AUTO_INCREMENT
Student name	varchar(250)	
Roll ID	varchar(250)	
Email	varchar(250)	
Gender	varchar(250)	
DOB	varchar(250)	
Class_id	varchar(250)	
Status	varchar(250)	
Date	Time stamp	

3.6.4 Database Entity Relationship Diagram

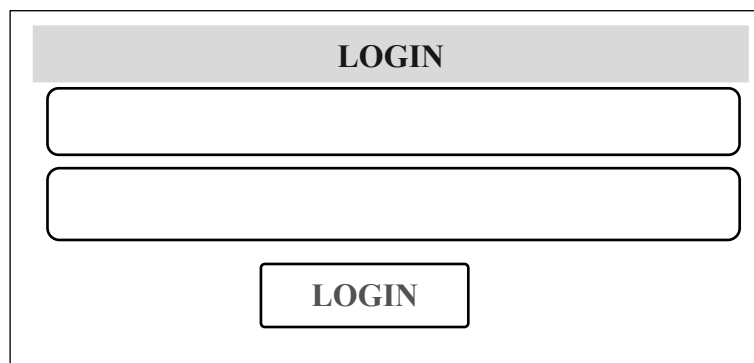
**Figure 3.4: Database Entity Relationship Diagram**

3.6.5 Input and Output Design



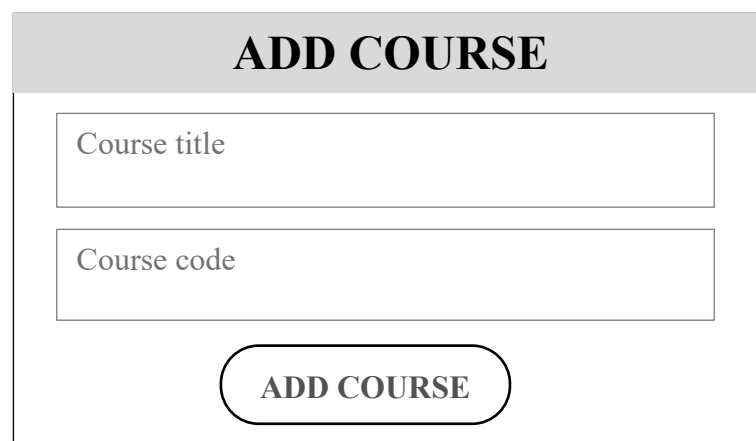
The 'ADD CLASS' form features a grey header bar with the title 'ADD CLASS' in bold. Below the header, there are three stacked rectangular input fields with labels 'Class Name', 'Class Numeric value', and 'Section'. At the bottom of the form is a rounded rectangular button labeled 'ADD CLASS'.

Figure 3.5: Add Class



The 'LOGIN' form has a grey header bar with the title 'LOGIN' in bold. It contains two stacked rectangular input fields. Below these fields is a rectangular button labeled 'LOGIN'.

Figure 3.6: Login form



The 'ADD COURSE' form has a grey header bar with the title 'ADD COURSE' in bold. It includes two stacked rectangular input fields labeled 'Course title' and 'Course code'. At the bottom is a rounded rectangular button labeled 'ADD COURSE'.

Figure 3.7: Add Course

ADD RESULT

Select Class

Select Student

Course 1

Course 2

Course 3

Marks

Marks

Marks

Course 4

Course 5

Course 6

Marks

Marks

Marks

Course 7

Course 8

Course 9

Marks

Marks

Marks

ADD AREA

Figure 3.8: Add Result

3.6.6 Report Layout

STUDENT RESULT DETAILS		
Student Name: ABBAS HARUNA		
Student Roll Id: 2025		
Student Class: First(C)		
#	Course	Marks
1	English	98
2	Music	88
3	Science	90
Total Marks		276 out of 300
Percentage		92 %

Figure 3.9: Student Result

3.7 System Requirement Specification

3.7.1 Hardware Requirements

The software to be design needs the following hardware for an effective operation of the newly designed system.

- i. A system running on intel, P(R) duo core with higher processor
- ii. The-Random Access Memory (RAM) should be at least 512MB.
- iii. At least 80-GB hard disk.
- iv. A monitor.

3.7.2 Software Requirements

The software requirements include:

- i. A window 7 or higher version of operating system.
- ii. XAMP or WAMP for Database
- iii. PHP
- iv. MySQL
- v. Browser

3.7.3 Personnel Requirement

Any computer literate who has a technical knowhow of internet surfing can use the system because it is user friendly.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Introduction

The new system is designed using PHP and MySQL programming language for easy records inserting and updating. This system will help in managing and easily retrieving of information from the system for management purposes.

4.2 Results

4.2.1 Home page interface

This is the first interface that users will encounter upon accessing the system. The Home Page Interface of an Online Result Processing System serves as the primary entry point for users, providing access to key functionalities and information. It is designed to be user-friendly, responsive, and intuitive, ensuring smooth navigation for students, staff, and administrators. Typically, it features the institution's name/logo, a brief description of the system, and navigation links or buttons to modules such as Login, Check Result, Register, and Help/Support.

