

# FYP MANAGER: SIMPLIFYING PROJECT ALLOCATION AND SUPERVISION



*By*

[REDACTED]

**Register No. 2303921362221012**

**A project report submitted to the  
FACULTY OF INFORMATION AND COMMUNICATION  
ENGINEERING**

*In partial fulfillment of the requirements  
for the award of the degree of*

**MASTER OF COMPUTER APPLICATIONS**

**PSNA COLLEGE OF ENGINEERING AND TECHNOLOGY  
(An Autonomous Institution)**

**DINDIGUL – 624 622**

**ANNA UNIVERSITY, CHENNAI**

**JUNE 2025**

## **BONAFIDE CERTIFICATE**

Certified that this project report titled **FYP MANAGER: SIMPLIFYING PROJECT ALLOCATION AND SUPERVISION** is the bonafide work of [REDACTED] A (Reg. No.: 2303921362221012), who carried out the work under my supervision. Certified further that to the best of my knowledge and belief, the work reported herein does not form part of any other project report or dissertation on the basis of which a degree or an award was conferred on an earlier occasion on this or any other candidate.

Place: PSNACET

Date:

**Mrs. NIMMATI SATHEESH, M.C.A., MPhil., (Ph.D.),  
Supervisor,  
Assistant Professor,  
Department of MCA,  
PSNA College of Engineering and Technology,  
(An Autonomous Institution)  
Dindigul – 624 622.**

COUNTERSIGNED

**Dr. P. JAGANATHAN, M.C.A., Ph.D.,  
Professor and Head,  
Department of MCA,  
PSNA College of Engineering and Technology,  
(An Autonomous Institution)  
Dindigul – 624 622.**

**ANNA UNIVERSITY, CHENNAI - 600 025**

**VIVA VOCE EXAMINATION**

The Viva Voce Examination of this Project work submitted by

[REDACTED] A (Reg. No.: 2303921362221012), was held on .....

**INTERNAL EXAMINER**

**EXTERNAL EXAMINER**



The Mind  
building the future with mind...

Date: 16-06-2025

### Project Completion Certificate

This is to certify that Mr. [REDACTED] A (Reg.No:2303921362221012), a final year MCA student of PSNA College of Engineering and Technology (Autonomous), Dindigul, has been successfully completed the final semester project entitled “**FYP Manager: Simplifying Project Allocation and Supervision**” in our organization under the external guidance of Ms. Kokila A, during the period from **January 2025 to June 2025**.

Congratulations on successfully completing this project! May this achievement be a stepping stone toward unlocking your full potential and paving the way for future accomplishments.

We wish you all the best in your future endeavors.

Sincerely  
  
Project Coordinator

The Mind IT

The Mind IT

No.1, Ground Floor, Christ Church Complex, Chatram Bus Stand, Trichy, Tamil Nadu-620002.  
[www.themind.co.in](http://www.themind.co.in), [hr@themind.co.in](mailto:hr@themind.co.in), 0431-3559737.

## ABSTRACT

A “FYP Manager: Simplifying Project Allocation and Supervision”, Managing and controlling final year projects for students through traditional methods can be a time-consuming and cumbersome process for both students and faculty members. This project addresses these challenges by developing a comprehensive, web-based system designed to automate and optimize the management of student projects from initiation to completion. The system facilitates the entire project life cycle, starting with project title recommendations based on student preferences and faculty expertise. It also incorporates a prioritization mechanism for both project titles and student teams, ensuring efficient matching of students to suitable projects based on their skills and interests. One of the standout features of the system is its deduplication functionality, which automatically identifies and removes redundant or overlapping project proposals, thus streamlining the project selection and allocation process. This significantly reduces administrative workload, allowing faculty members and project coordinators to focus on more strategic tasks.

Furthermore, the system enables students to submit their project preferences, view available projects, and communicate directly with project advisors. Faculty members, including internal guides, HODs, and project coordinators, benefit from real time access to project progress reports, team performance, and project scoring, ensuring better decision-making and more timely interventions when necessary. The web application enhances collaboration between students, faculty advisors, and department heads, promoting a more organized, transparent, and efficient project management experience. By formalizing the project allocation and monitoring process, the system ensures consistency, reduces human error, and creates a professional environment for final year project management.

This project was implemented using Python and Flask for the backend, with HTML, CSS, and Bootstrap for the frontend, and MySQL for backend data storage. It includes key features such as user authentication, project title submission and approval, student-staff allocation, review scheduling and attendance marking, material submission tracking, plagiarism detection, and data-driven insights through semantic analysis using Sentence-BERT.

## **ACKNOWLEDGEMENT**

I express my humble thanks to the God Almighty, for the kindness, grace and Blessings showered on me to complete the project successfully.

I express my sincere thanks to **Tmt. K. DHANALAKSHMI AMMAL**, Chairperson and **Rtn. MPHF. R.S.K. RAGURAAM D.A.E., M. Com**, Pro-Chairman for permitting me to utilize all the necessary facilities of this institution.

I express my gratitude to our Principal **Dr. D. VASUDEVAN, M.E., Ph.D.**, for his guidance and advice to shape myself for the future career.

I convey my earnest thanks to **Dr. P. JAGANATHAN, M.C.A., Ph.D.**, Professor and Head, Department of Computer Applications, PSNA College of Engineering and Technology, for his invaluable guidance, support and suggestions throughout the duration of this project work.

I am highly indebted to my guide, **Mrs. NIMMATI SATHEESH, M.C.A., MPhil., (Ph.D.)**, Assistant Professor, Department of Computer Applications, PSNA College of Engineering and Technology, for her suggestions and constructive criticisms during the duration of my project.

I would not forget to remember **Ms. A. KOKILA, M.E.**, of The Mind IT for her encouragement and more over for her timely support and guidance.

I am thankful to all teaching and non-teaching staff of the Department of Computer Applications, PSNA College of Engineering and Technology who have helped me in successfully completing my project work in time. Last but not the least, I am grateful to my parents and friends for their constant support and encouragement.

## TABLE OF CONTENTS

<b>Chapter No.</b>	<b>Title</b>	<b>Page No.</b>
	<b>ABSTRACT</b>	v
	<b>ACKNOWLEDGEMENT</b>	vi
	<b>LIST OF TABLES</b>	ix
	<b>LIST OF FIGURES</b>	x
	<b>LIST OF ABBREVIATIONS</b>	xi
<b>1</b>	<b>INTRODUCTION</b>	1
	1.1 PROBLEM STATEMENT	1
	1.2 CHALLENGES	2
	1.3 AIM OF THE PROJECT	2
	1.4 CHAPTER PLAN	3
<b>2</b>	<b>SYSTEM ANALYSIS</b>	4
	2.1 LITERATURE SURVEY	4
	2.2 EXISTING SYSTEM	6
	2.3 PROPOSED SYSTEM	6
	2.4 REQUIREMENTS GATHERING	7
	2.5 FEASIBILITY STUDY	9
	2.6 COMPANY PROFILE	10
<b>3</b>	<b>SYSTEM DESIGN</b>	12
	3.1 MODULE DESIGN	12
	3.2 UML DIAGRAM	14
	3.3 ER DIAGRAM	19
	3.4 DATABASE DESIGN	20
<b>4</b>	<b>IMPLEMENTATION</b>	24
	4.1 BACKGROUND STUDY	25
	4.2 IMPLEMENTATION	29

