



Development of an Automated Course Allocation Management System: A Case Study of Computer Science Department

¹Usman, Abdullahi Mohammed, ²Ndubuisi, Akachukwu Samuel, ³Idris, Mohammed Kolo, ¹Siyaka, Opotu Hassan

¹Artificial Intelligent Department,

²Cyber Security Department, Federal Polytechnic, Bida, Niger State

³Computer Science Department, Federal University of Technology, Minna, Niger State

ABSTRACT

Efficient course allocation is a critical component of academic institutions, ensuring that lecturers are assigned to courses in alignment with their expertise field. However, manual course allocation processes are often time-consuming, prone to errors, and lack transparency. This project presents the Development of an Automated Course Allocation Management System, designed to streamline the course allocation process and enhance efficiency. The system is developed using HTML, CSS, Bootstrap, PHP, and MySQL, and it provides a web-based platform for administrators and lecturers. The Admin Dashboard allows administrators to manage courses, add lecturers, add courses, and automate the allocation process based on specific criteria, such as the lecturer's area of expertise. The system also includes a Lecturer Dashboard, where lecturers can view their assigned courses. Automating the allocation process reduces human errors, improves accuracy, and saves time, thereby enhancing the overall academic planning process. The system also promotes transparency and fairness in course allocation, as lecturers are matched to courses based on pre-defined rules. This project demonstrates the importance of integrating technology into academic management processes, offering a scalable solution that can be adapted to various institutions to improve course allocation efficiency and satisfaction among academic staff.

ARTICLE INFO

Article History

Received: July, 2024

Received in revised form: July, 2024

Accepted: November, 2024

Published online: December, 2024

KEYWORDS

Automated, Course Allocation, Management, System

INTRODUCTION

The allocation of courses to lecturers is a pivotal aspect of academic administration, directly affecting the quality of education and the efficiency of educational institutions. This process ensures that courses are assigned to the most qualified lecturers based on their expertise, availability, and institutional needs. Traditionally, course allocation was managed through manual methods, involving extensive paperwork, complex scheduling, and subjective decision-making. These methods, while straightforward, often led to inefficiencies, errors, and dissatisfaction among both lecturers and administrators (Becker & Andrews, 2015; Chan, Lee, & Smith, 2018).

Technological advancements have played a crucial role in this transition. Web-based platforms and cloud computing have revolutionized course allocation by providing real-time updates, accessibility, and scalability (Garcia & Lee, 2018; Patel & Kumar, 2020). Mobile applications have further contributed to this evolution by enabling lecturers and administrators to manage course allocations on-the-go (Chen, Lin, & Tan, 2019).

The shift from manual to automated course allocation is not merely a technological upgrade but a strategic move towards modernizing academic administration. By adopting an online course allocation management system, institutions can enhance their operational

Corresponding author: Usman, Abdullahi Mohammed

✉ usmannck@gmail.com

Artificial Intelligent Department, The Federal Polytechnic, Bida, Niger State.

© 2024. Faculty of Technology Education. ATBU Bauchi. All rights reserved



efficiency, improve the quality of course assignments, and ultimately provide a better educational experience for both lecturers and students. This transition aligns with broader trends in educational technology aimed at leveraging digital tools to solve traditional administrative challenges (Lopez, 2021).

The development of automated course allocation systems has been driven by significant advancements in technology. Web-based systems have become increasingly popular due to their accessibility and ease of use. These systems allow users to access course allocation tools from any location with an internet connection, facilitating remote and collaborative work (Garcia *et al.*, 2018). Cloud-based solutions have further enhanced the scalability and reliability of course allocation systems. By leveraging cloud infrastructure, these systems can handle large volumes of data and provide real-time updates, ensuring that allocation decisions are based on the most current information available (Patel *et al.*, 2020). Mobile applications represent another technological advancement in course allocation, offering increased convenience and flexibility for users. These applications enable lecturers and administrators to manage course allocations on the go, improving responsiveness and communication (Chen *et al.*, 2019).

REVIEW OF RELATED LITERATURE

Adams and Williams (2021) in their article *"Implementation of a Weighted Matching Algorithm for Course Allocation"* developed an algorithm to address the complexities of course allocation. Using a quantitative approach involving algorithm development and simulation, the study presents a weighted matching system to enhance fairness in assigning courses to lecturers based on specific criteria such as qualifications, preferences, and workload. A key strength of this research is its novel algorithm, which offers a systematic solution to improve fairness in course allocations. However, the study's limitation lies in its lack of real-world testing, as the results are limited to theoretical simulations without practical application.

