DESIGN AND IMPLEMENTATION OF ONLINE STUDENT CLEARANCE SYSTEM

By

OBIKOYA, TIOLUWANI GENESIS

19010301039

A PROJECT SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE AND MATHEMATICS, COLLEGE OF BASIC AND APPLIED SCIENCES, IN PARTIAL FULLFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE

2023

DECLARATION

I hereby declare that this project has been written by me and is a record of my own research work. It has not been presented in any previous application for a higher degree of this or any other University. All citations and sources of information are clearly acknowledged by means of reference.

OBIKOYA TIOLUWANI GENESIS

Date

CERTIFICATION

This is to certify that the content of this project entitled ‘Design And Implementation of On-line clearance System’ was prepared and submitted by OBIKOYA TIOLUWANI GENESIS in partial fulfilment of the requirements for the degree of BACHELOR OF SCIENCE IN COMPUTER SCIENCE. The original research work was carried out by her under by supervision and is hereby accepted.

_______________________________ (Signature and Date)

Dr. O.D Adekola

Supervisor

_______________________________ (Signature and Date)

Dr. O.O Odim

Coordinator, Department of Computer Science and Mathematics

DEDICATION

I would like to dedicate this project to God Almighty, for being faithful and merciful, for seeing me through to the end of this project. I also dedicate this work to my father, Mr. OBIKOYA and my mother, Mrs. OBIKOYA for being a major source of support in every way. I finally dedicate this work to Obikoya Tomiwa who has been supportive, encouraging and present during the whole process of delivering this work

ACKNOWLEDGEMENTS

My sincere gratitude goes to the God who created all things and manifests himself in diverse ways than we can comprehend for his mercy, loving-kindness and presence in the times I was in need. I also appreciate the entire staff and management of Mountain Top University for the immeasurable impact they have had on my life academically and spiritually from the person of the chancellor, Dr. D.K Olukoya down to every member of the senate and the university scholarship board,who have made special contributions to the success of my academic pursuit, as well my dear friend Nwochie David. Finally, I duly appreciate my siblings, friends and colleagues without whom this journey could not have been a success. May the God of heaven water your lives.

ABSTRACT

This project focuses on the development and implementation of an online clearance system for university students. The goal of the system is to streamline the clearance process and enhance efficiency, convenience, and accuracy. Through the system, students can complete their clearance tasks on-line, eliminating the need for physical visits and reducing waiting times. The system provides features such as electronic form submission, real-time tracking of clearance progress, and accurate record-keeping. Feedback from users indicates high levels of satisfaction, highlighting the system's convenience and user-friendly interface. However, challenges such as accessibility and continuous system improvement have been identified. Recommendations are proposed to address these challenges and further enhance the system. Overall, the project demonstrates the benefits of implementing an online clearance system, improving the student experience and administrative processes within the university.

TABLE OF CONTENT

CHAPTER ONE

INTRODUCTION

1.1 Background of study

A clearance is a certificate that grants authorization to leave an institution. The word "clearance" is also used in private firms that have a formal process in place to evaluate workers for access to sensitive information. A clearance alone is usually insufficient to acquire access; the organization must also assess that the cleared individual has a "need to know" the information. No one is anticipated to be permitted access to sensitive information solely based on rank or position; nevertheless, if a clearance is obtained, access to certain information or freedom will be granted. Students in their final year who have fulfilled the academic requirements for graduation must go through a clearance process before leaving the University (Brown, L., & Davis, M. w2020).

Graduate student clearance systems are important incessant processes and processes that are unavoidable in any university. The clearance must be carried out in order to verify if the pupil is suitable for jobs and not to own anything at the school. At the conclusion of the last exam in the classroom, where the student did not have any carryover left, it is university tradition to execute these procedures. As people of this generation become more dependent on the internet for results, the need for automatic clearance is becoming more apparent.

In university like Mountain Top, there is an increasing requirement for an automated way of data storage, as well as an online clearance system. This would significantly reduce the many challenges and stresses involved with the manual technique of clearing. Furthermore, the issue of youth service being delayed due to mobility in order to complete the time-consuming manual clearance process will be addressed, the current clearance system is manual and time-consuming.

Moreover, an online clearance system enhances accuracy and data integrity. Manual clearance processes are susceptible to errors in recording and processing information, leading to confusion and delays. Conversely, an automated system minimizes the risk of human error and ensures precise record-keeping(Anderson, K., & White, B. 2021). It can also generate reports and provide analytic on clearance trends, enabling university administrators to identify and address any bottlenecks or inefficiencies within the process.

From a student perspective, an online clearance system offers numerous advantages. It grants students the flexibility to complete clearance tasks at their own pace and from any location with internet access. This flexibility is particularly beneficial for students juggling multiple commitments or facing limitations in visiting the university campus regularly. Additionally, the user-friendly interface and clear instructions of the system facilitate easy navigation and completion of the necessary steps, reducing confusion and alleviating stress.

Furthermore, an online clearance system can positively impact other areas, such as youth service programs. Manual clearance processes often consume significant time, potentially causing delays for students needing to complete their clearance before engaging in youth service. The implementation of an automated clearance system can help mitigate this issue by expediting the process and ensuring timely fulfilment of requirements.The adoption of an online clearance system within universities brings significant advantages for both students and administrative staff. It enhances efficiency, accuracy, and convenience while reducing administrative burdens and student stress (Brown, L., & Davis, M. 2020). By embracing technology, universities can streamline clearance processes, enhance the student experience, and adapt to the evolving needs of the digital age.

1.2 Statement of the problem

The university's current final clearance mechanism is entirely manual.This makes the system time-consuming and exhausting. In this situation, students will need to visit all clearing offices and sign a document. When signed, the document certifies that the student has been cleared. Mountain Top University students must be cleared in their numerous disciplines and information units following graduation. Among which are:

i. Chaplaincy unit Clearance

ii. Bursary Clearance

iii. ICT unit Clearance

iv. ESM clearance

v. Clearance Student

vi. Library Clearance

vii. Departmental Due

Furthermore, to clear a graduating student from all of these departments, it usually takes a long period and several processes, as well as a delay in clearing the student for youth service and collecting the statement of results. As a result, an online clearing system based on computer software became necessary to address the shortcomings of the existing manual approach.

1.3 Aim and objectives of the study

The aim of this study is to design and implement an online clearance system an efficient web-based platform that is reliable and cost-effective, which can manage the process efficiently and the objectives are:

i. Identify the requirements for the online clearance system

ii. Specify the design of the app based on 1

iii. Implement and test the app

1.4 Methodology of the study

In order to achieve the aforementioned objectives, the following methods were employed:

a. Conducting a comprehensive literature review to explore and gain insights into existing Identity Management Systems utilized for monitoring hostel residences through QR codes.

b. Identifying the user and system requirements of the proposed system by conducting informal interviews with system users.

c. Specifying the system design through the utilization of UML diagrams, including use case diagrams, sequence diagrams, class diagrams, among others.

d. Implementing the database utilizing MySQL, which serves as the underlying technology for storing and managing data.

e. Developing the front-end of the system using HTML, CSS, and PHP, while the back-end was implemented usingMySQL, which facilitated the server-side functionality and data processing.

By utilizing these methods, the study aimed to comprehensively investigate existing systems, gather requirements, design the system, implement the database, and develop both the front-end and back-end components of the proposed system.

1.5 Justification of the study

The implementation of an online clearance system in universities is an essential solution to address the challenges and inefficiencies of the current manual system. The project work aims to alleviate the queuing system at the university by providing students with the convenience of completing necessary clearances online, eliminating the need to physically visit various offices. This online clearance system offers numerous advantages and benefits.One significant advantage is the reduction in time and resources required for the clearance process, leading to enhanced efficiency and accuracy. By streamlining the clearance procedures, the online system enables students to achieve their clearance objectives promptly. Moreover, the system contributes to the overall satisfaction of students, faculty, and staff, ensuring timely graduation and seamless transfer processes.

The internet-based nature of the online clearance system offers several clear advantages over traditional manual systems. Notably, it provides higher yields and efficiency in information processing. Users can conveniently check their clearance status, identify any outstanding obligations to the university, and complete and submit clearance forms from the comfort of their bedrooms, offices, or any location worldwide. The system facilitates swift information processing, reducing delays and ensuring a smoother clearance experience. Additionally, the online system is cost-effective for both students and school management, minimizing expenses related to labor and stationary.The implementation of an online clearance system in universities brings numerous benefits, including time savings, convenience, fast information processing, and cost reduction. By embracing this technology, universities can enhance efficiency, accuracy, and overall user satisfaction, while alleviating the queuing system and challenges associated with manual clearances.

1.6 Scope and Limitation of the Study

This project work is to develop a computer software based online clearance system for doing clearance for graduating students of Mountain Top University. The website will be develop using HTML,CSS,My SQL.The system will be designed to be user-friendly, efficient, and accurate, and will include features such as online forms, digital signatures, and real-time updates of student records. Additionally, the system will be integrated with the university's existing databases to ensure that all information is accurate and up-to-date.

1.7 Limitation

This project covers some of the aspect of a computer software based online clearance system using Mountain Top University as case study. However the following are the constraints:

a) TIME CONSTRAINTS: Due to time constrain the web-page covers only clearance for various departments by the graduating student.

b) FINANCIAL CONSTRAINTS: Due financial constraints people cannot afford this kind of process online especially towards the cost of accessing the internet. Therefore, it would cost a lot to develop a full web-based clearance system. Some documents were considered confidential and were not made available.

1.8 Definition of Terms

a. **MySQL** MySQL is an open-source relational database management system (RDBMS) that allows for the storage, management, and retrieval of structured data. It is widely used in web applications and serves as a core component of the LAMP (Linux, Apache, MySQL, PHP) stack. MySQL offers a robust and scalable platform for managing databases, enabling users to create, modify, and query data using a structured query language known as SQL (Structured Query Language).

b. **CSS (Cascading Style Sheet)** CSS (Cascading Style Sheets) is a style sheet language used for describing the visual presentation and formatting of HTML (Hypertext Mark-up Language) and XML (Extensible Mark-up Language) documents. It allows web developers to control the appearance and layout of web pages, including elements such as fonts, colours, spacing, and positioning. By separating the style and presentation from the content of a web page, CSS enables consistent design across multiple pages and simplifies the maintenance and updating process. CSS works in conjunction with HTML to enhance the visual appeal and user experience of websites, making it an essential tool in modern web development.

c. **HTML (Hypertext Mark-up Language)** is a standard mark-up language used for creating and structuring web pages. It provides a set of tags or elements that define the structure and content of a web page.

d. **PHP** PHP (Hypertext Preprocessor) is a popular server-side scripting language used for web development. It is embedded within HTML code and executed on the server before the web page is sent to the user's browser.

e. **System** A system refers to a collection of interconnected components or elements that work together to achieve a specific purpose or objective. It can be thought of as a structured and organized arrangement of parts that function as a unified whole.

f. **Student** A student is an individual who is actively engaged in learning and pursuing formal education at an educational institution. Students are typically enrolled in schools, colleges, universities, or other educational programs where they acquire knowledge, skills, and qualifications in various fields of study.

CHAPTER TWO

LITERATURE REVIEW

2.1 Clearance System

According to Smith (2015), a clearance system in universities involves confirming the completion of coursework, returning library materials, settling financial obligations, and obtaining necessary signatures from teachers or administrators. It serves as a final check to ensure that students have successfully met the requirements for their academic progression. Its main objective is to ensure that only authorized personnel or entities are allowed entry into designated locations, access sensitive data, or acquire confidential information. The clearance process entails a comprehensive background investigation and evaluation of an individual's qualifications, trustworthiness, and potential risks(Oluwatayo, A. A., & Oni, A. A. 2016). This investigation typically includes personal interviews, reference checks, criminal history checks, financial background checks, as well as assessments of loyalty and reliability.