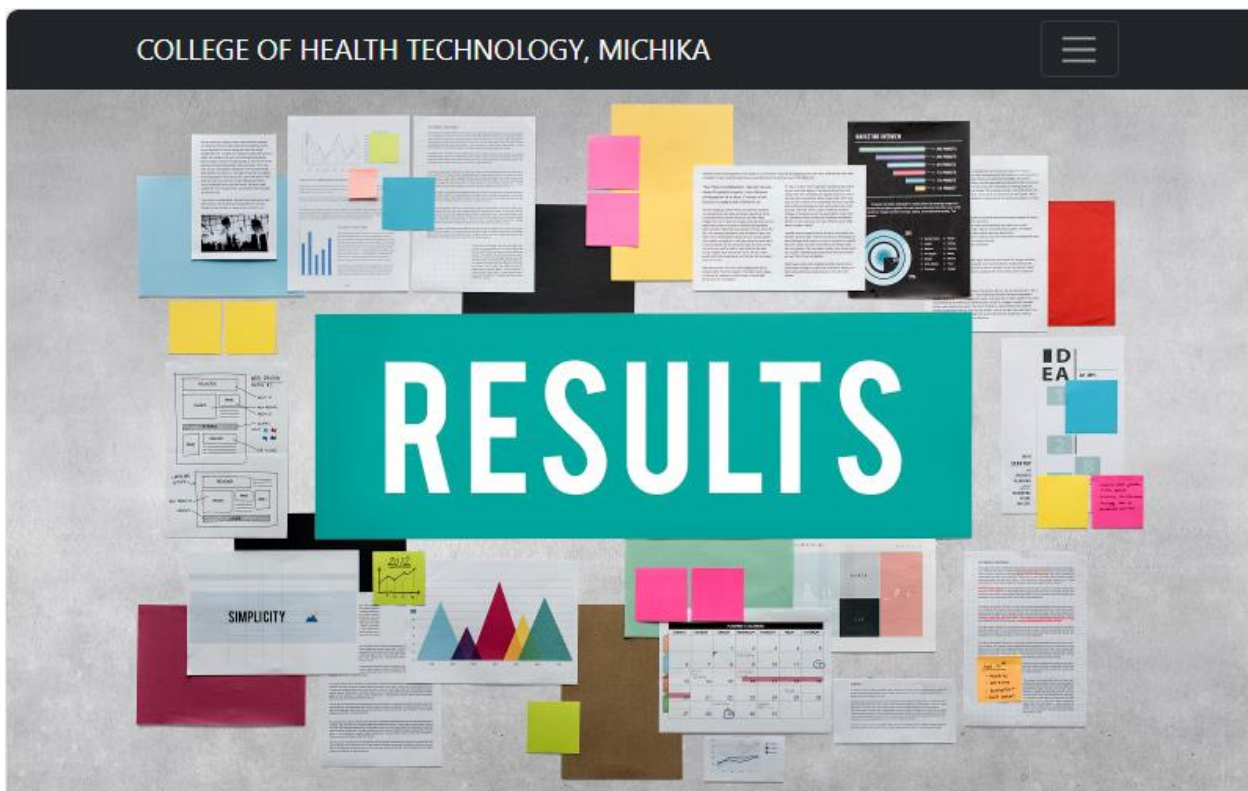
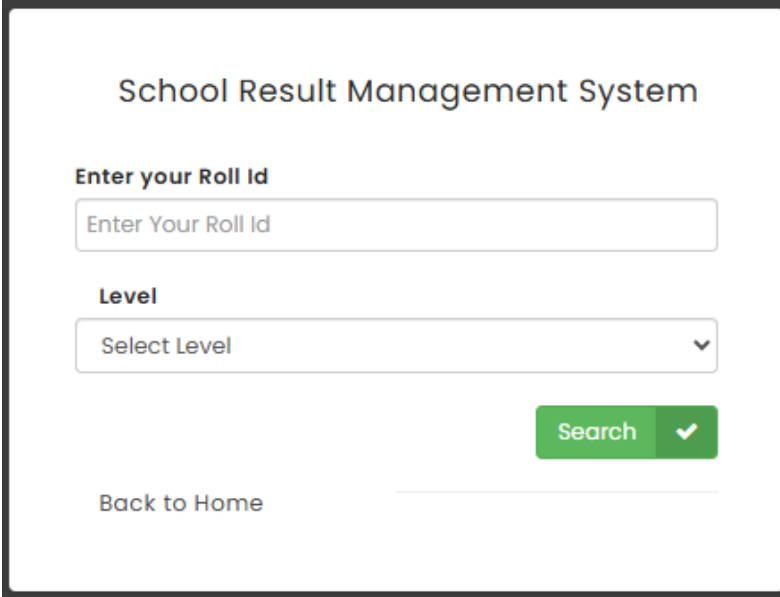


Figure 4.1: Welcome Interface

4.2.2 Login interface

This interface is designed for authentication purposes, allowing only authorized users, such as administrators or project coordinators, to gain access. Users must enter their credentials, such as

a username and password, to log into the system before performing any operations related to project allocation and verification.