Chen, *et al.*, (2019), in their work *"Mobile Applications for Educational Administration: Enhancing Course Allocation"*, explore the role of mobile platforms in improving administrative tasks like course allocation. The authors adopt a qualitative case study methodology, highlighting the benefits of mobile applications in enhancing accessibility and convenience for both lecturers and administrators. The study's strength is in demonstrating how mobile technology simplifies educational administration. However, it has limitations in terms of sample size and its focus on specific mobile platforms, making it less generalizable across various educational settings. Garcia *et al.*, (2018) conducted a comparative study in their article *"Web-Based vs. Desktop-Based Systems in Educational Settings"*, comparing the efficiency and flexibility of web-based systems against traditional desktop-based systems. Their study reveals that web-based systems offer greater flexibility, ease of access, and user engagement, especially in educational administration. However, one notable limitation is that the study does not consider the rise of mobile applications, which have since revolutionized educational management systems.

Jones and Clark (2020) emphasize the importance of transparency and student satisfaction in their article *"Enhancing Transparency in Course Allocation Through Automated Systems."* Using mixed-methods that combine surveys with a system implementation, they examine how automated systems can make the course allocation process more transparent and fairer for all stakeholders. The study's strength lies in its focus on student satisfaction and transparency, key factors often overlooked in course allocation processes. Nonetheless, the reliance on self-reported data introduces a potential bias in their findings, which could affect the overall validity of the conclusions.

In *"A Case Study of Automated Course Allocation at the University of ABC,"* Lee and Nguyen (2019) provide practical insights into the implementation of an automated course allocation system. Through a case study approach, they analyze the challenges and benefits experienced by the institution during and after the



implementation. Their findings highlight the effectiveness of automated systems in managing course allocations and improving administrative efficiency. However, the limitation of this study is its narrow focus on a single institution, limiting the applicability of its findings to broader educational contexts.

Patel *et al.*, (2020), in their article "Cloud-Based Solutions for Scalable Educational Administration," explore the use of cloud technology to support the scalability of course allocation systems in educational settings. Their technical analysis, complemented by a case study, demonstrates the cost-effectiveness and scalability of cloud-based solutions. While the study highlights these strengths, it does not sufficiently address security and privacy concerns, which are critical considerations when using cloud platforms for sensitive educational data.

Lastly, Chan, *et al.*, (2018) in their article "Manual Course Allocation and Its Limitations in Modern Education," use a qualitative observational study and interviews to identify the significant limitations of manual course allocation. They point out issues like workload imbalance, human error, and inefficiency in manual allocation methods. While the study effectively identifies the key shortcomings of manual systems, it lacks quantitative data to back up its findings, which weakens the generalizability of the results.

METHODOLOGY

The methodology adopted in designing this research work are HTML and CSS for design of the interface, the PHP programming language for the operation of the system and MySQL will be used as database. The major aim of using this technology is to develop an interactive user interface and finally ensure that the application is very easy and user friendly.

Description and Analysis of the Existing System

In many institutions, the course allocation process begins with the collection of data regarding available courses, lecturer qualifications, and scheduling constraints. This data is typically gathered through forms or direct

communication and then manually reviewed by administrative staff. The existing system may utilize basic software tools like spreadsheets or standalone applications to assist in managing this information, but often, the process remains largely manual.

Description and Analysis of the Proposed System

The academy task allocation system was designed to make things easier for students each academy semester. Additionally, this method prioritizes the expert field courses that will be taught to students who registered for courses during the semester. Additionally, the approach assists higher ranking lecturers in receiving a minimum number of courses, thereby reducing their workload. Additionally, it helps research lecturers by assigning fewer courses based on the system-created profiles. Additionally, the system will be available online so that administrators and lecturers can access it from anywhere at any time.

System Design

The design of the new system includes the UML diagram, the specifications of the data for input into the system and output from it, Entity Relationship Diagram. It also includes the flowchart and diagrammatic overview of the whole system.

Unified Modeling Language (UML) Diagram

UML (Unified Modeling Language) diagrams as shown in figure 1 are used to visually represent the design of the system. Key UML diagrams for the online course allocation management system include:

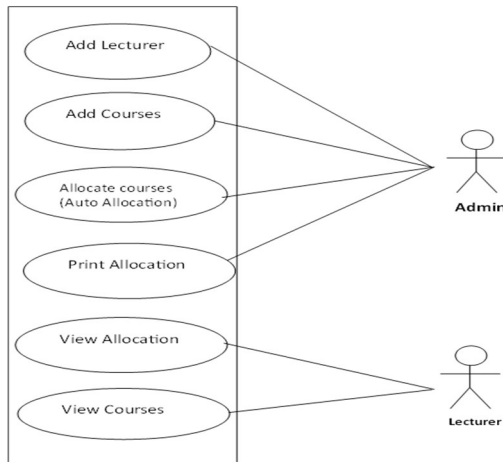


Figure 1 UML Diagram

Use Case Diagram: Illustrates the interactions between users (lecturers, administrators) and the system. It highlights the main functionalities such as Allocate courses, view allocation, Print allocation, viewing courses, adding of new lecturers and courses.