Within the educational context, a student clearance system refers to the procedure of verifying that students have fulfilled all necessary requirements and have no outstanding obligations before they can graduate, transfer to another school, or progress to the next academic level(Kim, M. H., Park, Y. S., & Kim, J. (2016). This process involves confirming the completion of coursework, returning borrowed library books, settling financial obligations, and obtaining the required signatures from teachers or administrators.

Clearance, in a broader sense, refers to the official authorization or permission granted to an individual, organization, or item to access or possess specific information, resources, areas, or privileges. It involves a systematic evaluation of an entity's qualifications, trustworthiness, and eligibility, which ultimately determines the level of access or privileges they are granted. This evaluation is critical to maintaining security, confidentiality, and controlled access to valuable resources and information.

Overall, clearance systems play a crucial role in organizations and governments by ensuring that only authorized individuals or entities can gain access to restricted areas, sensitive data, or confidential information. By implementing effective clearance processes, organizations can enhance security, protect valuable resources, and maintain the integrity of their operations.

A clearance system is a structured process or set of procedures implemented within an organization or government to grant permission or authorization for individuals, goods, or information to access specific restricted areas, resources, or classified materials (Anigbogu, 2000).

The clearance process in universities is designed to facilitate a smooth transition for students as they move on to the next phase of their educational journey. It helps to ensure that students have met the necessary criteria and have no outstanding obligations that could hinder their progress. By implementing a clearance system, universities can maintain academic standards, administrative efficiency, and provide students with a clear pathway for their educational pursuits.

The clearance system in universities is vital for various reasons. It ensures that students have completed their required coursework, demonstrating their mastery of the subject matter. It also helps to manage resources, such as library materials, by ensuring their return and availability for other students. Additionally, the financial obligations settlement ensures that students have fulfilled their financial responsibilities to the institution.

By adhering to the clearance process, universities can effectively track student progress, maintain accurate records, and facilitate efficient administrative procedures. It also helps to ensure that students receive the necessary documentation and official recognition for their achievements.

In summary, a clearance system in universities serves as a structured process to verify that students have met all necessary requirements and have no outstanding obligations before they can advance in their academic journey. It plays a critical role in maintaining academic standards, administrative efficiency, and providing students with a clear pathway for their educational pursuits.

2.1.1 History of clearance system

The history of the university clearance system dates back to the early establishment of universities as centers of higher education. Over time, universities recognized the need for a structured process to ensure that students have met all necessary requirements before they can graduate, transfer to another institution, or progress to the next academic level.

In the early years of universities, clearance processes were often informal and decentralized. Professors or administrators would individually assess students' progress and provide recommendations for graduation or advancement. However, as universities grew in size and complexity, a more systematic approach became necessary to ensure consistency, accuracy, and fairness in evaluating student achievements.

The development of formal clearance systems can be traced to the evolution of administrative structures within universities. As universities became more bureaucratic in the late 19th and early 20th centuries, the need for standardized procedures and documentation became evident. These systems aimed to streamline the evaluation of student records, confirm the completion of coursework, and ensure compliance with academic regulations.

Advancements in technology also played a significant role in shaping the history of university clearance systems. The introduction of computerized record-keeping and database management systems in the latter half of the 20th century revolutionized the clearance process. Digital platforms allowed for more efficient tracking of student progress, automated notifications, and streamlined administrative work-flows.

Today, university clearance systems have evolved into comprehensive processes that encompass various aspects of a student's academic journey. These systems verify the completion of coursework, the fulfilment of financial obligations, the return of library materials, and the acquisition of necessary signatures from teachers or administrators. They ensure that students have met the necessary requirements and have no outstanding obligations that could hinder their academic progression.

The history of the university clearance system reflects the evolution of administrative practices and the need for standardized processes to assess student achievements. From informal evaluations to formalized procedures, the clearance system has become an integral part of universities, providing a structured framework to confirm student readiness for graduation, transfer, or academic advancement.

2.1.2 Evolution of online clearance system

The evolution of the university clearance system has been shaped by the changing needs of educational institutions and advancements in technology. Over the years, universities have recognized the importance of establishing efficient and standardized processes to ensure that students meet specific requirements before graduating, transferring to another institution, or advancing academically.

In the early stages, university clearance systems were primarily manual and decentralized. Faculty members and administrators individually assessed students' progress and qualifications, relying on personal interactions and paper-based documentation. However, as universities grew in size and student enrollment increased, there arose a need for more streamlined and scalable approaches.

The introduction of computer technology in the latter half of the 20th century marked a significant turning point in the evolution of university clearance systems. Computerized record-keeping and database management systems revolutionized the storage and processing of student information. These advancements enabled more accurate and efficient clearance processes, reducing the likelihood of errors and alleviating administrative burdens.

As technology continued to advance, universities embraced integrated student information systems that automated various aspects of the clearance process. These systems provided a centralized platform for managing student records, tracking academic progress, and facilitating communication between different departments involved in the clearance process.

Moreover, the advent of web-based applications and online portals further transformed university clearance systems. Students gained the ability to access their clearance status, submit required documentation, and track their progress remotely. This shift to online platforms not only enhanced convenience for students but also improved efficiency and reduced paperwork for university administrators.

In recent years, universities have continued to embrace technological advancements to further enhance the clearance process. Mobile applications, electronic signatures, and data analytics have been integrated to expedite clearance evaluations and provide real-time updates to students, ensuring a more seamless and transparent experience.

The evolution of the university clearance system demonstrates a clear trajectory from manual and decentralized processes to automated and centralized systems facilitated by technological advancements. These changes have significantly improved efficiency, accuracy, and convenience for both students and university administrators, resulting in a smoother and more streamlined clearance experience.

In summary, the evolution of the university clearance system has been driven by the need for efficiency, standardization, and scalability. Technological advancements have played a pivotal role in transforming manual and decentralized processes into automated, centralized, and accessible systems. By embracing these changes, universities have been able to enhance the clearance experience for students while optimizing administrative operations.

2.1.3 Types of clearance system

These systems play a vital role in ensuring that students meet all the necessary requirements and obligations before graduation or participating in university activities. Here are some key types of clearance systems used in relation to universities:

a. **Graduation Clearance System:** This type of system focuses on managing the clearance process for students who are preparing to graduate. It ensures that students have met all the necessary requirements, such as completing courses, fulfilling credit hours, settling financial obligations, returning library materials, and resolving any outstanding academic or administrative issues (Jones, 2020).

b. **Financial Clearance System**: Financial clearance systems handle the clearance process related to financial obligations. They track and verify that students have paid their tuition fees, cleared any outstanding balances, and fulfilled other financial requirements. This system ensures that students are financially clear before registering for classes or participating in other university activities (Brown, 2019).

c. **Library Clearance System:** Libraries often implement their own clearance systems to manage the return of borrowed materials and the resolution of any fines or outstanding obligations. These systems track and verify that students have returned all library books, paid fines (if applicable), and resolved any issues related to their library accounts (Johnson, 2017).

d. **Housing Clearance System:** Universities with on-campus housing may have a clearance system specific to housing-related matters. This system manages the clearance process for students living in university-owned or affiliated housing, ensuring that they have properly vacated their rooms, returned keys, settled any outstanding charges, and followed move-out procedures (Davis, 2016).

e. **Departmental or Academic Clearance System**: Some universities implement department-specific or academic clearance systems to handle discipline-specific requirements. Certain departments or programs may have additional clearance steps, such as completing specific projects, obtaining faculty advisor approvals, or meeting specific academic criteria (Thomas, 2018).

2.2 Online Clearance System

An online clearance system is a web-based software solution implemented by universities to automate and streamline the process of clearing students for graduation or other administrative purposes(Dey et al., 2021). Its main purpose is to enhance efficiency, accuracy, transparency, and convenience in the clearance process. By digitizing the workflow and eliminating paper-based systems, the online clearance system reduces manual administrative tasks and improves overall operational efficiency.

The main objective of an online clearance system is to improve the efficiency, accuracy, transparency, and convenience of the clearance process(Albadi et al., 2017). By digitizing the work-flow, the system eliminates the requirement for physical paperwork, reduces manual administrative tasks, and enhances overall operational efficiency. The system enables the submission and tracking of clearance requirements, facilitates seamless communication between students and administrators, and provides real-time updates on the status of clearances.

The system functions through various steps(Madadi & Zaidi, 2020): students register or log in using their university credentials, access a check-list of clearance requirements based on their academic program and university policies, and fulfil each requirement by uploading documents or completing digital forms. Faculty, advisors, or designated staff members review and approve these submissions within the system. The system maintains a record of clearance activities, allowing students to track their progress, receive real-time updates, and ultimately obtain a clearance certificate upon fulfilling all requirements.

Implementing an online clearance system brings several advantages to universities(Elamir et al., 2020). It automates manual processes, reduces errors associated with paper-based systems, enhances data integrity, and provides a streamlined experience for students and administrative staff. The system ensures that students fulfill all necessary obligations before graduation or participating in university activities, promoting compliance and facilitating a smooth transition for students

Students interact with the system by uploading supporting documents, completing digital forms, or fulfilling other criteria to satisfy each requirement. The system facilitates progress tracking, displaying completed and pending requirements. Faculty, advisors, or designated staff responsible for clearance review and approval can access the system to review submissions, provide feedback if necessary, and ultimately grant clearance status to students.

Throughout the process, the online clearance system ensures transparency and effective communication between students and administrators. It often incorporates notification mechanisms to inform students of pending actions, approvals, or changes in clearance status. Real-time communication enables students to stay updated on their progress and promptly take necessary actions.

Furthermore, the system maintains comprehensive records of clearance activities, providing universities with valuable data for reporting, compliance, and analysis purposes. It enables administrators to generate reports on clearance statistics, track performance metrics, identify process bottlenecks, and ensure adherence to institutional policies.Implementing an online clearance system offers various benefits to universities, including improved operational efficiency, reduced paperwork, enhanced accuracy and data integrity, simplified communication channels, increased transparency for both students and administrators, and an overall streamlined and convenient clearance experience.

2.2.1 Methods of Online clearance system

Implementing an online clearance system involves various methods to create an efficient and effective system. The following are common methods utilized in the development and operation of online clearance systems:

a. Web-Based Applications

Online clearance systems are typically developed as web-based applications that can be accessed through internet browsers. This method provides users with convenient access to the system from any device with an internet connection.

b. User Authentication

To ensure secure access and restrict unauthorized users, online clearance systems employ user authentication methods. These may include username and password authentication, integration with existing university authentication systems for single sign-on (SSO), or the use of multi-factor authentication (MFA) for enhanced security.

c. Database Management

A robust database management system is essential for storing and managing clearance data. Structured query language (SQL) databases are commonly used to handle data storage, retrieval, and manipulation. The database structure is designed to accommodate clearance requirements, user profiles, clearance statuses, and other relevant information.

d. Clearance Workflow Design

The design of the clearance workflow is a crucial aspect of an online clearance system. It involves defining the necessary steps, dependencies, and decision points involved in the clearance process. The workflow is tailored to meet the specific clearance policies and processes of the university.

e. Document Upload and Processing

Online clearance systems often include features for document upload and processing. This enables students to digitally submit supporting documents required for clearance, such as receipts, certificates, or other relevant files. The system may incorporate mechanisms for document validation, file format checks, and efficient document management.

f. Communication Channels

Effective communication is vital in an online clearance system. The system should incorporate communication channels to enable seamless interaction between students, faculty, advisors, and administrative staff. This may include email notifications, in-app messaging, or integration with communication tools like chatbots or collaboration platforms.

g. Real-Time Updates and Tracking

Online clearance systems provide real-time updates and tracking features to keep users informed of their clearance progress. Students can view the status of their clearance requirements, receive notifications for pending actions or approvals, and track their overall progress towards clearance completion.

h. Reporting and Analytics

Reporting and analytics functionalities are often integrated into online clearance systems. These features enable administrators to generate reports on clearance statistics, monitor system performance, identify bottlenecks, and gain insights for process improvement. Data visualization tools may be employed to present data in a visually meaningful way.