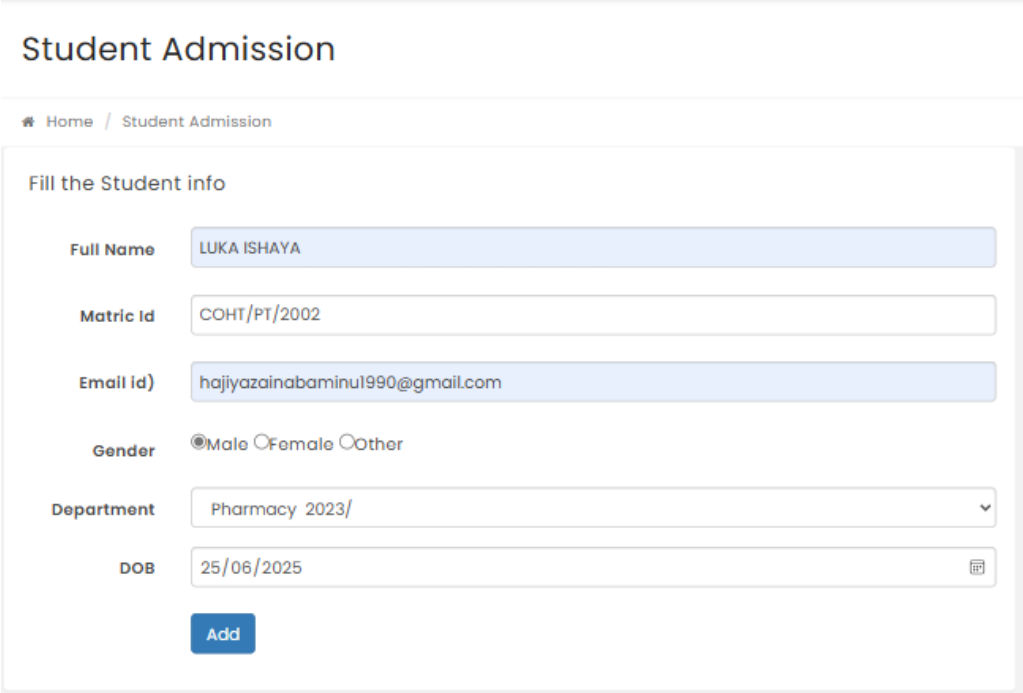


The image shows a login page for the 'School Result Management System'. It features a title at the top, followed by a label 'Enter your Roll Id' and a corresponding text input field. Below this is a 'Level' dropdown menu with 'Select Level' as the placeholder. To the right of these fields is a green 'Search' button with a checkmark icon. At the bottom left, there is a 'Back to Home' link.

Figure 4.2: Login page interface

4.2.3 Add Student Interface

This section enables the project coordinator or administrator to register new students into the system. The interface captures essential student details, such as name, matriculation number, department, and level, ensuring that only registered students can submit project topics for allocation and verification.



The image displays a 'Student Admission' form. At the top, there's a breadcrumb trail 'Home / Student Admission'. The main section is titled 'Fill the Student info' and contains several input fields: 'Full Name' (LUKA ISHAYA), 'Matric Id' (COHT/PT/2002), 'Email Id)' (hajiyazainabaminu1990@gmail.com), 'Gender' (radio buttons for Male, Female, Other), 'Department' (Pharmacy 2023/), and 'DOB' (25/06/2025). A blue 'Add' button is located at the bottom of the form.

Figure 4.3: Add Student interface

4.2.4 Add Department Interface

The Add Department Interface in a result processing system is designed to allow administrators or authorized users (such as ICT staff or academic officers) to input and manage departmental information within the platform.

Create Student Department

Home / Departments / Create Department

Create Student Department

Department Name

Community Health Extension Worker

MLT, CHEW, ENT, PT

Department Name in Numeric

100

100, 200, 300 etc

Session

2023/2024

Eg- 2024/2025, 2025/2026 etc

Submit ✓

Figure 4.4: Add Department

4.2.5 Registered Students Interface

This interface presents a comprehensive list of all students who have been successfully registered in the system. It allows the administrator to review student records, and query the database for information about a student.

View Students Info

Show 10 entries

Search:

#	Student Name	Roll Id	Class	Reg Date	Status	Action
1	Anuj kumar	10861	Fourth(C)	2024-04-24 11:30:57	Blocked	Edit View Result
2	amit kumar	2626	Sixth(B)	2024-04-22 11:30:57	Active	Edit View Result
3	rahul kumar	990	Seventh(B)	2024-04-24 11:30:57	Active	Edit View Result
4	sanjeev singh	122	Eight(A)	2024-04-25 11:30:57	Active	Edit View Result
5	Shiv Gupta	12345	Tenth(A)	2024-05-01 16:19:40	Active	Edit View Result
#	Student Name	Roll Id	Class	Reg Date	Status	Action

Showing 1 to 5 of 5 entries

Previous 1 Next

Figure 4.5: Registered Students Interface

4.2.6 Dashboard Interface

The dashboard interface shows all the activities and functions that the admin can perform on the system. It also shows a summary of all employees and promotion requests.