Database design involves creating a schema that efficiently organizes data. The main entities in the database and their relationships are captured in an Entity-Relationship (ER) diagram.

Entities and Relationships:

1. **User:** Represents system users (lecturers, administrators).
2. **Course:** Contains information about courses (course_id, course_name, credits).
3. **Lecturer:** Includes details about lecturers (lecturer_id, name, department).
4. **Allocation:** Tracks course allocations to lecturers (allocation_id, lecturer_id, course_id,).

Normalization: Ensures the database is free of redundancy and maintains data integrity. Tables are normalized to at least the third normal form (3NF). The database schema will consist of several tables related to the courses, lecturers, and allocations.

Database Design

Users Table 1: (To store admin login details)

Column	Data Type	Description
us_id	INT (PK)	Unique identifier for each user (admin)
Username	VARCHAR(100)	Username for the admin
Password	VARCHAR(255)	Password (should be hashed)
Role	VARCHAR(50)	Role of the user, e.g., 'admin'

This table stores information about admin users

Course Table 2: (To store course details)

Column	Data Type	Description
id_c	INT (PK)	Unique identifier for each course
sc_id	INT (FK)	Foreign key linking to the scope table (lecturer's area of specialization)
id_s	INT (FK)	Foreign key linking to the staff table (lecturer responsible for the course)
title2	VARCHAR(255)	Course title
Code	VARCHAR(100)	Course code
Unit	INT	Course unit load
L	INT	Number of lecture hours
P	INT	Number of practical hours
Ch	INT	Total credit hours
Lab	BOOLEAN	Indicator if the course has a lab component
lecturer	VARCHAR(255)	Name of the assigned lecturer (optional, may derive from staff table)

Corresponding author: Usman, Abdullahi Mohammed

✉ usmannck@gmail.com

Artificial Intelligent Department, The Federal Polytechnic, Bida, Niger State.

© 2024. Faculty of Technology Education. ATBU Bauchi. All rights reserved

This table stores information about courses, including lecturer assignment through foreign keys.

Staff Table 3: (To store lecturer details)

Column	Data Type	Description
id_s	INT (PK)	Unique identifier for each lecturer
sc_id	INT (FK)	Foreign key linking to the scope table for lecturer's area of specialization
uname	VARCHAR(100)	Username for lecturer login
Pass	VARCHAR(255)	Password for lecturer login (should be hashed)
Title	VARCHAR(50)	Academic title (e.g., Dr., Prof.)
fname	VARCHAR(100)	Lecturer's first name
Lname	VARCHAR(100)	Lecturer's last name
Area	VARCHAR(255)	Lecturer's area of expertise
Email	VARCHAR(100)	Lecturer's email address

This table stores information about lecturers, who are assigned to courses.

Scope Table 4: (To store areas of specialization for lecturers)

Column	Data Type	Description
sc_id	INT (PK)	Unique identifier for each area of specialization
Title	VARCHAR(255)	Title or name of the specialization

This table stores various specializations or subject areas for lecturers, and is referenced in both the course and staff tables

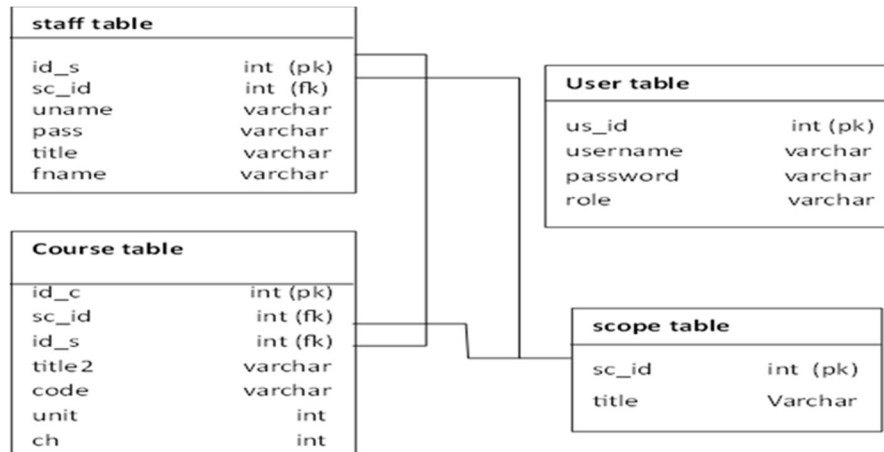


Figure 2 Entity Relationship Diagram

Entity-Relationship (ER) Diagram Overview:

1. **Users Table:** Holds information about admin users who manage the system.

2. **Staff Table:** Contains information about lecturers.
 3. **Scope Table:** Holds the area of specialization for lecturers and is used to match courses with appropriate staff.
 4. **Course Table:** Stores course information and links to lecturers and areas of specialization.
- The **course** table also references the **scope** table through the **sc_id** foreign key to ensure the course aligns with the lecturer's area of specialization.
 - The **staff** table references the **scope** table via **sc_id** to indicate the lecturer's specialization.

