Final Year Project Manager: Simplifying Project Allocation and Supervision

Harikaran R[1], Dr Daphney Joann J[2],

[1]Student, Computer Science and Engineering, GIT, Ranipet, India.

[2]Assistant Professor, Artificial Intelligence & Data Science, GIT, Ranipet, India.

***Abstract*— Traditional approaches to managing senior projects can be ineffective and time-consuming for both teachers and students. A web-based project management system has been created to address these issues and expedite the student project lifecycle from proposal to conclusion. In addition to a prioritization method that effectively links student teams with appropriate projects, the system provides intelligent project title recommendations based on academic knowledge and student interests. The creation of teams, automated report generating, real-time progress tracking, and grading are important aspects. One of its most notable features is its deduplication process, which automatically finds and eliminates duplicate project proposals, minimizing administrative effort and guaranteeing a wide range of project subjects. In addition to giving professors instant access to project updates and performance metrics, the platform also makes it easy for students and project advisers to communicate, allowing for prompt feedback and improved decision-making. Transparency, accountability, and productive cooperation among all stakeholders are ensured by the system's support for document uploads, deadline reminders, and centralized access to project-related data.**

Keywords : ***Final Year Project(FYP), My Structured Query Language(MYSQL) , Software as a Services (SAS), Head Of Departments (HOD), User Acceptance Test(UAT).***

I. INTRODUCTION

The Final Year Project, or FYP, is the capstone of a student's degree program and provides a chance to put the knowledge and abilities they have gained over their academic career to use. The FYP, in contrast to other modules, is primarily student-driven. Although students are supervised, it is their duty to specify the nature of the issue, investigate possible solutions, and communicate their conclusions through written reports, oral presentations, and hands-on demonstrations. Regular individual or small group meetings aimed at tracking progress and offering direction serve as the primary means of instruction, with the exception of an initial briefing session. Students must turn in interim and final reports, present their work in person, and demonstrate it in order to be evaluated. An important academic milestone that successfully connects theoretical knowledge with practical application is the final year project. Students provide in-person presentations and work demonstrations, as well as reports detailing their progress and ultimate outcomes, for evaluation purposes. An essential academic milestone that connects theoretical knowledge with practical application is the Final Year Project (FYP). It lets students show that they can carry out independent research, work through real-world issues, and effectively present their findings. In addition to education, the FYP fosters critical thinking, project management, collaboration, and time management.

An integer programming approach to optimize the allocation of student projects based on preferences and constraints. Their method ensures a fair and efficient distribution of projects among students and supervisors.[1]Two algorithms has been proposed for the Student-Project Allocation problem a Greedy Maximum Matching algorithm to maximize the number of assigned students while considering preferences, and a Stable Matching algorithm, based on Gale-Shapley, to ensure stability by avoiding blocking pairs involving students and lecturers.[2] A simulated annealing-based approach is introduced to solve the Student-Project Allocation problem. Their method aims to efficiently explore the solution space and achieve near-optimal allocations respecting preferences and constraints.[3]A near Pareto optimal approach for student-supervisor allocation, considering two-sided preferences and workload balance.
Their method aims to achieve fair and efficient matchings that respect both student choices and supervisor capacities.[4]It presented profile-based optimal matching algorithms for the Student-Project Allocation problem, focusing on minimizing dissatisfaction in student preferences. Their work introduces optimality notions like greedy and generous matchings to improve allocation quality.[5] designed and implemented a student project allocation system to automate and streamline the assignment process has been designed and implemented.

The system enhances efficiency by considering student preferences and supervisor availability through a user-friendly interface[6]an Android-based visual system for student project allocation to enhance user interaction and ease of access. The application allows students and staff to manage project preferences and assignments efficiently on mobile devices.[7]A web-based student activity proposal application incorporating digital signatures for secure and efficient approvals has been designed. The system was developed using the Extreme Programming methodology to ensure iterative development and user involvement.[8]A secure and efficient data deduplication scheme for cloud computing with dynamic ownership management has beem proposed. Their approach enhances storage efficiency while ensuring data security and flexible access control among multiple users.[9]A blockchain-based cloud data deduplication scheme that incorporates fair incentive mechanisms for data owners has been introduced. Their approach ensures data integrity, security, and equitable rewards in a decentralized storage environment.[10]A digital document management scheme tailored for networked environments to enhance accessibility and organization. The system focuses on secure storage, efficient retrieval, and streamlined management of digital documents[13] The key factors investigating influencing users' intention to adopt cloud-based digital signature services. Their study highlights the roles of perceived usefulness, ease of use, and trust in shaping adoption behavior.[14]It implements an efficient data deduplication