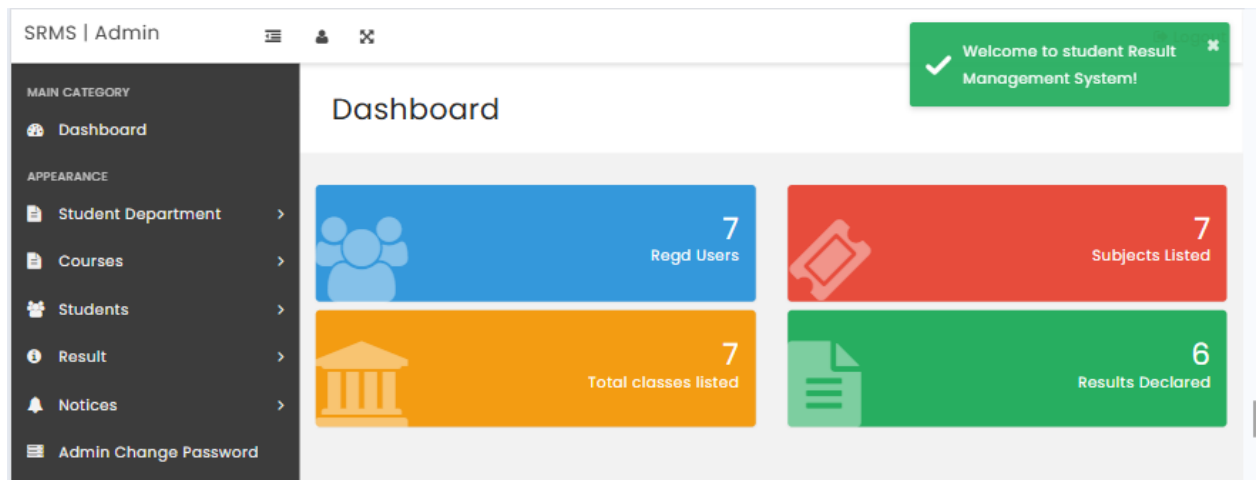


Figure 4.6: Dashboard interface

4.2.7 Result Processing Interface

The Result Processing Interface provides fields for entering scores for each student per course, automatically calculates grades based on predefined grading criteria, and stores the results securely in the system's database.

Declare Result

Home / Student Result

Class	Fourth Section-C
Student Name	Select Student
Subjects	English
	87
	Maths
	67
	Music
	87
	Science
	90

Declare Result

Figure 4.7: Result Processing interface

4.3 Discussion

The Result Processing System comprises several user-centered interfaces that facilitate smooth interaction between administrators, staff, and students. These interfaces are designed to ensure seamless navigation, accurate data entry, and effective result management across various academic departments. Each interface serves a specific function and contributes to the overall efficiency and usability of the system.

The Home Page Interface (Figure 4.1) is the initial point of interaction for all users accessing the system. It serves as the gateway to the platform, offering a visually appealing and informative landing page. This interface typically features the institution's name and logo, a brief system description, and navigation menus or buttons that link to core modules such as Login, Check Result, Register, and Help/Support. The design prioritizes user-friendliness and responsiveness to ensure accessibility across various devices, including smartphones and desktops. By providing a clear and intuitive layout, the home page improves user experience and serves as a central hub for accessing system functionalities.

The Login Interface (Figure 4.2) is a security feature that restricts system access to only authorized users. It prompts users—typically administrators, ICT staff, or academic officers—to enter valid usernames and passwords for authentication. Once logged in, users can access sensitive modules such as result entry, student data management, and departmental configuration. This interface helps maintain data privacy and system integrity by ensuring that only approved personnel can carry out administrative functions. It may also include features such as “Forgot Password” and login attempt tracking to enhance security.

The Add Student Interface (Figure 4.3) enables administrators to register new students into the system's database. This interface captures key student information, including full name, matriculation number, department, and level of study. Accurate student data is essential for effective result computation and departmental reporting. This interface often includes validation mechanisms to prevent duplicate entries and ensure the accuracy of the recorded data. By centralizing student registration, the system ensures that only recognized students can access academic records and results.

The Add Department Interface (Figure 4.4) allows system administrators to create and manage academic departments within the institution. This interface is particularly important for setting up the institutional structure, as it categorizes students and staff under appropriate academic units. Administrators can enter department names, assign them to corresponding faculties, and generate unique department codes. The interface helps streamline course assignments and result generation by ensuring that each student is associated with a specific academic department. This functionality

is critical in large institutions where multiple departments operate under different faculties or schools.