Relationships:

- The **course** table references the **staff** table through the **id_s** foreign key to assign a lecturer to a course.

Workflow Design

The workflow design defines the processes of interactions within the system as shown in figure 3.

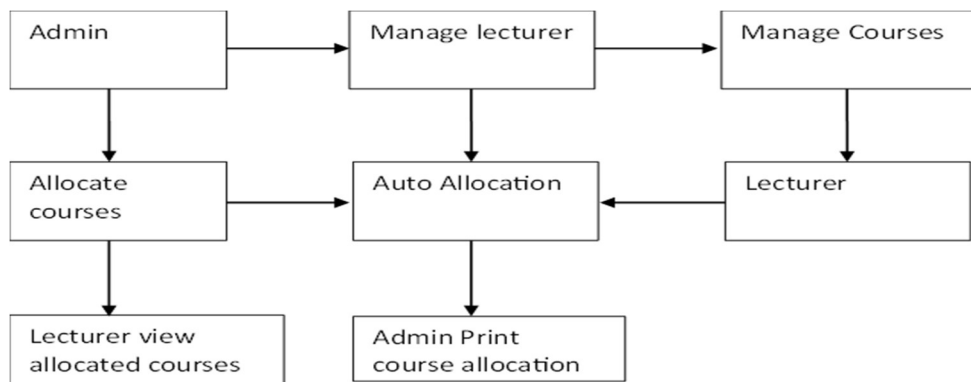


Figure 3: Workflow Diagram

Course Allocation:

1. Administrator logs in.
2. Administrator allocate course to lecturers.
3. Administrator can mange lecturers, that is to add new lecturer
4. Administrator can also manage courses, that is to add new course

Lecturer View:

1. Lecturer logs in.
2. System retrieves the lecturer's assigned courses from the database.
3. Lecturer views their schedule and course details on their dashboard.

System Architecture

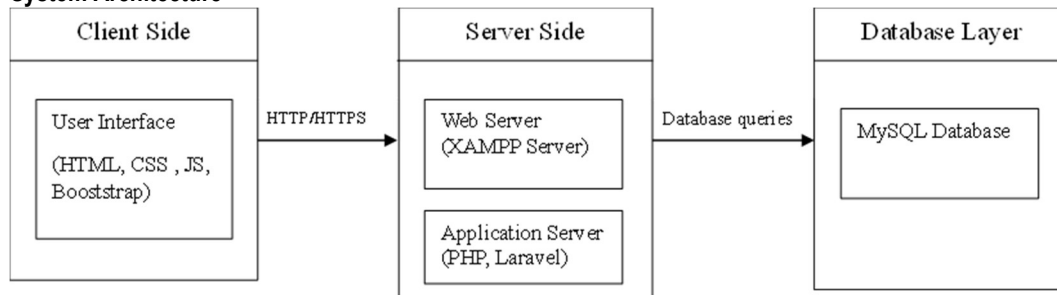


Figure 4: System Architecture

The proposed Online Course Allocation Management System is designed with a layered architecture, ensuring separation of concerns, scalability, and maintainability as shown in figure 4. The frontend provides an interactive user interface, while the backend handles business logic, authentication, and data management. This architecture ensures a robust, efficient, and user-friendly system for managing course allocations in an academic setting.

System Requirements

Every system needs some components, both hardware and software, that will enable it perform its primary tasks. These components needed by the system before it can run successfully is known as the system requirements. The following are the list of system required.

The computer to install and run the application should have a minimum of Intel Pentium IV and above processor, 1.3 GHz processor speed, 4GB RAM, 50 GB Hard Drive and 15" SVGA Colour Monitor as the hardware requirement while the software required for the implementation of the new system are Windows 7 or higher, XAMPP, Web Browser and Antivirus Program.