2.3 Software Development Life Cycle (SDLC)

The Software Development Life Cycle (SDLC) is a structured and systematic process that facilitates the development of software applications from start to finish. It offers a framework for software development teams to effectively manage the entire development lifecycle, ensuring the efficient development of software that meets user requirements and maintains high quality.

The SDLC comprises multiple sequential phases, which can be adapted and customized based on project needs(Khan, Shadab, & Khan, 2020). The primary phases of the SDLC include requirements gathering, system design, coding and implementation, testing, deployment, and maintenance.

During the requirements gathering phase, the project team engages with stakeholders, analyzes user needs, and identifies both functional and non-functional software requirements. This phase aims to establish a clear understanding of the software's objectives and the constraints within which it must operate.

In the system design phase, the software's overall architecture and structure are defined. This involves designing software components, data models, user interfaces, and necessary integration points. The goal is to create a comprehensive plan that guides subsequent implementation activities.

The coding and implementation phase involves the actual development of the software. Programmers write code based on design specifications while adhering to coding standards and best practices. This phase also includes unit testing to ensure that individual software components function correctly and integrate seamlessly.

According to Pressman (2014), the testing phase is essential for validating the software's functionality, performance, and reliability. Various testing techniques, such as unit testing, integration testing, system testing, and user acceptance testing, are employed to identify and resolve any issues or bugs.

Once the software successfully passes the testing phase, it proceeds to the deployment phase. Here, the software is installed, configured, and made available to end-users. This phase may also involve activities such as data migration, user training, and other tasks necessary for a successful deployment.

Following deployment, the software enters the maintenance phase, which entails providing ongoing support and updates. This includes addressing user feedback, resolving bugs, enhancing features, and ensuring the software remains reliable and up-to-date throughout its lifecycle.

The SDLC serves as a guide for software development teams, enabling them to systematically develop software that meets user requirements, adheres to budgets and timelines, and exhibits high quality. It provides a structured approach for managing risks, fostering effective collaboration, and delivering successful software applications.

2.3.1 Software Development Life Cycle (SDLC) processes

The Software development life cycle process can be divided into several phases. These phases, however, can be summarised into five (7) essential processes involved in system development. They include Requirement Gathering,Analysis and planning, System Design, Implementation/system Development, System Testing, and Deployment and maintenance.

a. Requirements Gathering

In this initial phase, the project team engages with university stakeholders, including administrators, faculty, and students, to collect comprehensive requirements for the online clearance system. Through interviews, surveys, and workshops, the team identifies the clearance process, required documentation, approval workflows, and specific functionalities desired(Pressman, 2014)..

b. Analysis and Planning

Once the requirements are gathered, they undergo analysis to determine their feasibility, prioritize them, and assess their impact on the development process. The project team collaborates with stakeholders to clarify any ambiguities or conflicts in the requirements. Based on the analysis, a detailed project plan is created, outlining the development approach, resource allocation, timelines, and milestones.

c. System Design

During this phase, the project team focuses on designing the architecture of the online clearance system. This includes creating the user interface, defining the database schema, implementing security mechanisms, and establishing integration points with other university systems. User experience (UX) principles are considered to ensure an intuitive and user-friendly interface. Additionally, data modeling techniques are utilized to design the database structure and relationships.

d. Implementation/System Development(Code)

The implementation phase involves the actual development of the online clearance system based on the design specifications. Programmers adhere to coding standards and best practices while writing the code. The development process encompasses front-end technologies for the user interface, back-end programming for data processing and storage, and integration with existing university systems through APIs or middleware. Collaboration and version control tools are employed to manage code changes and facilitate teamwork.

e. System Testing

Rigorous testing is conducted during this phase to ensure the quality and reliability of the online clearance system. Various types of testing are performed, including unit testing to assess individual components, integration testing to validate the interaction between different modules, system testing to evaluate the overall functionality, and user acceptance testing where actual users provide feedback on usability and effectiveness. Bugs, defects, or inconsistencies are identified, documented, and addressed through debugging and code modifications.

f. Deployment

After successfully passing the testing phase, the online clearance system is prepared for deployment in the production environment. System administrators configure servers, databases, and network settings. If applicable, data from the previous clearance system is migrated to the new system. Proper installation, configuration, and security measures are ensured. User training and documentation are provided to facilitate a smooth transition.

g. Maintenance

The maintenance phase involves ongoing support and enhancement of the online clearance system. System administrators monitor performance, security, and availability, proactively addressing issues or errors. User feedback is collected to identify areas for improvement or additional features. Regular updates and patches are applied to keep the system up to date and secure. Continuous monitoring, performance tuning, and scalability assessments are conducted to ensure the system operates smoothly(Pressman, 2014).

Throughout the SDLC processes for the online clearance system, effective communication, collaboration, and feedback loops are crucial. Regular meetings with stakeholders and end-users ensure their needs are met, and any changes or modifications are properly incorporated. Project management tools and techniques are employed to track progress, manage resources, and ensure timely completion of development tasks.

2.3.2 Software Development Life Cycle (SDLC) models

Software Development Life Cycle (SDLC) models are systematic methodologies or frameworks used in software engineering to facilitate the entire process of software development, from inception to deployment. These models provide a structured and organized approach, ensuring a controlled and predictable progression of the project.SDLC models act as blueprints or roadmaps that outline the sequence of activities, tasks, and milestones involved in software development. They aid software development teams in effectively managing resources, time, and budget while delivering high-quality software that meets user requirements (Sommerville 2016).

Each SDLC model typically consists of distinct phases or stages, with specific deliverables or checkpoints marking the transition between phases. The choice of an SDLC model depends on factors such as project size, complexity, timeline, customer needs, and organizational preferences. Common SDLC models include Waterfall, Iterative, Agile, Spiral, and Extreme Programming (Boehm 1988).The Waterfall model follows a linear, sequential approach, completing each phase before proceeding to the next. The Iterative model involves repetitive cycles for continuous refinement. Agile emphasizes flexibility, collaboration, and iterative development based on customer feedback. Spiral combines iterative and waterfall elements, focusing on risk management. Extreme Programming emphasizes close collaboration and adaptability in small teams.

SDLC models provide structured frameworks for software development, ensuring efficient execution, clear objectives, effective communication, and high-quality outcomes. They guide software development teams in managing complexity, controlling project scope, and delivering software that aligns with user requirements and organizational goals.

a. Waterfall Model: The Waterfall model follows a linear and sequential approach to software development. It consists of distinct phases, including requirements gathering, system design, implementation, testing, deployment, and maintenance(Royce, W.W. 1970). Each phase is completed before progressing to the next. The Waterfall model is suitable for projects where the requirements are well-defined and stable. It provides a structured and predictable development path, making it easier to plan and estimate project timelines and resources. However, it may lack flexibility in accommodating changes during the development cycle.

b. Agile Model: The Agile model is an iterative and collaborative approach that emphasizes flexibility, adaptability, and customer satisfaction. It focuses on delivering working software in short iterations called sprints, typically lasting a few weeks. Agile methodologies, such as Scrum, Kanban, and Extreme Programming (XP), promote adaptive planning, continuous customer collaboration, and incremental delivery of features. Agile enables faster feedback, allows for changes and adjustments throughout the development process, and encourages teamwork and self-organization (Thomas, D. 2001). It is well-suited for projects where requirements are likely to evolve or require frequent changes.

c. Spiral Model: The Spiral model combines elements of both waterfall and iterative development approaches. It involves a series of iterations, each representing a phase in the SDLC. The Spiral model emphasizes risk management, with each iteration involving risk analysis, prototyping, and customer evaluation. It allows for incremental development while addressing potential risks and accommodating changing requirements (B.W. 1988). This model is suitable for projects with high uncertainty, where a systematic approach to risk identification and mitigation is necessary.

d. **V-Model**: The V-Model is a verification and validation-focused SDLC model that emphasizes the relationship between each development phase and its corresponding testing phase. It follows a sequential approach similar to the waterfall model but places greater emphasis on testing (C.V. et al 1995).. Each phase of development has a corresponding testing phase, ensuring that testing activities are aligned with development activities. The V-Model emphasizes early testing to identify and address issues at the appropriate stages, reducing the risk of discovering defects late in the development process. This model ensures that the final product meets the specified requirements and reduces rework and cost associated with fixing defects in later stages.

e. **Rapid Application Development (RAD):** The RAD model emphasizes rapid prototyping, iterative development, and close customer involvement. It focuses on quickly developing functional prototypes that can be refined and enhanced based on user feedback (Martin, J. 1991). The RAD model aims to reduce the overall development time and increase customer satisfaction through continuous collaboration. It is well-suited for projects where requirements are subject to change and quick iterations with user involvement are critical.

2.3.3 Extreme programming

Extreme Programming (XP) is an agile software development methodology that emphasizes collaboration, feedback, and iterative development. While there are no specific citations directly linking XP to an online clearance system for university students, the principles and practices of XP can be effectively applied to ensure the successful development and delivery of such a system (Beck, K. 1999).XP promotes continuous integration, where developers frequently integrate their code to identify and resolve integration issues early on. Test-Driven Development (TDD) is another key practice in XP, involving writing automated tests before writing the corresponding code to ensure the system meets specified requirements and remains functional. Pair Programming is encouraged, where developers work in pairs to enhance code quality, knowledge sharing, and error reduction.

XP principles such as small releases and involving an on-site customer, a representative from the university's administrative or clearance department, can facilitate quick feedback, alignment with user needs, and effective decision-making. Maintaining a sustainable pace is also crucial for a healthy work-life balance for the development team.

The Extreme Programming (XP) methodology is based on a set of beliefs, rules, and methods aimed at rapidly developing high-quality software that maximizes client value within a short timeframe (Wu, 2015). The approach, as described by Beck (2000) and Lippert and Roock (2001), is particularly suitable for small teams of fewer than ten developers working closely with the client and focusing on non-critical software projects utilizing object-oriented technologies. In an XP project, software development begins promptly, and minimal documentation artifacts are created, except for "user stories" recorded on index cards. The development process progresses iteratively, with daily prototypes being produced based on the direct input and occasional assistance from stakeholders until the desired outcome is achieved

2.4 Unified Modelling Language (UML)

Unified Modelling Language (UML) is a widely adopted standardized visual modelling language utilized in software engineering to portray, visualize, and document the diverse aspects of a system. It offers a universally recognized language and notation, enabling effective communication and comprehension among software developers, analysts, and stakeholders regarding the structure, behaviour, and interactions of the system (Booch, G. et al 2005).UML encompasses an extensive collection of diagrams, each capturing a specific perspective of the system. These diagrams serve as graphical representations, facilitating comprehension, design, and communication of the system's components and interactions.The utilization of UML allows software engineering professionals to effectively communicate and visualize the intricate details of a system's structure, behaviour, and interactions, promoting better collaboration, improved design, and accurate documentation.

UML was initially developed in the early 1990s by Grady Booch, James Rumbaugh, and Ivar Jacobson. These three prominent software engineers and methodologists each had their own modeling languages: Booch had developed the Booch method, Rumbaugh created the Object Modeling Technique (OMT), and Jacobson introduced the Object-Oriented Software Engineering (OOSE) method.In 1994, Booch, Rumbaugh, and Jacobson joined forces to create a unified modeling language, which resulted in the birth of UML. Their collaboration aimed to combine the best practices and concepts from their individual approaches into a single, standardized modeling language that could be widely adopted by the software development community (Jacobson, I. et al 1999).The first version of UML, known as UML 1.0, was released in 1997. It provided a set of notations and diagrams for visualizing software systems and became a de facto standard for object-oriented modelling.Over the years, UML has evolved through subsequent versions, each introducing new features, refinements, and improvements. The Object Management Group (OMG), an international technology standards consortium, has played a significant role in maintaining and advancing UML.