<b>5</b>	<b>SYSTEM TESTING</b>	31
	<b>5.1 TESTING</b>	32
	<b>5.2 TEST CASES</b>	32
	<b>5.3 OUTCOME OF THE TEST</b>	33
<b>6</b>	<b>CONCLUSION</b>	35
	<b>6.1 CONCLUSION</b>	35
	<b>6.2 FUTURE ENHANCEMENTS</b>	35
	<b>APPENDIX A: SCREENSHOT</b>	36
	<b>APPENDIX B: SAMPLE CODE</b>	42
	<b>REFERENCES</b>	51

## **LIST OF TABLES**

<b>Table No.</b>	<b>Title</b>	<b>Page No.</b>
3.1	Project Coordinator Login	20
3.2	Staff Allocation	20
3.3	Staff Table	21
3.4	Digital Signature	21
3.5	Student Table	22
3.6	Review Details	23
5.1	Test Cases of FYP Manager	33

## **LIST OF FIGURES**

<b>Figure No.</b>	<b>Title</b>	<b>Page No.</b>
3.1	Use Case Diagram for Actors	16
3.2	Class Diagram for FYP Manager	17
3.3	Sequence Diagram for Actors	18
3.4	ER Diagram for FYP Tables	19
5.1	Test Outcome	34

## LIST OF ABBREVIATIONS

Acronym	Abbreviations
<b>API</b>	Application Programming Interface
<b>HTML</b>	Hypertext Markup Language
<b>SQL</b>	Structured Query Language
<b>UML</b>	Unified Modeling Language
<b>XML</b>	Extensible Markup Language
<b>HTTP</b>	Hypertext Transfer Protocol
<b>DBMS</b>	Database Management System
<b>CRUD</b>	Create, Read, Update and Delete
<b>GPL</b>	GNU General Public License
<b>LMS</b>	Learning Management System
<b>BERT</b>	Bidirectional Encoder Representations from Transformers

# **CHAPTER 1**

## **INTRODUCTION**

The final year project is the culmination of the degree as it gives students a chance to demonstrate all they have learned. The project module is very different from other modules. Although students are supervised, the onus is on the student to define the problem boundaries, to investigate possible solutions, and to present the results in writing, verbally and in action. Apart from an initial briefing session there are no formal lectures to attend. Teaching consists of regular individual/small group meetings to discuss progress. For assessment, students submit reports of their progress and final results, and give in-person presentations and demonstrations of their work.

### **1.1 PROBLEM STATEMENT**

The Final Year Project (FYP) management system and process encounter several challenges that can impact their effectiveness and efficiency. A key issue is the lack of clear guidelines, leading to confusion among students, faculty, and administrators. Inconsistent or vague instructions for FYP selection, execution, and assessment can hinder a smooth process. Additionally, limited access to resources, including research materials and technical support, may impede students in conducting high quality research, affecting the diversity of available project topics. Allocation problems pose another challenge, as mismatches can occur when projects are not assigned based on student preferences, skills, and academic backgrounds. Communication gaps exacerbate these issues, resulting in misunderstandings, missed deadlines, and inadequate support. Adequate mentorship is crucial, yet inadequate guidance from faculty advisors can affect the overall quality of FYP outcomes.

Addressing this requires clear expectations, regular meetings, and mentorship training for faculty members. Technological challenges, such as outdated software or insufficient infrastructure, can impede progress, particularly in fields requiring advanced technological tools. Proper timeline management is essential to avoid delays in project completion, and subjective assessment criteria may lead to bias in the evaluation process. Without a robust deduplication mechanism, multiple students may select similar projects, limiting the variety of topics. Moreover, inadequate monitoring tools make it challenging for faculty and administrators to track project

development, identify issues, and provide timely support. Limited collaboration platforms hinder the exchange of ideas, feedback, and resources crucial for project success. Addressing these challenges requires a holistic approach, encompassing clear guidelines, effective communication strategies, sufficient resources, and the implementation of efficient monitoring and assessment processes. Continuous improvement and feedback mechanisms are essential to refine the FYP management system for optimal outcomes. In light of these challenges, there is a compelling need to develop a web-based system that addresses these shortcomings and streamlines the FYP management process. The proposed system aims to enhance clarity, communication, and efficiency in project selection, allocation, and tracking, ultimately providing a more supportive and effective environment for both students and faculty involved in the FYP journey.

## **1.2 CHALLENGES**

During the development of the Final Year Project Manager web application, several challenges were encountered across different phases of the project. One of the major design challenges was managing different user roles Project Coordinator, Staff, and Students without implementing a traditional user registration or login system. This required careful planning to ensure that access to specific features was correctly controlled, while also avoiding complex authentication mechanisms. Handling dynamic data was another significant challenge. Data such as student allocations, project titles, review schedules, materials, and attendance needed to be fetched and displayed in real-time depending on the role accessing the system. Designing clean and efficient backend routes in Flask and integrating them smoothly with the frontend using JavaScript and Bootstrap demanded careful coordination between frontend and backend development.

## **1.3 AIM OF THE PROJECT**

The aim of this project is to develop and implement a web-based system for Final Year Project (FYP) management, addressing the current challenges and inefficiencies in the manual and traditional processes. The primary goal is to create a user-friendly platform that streamlines project selection, allocation, and tracking, ultimately enhancing the overall FYP experience for students, faculty advisors, and administrators.

- To optimize project allocation by implementing an algorithm based on student preferences and faculty expertise.
- To enhance guidelines and clarity for students, faculty, and administrators involved in the FYP process.
- To improve resource access and mentorship through enhanced communication channels.
- To upgrade technology infrastructure to support advanced tools and technologies for technical projects.
- To implement a deduplication mechanism to eliminate redundancy in project proposals and ensure topic diversity.
- To enhance the evaluation process with objective and transparent criteria for fair and constructive assessments.
- To create monitoring and collaboration tools for seamless communication and resource sharing.
- To automate notification systems to keep stakeholders informed about project updates, deadlines, and events.
- To improve user experience by designing a user-friendly interface for all users.
- To provide customizable reporting for project allocations, review outcomes, and overall progress.

## **1.4 CHAPTER PLAN**

The first chapter, Introduction deals with overview of the project which comprises of problem statement, challenges faced in the project and aim of the project. Chapter 2, System analysis deals with literature survey requirement gathering of the project and proposed system of the project. Chapter 3, System design is followed by chapter 2 which depicts module description, database design and UML diagram. Next Chapter 4, implementation depicts about frontend, backend software and steps to install the proposed software. The Chapter 5, deals with testing and outcome of test cases. The Chapter 6, Conclusion and Future Enhancement can be included in the project.

## **CHAPTER 2**

### **SYSTEM ANALYSIS**

System analysis in a software project is the process of carefully examining and understanding the desired system. Its goal is to identify requirements, functionalities, and constraints, leading to a well-informed and efficient design for the software solution. This phase is vital for creating a robust, user-friendly, and successful software system.

The proposed system a Final Year Project Manager Web Application aims to simplify and centralize the workflow involved in managing academic projects for students, guides, and coordinators. The analysis focuses on understanding the existing manual or partially automated process and transforming it into a digital platform that ensures coordination, transparency, and efficiency.

#### **2.1 LITERATURE SURVEY**

The literature review in the field of educational project management systems highlights the increasing shift towards digitization and automation in academic workflows. Traditional methods of managing student projects relying heavily on paper records, offline communication, and manual scheduling have proven to be inefficient, error-prone, and difficult to scale. The demand for streamlined platforms that facilitate interaction among coordinators, staff, and students has become evident, particularly in engineering and computer science programs. Several studies emphasize the importance of user - centered dashboard systems in educational technology.