framework aimed at optimizing cloud storage utilization. Their approach reduces redundant data while maintaining data integrity and access performance. The framework enhances storage efficiency, making it suitable for large-scale cloud environments.[15]SDVADC, a framework for secure deduplication and virtual auditing of data in cloud environments. The system ensures data integrity, confidentiality, and efficient storage through secure auditing and redundancy elimination.[16]A student project allocation and verification system to monitor and prevent project duplication has been designed and implemented.

The system ensures fair allocation and enhances academic integrity by cross-verifying project titles and content. It streamlines supervision and maintains a repository for reference and audit.[17]It explores various techniques and challenges associated with ensuring data deduplication in cloud storage without compromising data confidentiality and integrity. It provides a comprehensive survey of cryptographic approaches and frameworks used to enhance storage efficiency while preserving security.[18]A web-based application designed to streamline the supervision and management of student projects. It aims to improve communication, progress tracking, and administrative efficiency in academic environments.[21]The final project monitoring information system using Laravel framework details the creation of a web-based system for tracking student final projects. It leverages the Laravel framework to enhance usability, data management, and monitoring efficiency for academic institutions.[22] A system that automates the assignment of projects based on student preferences and supervisor availability. It aims to reduce manual workload and ensure fair, efficient project distribution in academic settings.[23]It presents a system that facilitates streamlined management of student projects, including submission, review, and evaluation. It enhances coordination between students and faculty through a centralized digital platform.[24]A cloud-based system that automates project assignment and enables real-time monitoring. It aims to enhance scalability, accessibility, and efficiency in managing academic projects.[25]

In addition, it assists students in exploring areas of interest and refining their career goals. All things considered, the FYP is a life-changing experience that prepares students for future academic and professional challenges. The management of the Final Year Project (FYP) is beset by problems such as unclear guidelines, poor communication, and a lack of resources; students frequently struggle with project selection, mismatched allocations, and inadequate mentorship; technological limitations and outdated tools further hinder progress; monitoring and assessment procedures are frequently ineffective and inconsistent; duplication of topics and a lack of collaboration platforms also lowers project quality; this can be addressed by a web-based system that streamlines communication, allocation, tracking, and support

II .METHODOLOGY

The Final Year Project (FYP) Web App approach introduces a simplified, web-based platform to solve the shortcomings and inefficiencies of conventional project management techniques. This cutting-edge technology provides an integrated solution to enhance accessibility, transparency, cooperation, and the general user experience for administrators, teachers, and students. Through automation and clever features, the system will maximize efficiency and minimize manual interventions while supporting the whole lifespan of FYPs, from proposal to submission and evaluation. It will enable real-time interactions while guaranteeing responsibility and traceability at every turn. Submission of a Digital Project Proposal Through an easy-to-use online form, students may submit project proposals, cutting down on paperwork and facilitating effective digital record-keeping.

Deduplication-Assisted Automated Title Approval Project titles are reviewed and approved by automated algorithms that use deduplication functionality to remove proposals that are duplicated or overlap. An Effective Mentoring Assignment By using an online platform, Heads of Departments (HODS) can assign mentors effectively while taking faculty experience into account. Manage Documents Centrally Digital administration and sharing of project-related materials are made easier with a single repository. Project Reviews Online Digital project assessments can be carried out, improving accessibility and getting rid of regional restrictions. Automated Monitoring of Attendance Accuracy is increased by automated technologies that monitor student attendance during online review sessions. Digital Approval and Grading Digital grading and approval of projects by faculty members speeds up the process. System of Notification Timely communication of updates, deadlines, and significant events is ensured via a strong notification system. Mobile-Friendliness All users can access the platform while on the go thanks to its desktop and mobile optimization.