The Registered Students Interface (Figure 4.5) provides a centralized view of all students who have been successfully added to the system. It allows administrators to search, view, edit, or verify student information. This interface supports database queries that help in tracking student records, checking registration status, or resolving data inconsistencies. It serves as a monitoring and verification tool for academic officers and ensures that only properly registered students appear in departmental result processing workflows.

The Dashboard Interface (Figure 4.6) acts as the control center of the system for administrators. It provides a summarized overview of system activities, including the total number of registered students, departments, and recent actions performed within the platform. From the dashboard, users can quickly access essential modules such as result entry, department management, student records, and analytics. The interface is designed for efficiency, displaying key performance indicators (KPIs) and recent alerts that assist administrators in tracking system usage and performance. This feature enhances decision-making and task management within the system.

The Result Processing Interface is a critical module in the Online Result Processing System that enables authorized academic staff (such as course lecturers or exam officers) to input, compute, and manage student academic results. This interface provides fields for entering scores for each student per course, automatically calculates grades based on predefined grading criteria, and stores the results securely in the system's database. It also allows for editing or updating of results when necessary and ensures that only authorized users can access and modify the data. The interface is designed to be user-friendly, accurate, and efficient, reducing the time and errors associated with manual result computation, as shown in Figure 4.7.

Each of these interfaces plays a distinct role in achieving the core objectives of the Online Result Processing System, including accuracy, transparency, data integrity, and ease of use. Together, they create a robust framework for managing student academic records and improving the efficiency of result handling in higher education institutions.

4.4 User manual

4.4.1 System Installation

The user manual is a clear and precise instruction on how a user can operate the propose system, without any stress and successful. The following steps required

- i. Start or boot the computer form the hard disk
- ii. Double click on the folder that program is been stored in the desktop
- iii. Double click on the program and allow it to load gently

- iv. A security unit will display where the user will specify the user name and password the click on OK.
- v. A welcome menu will be displayed where the user has options to select which operation to be performed.
- vi. To find information about player, select any name and search.
- vii. Click on exist on the welcome screen to exist from the program.

4.4.2 System Operational Guide

The following are the necessary steps to take in order to use the system efficiently and effectively.

- i. Load the url of the system <https://localhost/srms/> the welcome page will be displayed.
- ii. Click on the **Proceed** button to proceed to the main system.
- iii. Provide the login details by entering your username and password.
- iv. The various task that you can perform on the portal will be displayed on the sidebar of the dashboard.

CHAPTER FIVE

SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Summary

This study was carried out to design and implement an Online Result Processing System for the College of Health Technology, Michika, with the aim of automating the manual result management processes within the institution. The traditional system of processing student results was characterized by challenges such as delays, inaccuracies, lack of confidentiality, and high potential for human error. These problems often affected the credibility and efficiency of the result management process and limited accessibility for both staff and students.

To address these issues, a computerized result processing system was proposed and developed. The system provides functionalities for secure login, result entry by authorized personnel, automatic grade computation, result storage in a centralized database, and student access to view their results online. The design and development followed standard software engineering principles, incorporating user-friendly interfaces and a relational database structure to ensure data integrity, security, and scalability.

The system was tested and evaluated, and the results showed that it effectively reduces time spent on result compilation, minimizes errors, ensures confidentiality, and improves accessibility. This automation has the potential to increase staff productivity and student satisfaction significantly.

5.2 Conclusion

In conclusion, the implementation of the Online Result Processing System for the College of Health Technology, Michika, marks a significant improvement in academic result management. The new system replaces the error-prone manual process with an efficient, accurate, and user-friendly digital platform. It provides secure and timely access to academic records, minimizes administrative burden, and enhances the integrity and transparency of the result processing procedure.

The success of this system also highlights the importance of adopting modern information and communication technology (ICT) solutions in higher education institutions, especially in a time when automation and digitalization are crucial to institutional development. With proper training and continuous maintenance, the system can serve as a long-term solution to academic record-keeping challenges in the college and could be replicated in other similar institutions.

5.3 Recommendations

Based on the findings and successful implementation of this project, the following recommendations are made:

- i. The management of the College of Health Technology, Michika, should fully adopt and integrate the online result processing system across all departments for consistent and effective result management.
- ii. Continuous monitoring, testing, and maintenance should be carried out to ensure the system remains secure, reliable, and functional as academic requirements evolve.
- iii. Staff and students should undergo adequate training to familiarize themselves with the functionalities of the system to ensure its effective utilization.
- iv. A routine backup mechanism should be implemented to prevent data loss and allow for quick recovery in case of system failure or cyber threats.
- v. The system should be designed to accommodate future upgrades, such as integration with student registration, transcript generation, or mobile app access, to enhance its overall functionality.