Dey et al. (2022) demonstrate the importance of participatory design in creating teacher dashboards, ensuring the system meets user needs effectively [3].

Gomez-del Rio and Rodriguez (2022) show that project-based learning (PBL) environments enhance engineering design skills, supporting the need for integrated project management tool [5].

Guo et al. (2020) review PBL outcomes, confirming that well-structured platforms positively affect student learning and performance [6].

Hao et al. (2019) highlight the benefits of automated feedback systems in improving student revisions and learning outcomes [7].

Israel-Fishelson and Kohen-Vacs (2023) highlight the benefits of student-centered dashboards that enhance learning engagement and provide better academic progress tracking [8].

McKinney (2022) delivers a practical guide from the creator of pandas, offering updated, real-world examples and clear instruction on data wrangling with pandas, NumPy, and IPython, making it a go-to resource for data analysts [10].

Saito et al. (2021) validate the use of rubric-based evaluations in programming education, highlighting structured assessment's role in fair grading [13].

Veluvali and Surisetti (2022) emphasize that learning management systems (LMS) improve learner engagement through modular and interactive design [16].

Xu (2021) demonstrates that schema refinement, indexing strategies, and query tuning can dramatically boost MySQL throughput and slash latency in real-world applications [19].

Yun (2023) reviews the challenges of digitalization in education during COVID-19, emphasizing the need for intuitive and accessible digital platforms [20].

## Limitations of the Literature Survey

**1. Time-Consuming:** Conducting a comprehensive literature survey on graphical password authentication systems using clued click points requires substantial time to locate, read, and analyse relevant research papers and articles, given the breadth of the topic and evolving research landscape.

**2. Information Overload:** The extensive volume of literature available on graphical password authentication can overwhelm researchers, making it challenging to sift through and prioritize the most relevant and credible sources for inclusion in the survey.

**3. Publication Bias:** Literature surveys may be susceptible to publication bias, where studies with positive or significant results are more likely to be published, potentially skewing the overall findings towards certain perspectives or outcomes related to clued click point authentication.

**4. Limited Access:** Some relevant research articles may be inaccessible due to paywalls or restricted access, which can hinder the comprehensiveness of the survey and limit the diversity of perspectives included in the analysis

## 2.2 EXISTING SYSTEM

In the current manual submission process, students navigate a cumbersome journey of physically delivering project proposals to department offices, involving the printout of multiple hard copies. The title approval stage relies on faculty conducting manual reviews, offering handwritten feedback on printed documents. Mentor assignment is an in-person affair, adding paperwork and dependency on the availability and expertise of faculty members. Document sharing, whether physical or through email, lacks a centralized system, leading to scattered file management [6]. Project reviews involve in-person presentations, with feedback provided on printed evaluation sheets. Attendance tracking relies on manual sheets, introducing potential errors. Grading and approval are labour-intensive, involving manual assessment and physical sign-offs. Communication, predominantly face-to-face, via phone calls or emails, may lead to fragmented information exchange and delays.

### Limitations

- Manual processes result in a slow and inefficient workflow, impacting project timelines.
- Heavy reliance on physical paperwork leads to document overload and storage challenges.
- Face-to-face and email communication may cause gaps and delays in conveying important information.
- Approvals and signatures require physical presence, leading to delays and inconvenience.
- Tracking project progress is challenging due to manual record-keeping and scattered information.

## 2.3 PROPOSED SYSTEM

The proposed system for the Final Year Project (FYP) Web App aims to overcome the drawbacks of the traditional system by introducing a modern, web-based platform. The proposed system envisions a modern, efficient, and user-centric platform that revolutionizes FYP

management, offering enhanced collaboration, streamlined workflows, and improved accessibility for all stakeholders involved. Key features of the proposed system include:

- **Digital Project Proposal Submission**

Students can electronically submit project proposals, reducing reliance on physical paperwork.

- **Automated Title Approval with Deduplication**

Automated algorithms assist in the review and approval of project titles, incorporating deduplication functionality to eliminate redundant or overlapping proposals.

- **Efficient Mentor Assignment**

It can efficiently assign mentors through an online platform, considering faculty expertise.

- **Centralized Document Management**

A centralized repository facilitates digital sharing and management of project-related documents.

- **Online Project Reviews**

Project reviews can be conducted digitally, enhancing accessibility and eliminating geographical constraints.

- **Automated Attendance Tracking**

Automated systems track student attendance during online review sessions, improving accuracy.

## **2.4 REQUIREMENTS GATHERING**

Requirements gathering is the initial phase in the software development process where stakeholders' needs and expectations for a project are identified and documented. This process involves collecting detailed information about the desired features and functionalities of the system. It includes understanding the business objectives, user requirements, and technical constraints. Effective requirement gathering ensures that all parties have a clear and shared understanding of what the system should achieve. It lays the foundation for the project's scope, design, and development stages.

## **Hardware Configuration**

The below Hardware Specifications were used in both Server and Client machines when developing. It outlines the physical components required, such as processor type and speed, RAM, and storage capacity. It ensures the system's efficiency and performance.

Processor : Intel (R) Core (TM) i5

Processor Speed : 2.4 GHz

RAM : 8 GB

Hard Disk Drive : 250 GB

## **Software Configuration**

The below Software Specifications were used in machines when developing. It includes selecting and setting up software tools, operating systems, and applications necessary for development and deployment. It ensures compatibility and stability.

Operating System : Windows or Later

Frontend : HTML, CSS, Bootstrap

Programming Language : Python 3.7.4, Flask Framework

Backend : MYSQL

IDE : VS Code

Web Server : Wamp Server

Browser : Google Chrome

## 2.5 FEASIBILITY STUDY

The feasibility study is conducted to assess the viability of implementing the proposed Final Year Project Manager Web Application from different dimensions: technical, operational, economic, and legal. The aim is to ensure that the system can be successfully developed and deployed with the available resources.

### Technical Feasibility

**Technology Stack:** The project uses well-established technologies like Python Flask (backend), MySQL (database), HTML/CSS/Bootstrap (frontend), which are stable, widely supported, and suitable for web application development.

**Development Environment:** The system is being developed on commonly available hardware and software platforms, reducing the need for specialized infrastructure.

**Scalability:** The architecture is modular, allowing for future enhancements like file storage systems, dashboards, or analytics.

### Operational Feasibility

**User Roles:** The system is designed with clear, well-defined roles for coordinators, staff, and students, ensuring minimal confusion.

**Ease of Use:** The user interface is intuitive, using Bootstrap for responsive design. Non-technical users (students and faculty) can operate the system with minimal training.

**Workflow Optimization:** Automates repetitive tasks like review scheduling, project status updates, and attendance tracking.

### Economic Feasibility

**Development Cost:** Since open-source technologies are used (Flask, MySQL, Bootstrap), there is no licensing cost involved.

**Maintenance:** Low ongoing maintenance cost due to use of lightweight frameworks and modular design.

**Time Efficiency:** Reduces staff workload, paperwork, and review coordination efforts saving valuable time and resources in the long term.

## **2.6 COMPANY PROFILE**

The Mind IT is one of the few IT system integration, professional service and software development companies in Macedonia that works with Enterprise systems and companies which has sister concern in Trichy. As a privately owned company, The Mind IT provides IT Consultancy, software design and development as well as professional services and hardware deployment and maintenance to the following verticals: Government (Local and Central), Financial Services (insurance, banking and clearing house), Telecommunications, Energy and Utilities, Health Care and Education.



## **MISSION**

The Mind IT Solution' mission is to enhance the business operation of its clients by developing and/or implementing premium IT products and services.