System Flowchart

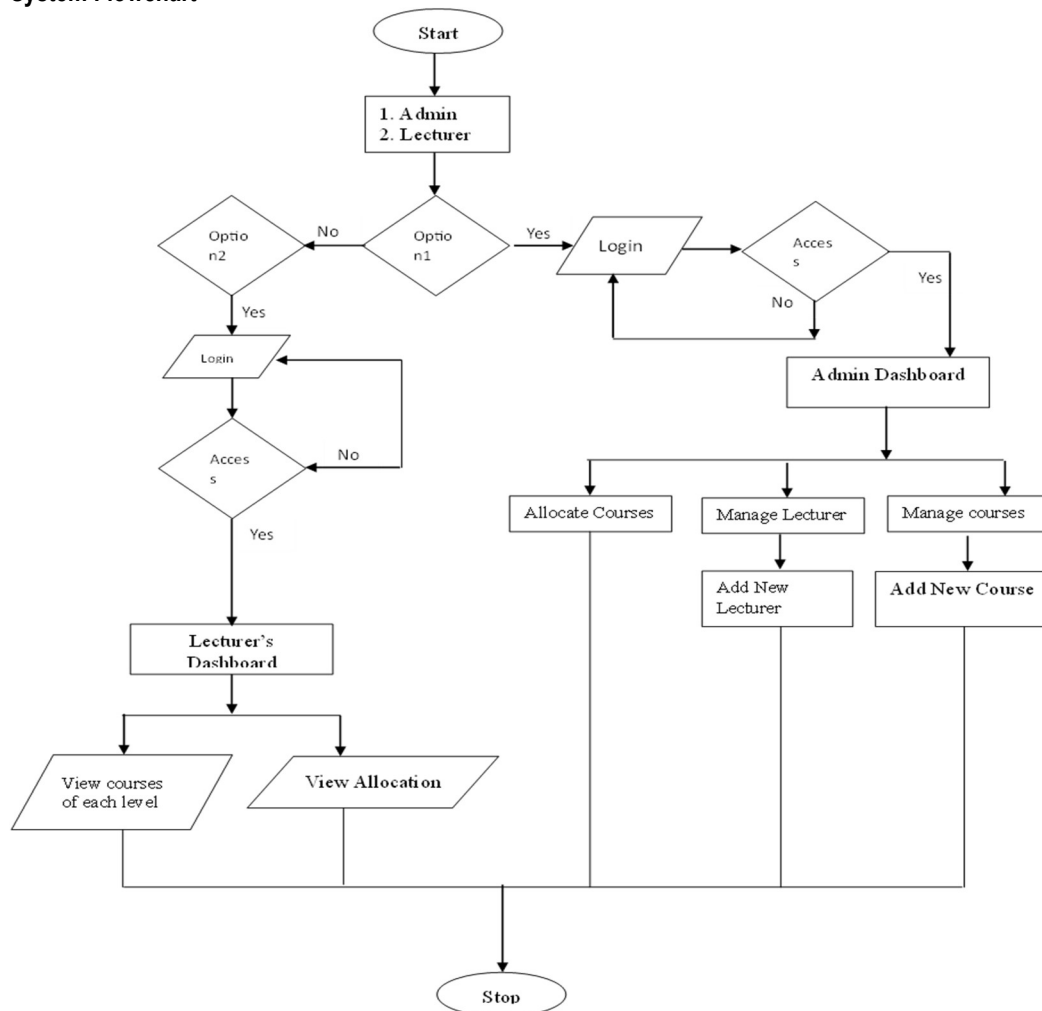


Figure 5: System Flowchart

The system flowchart provides a clear and structured overview of the processes involved in the Online Course Allocation Management System as shown in figure 5. It delineates the pathways taken by administrators and lecturers from login to performing specific tasks like managing courses, managing lecturers, allocating courses, viewing allocations, and updating availability. Each decision point and process ensures that the system operates smoothly and efficiently, providing a user-friendly experience while maintaining data integrity and security.

SYSTEM IMPLEMENTATION

During the system's development, several key features were implemented. The Admin Dashboard allows administrators (course managers) to input course and lecturer details, set allocation criteria. The Lecturer Dashboard enables lecturers to view their assigned courses, and also view the courses of each level. The system simplifies course management, reduces human error, and saves time.

Corresponding author: Usman, Abdullahi Mohammed

✉ usmannck@gmail.com

Artificial Intelligent Department, The Federal Polytechnic, Bida, Niger State.

© 2024. Faculty of Technology Education. ATBU Bauchi. All rights reserved

System Interface

The system main interface Home Page as shown in figure 6 is the main entry point of the system and offers users a simple, intuitive

interface to access different functionalities based on their roles (Admin or Lecturer).