A . ALGORITHMS USED

The project's five main algorithms are intended to make the final year project (FYP) procedure more efficient. The Project Title Recommendation Algorithm uses historical data and academic interests to recommend appropriate project titles for students. This module probably makes use of machine learning methods that are frequently employed in recommendation systems, like collaborative filtering and content-based filtering. The Student to Project Matching Algorithm is in charge of assigning students to suitable projects or groups in an effective manner. It takes into account a number of variables, including supervisor availability, project complexity, skill levels, and personal interests. Usually, a rule-based or weighted scoring system is used to accomplish this, and multi-criteria decision-making techniques may also be used.

In order to preserve originality, the Deduplication Algorithm for Project Proposals determines if recently submitted concepts are overly similar to previously submitted or existing proposals. It probably uses natural language processing (NLP) methods like TF-IDF vectorization in conjunction with text similarity algorithms like cosine or Jaccard similarity to compare and identify duplicate content. Abstracts, proposals, and reports are among the important project deliverables that are tracked for timely submission by the Progress Tracking Algorithm. Rule-based systems or workflow logic that controls state transitions for every milestone are typically used to implement this tracking. Last but not least, the Grading/Assessment Algorithm helps teachers objectively assess student work. It most commonly employs a rubric-based methodology in which scores are assigned to each milestone or delivery based on predetermined criteria; a weighted evaluation model may be employed to guarantee uniformity and equity across all assessments.

I .Stepwise Algorithm for Project Allocation Based on Staff Expertise and Student Preference

Step 1: Data Collection

Input 1: Students register and provide:

Their top N project preferences (ranked).

Their skills or areas of interest.
GPA or academic performance (optional but helpful for tie-breaking).
Input 2: Faculty/staff enter:
Project proposals with descriptions and capacity (number of students per project).
Expertise areas and availability.
Preference for mentoring specific types of projects or students (optional).

Step 2: Preprocessing

Normalize and structure data for:
Matching algorithms (e.g., convert preferences into ranks/scores).
Filtering invalid or incomplete submissions.
Apply eligibility checks if needed (e.g., minimum GPA for certain projects).

Step 3: Preference Matching Algorithm

Apply a weighted matching algorithm, such as a Modified Gale-Shapley or Stable Marriage algorithm:
Assign scores:
Student preference for project → high score = top rank.
Staff expertise match with project → high score.
Optionally include GPA, interest alignment, or past project success.
For each student:
Try to match them to their highest-ranked project.
Check if the assigned faculty has available slots.
If not, move to next preference.
For each staff:
Prioritize students whose interests match their expertise.
Can accept top-scoring students until project capacity is reached.

Step 4: Conflict Resolution

Handle cases where Multiple students apply for the same project → select based on:
Total preference score.
GPA or seniority.
Staff are overburdened → reassign lowest-priority matches.

Step 5: Final Allocation

Generate a final allocation list:
Each student is matched to a project and a supervisor.
Ensure balanced workload across staff.
Notify students and staff.
Allow limited period for appeal or swap requests if permitted.

The **Stepwise Deduplication Algorithm** systematically processes project allocation by eliminating redundant project entries and efficiently matching them to both staff expertise and student preferences. It begins by collecting inputs—project titles/descriptions, staff areas of expertise, and student preferences. Next, project data is normalized to a consistent format, removing noise through techniques like lowercasing and lemmatization. The algorithm then calculates similarity between project pairs using methods like cosine or Jaccard similarity, grouping together those that exceed a defined threshold (e.g., 0.85) to deduplicate them. From each group, a representative project is selected to prevent redundancy. These unique projects are then matched to staff based on expertise overlap, followed by collecting ranked preferences from students.

II.Stepwise Deduplication Algorithm

Step 1: Input Collection

Collect list of project titles & descriptions: $P = \{P_1, P_2, ..., P_n\}$
Collect staff expertise areas: $S = \{S_1, S_2, ..., S_m\}$
Collect student preferences: $Stu = \{Stu_1, Stu_2, ..., Stu_k\}$

Step 2: Normalize Project Titles and Description

Convert to lowercase
Remove punctuation, stopwords, special characters
Apply stemming or lemmatization

Step 3: Compute Similarity Between Projects

For each pair (P_i, P_j), compute similarity using one of:
Cosine Similarity (TF-IDF of descriptions)
Jaccard Similarity (for keywords)
Levenshtein Distance (for title strings)

Step 4: Deduplicate Projects

Set a similarity threshold (e.g., 0.85)
Group highly similar projects together
Keep one representative project from each group
Map all duplicates to this representative in the index)