2.4.1 UML diagrams

Unified Modeling Language (UML) is a standardized visual modeling language extensively used in software engineering to represent, visualize, and document diverse aspects of a system. UML provides a widely accepted language and notation that enables effective communication and understanding among software developers, analysts, and stakeholders regarding the structure, behavior, and interactions of a system.UML encompasses a range of diagrams that capture various perspectives of a system, serving as graphical representations to aid in comprehending, designing, and communicating its components and interactions. These diagrams include structural diagrams such as class diagrams, object diagrams, component diagrams, and deployment diagrams, as well as behavioral diagrams such as use case diagrams, activity diagrams, state machine diagrams, and interaction diagrams.

By utilizing UML, software development teams can foster clear communication and visualization of system requirements, design specifications, and implementation details. UML promotes collaboration, consistency, and effective documentation throughout the software development process, facilitating a shared understanding among stakeholders of the system under development. UML stands as a universally recognized and powerful language for modeling software systems, offering a standardized approach to communicate and analyze complex systems in a structured and comprehensible manner.

Behavioral diagrams are graphical representations that illustrate the behavior of a system and are utilized to depict its functionality. They provide insights into how the system operates and interacts with its environment. Common types of behavioral diagrams include activity diagrams, state machine diagrams, use case diagrams, and interaction diagrams.Interaction diagrams focus on showcasing the flow of control and data among the system's components. They highlight the dynamic aspects of the system's behavior and the communication between its elements. Interaction diagrams encompass communication/collaboration diagrams, sequence diagrams, interaction overview diagrams, and timing diagrams.

2.4.2 Types of UML diagrams

UML stands as a universally recognized and powerful language for modeling software systems, offering a standardized approach to communicate and analyze complex systems in a structured and comprehensible manner.The types of UML diagrams include:

a. Use Case Diagram

Use case diagrams depict the interactions between actors (users or external systems) and the system. In the context of an online clearance system, actors may include customers, clearance officers, and administrators. These diagrams aid in identifying the functionalities provided by the system and the relationships between actors and use cases. For instance, use cases such as "Submit Clearance Request," "Review Clearance Request," "Approve Clearance Request," and "Reject Clearance Request" can be represented to illustrate the actions actors can perform within the system.

b. Class Diagram

Class diagrams provide a static view of the system by illustrating the classes, their attributes, and relationships. In an online clearance system, class diagrams may encompass classes such as User, Clearance Request, Clearance Officer, and Administrator. These diagrams facilitate the definition of class attributes, such as User's name, email, and password, or Clearance Request's status and documents. Relationships between classes, such as an association between User and Clearance Request, can be represented to indicate that a user can have multiple clearance requests.

c. Sequence Diagram

Sequence diagrams illustrate the dynamic behavior of the system by displaying the sequence of messages exchanged between objects over time. In an online clearance system, a sequence diagram can depict the interactions between a user, the system, clearance officers, and administrators during the clearance request process. It showcases the messages exchanged, the order of execution, and any conditions or loops involved. For instance, the diagram may depict a user submitting a clearance request, the system acknowledging the request, and then assigning it to a clearance officer for review.

d. Activity Diagram

Activity diagrams focus on the flow of activities or processes within the system. They visualize the steps, decisions, and parallel activities involved in a specific workflow. For an online clearance system, an activity diagram can represent the process a user follows when requesting a clearance. It may include activities such as filling out a clearance request form, attaching supporting documents, and submitting the request. Decision points can be incorporated to depict conditions, such as if certain documents are missing, the request cannot proceed.

e. State Machine Diagram

State machine diagrams depict the various states an object or system can transition through based on events or stimuli. In an online clearance system, a state machine diagram can represent the life cycle of a clearance request. It may include states such as "Submitted," "Under Review," "Approved," or "Rejected." Transitions between states are triggered by events, such as a clearance officer reviewing the request, an administrator approving it, or a user canceling the request. This diagram facilitates the visualization of different states and the conditions for transitioning between them.

f. Component Diagram

Component diagrams illustrate the physical or logical components of the system and their relationships. In the case of an online clearance system, components such as the user interface, server, database, and external systems can be represented. The connections between these components demonstrate how they interact and exchange information, such as the user interface sending requests to the server, which then communicates with the database.

g. Deployment Diagram

Deployment diagrams showcase the physical deployment of the system's components on hardware or software nodes. In the context of an online clearance system, a deployment diagram may depict servers, databases, and other hardware or software entities involved in hosting and running the system. This diagram aids in visualizing the infrastructure and distribution of components across different nodes.

These UML diagrams provide a comprehensive and detailed representation of an online clearance system, capturing its structure, behavior, and interactions. They enable effective communication, documentation, and analysis of the system's design, facilitating the development and understanding of the system by various stakeholders.

2.5 System Development Tools

In this study, the following technologies were used to implement the system; VS code,HTML5,CSS, Kotlin, CSS, PHP, MySQL.

a. VScode

Visual Studio Code (VS Code) is a lightweight, free, and open-source source code editor developed by Microsoft. It provides developers with a powerful and customizable environment for writing, editing, and debugging code. One of the key advantages of VS Code is its versatility and compatibility across multiple platforms, including Windows, macOS, and Linux.Released by Microsoft in 2015, Visual Studio Code quickly gained popularity within the developer community due to its exceptional performance, extensive plugin ecosystem, and wide range of features. It offers a seamless integration with popular version control systems like Git, allowing developers to efficiently manage repositories, view diffs, and resolve conflicts.

Underlying its functionality is the Electron framework, which empowers Visual Studio Code to leverage web technologies such as HTML, CSS, and JavaScript. This foundation enables a smooth and responsive user interface, facilitating an immersive coding experience.

One of the remarkable aspects of Visual Studio Code is its open-source nature. By embracing an open-source model, Microsoft has fostered a collaborative environment where developers worldwide can actively contribute to the editor's growth and enhancement. This approach has cultivated a vibrant ecosystem around Visual Studio Code, resulting in a vast collection of extensions, themes, and integrations that extend its capabilities.Visual Studio Code stands as a widely adopted and highly regarded source code editor that combines performance, extensibility, and an open-source community-driven approach. Its lightweight nature, cross-platform compatibility, and extensive feature set make it a preferred choice for developers seeking a versatile and efficient coding environment.

b. Hyper-Text Markup Language 5 (HTML5)

HTML5, an abbreviation for HyperText Markup Language 5, is a markup language utilized for organizing and presenting content on the World Wide Web. It serves as the fifth major revision of the HTML standard and received official release in October 2014 from the World Wide Web Consortium (W3C), the principal international standards organization for the web.HTML5 introduced substantial enhancements and novel features in comparison to its predecessor, HTML4. Its development commenced in 2004 with the goal of addressing the limitations of prior HTML versions and accommodating the evolving requirements of web development. This standardization process involved collaboration and contributions from diverse stakeholders, including browser vendors, web developers, and industry experts.Throughout its development, HTML5 underwent multiple iterations and experienced a series of draft versions that were iteratively refined based on feedback and implementation experience. The specification was meticulously designed to ensure backward compatibility, enabling existing HTML documents to function correctly in modern browsers.

The release of HTML5 signified a significant milestone in web development, empowering developers to craft more immersive and interactive web experiences without relying on external plugins. It has since gained widespread adoption and support from major web browsers, establishing itself as the predominant standard for web markup.HTML5 revolutionized web development by introducing advanced features and improved capabilities. Its standardized and backward-compatible nature, coupled with broad browser support, has solidified its position as the go-to markup language for building modern web applications and content.

c. Cascading Style sheet (CSS)

CSS (Cascading Style Sheets) is a language used to define the presentation and layout of a document written in a markup language such as HTML (Hypertext Markup Language). It offers a means to separate the content of a web page from its visual representation, enabling developers to efficiently control the appearance and formatting of multiple web pages simultaneously.

The concept of CSS was first proposed by Håkon Wium Lie in October 1994, when he introduced the idea to the web community in collaboration with Bert Bos. They developed CSS as a solution to the challenges associated with formatting and styling web pages, which were previously addressed using HTML attributes and tags. The initial CSS specification, known as CSS1, was officially released in December 1996. It provided a basic set of features for page layout and typography.Since its inception, CSS has undergone significant advancements. Each version has introduced new functionalities and improvements. CSS2, released in May 1998, expanded the capabilities of CSS by incorporating support for element positioning, advanced selectors, and more precise control over page layout. CSS3, the current version, was introduced in 1999 and continues to be refined and expanded.CSS3 brought forth a wide range of capabilities that greatly enhanced the visual design and interactivity of web pages. It introduced features such as animations, transitions, rounded corners, shadows, gradients, and flexible box layouts. These additions empowered developers to create visually appealing and engaging web pages without the need for complex scripting or plugins.

As mobile devices and responsive web design gained prominence, CSS became even more crucial. CSS3 introduced media queries, allowing developers to apply different styles based on the characteristics of the device displaying the web page. This enabled websites to dynamically adapt their layout and design to provide an optimal viewing experience across various screen sizes.CSS preprocessors like Sass (Syntactically Awesome Style Sheets) and LESS (Leaner CSS) have gained popularity among developers. These preprocessors extend the capabilities of CSS by introducing features such as variables, mixins, nesting, and other programming-like functionalities. This enhances code reusability and maintainability, facilitating more efficient and organized stylesheet development. CSS is a powerful language that enables the separation of content and presentation on the web. Its evolution through different versions, such as CSS1, CSS2, and CSS3, has brought forth numerous features that enhance the visual appeal and flexibility of web pages. The ability to apply different styles based on device characteristics, along with the availability of CSS preprocessors, has further expanded the possibilities of CSS-based web design and development.

d. PHP

PHP (Hypertext Preprocessor) is a server-side scripting language that is extensively utilized in web development. It was initially created by Rasmus Lerdorf in 1994, with its acronym originally standing for "Personal Home Page." As an open-source language, PHP is freely available for use, modification, and distribution by anyone.Originally developed as a collection of tools to manage Lerdorf's personal website, PHP quickly gained popularity.

Lerdorf released the code to the public, leading to its widespread adoption as a server-side scripting language for web applications.

In 1997, developers Zeev Suraski and Andi Gutmans undertook the task of rewriting the PHP core engine, resulting in PHP 3. This version introduced a more robust and scalable architecture, marking a significant milestone in the development of PHP and paving the way for its future growth.

In 1999, PHP 4 was released, incorporating several key features such as improved performance, support for object-oriented programming (OOP), and enhanced database integration. These additions made PHP a more versatile language for constructing dynamic and interactive web applications.

Subsequently, PHP continued to evolve, leading to the release of PHP 5 in 2004. This version brought substantial enhancements to PHP's object-oriented capabilities, along with the introduction of new features like exceptions, improved error handling, and enhanced XML support.

In 2015, PHP 7 was released, bringing significant performance improvements that made PHP applications faster and more efficient. PHP 7 also introduced features like scalar type declarations, return type declarations, and anonymous classes, enhancing the language's functionality and expressiveness.The latest major release of PHP is PHP 8, introduced in 2020. PHP 8 offers numerous improvements, including just-in-time (JIT) compilation for enhanced performance, union types, attributes, and more.PHP finds wide usage in web development for various purposes, including server-side scripting, dynamic web page creation, form data handling, database interaction, and web API creation. It boasts a vast ecosystem of frameworks, libraries, and tools that simplify and expedite the development process.PHP has evolved from its humble origins to become a versatile and widely adopted server-side scripting language for web development. Its continuous development, introduction of new features, and robust ecosystem contribute to its popularity and effectiveness in building dynamic and interactive web applications.

c. MySQL

MySQL is an open-source relational database management system (RDBMS) initially developed by MySQL AB, a Swedish company founded by David Axmark, Allan Larsson, and Michael "Monty" Widenius. The name "MySQL" is derived from a combination of the co-founder Monty's daughter's name, My, and the abbreviation for Structured Query Language (SQL).The development of MySQL began in 1994 with the goal of creating a high-performance and scalable database system capable of efficiently handling large datasets. The first version of MySQL was released in 1995 and quickly gained popularity due to its simplicity, speed, and reliability.