- Providing high quality software development services, professional consulting and development outsourcing that would improve our customers' operations
- Making access to information easier and securer (Enterprise Business)
- Improving communication and data exchange (Business to Business)
- Providing our customers with a value for money and exceptional service experience is our top priority.
- Leveraging innovative technologies and agile methodologies to deliver scalable and future-ready digital solutions tailored to client needs.

## **VISION**

The Mind IT Solution is a leading IT company for Consulting Services and Deployment of best of breed Business Solutions to top tier domestic and international customers.

## **SERVICES OFFERED**

- **Web Development** - Create an impactful web app that fits your brand and industry within a shorter time frame.
- **Database Analysis** - Enhance data quality, customizable reports enhanced with interactive dashboards across web and mobile.
- **Mobile App Development** - Build powerful and user-friendly mobile apps tailored to your business needs, ensuring seamless performance and engagement across both Android and iOS platforms.
- **Server Security** - Server security covers the processes and tools used to protect the valuable data and assets held on an organization's servers.
- **UX/UI Strategy** – Build the product you need on time with an experienced team that uses a clear and effective design process.

This chapter outlines the system analysis phase, emphasizing its role in building an efficient Final Year Project Manager system. It reviews literature on digital education tools and evaluates current project handling methods. The proposed web-based system is introduced, offering features like title submission, staff allocation, review scheduling, and evaluation. The chapter concludes with a feasibility study and institutional background, while the next chapter focuses on system design, including architecture, data flow, and user interface.

# **CHAPTER 3**

## **SYSTEM DESIGN**

System design concentrates on moving from problem domain to solution domain. The important phase is composed of several steps. It provides the understanding and procedural details necessary for implementing the system recommended in the feasibility study. Emphasis is on translating the performance requirements into design specification. Software design has evolved from an intuitive art dependent on experience to a science, which provides systematic techniques for the software definition. Software design is a first step in the development phase of the software life cycle.

### **3.1 MODULE DESIGN**

Module design in the FYP Manager focuses on organizing the system into modular components, each fulfilling specific functionalities critical for secure user authentication and seamless system operation.

Key principles in module design include high cohesion and low coupling, ensuring that modules are self-contained and interdependent only where necessary. This promotes flexibility, reusability, and easier maintenance across the system.

During module design, software architects define clear interfaces that specify how modules communicate and interact, ensuring seamless integration and interoperability [4]. This approach facilitates modular programming practices, enabling developers to work on independent modules concurrently and simplify maintenance tasks.

Effective module design enhances the scalability of software systems, as modules can be easily added, modified, or replaced without affecting other parts of the system [5].

By promoting encapsulation and abstraction, module design helps manage complexity, improve code readability, and support systematic testing and debugging processes.

## **1. Project Coordinator Module**

Responsible for managing the overall system and assigning roles.

### **Features:**

- Add staff and students.
- Allocate students to guides (staff).
- Schedule PRC Meeting.
- Schedule review dates.
- Add instruction for every review.
- View all projects and their statuses.
- Monitor review attendance and feedback.
- Access submitted materials from students.
- Generate report for every review.
- Generate attendance sheet for each review.
- Manage and edit staff details.
- Manage and edit student details.

## **2. Staff (Guide) Module**

Accessible only by faculty members assigned as project guides.

### **Features:**

- View list of allocated students.
- View submitted project titles and abstracts.
- Approve or reject project titles with optional feedback.
- Digitally sign approved abstracts (auto-sign using staff signature).
- Attend PRC meetings and participate in review panels.
- View scheduled review dates and meetings.
- Access materials submitted by students for each review.
- Mark attendance for each student during reviews.
- Provide written feedback/comments for each student per review.

- Edit/update previously submitted evaluations.
- Manage personal profile and update password.
- Evaluate students for each review round based on following criteria,
  1. Presentation (Max 40)
  2. Q/A (Max 40)
  3. Attendance (Max 20)
  4. Total auto-calculated

### **3. Student Module**

Used by students to submit project-related details and materials.

**Features:**

- Submit project title and description for approval.
- View status of submitted project (Pending / Approved / Rejected).
- View approved project title and signed abstract.
- Attend scheduled review sessions.
- Upload materials for each review round.
- View uploaded materials and submission status.
- Track review attendance status (Present / Absent / NA).
- View marks and feedback/comments given by staff for each review.
- Manage personal profile and update password.

#### **3.2 UML DIAGRAM**

The UML is a general-purpose modelling language. The main aim of UML is to define a standard way to visualize the way a system has been designed. It is quite similar to blueprints used in other fields of engineering [17].

UML is not a programming language. it is rather a visual language. UML diagrams to portray the behaviour and structure of a system.

UML helps software engineers, businessmen and system architects with modelling, design and analysis.

The OMG adopted Unified Modelling Language as a standard in 1997. It's been managed by OMG ever since. ISO published UML as an approved standard in 2005.

UML has been revised over the years and is reviewed periodically. Complex applications need collaboration and planning from multiple teams and hence require a clear and concise way to communicate amongst them.

Businessmen do not understand code. So, UML becomes essential to communicate with non-programmer essential requirements, functionalities and processes of the system. UML is linked with object-oriented design and analysis.

UML makes the use of elements and forms associations between them to form diagrams.

Diagrams in UML can be broadly classified as:

- **Structural Diagrams:** Capture static aspects or structure of a system. Structural Diagrams include Component Diagrams, Object Diagrams, Class Diagrams and Deployment Diagrams.
- **Behaviour Diagrams:** Capture dynamic aspects or behaviour of the system. Behaviour diagrams include Use Case Diagrams, State Diagrams, Activity Diagrams and Interaction Diagrams.

The most frequently used diagram in software development is

- Use Case Diagram
- Class Diagram
- Sequence Diagram

### 3.2.1 Use Case Diagram

A use case diagram is a graphic depiction of the interactions among the elements of a system. A use case is a methodology used in system analysis to identify, clarify, and organize system requirements. The actors, usually individuals involved with the system defined according to their roles.



Fig. 3.1 Use Case Diagram for Actors

Fig. 3.1 illustrates the interactions of three user roles - Administrator, Staff, and Student with the FYP Manager.

### 3.2.2 Class Diagram

The class diagram is the main building block of object-oriented modelling. It is used for general conceptual modelling of the structure of the application, and for detailed modelling, translating the models into programming code. Class diagrams can also be used for data modelling.