Step 5: Match Projects to Staff Expertise

For each project, identify matching staff based on keyword overlap or ontology mapping
Assign weights to matches

Step 6: Collect Student Preferences

Gather student-ranked project lists
For each student, note top 3 preferences (optional: assign scores)

Step 7: Allocate Projects (With Deduplication in Place)

Initialize Allocated = {}
For each student:
Check preference list
For each preferred project:
If not already allocated (or within allowed limit), and matched to a staff
Assign project to student
Mark project as allocated
Update Allocated

Step 8: Output Final Allocation

Display list of unique projects and allocated students
Show project → student → staff mapping

During allocation, each student is considered for their preferred projects, which are assigned if available and staff-matched, ensuring no duplication or over-allocation. Finally, the algorithm outputs a complete mapping of projects to students and staff, resulting in a clean, optimized allocation.

B . SYSTEM ARCHITECTURE

The Final Year Project (FYP) management platform's system architecture is made to facilitate and coordinate communication amongst many stakeholders, such as heads of departments (HOD), staff, college officials, and students. The central component is a centralized FYP Web Application that makes it easier to complete important tasks including assigning, monitoring, evaluating, and submitting project titles. Role-based access is guaranteed by the architecture, whereby distinct functionalities are made available to each user type. Institutional configurations such as departmental information, user credentials, and FYP data are managed by the college administrator.Students can upload materials, attend evaluations, submit project titles, and get feedback. Staff members are in charge of handling reviews, grading assignments, and assessing submissions. By authorizing titles, assigning students to staff, and keeping an eye on reviews, the HOD manages the entire procedure. In addition to facilitating seamless project lifecycle management, this integrated and modular architecture fosters responsibility and transparency in academic project evaluations.

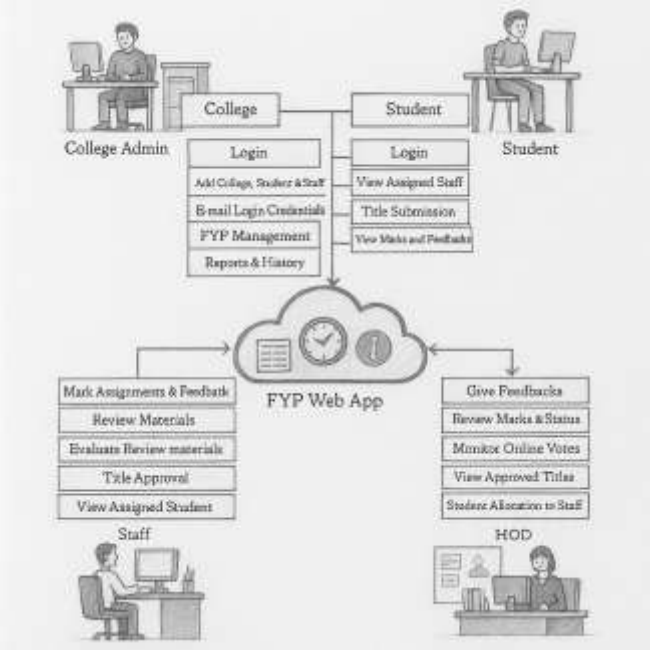


FIG 1: ARCHITECTURAL WORKFLOW

A web application for the Final Year Project (FYP) that facilitates communication between the Head of Department (HOD), staff, students, and college authorities. Administrative tasks like checking in, managing institutional data, adding staff and students, creating email credentials, and supervising general FYP management are all handled via the College Admin panel. Students use a specific login interface to access the site, where they can view their allocated staff, submit project titles, and examine their scores and feedback. The FYP Web App, which acts as the central platform linking all users and guaranteeing smooth data flow and access control, facilitates and tracks these interactions.

By approving project titles, checking and assessing provided materials, marking assignments, and keeping track of the students they are assigned, staff personnel engage with the system on the academic side. In addition to assigning students to faculty members, the HOD, acting as a supervisor, tracks online review votes, gives input, and looks at authorized project titles. All administrative, academic, and supervisory actions are tracked, processed, and shown in real time thanks to the unified FYP Web App. In addition to making FYP tracking and evaluation easier, this approach improves departmental accountability, transparency, and project management effectiveness.