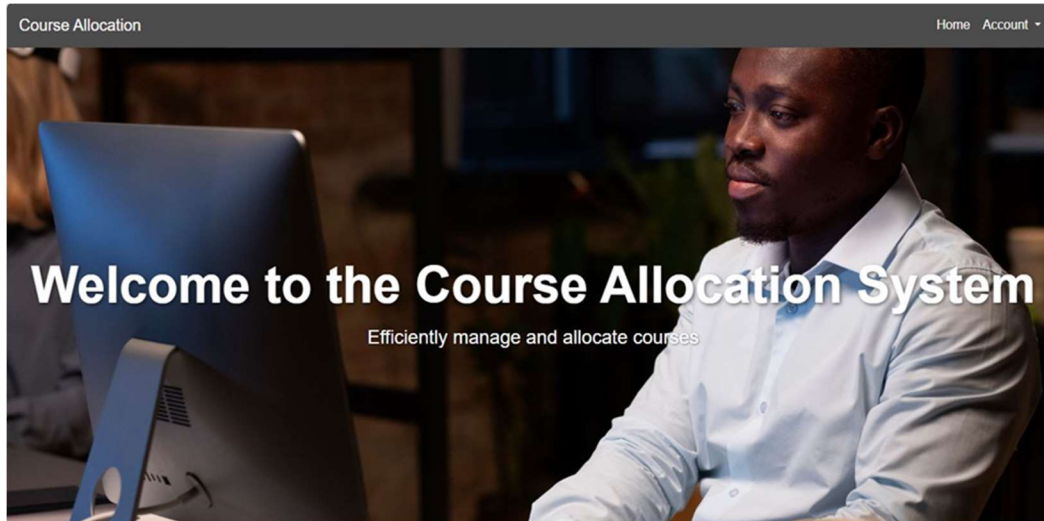


Figure 6: Home Page

The Admin Dashboard as shown in figure 7 provide an interface for an automated course allocation management system typically includes the following features:

1. Auto Allocate Course: This feature allows administrators to automatically assign lecturers to courses based on predefined criteria. With a simple click of a button, the system uses algorithms to ensure optimal allocation, saving time and reducing human error.
2. Reset Course: The "Reset Course" feature enables administrators to clear

or reset the current course allocation settings. This is useful if changes are needed, such as re-assigning courses or modifying lecturer schedules. It ensures that the system can be quickly restored to its original state for adjustments or updates.

3. Print Course Allocation: This feature allows administrators to generate and print a detailed report of course allocations. It typically includes information about which lecturers are assigned to specific courses, course schedules, and any other relevant data.

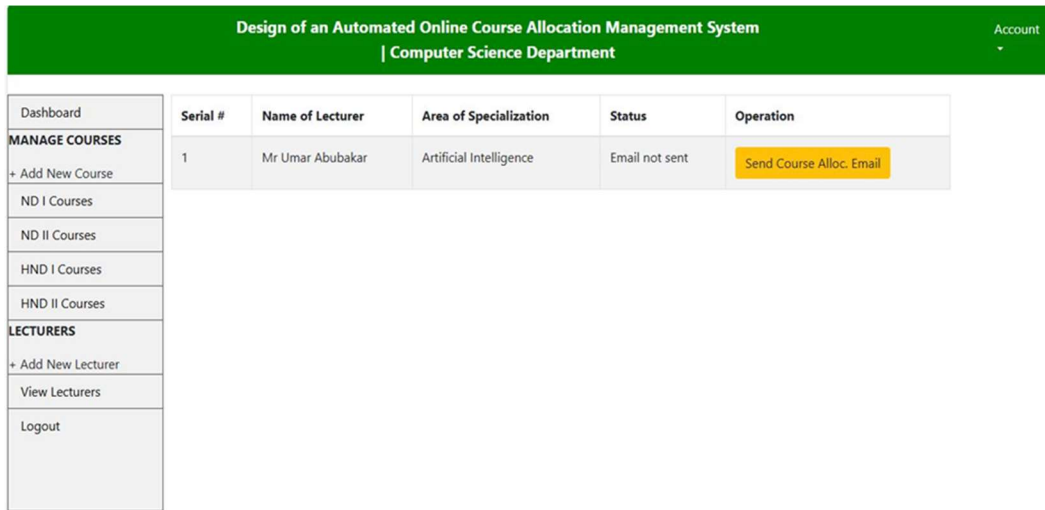


Figure 7: Admin Dashboard

The **Admin View Lecturer** (To View the Lecturer Assigned to a Specific Course) this feature allows administrators to view which lecturers have been assigned to specific courses

as shown in figure 8. This functionality is essential for managing and monitoring the allocation process, ensuring that the right lecturer is assigned to each course based on area of specialization.