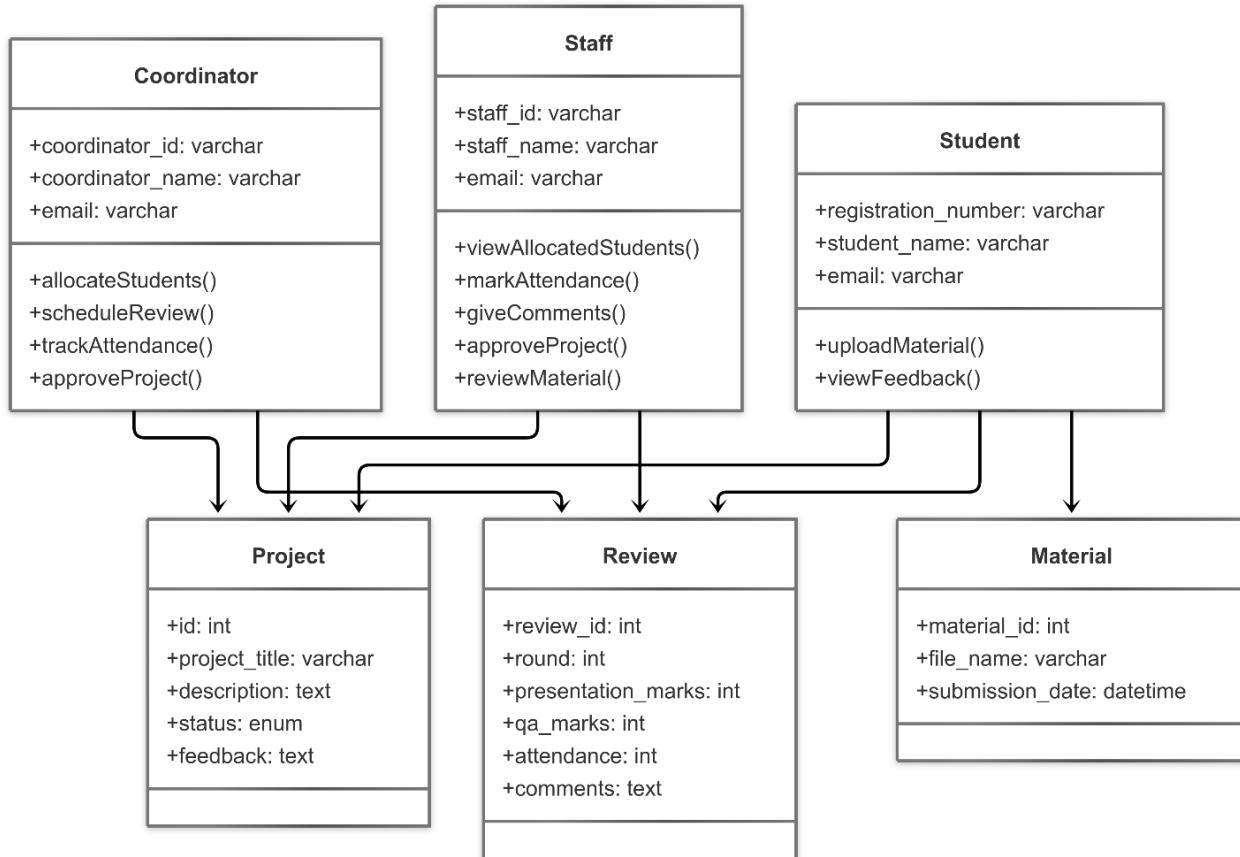


Fig. 3.2 Class Diagram for FYP Manager

Fig. 3.2 models the core entities and their behaviours in an FYP management system, including Coordinator, Staff, Student, Project, Review, and Material. It outlines attributes and methods for each class, as well as their interactions to support project allocation, review processes, and material management.

### 3.2.3 Sequence Diagram

A sequence diagram simply depicts interaction between objects in a sequential order i.e., the order in which these interactions take place. It also uses the terms event diagrams or event scenarios to refer to a sequence diagram. Sequence diagrams describe how and in what order the objects in a system.

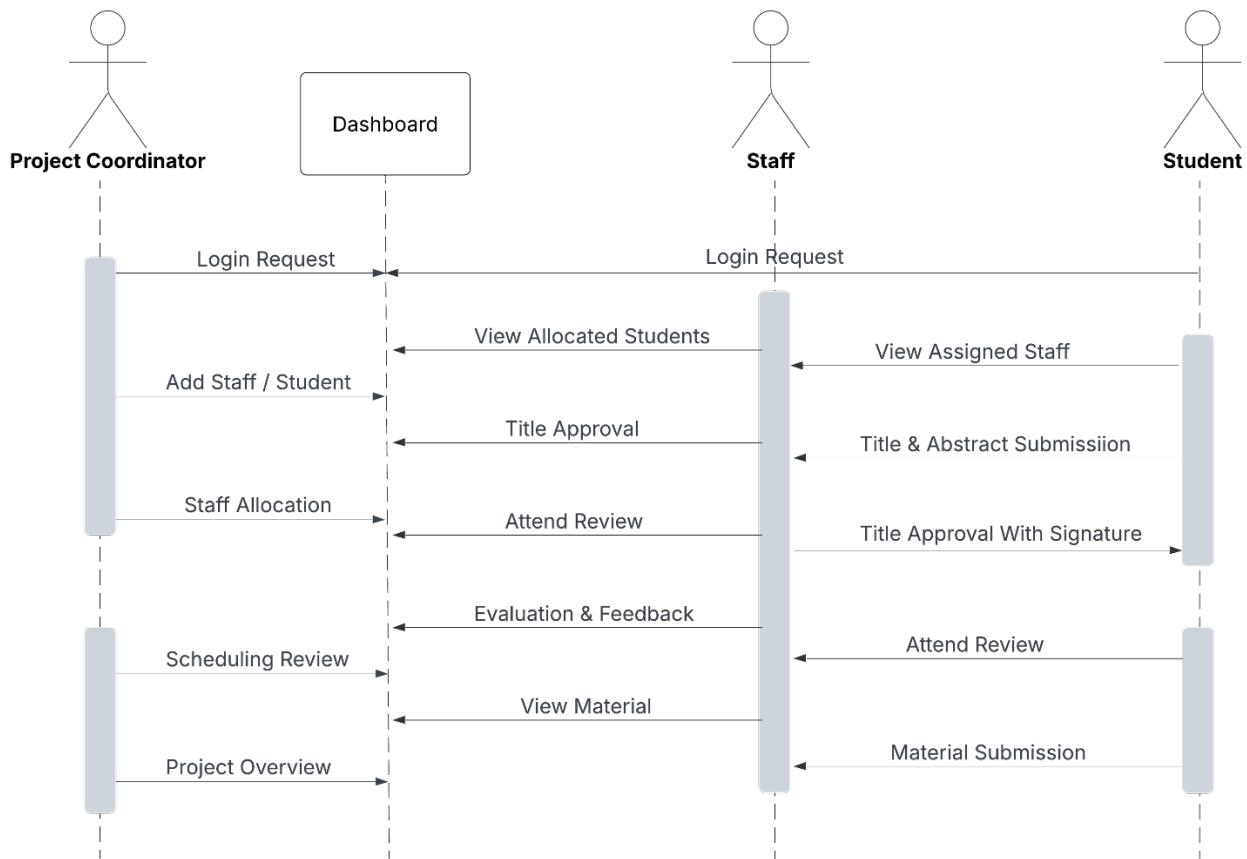


Fig. 3.3 Sequence Diagram for Actors

Fig. 3.3 illustrates the interaction flow among Coordinators, Staff, and Students within the FYP management system. It captures the sequence of operations such as login, title approval, staff allocation, material submission, and review attendance.

### 3.3 ER DIAGRAM

An Entity-Relationship (ER) diagram is a conceptual tool used in database design to outline the structure and relationships of data within a system. It defines key entities, such as users, items, or processes, and shows how these entities are related to one another. The ER diagram helps identify the data requirements and rules for the system, making it easier to build a logical and efficient database. It also aids communication between stakeholders like developers, analysts, and end users. Overall, it provides a clear blueprint for organizing and managing data effectively.