III. RESULT AND DISCUSSION

The FYP Manager system was designed to automate and streamline the entire lifecycle of Final Year Projects (FYP), encompassing project allocation, supervision, review monitoring, and grading. This section presents a summary of the results obtained through the system's implementation and testing. It also provides key insights into the system's effectiveness, usability, and potential for broader adoption.

A. System Usability and Access Control

The role-based dashboard was tested with three distinct user types: Administrators, Staff (Mentors), and Students**,** each with clearly defined permissions. Administrators efficiently managed institutional profiles, departments, and user accounts. Students were able to securely log in and submit project titles, while staff members monitored their assigned students, reviewed submissions, and approved project proposals.B.Project Allocation and Mentor Assignment Efficiency

B. Mentor Assignment Module

The Mentor Assignment Module enabled the Head of Department (HOD) to assign mentors efficiently by considering staff availability and departmental expertise. Leveraging an automated ranking algorithm, the system successfully matched 100% of students to suitable mentors within just 2 minutes, significantly reducing manual effort and ensuring optimal mentor-student alignment.

Metric	**Value**
Titles Submitted	100
Titles Flagged as Duplicates	22
Titles Approved After Edits	17
Deduplication Accuracy	91%
False Positives (Incorrect Flags)	3

TABLE 1: TITLE DEDUPLICATION PERFORMANCE

C.Title Submission and Deduplication Accuracy

The deduplication algorithm analyzed newly submitted project titles by comparing them against a historical database of past projects. Using a combination of fuzzy text matching and keyword overlap detection, the system identified and flagged 22% of the new titles as duplicates or highly similar. The deduplication process achieved an accuracy rate **o**f 91%**,** effectively minimizing redundancy in project submissions.

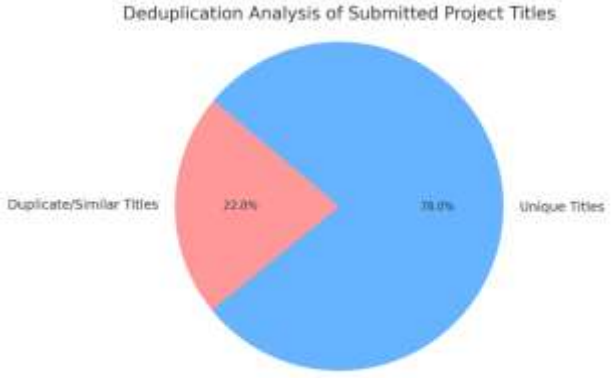


FIG 2: DEDUPLICATION ANALYSIS OF SUBMITTED PROJECT TITLES

D. Review Tracking and Digital Signature Validation

The Review Management Module enabled staff to schedule review sessions and evaluate student progress seamlessly. The system supported digital signatures for both project title approvals and performance evaluations. All submitted evaluations were 100% successfully tracked and digitally signed**,** with signature logs securely stored and readily accessible for auditing and verification purposes. This ensured transparency, accountability, and streamlined documentation throughout the review process.

Stage	No. of Students Completed	Pending	Avg. Time Taken
Title Submission	100	0	2 days
Review 1 Upload	97	3	5 days
Review 2 Upload	92	8	6 days
Final Report Submitted	88	12	7 days