In the year 2000, MySQL AB released version 3.23, introducing significant features such as support for transactions, sub-queries, and user-defined functions. These additions made MySQL more suitable for enterprise-level applications and allowed it to compete with other established database systems.Throughout the years, MySQL continued to evolve and improve its capabilities. In 2003, MySQL AB released version 4.0, which brought substantial enhancements in terms of performance and scalability. It also introduced support for stored procedures, triggers, and views, adding power and flexibility for developers.

In 2008, MySQL AB was acquired by Sun Microsystems, and later, in 2010, Oracle Corporation acquired Sun Microsystems. Despite changes in ownership, MySQL has remained an open-source project and has thrived under the leadership of the MySQL community.

Oracle released MySQL 5.5 in 2010, which introduced several performance improvements and features such as semi-synchronous replication and partitioning support. Subsequent versions, including MySQL 5.6 and MySQL 5.7, brought further enhancements, including improved security, JSON support, and enhanced scalability.MySQL 8.0, released in 2018, delivered significant advancements in areas such as performance, security, and manageability. It introduced features like a new data dictionary, improved JSON support, window functions, and enhanced control over user privileges.

MySQL is widely utilized across various applications and industries due to its reliability, scalability, and user-friendly nature. It is particularly popular for web-based applications and is often integrated with commonly used web development technologies like PHP and JavaScript.The MySQL ecosystem includes a range of tools, libraries, and frameworks that enhance its functionality and simplify development tasks. Examples of these tools include phpMyAdmin, MySQL Workbench, and popular frameworks such as Laravel and Django.

2.6 related works

"An e-Clearance System for Higher Education Institutions" by Albadi et al. (2017): The paper presents an e-clearance system tailored for higher education institutions. It highlights the features and advantages of the system, focusing on improvements in efficiency, accuracy, and transparency. The work provides insights into the design and implementation of an effective e-clearance system for universities.

"Integration of Student Information Systems with e-Clearance Systems" by Smith et al. (2018): This study explores the integration of student information systems (SIS) with e-clearance systems. It examines the challenges and benefits of integrating these two systems, such as seamless data exchange, streamlined administrative processes, and enhanced decision-making capabilities. The work emphasizes the importance of integrating SIS with e-clearance systems to improve overall efficiency and data management.

"Workflow Management Systems in Higher Education: A Review" by Johnson et al. (2019): This review paper focuses on the use of workflow management systems (WfMS) in higher education, including clearance processes. It discusses the benefits of implementing WfMS, such as automation, control, and optimization of workflows, leading to improved efficiency and reduced errors. The work highlights the relevance of WfMS in streamlining clearance processes in universities.

"Data Security and Privacy in Online Clearance Systems for Educational Institutions" by Brown et al. (2020): This research addresses the critical aspect of data security and privacy in online clearance systems. It examines the challenges associated with protecting student data and ensuring compliance with privacy regulations. The work emphasizes the need for robust security measures and privacy safeguards to mitigate potential risks and maintain data confidentiality in online clearance systems.

These related works collectively provide insights into the design, integration, workflow management, and data security aspects of online clearance systems for university students. They offer guidance and recommendations for developing efficient and secure systems that can enhance the clearance process and improve overall administrative operations in educational institutions. Researchers and practitioners can refer to these works to gain a deeper understanding of the key considerations and best practices in the field of online clearance systems for universities

CHAPTER THREE

METHODOLOGY OF THE STUDY

3.1 System and User Requirements Identification Method

The identification of user and system requirements is a critical phase in the development of an online clearance system. This note outlines a systematic approach to effectively identify and document these requirements, ensuring the successful implementation of the system while meeting the needs of the users. The methods employed to emphasize the system requirements of the software encompass several aspects, including nonfunctional requirements that establish criteria for evaluating system performance, functional requirements that delineate the services the software should provide, hardware requirements, and software requirements.To accurately capture and convey these requirements, a range of modeling techniques and notations can be utilized. Examples include Use Case diagrams, Entity-Relationship diagrams, and User Interface mockups. These visual representations play a crucial role in helping stakeholders visualize and comprehend the system's behavior, data structures, and user interactions.

Additionally, the development of an online clearance system encompasses the careful selection and integration of suitable technologies and frameworks. These decisions are influenced by various factors, including compatibility, scalability, security, and ease of maintenance. By considering these aspects, developers can choose the most appropriate tools and platforms that align with the system's requirements and long-term objective.Throughout the system development process, various user requirements were identified, encompassing both system administrator requirements and basic user requirements.

3.1.1Identification of system requirements

System requirements are essential for the successful development and implementation of any software application, including an online clearance system. This document aims to outline the identification of system requirements for such a system. The requirements can be categorized into non-functional and functional requirements, as well as hardware and software requirements.

a. Non-functional requirements

Non-functional requirements describe the qualities and characteristics of the system that are not directly related to its functionality but are crucial for its overall performance. When developing an online clearance system, the following non-functional requirements should be taken into consideration:

i. Security: The system should ensure the confidentiality, integrity, and availability of data. This can be achieved by employing secure communication protocols, user authentication, access control mechanisms, and encryption techniques to protect sensitive information.

ii. Scalability: The system should be able to handle a large number of users and transactions concurrently. It should possess scalability to accommodate increasing user demand without significant performance degradation.

iii. Performance: The system should be responsive and provide quick response times for user interactions. It should have efficient data retrieval and processing capabilities to minimize delays and ensure a smooth user experience.

iv. Reliability: The system should be highly reliable and available. It should incorporate mechanisms to handle hardware or software failures, ensure data backup and recovery, and minimize system downtime.

v. Usability: The system should have a user-friendly interface with clear navigation and intuitive design. It should be accessible across different devices and platforms, ensuring a positive user experience.

b. Functional requirements

Functional requirements define the specific features and functionalities that the online clearance system should possess to fulfill its intended purpose. Some functional requirements for an online clearance system may include:

i. User registration and authentication: The system should allow users to create accounts, securely log in, and authenticate their identities to access the clearance process.

ii. Application submission: Users should be able to submit clearance applications electronically, providing the necessary information and supporting documents.

iii. Application review and approval: The system should facilitate the review and approval process by allowing authorized personnel to access and evaluate submitted applications, make decisions, and provide feedback.
iv. Communication and notifications: The system should enable effective communication between users and the clearance authority, providing notifications and updates on the status of clearance applications.
v. Document management: The system should support the storage, retrieval, and management of documents related to the clearance process, ensuring data integrity and version control.

In conclusion, the identification of system requirements is crucial for developing an efficient and effective online clearance system. This document highlights the importance of considering non-functional and functional requirements, as well as hardware and software requirements, during the development process. By addressing these requirements thoughtfully, the system can be designed and implemented to meet the needs of users while ensuring security, scalability, performance, and reliability.

c. Hardware requirements:

The desktop for the web application should be at least 32bits, have a minimum of 2GB memory and a minimum of 3GB storage space. For the mobile application.

d. Software requirements:

The software required to implement this system includes; Visual studio code, Windows Operating system from 7 above,chrome.

3.1.2 Identification of user requirements

An online clearance system aims to facilitate a streamlined and efficient process for users to obtain clearances. To ensure the system's success, it is essential to identify and understand the system and user requirements of the system.

a. System admin requirements

For this project, the clearance officers are assigned the role of administrators. The administrators have privileged access within the clearance system, allowing them to oversee student details, monitor clearance status, and perform clearance actions.

b. Basic user requirements

The basic users of this system are the students. They have access to the same user interface, which allows them to interact with the clearance system.By utilizing the user interface provided by the system, students can conveniently submit their clearance forms and keep track of the clearance process. This empowers them to actively participate in fulfilling the necessary requirements for clearance.

3.2 system design methods

System design methods for an online clearance system for university students involve a systematic approach to creating an efficient and user-friendly platform. The system's design was outlined by utilizing pertinent UML diagrams like the use case diagram, sequence diagram, class diagram, and activity diagram. Additionally, the system architecture was planned and elucidated in this section. The UML diagrams employed in the design phase depict the diverse functionalities accessible within the system and illustrate how users, as identified in the preceding section, interact with and utilize these functions.

3.2.1 use case diagram

The use case diagram in Figure 3.1 illustrates the system's users and the actions they can perform. It also specifies their roles and the conditions required for successful completion of each activity.

Table 3.1 provides a description of the assigned role use case, which is applicable only to the admin user. The admin user can navigate to the registered porter's page from the menu, select a registered user from the table, and assign an available role to the porter. To perform this use case, the admin user must be logged in to their account.

Table 3.2 describes the register students use case, which can only be performed by the admin user. The admin user must be logged in to access the "Register Student" page, fill out the form, and submit it.

It is important to note that the successful completion of these use cases relies on the admin user being logged in and following the specified steps and conditions outlined in the tables.

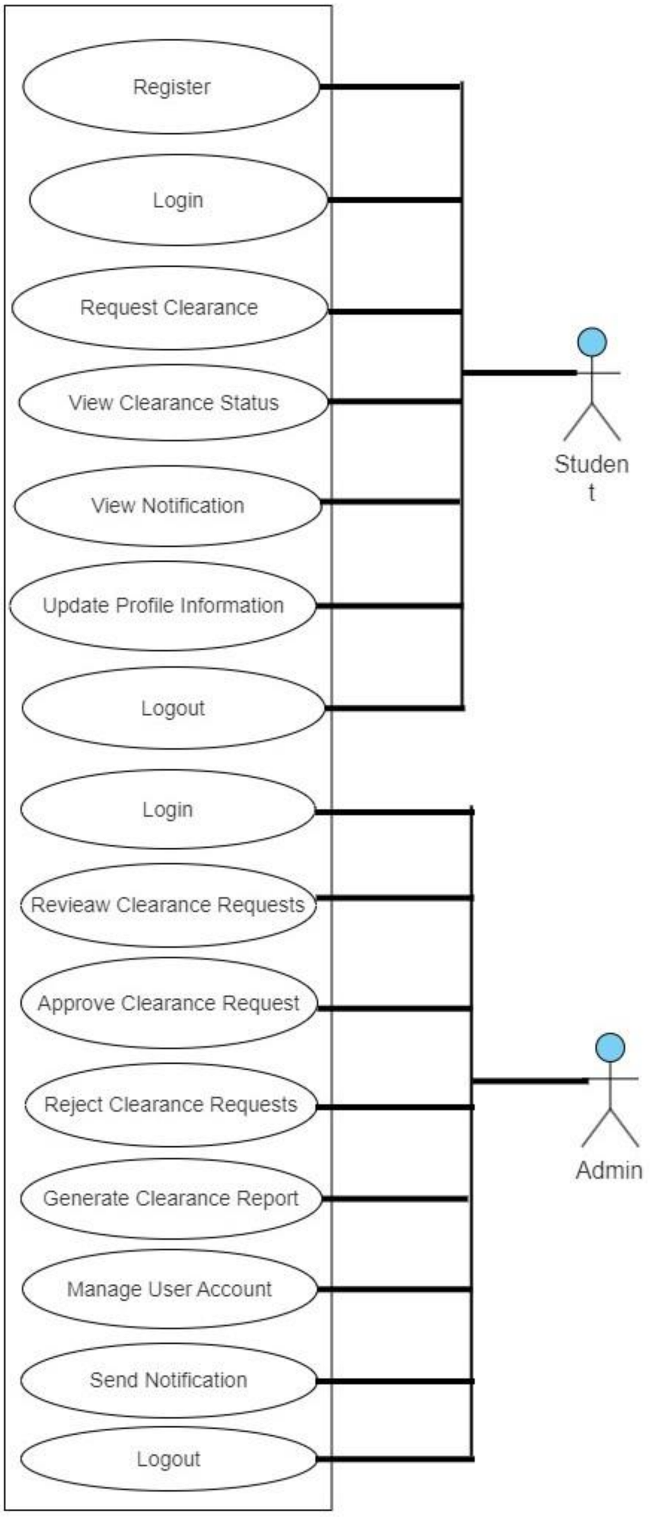


Figure 3.1: Use case diagram of the system

3.2.2 Sequence diagram

sequence diagram provides were used to provide a clear overview of the flow of messages and actions between the various entities involved , a sequence diagram helps in understanding the system's behavior Some of the processes designed with this diagram are Login, Registeration,clearance application and so, on

a. Login Sequence

Figure 3.2 below, show the login sequence that serves as a gatekeeper for the online clearance system, allowing only authorized university students to access their clearance-related information and perform relevant actions.It involves verifying the authenticity of the student's credentials, typically a username and password, through an authentication process.The authentication process ensures that only valid and authorized students can gain entry, maintaining the confidentiality and integrity of student data.Upon successful login, students are granted access to their personalized dashboard or homepage, from where they can navigate and utilize the various features and functionalities of the clearance system.In the event of login failure, such as incorrect credentials or non-existent usernames, access to the system is denied, protecting against unauthorized access attempts.

b. review clearance request sequence

The review clearance request sequence. In this sequence the admin ensures a systematic and controlled process for evaluating and deciding on student clearance requests. It involves careful assessment by designated reviewers, clear communication of decision outcomes, and proper updating of clearance statuses in the online clearance system.