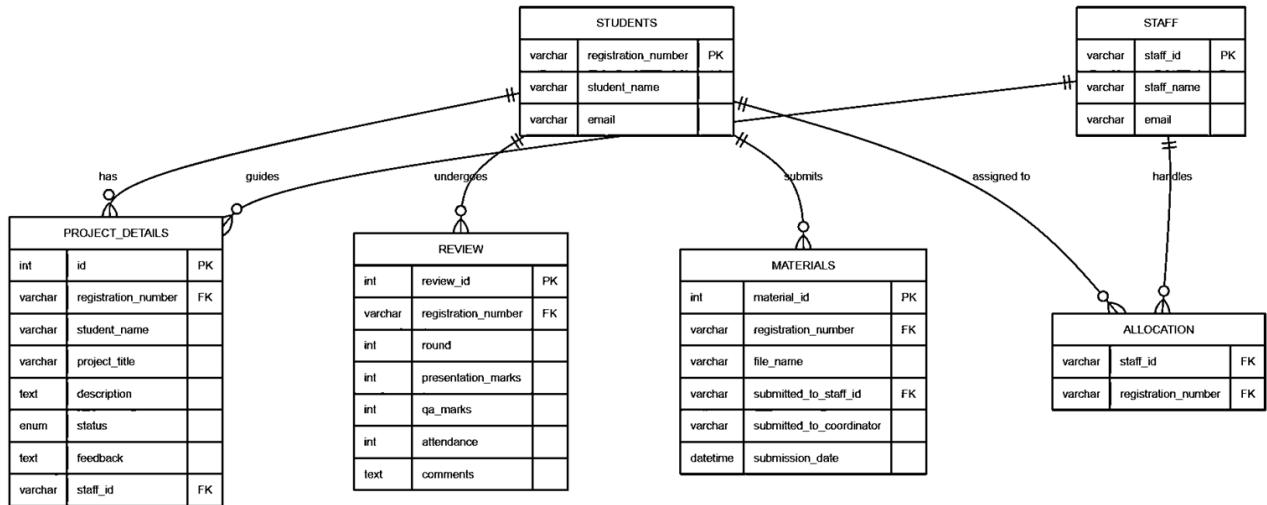


Fig. 3.4 ER Diagram for FYP Tables

Fig. 3.4 represents the relationships such as project guidance, material submission, student to staff allocation, and review participation to ensure organized project tracking and evaluation.

### **3.4 DATABASE DESIGN**

Database design is the process of organizing data into a structured schema. This involves defining tables, columns, relationships, and constraints to manage data efficiently.

Proper database design improves data integrity, making it easier to maintain and update information. This results in a more reliable and effective database system overall [8]. By focusing on efficient data management, optimization, and minimal redundancy, database design plays a crucial role in the overall functionality and reliability of any data driven application or system.

Table 3.1 Project Coordinator Login

S. No	Field	Data Type	Field Size	Constraint	Description
1	username	varchar	20	Not Null	Coordinator Username
2	password	varchar	20	Not Null	Coordinator Password

Table 3.1 describes the project coordinator login credentials i.e. username and password of the project coordinator.

Table 3.2 Staff Allocation

S. No	Field	Data Type	Field Size	Constraint	Description
1	id	Int	11	Primary Key	Id
2	department	varchar	20	Null	Department
3	staff_id	varchar	20	Null	Staff id
4	register_number	varchar	20	Null	Register number

Table 3.2 describes the staff who is assigned for particular student with information like staff id, department, student registration number.

Table 3.3 Staff Table

S. No	Field	Data Type	Field Size	Constraint	Description
1	id	int	11	Null	ID
2	name	varchar	20	Null	Name
3	mobile	bigint	20	Null	Mobile Number
4	email	varchar	40	Null	Email Id
5	location	varchar	20	Null	Location
6	staff_id	varchar	20	Primary Key	Staff Id
7	password	varchar	20	Null	Login password
8	staff_type	varchar	20	Null	Staff Id
9	date_time	timestamp	8	Null	Date time

Table 3.3 describes the staff information with basic details like id, name, mobile, email and login credentials.

Table 3.4 Digital Signature

S. No	Field	Data Type	Field Size	Constraint	Description
1	id	int	11	Primary Key	Id
2	department	varchar	20	Null	Department
3	staff_id	varchar	20	Null	Staff id
4	signature_link	varchar	20	Null	Signature link

Table 3.4 holds the digital signature of the staff which is used for approving the title along with the signature of the staff.

Table 3.5 Student Table

S. No	Field	Data Type	Field Size	Constraint	Description
1	location_id	int	11	Primary Key	Id
2	Name	varchar	20	Null	Name
3	register_number	varchar	20	Not Null	Register number
4	gender	varchar	20	Null	Gender
5	Dob	varchar	15	Null	Dob
6	mobile	bigint	20	Null	Mobile
7	Email	varchar	40	Null	Email
8	address	varchar	50	Null	Address
9	department	varchar	20	Null	Department
10	semester	varchar	50	Null	Semester
11	Year	varchar	20	Null	Year
12	username	varchar	30	Null	Username
13	password	varchar	20	Null	Password
14	date time	timestamp	20	Null	Date time
15	passout_year	int	8	Null	Passout year

Table 3.5 describes the student information with details like id, name, mobile, email, department, batch and login credentials.

Table 3.6 Review Details

S. No	Field	Data Type	Field Size	Constraint	Description
1	id	int	11	Null	id
2	department	varchar	20	Foreign Key	Department
3	register_number	varchar	20	Null	Register Number
4	upload_file_link	varchar	20	Null	Upload file link
5	description	varchar	30	Null	Description
6	staff_replay	varchar	30	Null	Staff replay
7	review_id	varchar	20	Null	Review id
8	date_time	varchar	20	Null	Date time

Table 3.6 describes the details about the review held along with the review materials uploaded by the students.

This chapter details the transition from system analysis to implementation using structured design techniques. It includes UML diagrams such as use case, class, sequence, and activity diagrams to illustrate system behaviour and architecture. The chapter also presents the ER diagram and database design, outlining the structure and relationships of core data entities. Additionally, it explains the module design, covering key components like user authentication, student-staff allocation, project submission, review scheduling, material tracking, and evaluation management. The next chapter focuses on system planning, existing system constraints, and the design of implementation methods.

## **CHAPTER 4**

### **IMPLEMENTATION**

Implementation is the stage in the project where the theoretical design is turned into a working system. The most critical stage is achieving a successful system and in giving confidence on the new system for the users, what it will work efficient and effectively. It involves careful planning, investing of the current system, and its constraints on implementation, design of methods to achieve the changeover methods.

The implementation process begins with preparing a plan for the implementation of the system. According to the plan, the activities are to be carried out in these plans; discussion has been made regarding the equipment, resources and how to test activities [7].

The coding step translates a detail design representation into a programming language realization. Programming languages are vehicles for communication between human and computers programming language characteristics and coding style can profoundly affect software quality and maintainability. The coding is done with the following characteristics in mind.

- Ease of design to code translation.
- Code efficiency.
- Memory efficiency.
- Maintainability.

Implementation is the stage of the project when the theoretical design is turned out into a working system. Thus, it can be considered to be the most critical stage in achieving a successful new system and in giving the user, confidence that the new system will work and be effective [11]. The implementation stage involves careful planning, investigation of the existing system and its constraints on implementation, designing of methods to achieve changeover and evaluation of changeover methods.

## **4.1 BACKGROUND STUDY**

### **Python 3.7.4**

Python is a general-purpose interpreted, interactive, object-oriented, and high-level programming language. It was created by Guido van Rossum during 1985- 1990. Like Perl, Python source code is also available under the GNU General Public License (GPL). This tutorial gives enough understanding on [10].

Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages. Python is a MUST for students and working professionals to become a great Software Engineer specially when they are working in Web Development Domain.