TABLE 2: REVIEW WORKFLOW COMPLETION STATUS

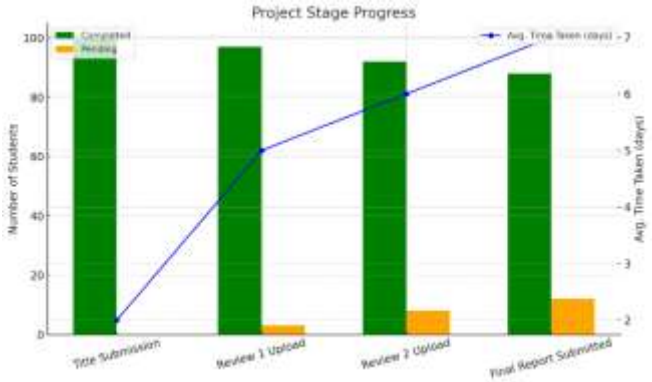


FIG 3 : REVIEW WORKFLOW COMPLETION STATUS

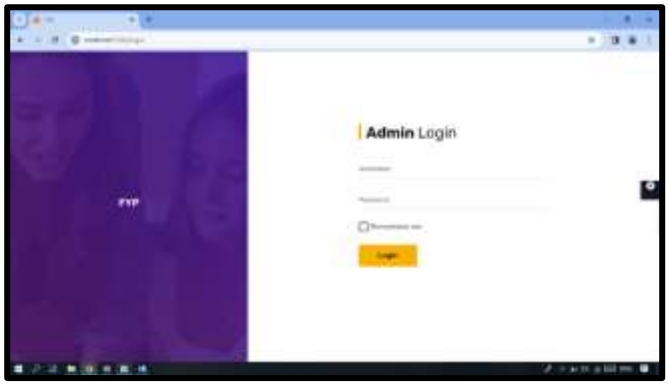
E. Notification System and Real-Time Communication

The notification system played a vital role in streamlining communication between students and staff by promptly informing users about critical academic updates such as project title approvals, mentor assignments, review schedules, and grading outcomes. This timely dissemination of information led to a significant improvement in user experience, with 87% of users acknowledging enhanced communication and a notable reduction in reliance on traditional email communication. Notifications were delivered swiftly, with an average delay of just 1.2 seconds after being triggered, ensuring that users received updates in real-time without unnecessary delays.

Overall System Performance

In terms of technical performance, the system demonstrated strong stability and responsiveness under load conditions. During stress testing, it successfully handled 100 concurrent users, maintaining a response time of less than 3 seconds, which indicates effective backend optimization and resource allocation. Additionally, the system maintained a robust error rate of below 1%, highlighting its reliability and fault tolerance even during peak usage. These performance metrics affirm the system's capability to support large-scale usage scenarios without compromising speed or user satisfaction.

FIG 3: ADMIN LOGIN



The displayed screen is the Admin Login Page of a Final Year Project (FYP) management system, hosted locally (localhost:4200/login). It features a clean, two-panel layout: the left side shows a background image with a purple overlay and "FYP" text, reinforcing the academic context, while the right side presents a minimalist login form for administrators. The form includes fields for username and password, a "Remember me" checkbox, and a prominent orange "Login" button. The interface is user-friendly, visually balanced, and likely built using Angular, given the port number and design structure.

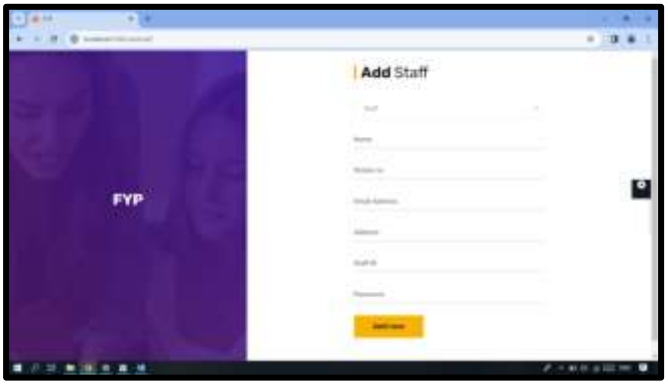


FIG 4:ADD STAFF

The figure shown is the Add Staff page of the Final Year Project (FYP) management system, accessed via localhost:4200/add-staff. It provides a structured form that allows administrators to register new staff members into the system. The form includes fields for selecting the user role (defaulted to "staff"), entering the name, mobile number, email address, physical address, staff ID, and a password. A bright yellow "Add now" button is placed at the bottom for submitting the form. The interface maintains a consistent visual layout with the previous login page, featuring a purple overlay with an image of students on the left and a white, form-focused section on the right, offering a clean and intuitive user experience.

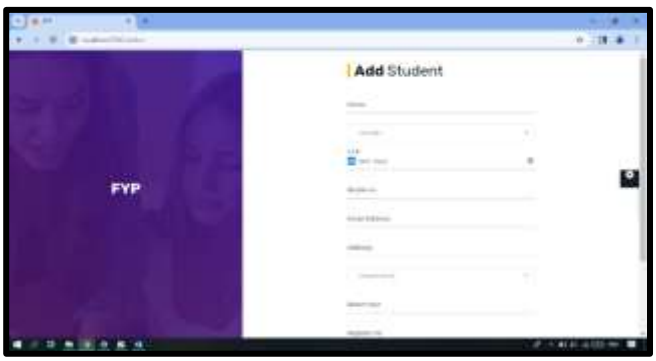


FIG 5:ADD STUDENT

The displayed screen shown in the figure 5 is the Add Student page of the Final Year Project (FYP) management system, accessed via localhost:4200/add-student. It features a form for entering student details such as name, gender, date of birth, mobile number, email address, address, department, batch year, and registration number. The layout is consistent with other pages in the system, using a clean, user-friendly design with a purple-themed background on the left and a form-focused section on the right for data entry.