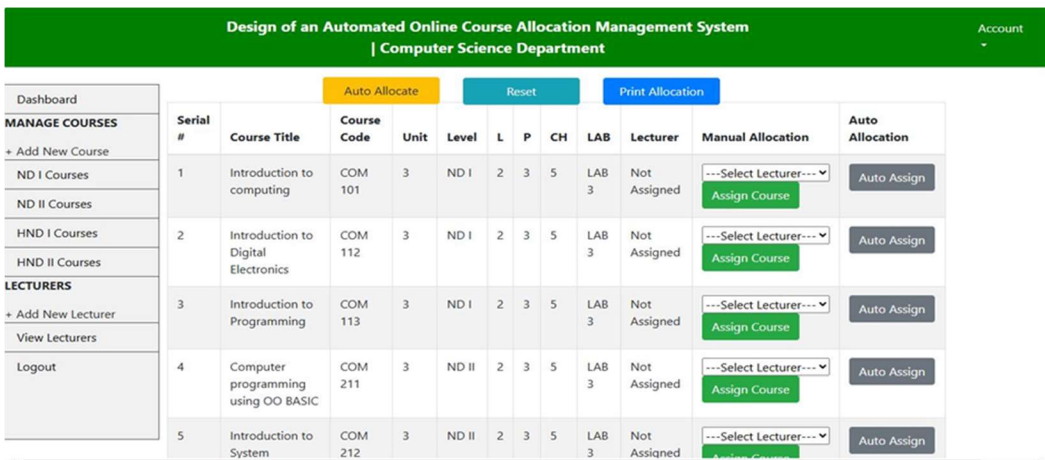


Figure 8: Admin View Lecturer

The **Admin Add Lecturer** feature in the automated course allocation system allows administrators to add new lecturer information into the system as shown in figure 9. This is crucial for managing lecturer data and ensuring that the system remains up-to-date with current staff. The

feature typically includes the following key fields and functionalities:

Key Elements of Admin Add Lecturer Interface:

1. **User Name:** Administrators can input a unique username for the lecturer. This is typically used for system login and identification.

2. **Password:** The admin can set a password for the lecturer to allow access to the system. This ensures secure access for each user and protects sensitive information.
3. **Title:** This field allows the admin to select or enter the lecturer's title, such as Dr., Mr., Mrs., etc. This adds a professional designation to the lecturer's profile.
4. **First Name:** The admin can input the lecturer's first name, which is used to personalize communication and ensure correct identification.
5. **Last Name:** The last name or surname of the lecturer is entered here to complete their full name.
6. **Area of Specialization:** The admin can specify the lecturer's area of expertise or academic focus (e.g., Computer Science, Mathematics, Chemistry). This is essential for ensuring that the lecturer is assigned to appropriate courses.
7. **Email:** The email address of the lecturer is recorded, enabling communication with the lecturer about course assignments, schedules, or other important updates.

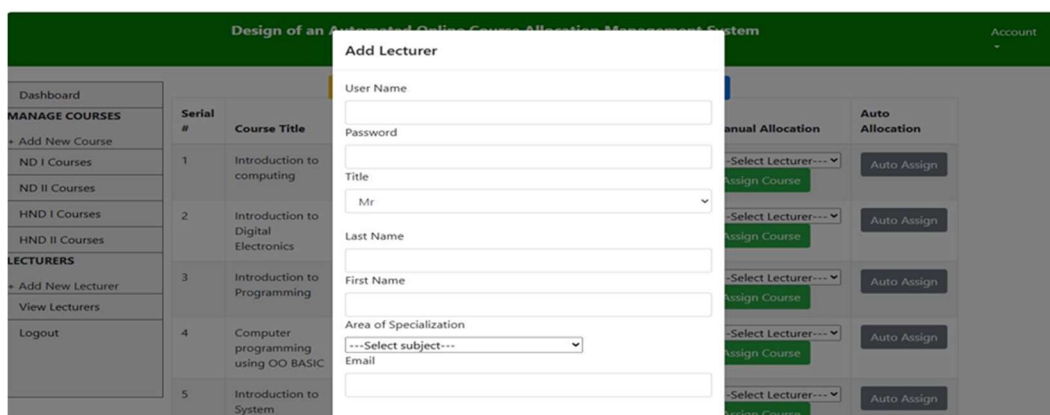


Figure: 9: Admin Add Lecturer

The **Admin Add Course** feature in an automated course allocation system allows administrators to add new courses to the system as shown in figure 10. This functionality is essential for keeping the course catalog up to date and ensuring that course allocation can be accurately managed. The feature typically includes the following key fields:

Key Elements of Admin Add Course Interface:

1. **Course Title:** The admin enters the title or name of the course (e.g., "Introduction to Computer Science" or "Advanced Mathematics"). This is the primary identifier for the course.
2. **Course Code:** A unique code for the course (e.g., "CS101" or "MTH303") is entered. The course code helps in identifying the course in the system, preventing confusion with other similar courses.
3. **Course Unit:** The admin specifies the number of units the course carries (e.g., 3 units, 4 units). This field helps in calculating workload and determining the credit value of the course.
4. **Course Scope:** This field provides a description or summary of the content covered in the course. It gives both students and lecturers a clear idea of the subject matter and objectives of the course.

5. **L (Lectures):** The admin specifies the number of hours per week dedicated to lectures for the course (e.g., 3 hours). This helps in planning and scheduling the course and understanding its time commitment.
6. **P (Practical Sessions):** This field is used to enter the number of hours per week allocated to practical or tutorial sessions for the course (e.g., 2 hours). This is important for courses that require hands-on learning or lab work.
7. **CH (Contact Hours):** Contact hours represent the total number of hours spent in direct interaction with the lecturer, both during lectures and practical sessions. This field combines the lecture and practical session hours.
8. **LAB:** The admin enters the number of hours dedicated specifically to laboratory work if applicable (e.g., for science or engineering courses). This helps in ensuring that lab sessions are scheduled and resources are allocated properly.