Python is currently the most widely used multi-purpose, high-level programming language. Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java. Programmers have to type relatively less and indentation requirement of the language, makes them readable all the time [10]. Python language is being used by almost all tech-giant companies like Google, Amazon, Facebook, Instagram, Dropbox, Uber... etc. The biggest strength of Python is huge collection of standard libraries which can be used for the following:

- Machine Learning
- GUI Applications (like Kivy, Tkinter, PyQt etc.)
- Web frameworks like Django (used by YouTube, Instagram, Dropbox)
- Image processing (like OpenCV, Pillow)
- Web scraping (like Scrapy, Beautiful Soup, Selenium)
- Test frameworks
- Multimedia

## **Pandas**

Pandas is a fast, powerful, flexible and easy to use open-source data analysis and manipulation tool, built on top of the Python programming language. pandas is a Python package that provides fast, flexible, and expressive data structures designed to make working with "relational" or "labeled" data both easy and intuitive [10]. It aims to be the fundamental high-level building block for doing practical, real world data analysis in Python.

Pandas is mainly used for data analysis and associated manipulation of tabular data in Data frames. Pandas allows importing data from various file formats such as comma-separated values, JSON, Parquet, SQL database tables or queries, and Microsoft Excel. Pandas allows various data manipulation operations such as merging, reshaping, selecting, as well as data cleaning, and data wrangling features. The development of pandas introduced into Python many comparable features of working with Data frames that were established in the R programming language. The panda's library is built upon another library NumPy, which is oriented to efficiently working with arrays instead of the features of working on Data frames.

## **NumPy**

NumPy, which stands for Numerical Python, is a library consisting of multidimensional array objects and a collection of routines for processing those arrays. Using NumPy, mathematical and logical operations on arrays can be performed [10].

NumPy is a general-purpose array-processing package. It provides a high-performance multidimensional array object, and tools for working with these arrays.

## **Matplotlib**

Matplotlib is a comprehensive library for creating static, animated, and interactive visualizations in Python. Matplotlib makes easy things easy and hard things possible.

Matplotlib is a plotting library for the Python programming language and its numerical mathematics extension NumPy. It provides an object-oriented API for embedding plots into applications using general-purpose GUI toolkits like Tkinter, wxPython, Qt, or GTK.

## **Seaborn**

Seaborn is a library for making statistical graphics in Python. It builds on top of matplotlib and integrates closely with pandas data structures. Visualization is the central part of Seaborn which helps in exploration and understanding of data.

Seaborn offers the following functionalities:

- Dataset oriented API to determine the relationship between variables.
- Automatic estimation and plotting of linear regression plots.
- It supports high-level abstractions for multi-plot grids.
- Visualizing univariate and bivariate distribution.

## **Scikit Learn**

Scikit-learn is a Python module for machine learning built on top of SciPy and is distributed under the 3-Clause BSD license.

Scikit-learn (formerly scikits. learn and also known as sklearn) is a free software machine learning library for the Python programming language. It features various classification, regression and clustering algorithms including support-vector machines, random forests, gradient boosting, k-means and DBSCAN, and is designed to interoperate with the Python numerical and scientific libraries NumPy and SciPy [10].

## **MySQL**

MySQL is a relational database management system based on the Structured Query Language, which is the popular language for accessing and managing the records in the database. MySQL is open-source and free software under the GNU license. It is supported by Oracle Company. MySQL database that provides for how to manage database and to manipulate data with the help of various SQL queries [15]. These queries are: insert records, update records, delete records, select records, create tables, drop tables, etc. There are also given MySQL interview questions to help you better understand the MySQL database.

MySQL is currently the most popular database management system software used for managing the relational database. It is open-source database software, which is supported by

Oracle Company. It is fast, scalable, and easy to use database management system in comparison with Microsoft SQL Server and Oracle Database. It is commonly used in conjunction with PHP scripts for creating powerful and dynamic server-side or web-based enterprise applications.

## **WampServer**

WampServer is a Windows web development environment. It allows you to create web applications with Apache2, PHP and a MySQL database. Alongside, PhpMyAdmin allows you to manage easily your database.

WampServer is a reliable web development software program that lets you create web apps with MYSQL database and PHP Apache2. With an intuitive interface, the application features numerous functionalities and makes it the preferred choice of developers from around the world. The software is free to use and doesn't require a payment or subscription.

## **Bootstrap 4**

Bootstrap is a free and open-source tool collection for creating responsive websites and web applications. It is the most popular HTML, CSS, and JavaScript framework for developing responsive, mobile-first websites [14]. It solves many problems which we had once, one of which is the cross-browser compatibility issue. Nowadays, the websites are perfect for all the browsers (IE, Firefox, and Chrome) and for all sizes of screens (Desktop, Tablets, Phablets, and Phones).

## **Flask**

Flask is a web framework. This means flask provides you with tools, libraries and technologies that allow you to build a web application [9]. This web application can be some web pages, a blog, a wiki or go as big as a web-based calendar application or a commercial website.

Flask is often referred to as a micro framework. It aims to keep the core of an application simple yet extensible. Flask does not have built-in abstraction layer for database handling, nor does it have formed a validation support. Instead, Flask supports the extensions to add such functionality to the application.

## **4.2 IMPLEMENTATION**

### **Introduction To MySQL**

#### **MySQL**

MySQL is a robust SQL database server developed and maintained by T.c.X DataKonsultAB of Stockholm, Sweden. Publicly available since 1995, MySQL has risen to become one of the most popular database servers in the world, this popularity due in part to the server's speed, robustness, and flexible licensing policy [19]. Given the merits of MySQL's characteristics, coupled with a vast and extremely easy-to use set of predefined interfacing functions, MySQL has arguably most-popular database counterpart.

#### **Installation**

MySQL is so popular among PHP users that support for the DB server is automatically built into the PHP distribution. Therefore, the only task left to deal with is the proper installation of the MySQL package. MySQL is compatible with practically every major operating system, including, among others, FreeBSD, Solaris, UNIX, Linux, and the various Windows versions. While the licensing policy is considerably more flexible than that of other database servers, I strongly suggest taking some time to read through the licensing information found at the MySQL site.

#### **Configuring MySQL**

It is time to configure the MySQL server after the installation has completed successfully. This process largely consists of creating new databases and configuring the MySQL privilege tables. The privilege tables control the MySQL database access permissions. Correct configuration of these tables is pivotal to securing your database system, and therefore it is imperative that you fully understand the details of the privilege system before launching your site into a production environment.

However, a number of resources available on the web are geared toward bringing MySQL users up to speed. Once you have correctly installed and configured the MySQL distribution, it's time to begin experimenting with Web-based databasing! The next section turns our attention towards exactly this matter, starting with an introduction of PHP's MySQL functionality.

## Using Python MySQL Connector in Flask for MySQL Database Interaction

Python MySQL Connector is an official library provided by Oracle that allows Python applications to interact with MySQL databases. It acts as a bridge between Python code and the MySQL server, enabling data operations like inserting, reading, updating, and deleting.

### Key Functions

- **Database Connection:** The connector establishes a session with the MySQL server using credentials such as host, username, password, and database name.
- **Cursor Creation:** After connecting, it provides a cursor object. This cursor is responsible for executing SQL commands and retrieving results.
- **Query Execution:** You can send SQL statements through the cursor. The connector handles communication with the MySQL engine and ensures secure execution.
- **Data Retrieval:** Once a SELECT query is executed, the connector allows fetching the results either row-by-row or all at once.
- **Transaction Management:** The connector supports transactions. You can commit to save changes permanently or roll back to undo recent changes if an error occurs.
- **Error Handling:** It provides a structured way to catch and manage database-related errors, like connection failures or invalid queries.
- **Integration:** It integrates easily with Python frameworks like Flask, Django, or standalone scripts, making it suitable for both small and large applications.