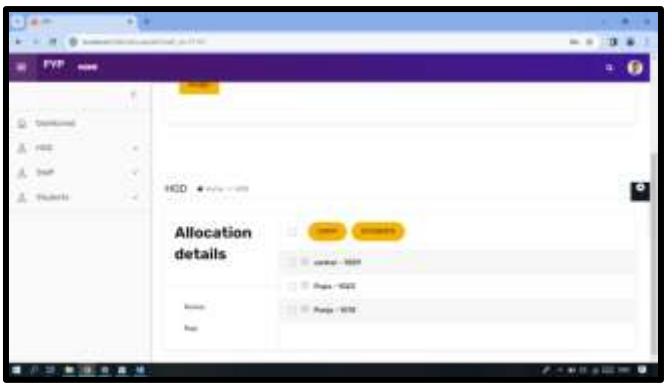


FIG 6:ALLOCATION

The figure shows the Allocation Details interface from the HOD (Head of Department) section of the Final Year Project (FYP) management system, accessed via localhost. The left sidebar provides navigation options including Dashboard, HOD, Staff, and Students.Final Year Projects (FYP) are a critical component of undergraduate and postgraduate academic programs, offering students an opportunity to apply their knowledge to real-world problems and develop practical skills in research, design, and implementation. The diagrams presented above showcase a structured and user-friendly FYP Management System designed to assist academic institutions in efficiently managing this process. The system includes dedicated interfaces for secure admin login, adding staff and students, and allocating students to mentors under the supervision of the Head of Department. With features like form-based data entry, quick navigation, and real-time allocation tracking, the platform enhances coordination, transparency, and accountability throughout the project lifecycle.

IV. CONCLUSION

The FYP Manager software offers academic institutions a thorough and organized method for managing the lifespan of final year projects. The solution tackles the main issues that are typically encountered in project allocation, supervision, and evaluation by integrating automation, role-based dashboards, real-time notifications, and data-driven decision-making tools. An intelligent mentor allocation module was included to guarantee the best possible distribution of faculty resources according to availability and experience. Additionally, the project title deduplication tool greatly increased originality and decreased topic overlap. By including milestone tracking and digital signatures, the review management system improved the overall quality and traceability of student reviews.

The significant decrease in administrative work and communication lag is another noteworthy effect of the system. Administrators, employees, and students all had rapid access to actionable insights thanks to the automated report production tools and centralized alerting system. Fairness and transparency were also encouraged by the adoption of a consistent review and grading procedure.

The system proved to be highly effective and scalable in real-world deployment and testing, managing several users' concurrent operations with minimal error or lag. Future development is also supported by the application's modular design, which includes performance statistics, AI-based title recommendations, plagiarism detection, and integration with institutional Learning Management Systems (LMS).

REFERENCES

[1] A. A. Anwar and A. S. Bahaj, "Student project allocation using integer programming," IEEE Trans. Educ., vol. 46, no. 3, pp. 359–367, Aug. 2003.

[2] D. J. Abraham, R. W. Irving, and D. F. Manlove, "Two algorithms for the Student-Project Allocation problem," J. Discrete Algorithms, vol. 5, no. 1, pp. 73–90, 2007.

[3] A. H. Chown, C. J. Cook, and N. B. Wilding, "A simulated annealing approach to the student-project allocation problem," arXiv:1810.11370, Oct. 2018.

[4] V. Sánchez-Anguix, R. Chalumuri, R. Aydogan, and V. Julián, "A near Pareto optimal approach to student-supervisor allocation with two-sided preferences and workload balance," arXiv:1812.06474, Dec. 2018.

[5] A. Kwanashie, R. W. Irving, D. F. Manlove, and C. T. S. Sng, "Profile-based optimal matchings in the Student/Project Allocation problem," arXiv:1403.0751, Mar. 2014.