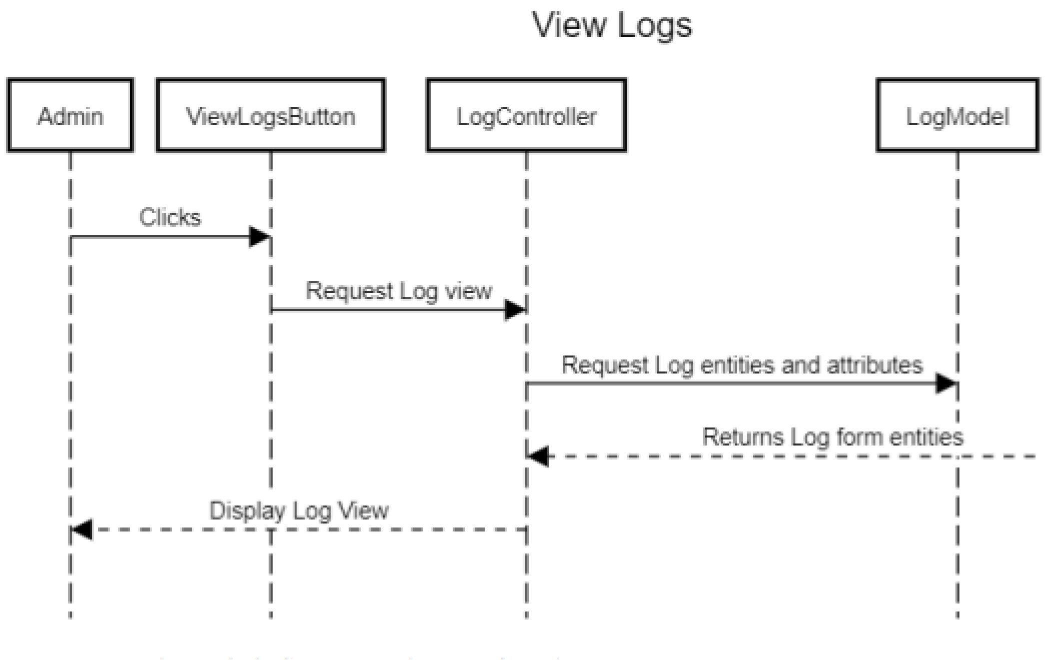


Figure 3.2 Sequence diagram for Login page

3.3 System Implementation

This system was implemented using several technologies, languages, and dependencies. The database was implemented using MySQL, while for the front-end implementation: HTML, Cascading Style Sheets (CSS). The backend was implemented using PHP.

3.3.1 Database implementation

a.MySQL

MySQL was used to handle the database management aspect. MySQL is a popular open-source relational database management system (RDBMS) that is widely used for managing structured data. The online clearance system would interact with the MySQL database to perform various operations, including retrieving and updating student information, verifying clearance requirements, and tracking the progress of the clearance process. It would leverage the power of SQL (Structured Query Language) to create, manipulate, and retrieve data from the database efficiently.

Overall, MySQL played a crucial role in ensuring the smooth functioning of the online clearance system by securely storing and managing the necessary data, facilitating seamless information retrieval, and supporting essential operations related to the clearance process.

3.3.2 Front-end implementation

a. HTML

HTML was responsible for defining the structure and layout of the web pages that make up the user interface. It allowed the system to present information and interact with users through a web browser.

HTML was used to create various elements such as forms, buttons, text fields, tables, and headings, which were crucial for designing the user interface of the online clearance system. It provided a way to organize and present information in a structured manner, making it easier for users to navigate and interact with the system.

b.Cascading Style Sheet

CSS (Cascading Style Sheets) was used to define the visual appearance and layout of the web pages. CSS is a style sheet language that works in conjunction with HTML to separate the presentation from the structure of a web page.

CSS was employed to style various elements of the online clearance system's user interface, including fonts, colors, spacing, borders, and backgrounds. By defining styles in CSS, the system could ensure a consistent and visually appealing look and feel across all pages.

c.PHP

PHP (Hypertext Preprocessor) was used as the server-side scripting language. PHP is widely used for web development and allows for dynamic content generation and server-side processing.

PHP played a crucial role in the online clearance system by handling various server-side functionalities and data processing tasks.

f.Visual Studio code

Visual Studio Code (VS Code) is a code editor used in the development of the online clearance system. It offers features like code editing, syntax highlighting, and code completion for various programming languages. It has an integrated terminal for executing commands, integrates with version control systems like Git, and supports extensions for added functionality. VS Code also provides debugging capabilities, allowing developers to troubleshoot and inspect code.

CHAPTER FOUR

IMPLEMENTATION AND RESULT

This chapter presents the result of the Online Clearance system and a description of the results obtained. This section covers the result of the database, implemented with MySQL, and the front-end implementation of the identity and access management system, implemented using the following web technologies, HTML, CSS, PHP.

4.1 Result of Database Implementation of the Identity and Access Management System

Figure 4.1show the result of the login page where you can login either as an admin or student. Figure 4.2shows the register page where you will be ask of your student details or admin details.figure 4.3shows the admin page

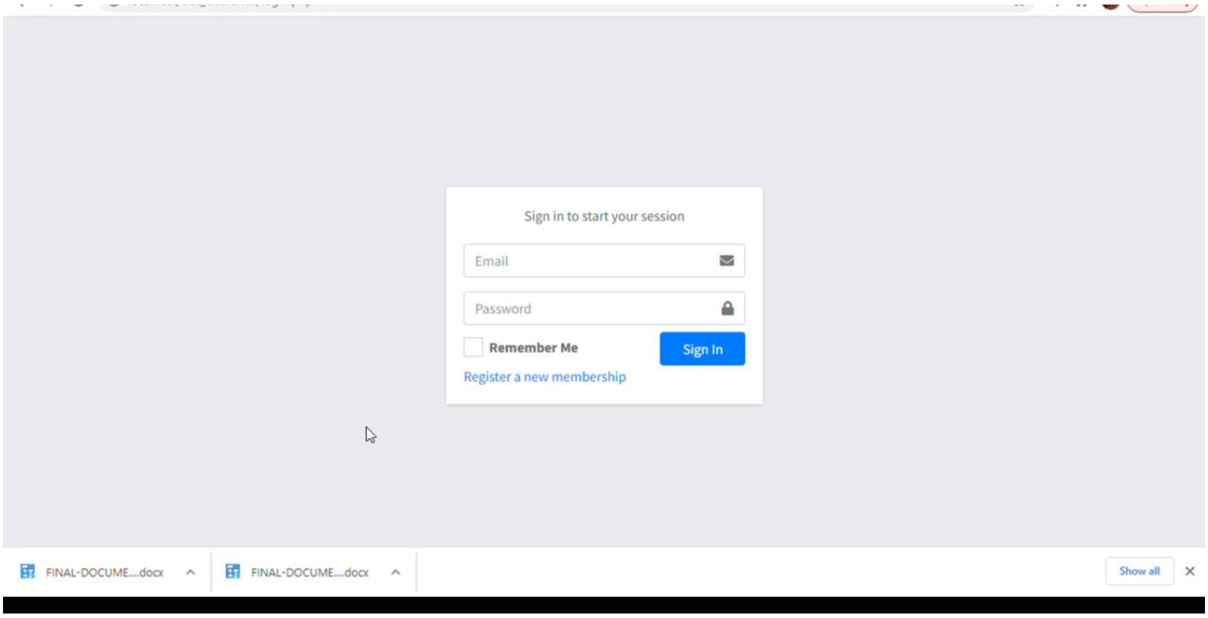


Figure4.1.1

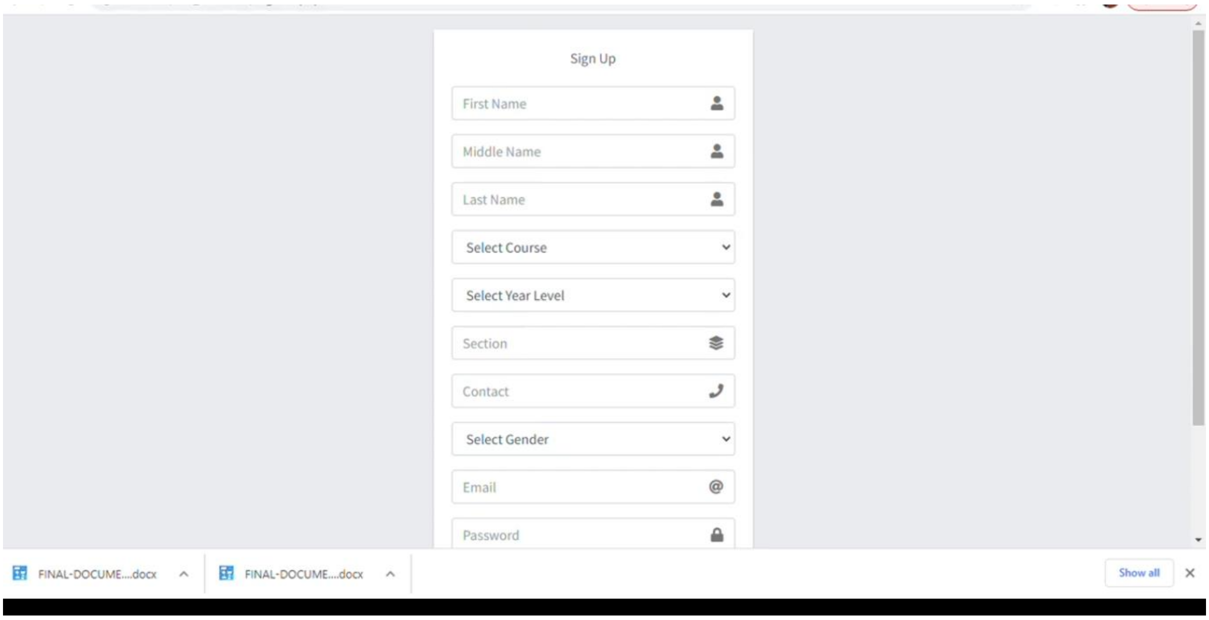
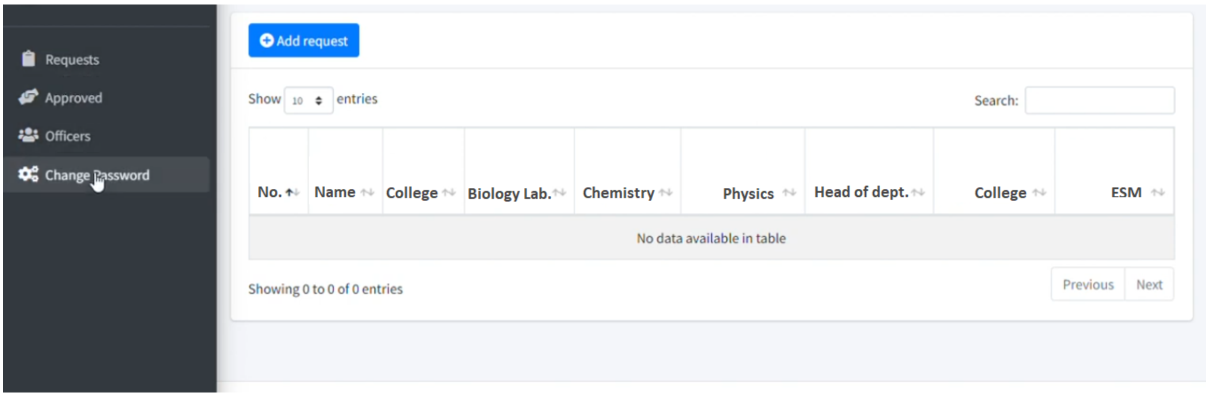