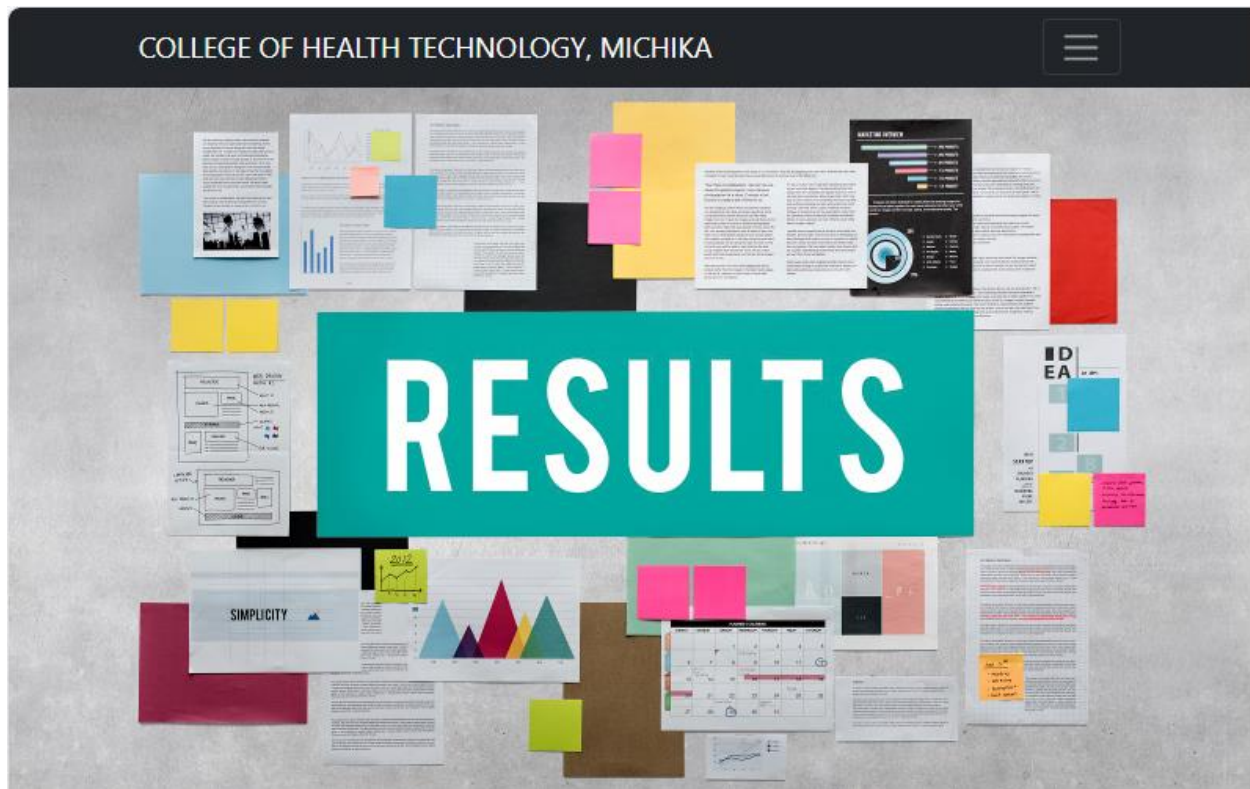
REFERENCES

- Almorsy, M., Grundy, J., & Müller, I. (2016). *An analysis of the cloud computing security problem*. Future Generation Computer Systems, 62, 98–116.
- Anderson, P., & White, L. (2023). *Disaster recovery and data protection in educational cloud systems*. Journal of Cloud Computing Research, 15(2), 101–118.
- Davis, F. D. (1989). *Perceived usefulness, perceived ease of use, and user acceptance of information technology*. MIS Quarterly, 13(3), 319–340. <https://doi.org/10.2307/249008>
- El-Gazzar, R., Hustad, E., & Olsen, D. H. (2020). *Role-based access control for academic result processing in cloud computing*. International Journal of Cloud Applications and Computing, 10(3), 45–59. <https://doi.org/10.4018/IJCAC.2020070103>
- Hashem, I. A. T., Yaqoob, I., Anuar, N. B., Mokhtar, S., Gani, A., & Khan, S. U. (2015). *The rise of “big data” on cloud computing: Review and open research issues*. Information Systems, 47, 98–115. <https://doi.org/10.1016/j.is.2014.07.006>
- Hughes, R. (2019). *Cloud computing and cost efficiency in higher education institutions*. Journal of Educational Technology Systems, 48(4), 555–570.
- Johnson, A., Peters, M., & Liu, S. (2021). *Blockchain-based frameworks for secure academic result processing*. Journal of Educational Technology Systems, 50(1), 60–78.
- Johnson, T., & Brown, G. (2022). *Design and implementation of secure result processing systems in the cloud*. International Journal of Information Security in Education, 12(1), 88–102.
- Kim, S., & Lee, J. (2021). *Data encryption strategies in cloud-based educational platforms*. International Journal of Computer Security, 29(3), 345–362. <https://doi.org/10.1016/j.ijcs.2021.01.012>
- Kumar, R., & Lee, J. (2020). *Cloud computing in education: A comparative study of cloud service providers*. Education and Information Technologies, 25(4), 2719–2743.
- Mell, P., & Grance, T. (2011). *The NIST definition of cloud computing* (Special Publication 800-145). National Institute of Standards and Technology.
- Miller, H., & Evans, D. (2022). *Cost-effective strategies for cloud adoption in academic institutions*. Journal of Cloud Economics, 10(2), 133–147.
- Patel, V., Sharma, P., & Roy, D. (2022). *Integrating artificial intelligence in cloud-based academic result processing systems*. International Journal of Educational Technology in Higher Education, 19(1), 89–104. <https://doi.org/10.1186/s41239-022-00356-4>
- Patil, R., Deshmukh, S., & Roy, P. (2024). *Multi-factor authentication techniques in academic cloud systems*. Journal of Information Assurance, 14(1), 22–37.
- Rahman, F., Adekunle, T., & Musa, L. (2023). *Enhancing accessibility and inclusivity in academic result systems through cloud computing*. Journal of Cloud Technology in Education, 18(2), 33–47.