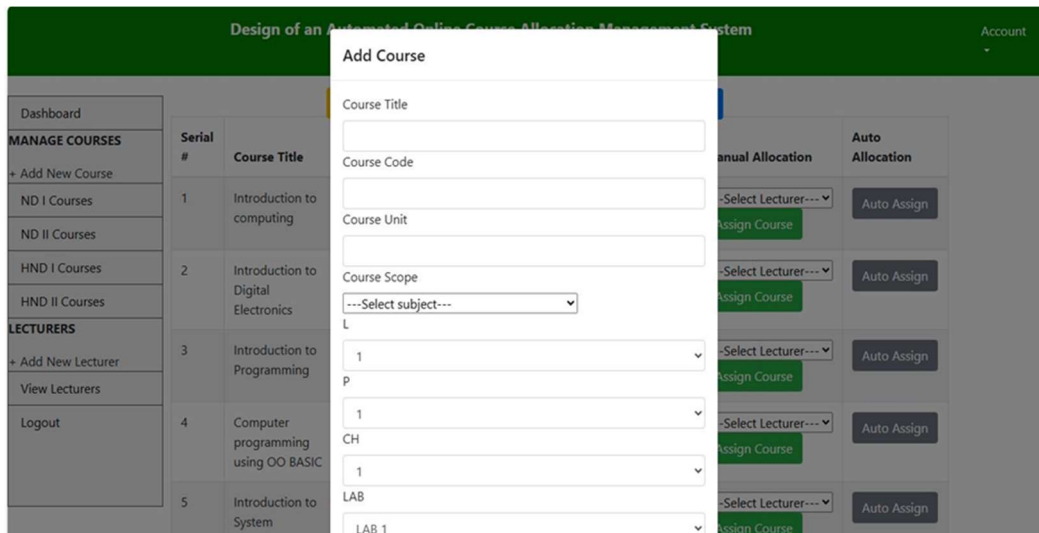


Figure 10: Admin Add Course

CONCLUSION

The implementation of the Automated Course Allocation Management System marks a significant step forward in streamlining the course allocation process in educational institutions. By automating tasks such as assigning courses to lecturers based on workload, specialization, and availability, the system ensures transparency, efficiency, and fairness in course management. The web-based platform provides both administrators and lecturers with easy access to course allocation information, making it more

accessible and reducing the burden of manual processes.

The system's ability to store, manage, and allocate data in real-time allows for quick decision-making and enables better management of resources within an academic environment. Overall, this system achieves its goal of improving efficiency, eliminating biases in course allocation, and providing a more structured approach to managing academic responsibilities.

RECOMMENDATIONS

The Automated Course Allocation Management System was developed to address inefficiencies in the manual process of assigning



courses to lecturers. The system automates the course allocation process based on predefined criteria such as lecturers' areas of specialization, availability, and workload, leading to more effective and fair distribution of teaching responsibilities. The system was designed using modern web technologies, including HTML, CSS, Bootstrap, PHP, and MySQL, to ensure a user-friendly interface and smooth interaction between the system and its database. Base on the listed advantages of the system over the previous method, the system is here by recommended to the institution for efficiency and effective work process.

REFERENCES

- Adams, J., & Williams, K. (2021). Implementation of a weighted matching algorithm for course allocation. *Journal of Educational Technology*, 15(3), 145-156.
- Becker, M., & Andrews, L. (2015). Historical perspectives on course allocation methods in higher education. *Educational Management Quarterly*, 22(3), 112-127.
- Chan, T., Lee, H., & Smith, R. (2018). Manual course allocation and its limitations in modern education. *Journal of Academic Administration*, 12(2), 88-97.
- Chen, Y., Lin, S., & Tan, J. (2019). Mobile applications for educational administration: Enhancing course allocation. *International Journal of Mobile Computing*, 22(4), 267-278.
- Garcia, M., & Lee, P. (2018). Web-based vs. desktop-based systems in educational settings. *International Journal of Educational Technology*, 13(4), 255-268.
- Johnson, S. (2016). The Time-Consuming Nature of Manual Processes. *Administrative Studies Quarterly*, 11(4), 58-72.
- Jones, D., & Clark, E. (2020). Enhancing transparency in course allocation through automated systems. *Journal of Higher Education Management*, 36(1), 100-115.
- Lee, J., & Nguyen, P. (2019). A case study of automated course allocation at the University of ABC. *Education Management Review*, 29(4), 309-324.
- Lopez, A. (2021). Scalability of Automated Systems. *Journal of Systems Architecture*, 17(1), 90-102.
- Patel, R., & Kumar, S. (2020). Cloud-based solutions for scalable educational administration. *Computing in Education*, 22(7), 441-458.