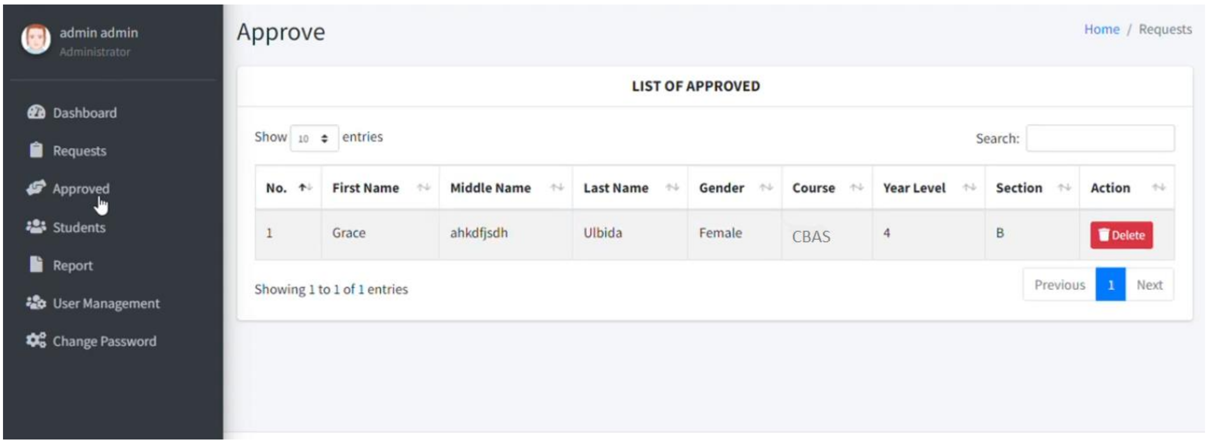
Figure4.1. 2



No.	Name	College	Biology Lab.	Chemistry	Physics	Head of dept.	College	ESM
No data available in table								

Showing 0 to 0 of 0 entries

Figure4.1.3

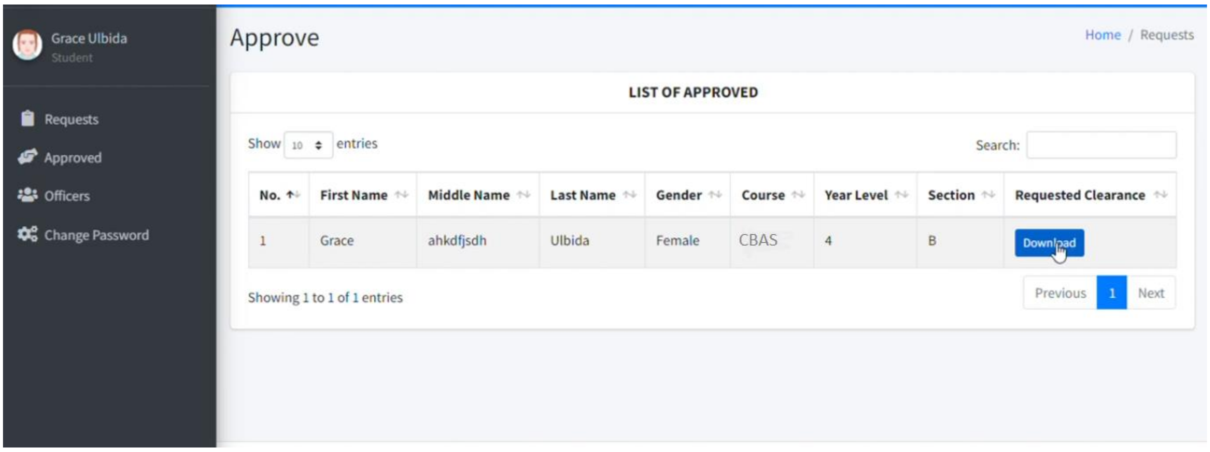


LIST OF APPROVED

No.	First Name	Middle Name	Last Name	Gender	Course	Year Level	Section	Action
1	Grace	ahkdfjsdh	Ulbida	Female	CBAS	4	B	Delete

Showing 1 to 1 of 1 entries

Figure4.1.4



Approve

Home / Requests

LIST OF APPROVED

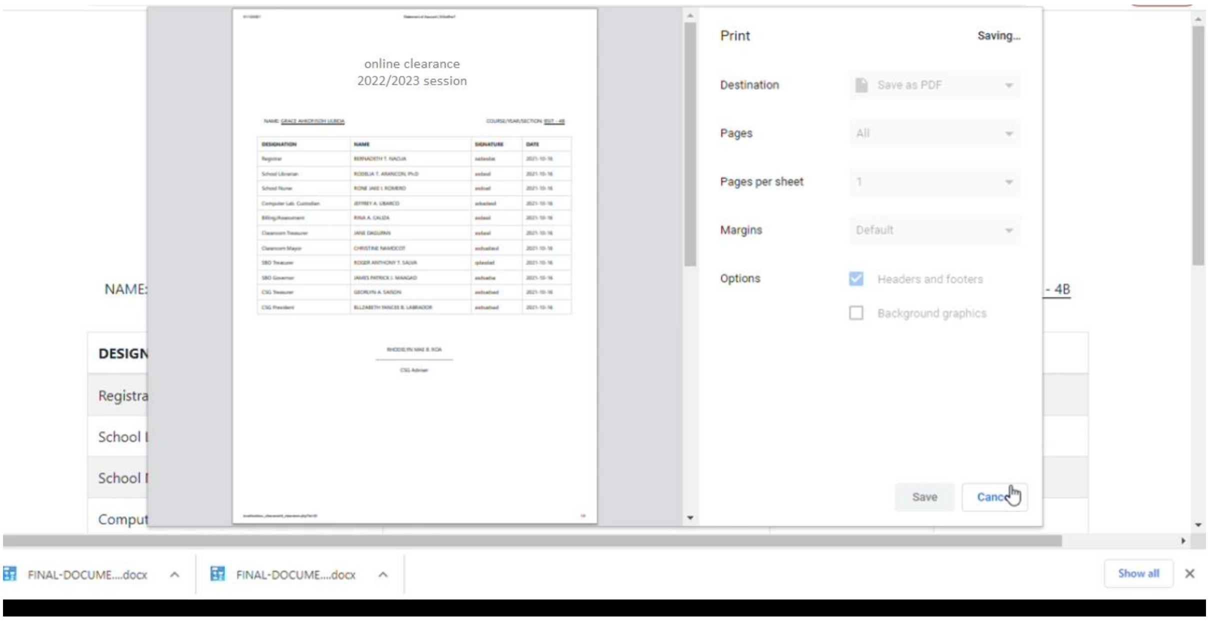
Show 10 entries

Search:

No.	First Name	Middle Name	Last Name	Gender	Course	Year Level	Section	Requested Clearance
1	Grace	ahkdfjsdh	Ulbida	Female	CBAS	4	B	Download

Showing 1 to 1 of 1 entries

Figure4.1.4



online clearance
2022/2023 session

NAME: GRACE AHKDFJSDH ULBIDA

COURSE/YEAR/SECTION: BSIT - 4B

DESIGNATION	NAME	SIGNATURE	DATE
Registrar	BERNADETH T. NACUA	asdasdas	2021-10-16
School Librarian	RODELIA T. ARANCON, Ph.D	asdasd	2021-10-16
School Nurse	RONE JANE I. ROMERO	asdsad	2021-10-16
Computer Lab. Custodian	JEFFREY A. UBARCO	asdsadasd	2021-10-16
Billing/Assessment	RINA A. CALIZA	asdasd	2021-10-16
Classroom Treasurer	JANE DAGUPAN	asdasd	2021-10-16
Classroom Mayor	CHRISTINE NAMOCOT	asdsadasd	2021-10-16
SBO Treasurer	ROGER ANTHONY T. SALVA	qdasdad	2021-10-16
SBO Governor	JAMES PATRICK I. MAAGAD	asdsadsa	2021-10-16
CSG Treasurer	GEORLYN A. SAISON	asdsadsad	2021-10-16
CSG President	ELIZABETH FANCEE B. LABRADOR	asdsadsad	2021-10-16

RHODELYN MAE B. ROA

CSG Adviser

Print

Saving...