- Rogers, E. M. (1995). *Diffusion of innovations* (4th ed.). New York: Free Press.
- Smith, H., & Brown, G. (2022). *Securing academic records with cloud-based result processing systems*. *International Journal of Academic Research in Education*, 11(1), 45–62.
- Smith, J. (2023). *Cybersecurity challenges in cloud-based education systems*. *Cybersecurity in Education Journal*, 8(1), 12–28.
- Williams, R., Thomas, K., & Adebayo, J. (2021). *Cloud computing for academic record management: Opportunities and challenges*. *Journal of Information Systems in Education*, 28(3), 211–228. <https://doi.org/10.1007/s42438-021-00219-2>
- Zhang, Y., Lin, X., & Wang, Q. (2020). *Scalability and performance analysis of cloud-based educational systems*. *Journal of Cloud Computing: Advances, Systems and Applications*, 9(1), 1–16. <https://doi.org/10.1186/s13677-020-00177-1>

APPENDIX A

Home page interface



Login interface

The image displays a login interface for a "School Result Management System". The interface is contained within a white rectangular box with a dark border. At the top, the title "School Result Management System" is centered. Below the title, there is a section for user authentication. It starts with the label "Enter your Roll Id" followed by a text input field containing the placeholder text "Enter Your Roll Id". Below this is a label "Level" followed by a dropdown menu with the text "Select Level" and a downward arrow icon. To the right of these fields is a green button with the text "Search" and a white checkmark icon. At the bottom left of the form, there is a link labeled "Back to Home" with a horizontal line underneath it.

Add Student Interface

Student Admission

Home / Student Admission

Fill the Student info

Full Name

Matric id

Email id)

Gender ☒ Male ☐ Female ☐ Other

Department

DOB

Add Department Interface

Create Student Department

Home / Departments / Create Department

Create Student Department

Department Name

MLT, CHEW, ENT, PT

Department Name in Numeric

100, 200, 300 etc

Session

Eg- 2024/2025, 2025/2026 etc

☒

Registered Students Interface

View Students Info

Show entries

Search:

#	Student Name	Roll Id	Class	Reg Date	Status	Action
1	Anuj kumar	10861	Fourth(C)	2024-04-24 11:30:57	Blocked	Edit View Result
2	amit kumar	2626	Sixth(B)	2024-04-22 11:30:57	Active	Edit View Result
3	rahul kumar	990	Seventh(B)	2024-04-24 11:30:57	Active	Edit View Result
4	sanjeev singh	122	Eight(A)	2024-04-25 11:30:57	Active	Edit View Result
5	Shiv Gupta	12345	Tenth(A)	2024-05-01 16:19:40	Active	Edit View Result
#	Student Name	Roll Id	Class	Reg Date	Status	Action

Showing 1 to 5 of 5 entries

Previous **1** Next

Dashboard Interface

SRMS | Admin

Dashboard

Welcome to student Result Management System!

MAIN CATEGORY

- Dashboard

APPEARANCE

- Student Department
- Courses
- Students
- Result
- Notices
- Admin Change Password

7 Regd Users

7 Subjects Listed

7 Total classes listed

6 Results Declared

Result Processing Interface

Declare Result

Home / Student Result

Class:

Student Name:

Subjects: English

87

Maths

67

Music

87

Science

90

[Declare Result](#)