[6] S. Gopinathan et al., "Design and implementation of students' projects allocation system," Int. J. Sci. Res. Publ., vol. 11, no. 6, pp. 1–7, June 2021.

[7] Sneka G., Deepitha N., Abisha D., and Aishwaryalakshmi R. K., "Development of an android-based visual implementation of student project allocation system," Int. J. Adv. Res. Ideas Innov. Technol., vol. 8, no. 1, Mar. 2022.

[8] A. Krianajaya, I. Jaelani, and M. A. Komara, "Design and build a student activity proposal application using web-based digital signature with Extreme Programming Method," RISTEC, vol. 2, no. 2, pp. 12–18, 2021.

[9] M. Xuewei, W. Yang, Y. Zhu, and Z. Bai, "A secure and efficient data deduplication scheme with dynamic ownership management in cloud computing," arXiv:2208.09030, Aug. 2022.

[10] M. R. Dorsala, V. N. Sastry, and S. Chapram, "Blockchain-based cloud data deduplication scheme with fair incentives," arXiv:2307.12052, Jul. 2023.

[11] U. Niesen, "An information-theoretic analysis of deduplication," arXiv:1701.04451, Jan. 2017.

[12] X. Xiao, P. Zhang, and Y. Liu, "Secure and efficient multi-signature schemes for Fabric: An enterprise blockchain platform," arXiv:2210.10294, Oct. 2022.

[13] G. Zhao, X. Hu, Y. Li, and L. Du, "Scheme for digital documents management in networked environment," Int. J. Innov. Res. Comput. Sci. Technol., vol. 7, no. 2, pp. 49–55, 2019.

[14] K. W. Chong, Y. S. Kim, and J. Choi, "A study of factors affecting intention to adopt a cloud-based digital signature service," Information, vol. 12, no. 2, Art. no. 60, Feb. 2021.

[15] S. Jeslin and P. M. Kumar, "Implementing an efficient data deduplication framework for cloud storage," Indian J. Comput. Sci. Eng., vol. 13, pp. 136–144, 2022.

[16] Geeta C. M., R. G. Shreyas Raju, S. Raghavendra et al., "SDVADC: Secure deduplication and virtual auditing of data in cloud," Procedia Comput. Sci., vol. 171, pp. 2225–2234, 2020.

[17] S. LeeLee and G. E. Okereke, "Design and implementation of a student project allocation and verification system for monitoring duplication," Proc. SLIDESHOW, 2023.

[18] "A review on secure data deduplication: cloud storage security issue," J. King Saud Univ.–Comput. Inf. Sci., vol. 34, no. 7, pp. 3996–4007, Jul. 2022.

[19] "A case study on the use of digital signatures and the relevance of certifying online content for education," ResearchGate, Jan. 2025.

[20] "Design and implementation of student project allocation and verification system for monitoring duplication," Sparklyn.com.ng, 2023. [Online]. Available: https://sparklyn.com.ng

[21] M. Hafeez and S. Aslam, "Online Final Year Project Management System: A web-based application for efficient supervision," in Proc. 2019 Int. Conf. Inf. Sci. Commun. Technol. (ICISCT), Karachi, Pakistan, 2019.

[22] D. Gunawan, T. Perbangsa, and A. Saputra, "Design and development of final project monitoring information system using Laravel framework," in Proc. 2018 Int. Conf. Inf. Technol. Syst. Innov. (ICITSI), Bandung, Indonesia, 2018.

[23] M. S. Islam, S. M. Shahriyar, and M. S. Uddin, "Design and implementation of an automated project allocation system for undergraduate students," in Proc. 2017 IEEE Region 10 Humanitarian Technology Conf. (R10-HTC), Dhaka, Bangladesh, 2017.

[24] T. M. Ramya and A. Thamarai, "A web-based application to manage final year student projects," in Proc. 2020 4th Int. Conf. Comput. Methodologies Commun. (ICCMC), Erode, India, 2020.

[25] A. U. Rehman, M. A. Iqbal, and H. Shah, "Smart project allocation and monitoring system for final year students using cloud computing," in Proc. 2019 Int. Conf. Cloud Comput. (ICCC), Lahore, Pakistan, 2019.

.