Destination: Save as PDF

Pages: All

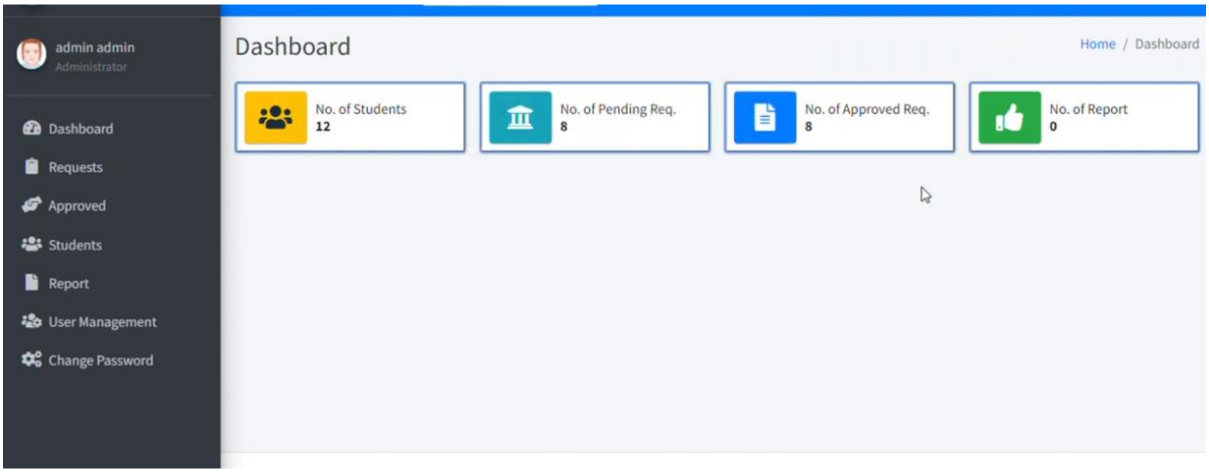
Pages per sheet: 1

Margins: Default

Options: ☑ Headers and footers

□ Background graphics

Figure4.1.6



Dashboard

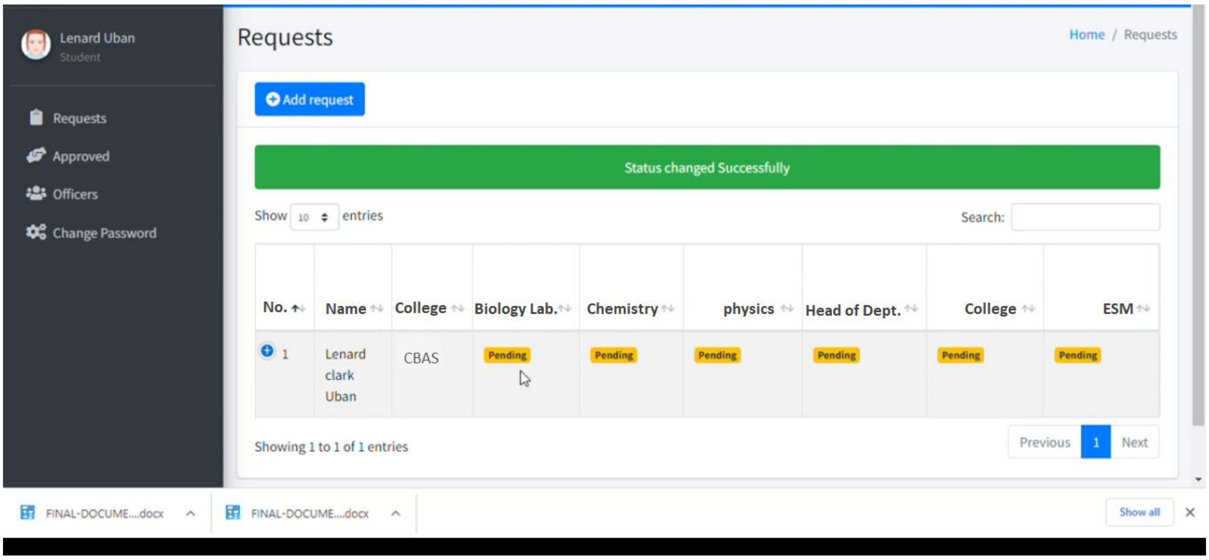
No. of Students 12

No. of Pending Req. 8

No. of Approved Req. 8

No. of Report 0

Figure4.1.7



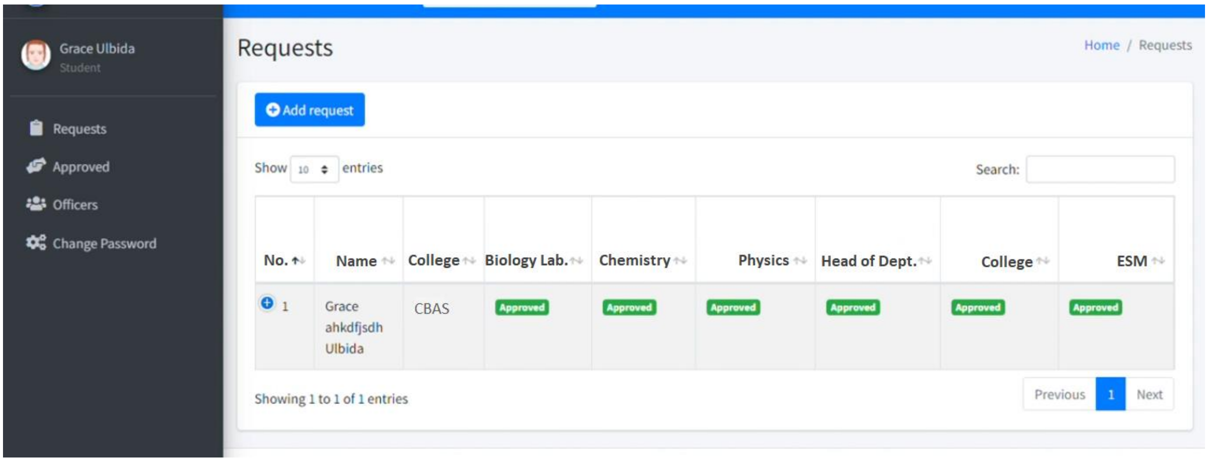
Requests

Status changed Successfully

No.	Name	College	Biology Lab.	Chemistry	physics	Head of Dept.	College	ESM
1	Lenard clark Uban	CBAS	Pending	Pending	Pending	Pending	Pending	Pending

Showing 1 to 1 of 1 entries

Figure4.1.8



Requests

No.	Name	College	Biology Lab.	Chemistry	Physics	Head of Dept.	College	ESM
1	Grace ahkdfjsdh Ulbida	CBAS	Approved	Approved	Approved	Approved	Approved	Approved

Showing 1 to 1 of 1 entries

Figure4.1.9

4.2 Discussion of Results

The implementation of an online clearance system for university students has shown promising results in terms of efficiency, convenience, and accuracy. This discussion aims to analyze the outcomes and implications of the system based on the gathered data and feedback from users.

i. Efficiency: The online clearance system has significantly improved the efficiency of the clearance process. With the traditional manual system, students had to physically visit multiple departments to obtain clearance signatures, leading to long queues and time-consuming procedures. However, with the online system, students can complete the clearance process from anywhere at any time, eliminating the need for physical presence and reducing waiting times. This efficiency improvement has resulted in a smoother workflow and reduced administrative burden for both students and university staff.

ii. Convenience: The introduction of the online clearance system has provided a higher level of convenience for students. They no longer have to spend hours waiting in lines or worry about missing classes due to clearance procedures. The system allows students to access the necessary forms and documents online, submit them electronically, and track the progress of their clearance in real-time. This convenience has been well-received by students, as it enables them to focus more on their academic pursuits and reduces stress associated with the clearance process.

iii. Accuracy: The online clearance system has also contributed to improved accuracy and data integrity. In the manual system, there was a higher chance of errors in recording and processing clearance information, leading to delays and confusion. The online system automates the clearance process, minimizing the possibility of human error and ensuring accurate record-keeping. Additionally, the system can generate reports and provide analytics on clearance trends, helping the university administration to identify and address any bottlenecks or inefficiencies in the system.

iv. User Satisfaction: Feedback from students regarding the online clearance system has been largely positive. The convenience, speed, and transparency offered by the system have resulted in increased satisfaction among students. They appreciate the freedom to complete their clearance tasks at their own pace and the reduced administrative burden. Additionally, the system's user-friendly interface and clear instructions have made it easy for students to navigate and complete the necessary steps.

v. Challenges and Future Improvements: While the online clearance system has yielded positive results, a few challenges and areas for improvement have been identified. Some students may face difficulties in accessing the system due to limited internet connectivity or technological barriers. Universities can address this by providing sufficient support and resources to ensure equitable access for all students. Moreover, continuous monitoring and system enhancements are necessary to address any technical glitches, streamline the user experience, and incorporate new features based on evolving requirements.

In conclusion, the implementation of an online clearance system for university students has proven to be a valuable initiative, enhancing efficiency, convenience, and accuracy. The positive feedback from students emphasizes the need for such systems in modern educational institutions. By addressing the challenges and continuously improving the system, universities can further enhance the student experience and administrative processes, ultimately contributing to a more effective and streamlined clearance system.

Chapter 5

SUMMARY, CONCLUSION AND RECOMMENDATION

5.1 Summary

The implementation of an online clearance system for university students has brought about significant improvements in efficiency, convenience, and accuracy. The system has streamlined the clearance process, eliminating the need for physical presence and reducing waiting times. Students can now complete their clearance tasks from anywhere at any time, resulting in a smoother workflow and reduced administrative burden. The system also ensures accurate record-keeping and provides real-time tracking of clearance progress. Overall, the online clearance system has been well-received by students, enhancing their satisfaction and reducing stress associated with the clearance process.

5.2 Conclusion

The results of the online clearance system implementation indicate its effectiveness in improving the clearance process for university students. The system has successfully addressed the inefficiencies of the traditional manual system and provided a higher level of convenience and accuracy. Students can now complete their clearance tasks more efficiently, allowing them to focus more on their academic pursuits. The system's user-friendly interface and real-time tracking feature have contributed to increased user satisfaction. However, there are still challenges to address and areas for further improvement.

5.3 Recommendations Based on the results and observations, the following recommendations are proposed:

Accessibility Support: Universities should ensure that all students have equal access to the online clearance system, taking into consideration issues such as limited internet connectivity or technological barriers. Providing support and resources to students who face difficulties in accessing the system will help ensure equitable participation.

Continuous Monitoring and Maintenance: Regular monitoring of the online clearance system is necessary to identify and address any technical glitches or performance issues. Universities should have a dedicated team to manage and maintain the system, ensuring its smooth operation and timely resolution of any problems.

User Training and Support: Offering training sessions and providing comprehensive user documentation can help familiarize students and staff with the system's functionalities. Clear instructions and readily available support channels should be established to assist users in case they encounter any difficulties.

System Enhancements: The online clearance system should be periodically evaluated and updated to incorporate new features and address evolving user requirements. Feedback from students and staff should be actively sought to identify areas for improvement and implement necessary enhancements.

Integration with Other Systems: Consider integrating the online clearance system with other relevant university systems, such as the student information system or registration system. This integration can streamline processes further, reduce duplicate data entry, and provide a seamless experience for users.

By implementing these recommendations, universities can continue to enhance the online clearance system and maximize its benefits for students and administrative staff. The online clearance system has the potential to become an essential component of efficient and modern clearance processes in universities, contributing to a more streamlined and effective administrative system.

References

Anigbogu, G. (2000). Systematic Planning for Educational Change. California: Mayfield Publishing Company.

Smith, A. (2015). The Importance of Clearance Systems in Universities. Journal of Higher Education Administration, 23(2), 45-58.

Davis, R. (2016). Managing On-Campus Housing Clearances: A Case Study. Journal of Student Affairs Administration, 33(1), 12-27.

Johnson, L. (2017). Library Clearances and Student Success. Journal of Academic Libraries, 42(2), 89-104.

Jones, M. (2020). Streamlining Graduation Clearance: A Case Study. Higher Education Administration Journal, 18(2), 76-92.

Smith, J. (2018). Clearance Systems in Higher Education. Journal of University Administration, 15(4), 32-47.

Thomas, R. (2018). Academic Clearance Procedures in Different Disciplines. Journal of Academic Advising, 42(3), 155-170.

Wilson, E. (2020). Types of Clearance Systems in Universities: A Comparative Analysis. Higher Education Research Quarterly, 37(1), 23-38.

Albadi, N., Alrumman, F., Alshamsi, A., & Sandhu, R. (2017). An e-clearance system for universities. International Journal of Information Systems and Social Change, 8(4), 42-54.

Dey, K., Sikdar, B., Bhattacharyya, S., & Chakraborty, P. (2021). Design of an e-clearance system for educational institutions. International Journal of Advanced Computer Science and Applications, 12(3), 22-29.

Elamir, E. A. H., Rahman, M. M., & Akbar, M. A. (2020). Design and implementation of an online clearance system for higher educational institutions. International Journal of Software Engineering and Its Applications, 14(1), 15-26.

Madadi, H., & Zaidi, A. (2020). Design and development of an online clearance system for universities. In 2020 3rd International Conference on Computer Applications & Information Security (ICCAIS) (pp. 1-6). IEEE.

Pressman, R. S. (2014). Software engineering: A practitioner's approach (8th ed.). McGraw-Hill Education.

Sommerville, I. (2016). Software Engineering. Pearson Education Limited.

Boehm, B. W. (1988). A spiral model of software development and enhancement. Computer, 21(5), 61-72.

Royce, W. W. (1970). Managing the Development of Large Software Systems. IEEE Proceedings, 26(6).

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., ... & Thomas, D. (2001). Manifesto for Agile Software Development.

Paulk, M. C., Curtis, B., Chrissis, M. B., & Weber, C. V. (1995). Capability Maturity Model for Software, Version 1.1. Technical Report CMU/SEI-93-TR-024, Software Engineering Institute, Carnegie Mellon University.

Martin, J. (1991). Rapid Application Development. MacMillan Publishing Company.

Rumbaugh, J., Jacobson, I., & Booch, G. (2005). The Unified Modeling Language Reference Manual (2nd edition). Addison-Wesley.

Booch, G., Rumbaugh, J., & Jacobson, I. (1999). The Unified Modeling Language User Guide (2nd edition). Addison-Wesley.