APPENDIX B

PROGRAM CODE

```
<?php
error_reporting(0);
include('includes/config.php');
?>
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no" />
    <meta name="description" content="" />
    <meta name="author" content="" />
    <title>Student Result Management System</title>
    <!-- Favicon-->
    <link rel="icon" type="image/x-icon" href="assets/favicon.ico" />
    <!-- Core theme CSS (includes Bootstrap)-->
    <link href="css/styles.css" rel="stylesheet" />
  </head>
  <body>
    <!-- Responsive navbar-->
    <nav class="navbar navbar-expand-lg navbar-dark bg-dark">
      <div class="container">
        <a class="navbar-brand" href="index.php">COLLEGE OF HEALTH
TECHNOLOGY, MICHKA</a>
        <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-
expanded="false" aria-label="Toggle navigation"><span class="navbar-toggler-
icon"></span></button>
        <div class="collapse navbar-collapse" id="navbarSupportedContent">
          <ul class="navbar-nav ms-auto mb-2 mb-lg-0">
            <li class="nav-item"><a class="nav-link active" aria-current="page"
href="#">Home</a></li>
            <li class="nav-item"><a class="nav-link active" href="find-
result.php">Students</a></li>
            <li class="nav-item"><a class="nav-link active" href="admin-
login.php">Admin</a></li>
          </ul>
        </div>
      </div>
    </nav>
    <!-- Header - set the background image for the header in the line below-->
    <header class="py-5 bg-image-full" style="background-image: url('images/background-
image.jpg')">

    </header>
    <!-- Content section-->
```

```

<section class="py-5">
  <div class="container my-5">
    <div class="row justify-content-center">
      <div class="col-lg-6">
        <h2>Notice Board</h2>
        <hr color="#000" />
        <marquee direction="up" onmouseover="this.stop();"
onmouseout="this.start();">
          <ul>
            <?php $sql = "SELECT * from tblnotice";
$query = $dbh->prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$cnt=1;
if($query->rowCount() > 0)
{
foreach($results as $result)
{ ?>
<li><a href="notice-details.php?nid=?php echo htmlentities($result->id);?>"
target="_blank"><?php echo htmlentities($result->noticeTitle);?></li>
<?php }} ?>

          </ul>
        </marquee>

      </div>
    </div>
  </div>
</section>

<!-- Footer-->
<footer class="py-5 bg-dark">
  <div class="container"><p class="m-0 text-center text-white"> Student Result
Management System</p></div>
</footer>
<!-- Bootstrap core JS-->
<script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.1.3/dist/js/bootstrap.bundle.min.js"></script>
<!-- Core theme JS-->
<script src="js/scripts.js"></script>
</body>
</html>

<?php
$sql1 ="SELECT StudentId from tblstudents ";
$query1 = $dbh -> prepare($sql1);
$query1->execute();
$results1=$query1->fetchAll(PDO::FETCH_OBJ);

```

```

$totalstudents=$query1->rowCount();
?>

        <span class="number counter"><?php echo
htmlentities($totalstudents);?></span>
        <span class="name">Regd Users</span>
        <span class="bg-icon"><i class="fa fa-users"></i></span>
    </a>
    <!-- /.dashboard-stat -->
</div>
<!-- /.col-lg-3 col-md-3 col-sm-6 col-xs-12 -->

<div class="col-lg-6 col-md-6 col-sm-6 col-xs-12" >
    <a class="dashboard-stat bg-danger" href="manage-subjects.php">
<?php
$sql ="SELECT id from tblsubjects ";
$query = $dbh -> prepare($sql);
$query->execute();
$results=$query->fetchAll(PDO::FETCH_OBJ);
$totalsubjects=$query->rowCount();
?>

        <span class="number counter"><?php echo
htmlentities($totalsubjects);?></span>
        <span class="name">Subjects Listed</span>
        <span class="bg-icon"><i class="fa fa-ticket"></i></span>
    </a>
    <!-- /.dashboard-stat -->
</div>
<!-- /.col-lg-3 col-md-3 col-sm-6 col-xs-12 -->

<div class="col-lg-6 col-md-6 col-sm-6 col-xs-12" style="margin-
top:1%;">
    <a class="dashboard-stat bg-warning" href="manage-classes.php">
    <?php
$sql2 ="SELECT id from tblclasses ";
$query2 = $dbh -> prepare($sql2);
$query2->execute();
$results2=$query2->fetchAll(PDO::FETCH_OBJ);
$totalclasses=$query2->rowCount();
?>

        <span class="number counter"><?php echo
htmlentities($totalclasses);?></span>
        <span class="name">Total classes listed</span>
        <span class="bg-icon"><i class="fa fa-bank"></i></span>
    </a>
    <!-- /.dashboard-stat -->
</div>
<!-- /.col-lg-3 col-md-3 col-sm-6 col-xs-12 -->

```

```

        <div class="col-lg-6 col-md-6 col-sm-6 col-xs-12" style="margin-
top:1%">
            <a class="dashboard-stat bg-success" href="manage-results.php">
                <?php
$sql3="SELECT distinct StudentId from tblresult ";
$query3 = $dbh -> prepare($sql3);
$query3->execute();
$results3=$query3->fetchAll(PDO::FETCH_OBJ);
$totalresults=$query3->rowCount();
?>

                <span class="number counter"><?php echo
htmlentities($totalresults);?></span>
                <span class="name">Results Declared</span>
                <span class="bg-icon"><i class="fa fa-file-text"></i></span>
            </a>
        <!-- /.dashboard-stat -->
    </div>
    <!-- /.row -->
</div>
<!-- /.container-fluid -->
</section>
<!-- /.section -->

</div>
<!-- /.main-page -->

</div>
<!-- /.content-container -->
</div>
<!-- /.content-wrapper -->

</div>
<!-- /.main-wrapper -->
    });
</script>
</body>
</html>
<?php } ?>

```