This chapter explains the shift from design to deployment through planning, coding, and testing. It covers the development environment, tools like Python, Flask, MySQL, and integration with technologies such as SMTP and Sentence-BERT. It concludes with database setup and Python-MySQL integration for dynamic features. The next chapter elaborates on system testing, including bug fixing, validation, and ensuring the system meets user requirements through test cases.

## **CHAPTER 5**

### **SYSTEM TESTING**

System Testing is an important stage in any system development life cycle. Testing is a process of executing a program with the intention of finding errors. The importance of software testing and its implications with respect to software quality cannot be overemphasized. Software testing is a critical element of software quality assurance and represents the ultimate review of specification, design and coding. A good test case is one that has a high probability of finding a yet undiscovered error [1].

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and or a finished product. It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of tests. Each test type addresses a specific testing requirement.

Testing is the set of activities that can be planned in advance and conducted systematically. Different test conditions should be thoroughly checked and the bugs detected should be fixed. The testing strategies formed by the user are performed to prove that the software is free and clear from errors. To do this, there are many ways of testing the system's reliability, completeness and maintainability.

The important phase of software development is concerned with translating the design specification into the error-free source code. Testing is carried out to ensure that the system does not fail, that it meets the specification and it satisfies the user. The system testing was carried out in a systematic manner with a test data containing all possible combinations of data to check the features of the system. A test data was prepared for each module, which took care of all the modules of the program.

System Testing is an important stage where the system developed is tested with duplicate or original data. It is a process of executing a program with the intent of finding an error. It is a critical process that can consume fifty percent of the development time.

## **5.1 TESTING**

The testing phase involves the testing of the developed system using various kinds of data. An elaborated testing of data is prepared, and a system is tested using the test data. While testing, errors are noted, and corrections will be done [2].

### **Unit Testing**

In the unit testing the analyst tests the program making up a system. The software units in a system are the modules and routines that are assembled and integrated to perform a specific function. In a large system, many modules on different levels are needed.

Unit testing can be performed from the bottom up starting with the smallest and lowest level modules and proceeding one at a time. For each module in a bottom-up testing, a short program executes the module and provides the needed data.

### **Integration Testing**

Integration testing is a systematic technique for constructing the program structure while conducting test to uncover errors associate with interfacing. Objectives are used to take unit test modules and built program structure that has been directed by design. The integration testing is performed for this Multi Cloud when all the modules where to make it a complete system. After integration the project works successfully.

### **Black Box Testing**

This method treats the coded module as a black box. The module runs with inputs that are likely to cause errors. Then the output is checked to see if any error occurred. This method cannot be used to test all errors, because some errors may depend on the code or algorithm used to implement the module.

## **5.2 TEST CASES**

A test case is a documented set of conditions or inputs along with the expected results that are used to determine whether a system or application is functioning correctly. It is designed to ensure that all aspects of the system are tested and validated. Each test case includes a unique identifier, a description, input data, the expected outcome, and the actual result of the test, which

is marked as pass or fail. Well-defined test cases help in identifying defects and verifying that the system meets its requirements.

Table 5.1 Test Cases of FYP Manager

TEST CASE ID	DESCRIPTION	EXPECTED RESULT	ACTUAL RESULT	STATUS
TC001	Valid credentials for Project Coordinator login	Secure access granted	Successful login	Pass
TC002	Adding Staff Details	Successfully added	Successfully added	Pass
TC003	Adding Student Details	Successfully added	Successfully added	Pass
TC004	Assign final year project students to staff without selecting students	Error Allocating Students	Allocation completed without errors	Fail
TC005	Invalid credentials for Guide / Staff login	Secure access denied	Login failed	Pass

Table 5.1 describe the software feature that has been tested from the perspective of the user.

### 5.3 OUTCOME OF THE TEST

The outcomes of the test for a FYP Manager using Flask and MySQL will depend on the specific test scenarios and criteria set for evaluation [12]. Here are some potential outcomes that can be observed during the testing process:

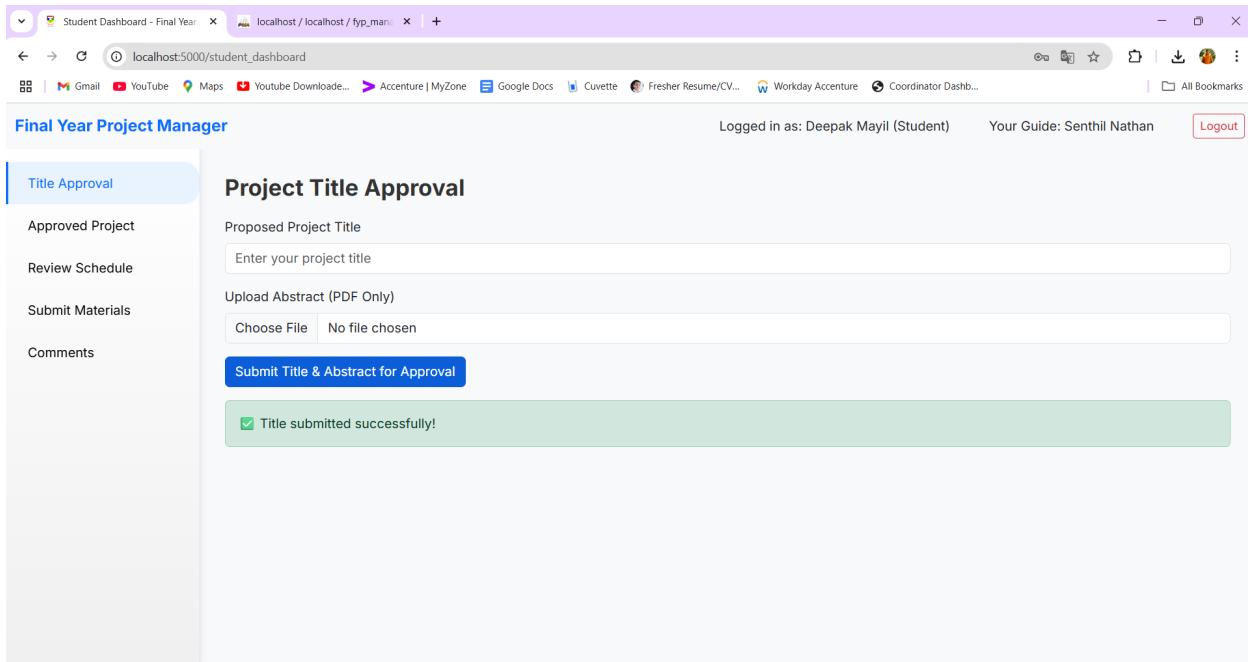


Fig. 5.1 Test Outcome

Fig. 5.1 describes that title has been successfully submitted a project title and abstract for approval, as confirmed by the green success message. The interface includes navigation options such as Title Approval, Approved Project, Review Schedule, Submit Materials, and Comments.

This chapter highlights the role of testing in ensuring system reliability and user satisfaction. It covers unit testing, integration testing, and black box testing, along with detailed test case documentation and evaluation. Test outcomes are summarized based on real-time academic use cases to validate overall system performance. The next chapter presents the conclusion, summarizing key findings, project outcomes, and possible future enhancements, emphasizing the system's role in streamlining academic project management and improving coordination.

# **CHAPTER 6**

## **CONCLUSION**

### **6.1 CONCLUSION**

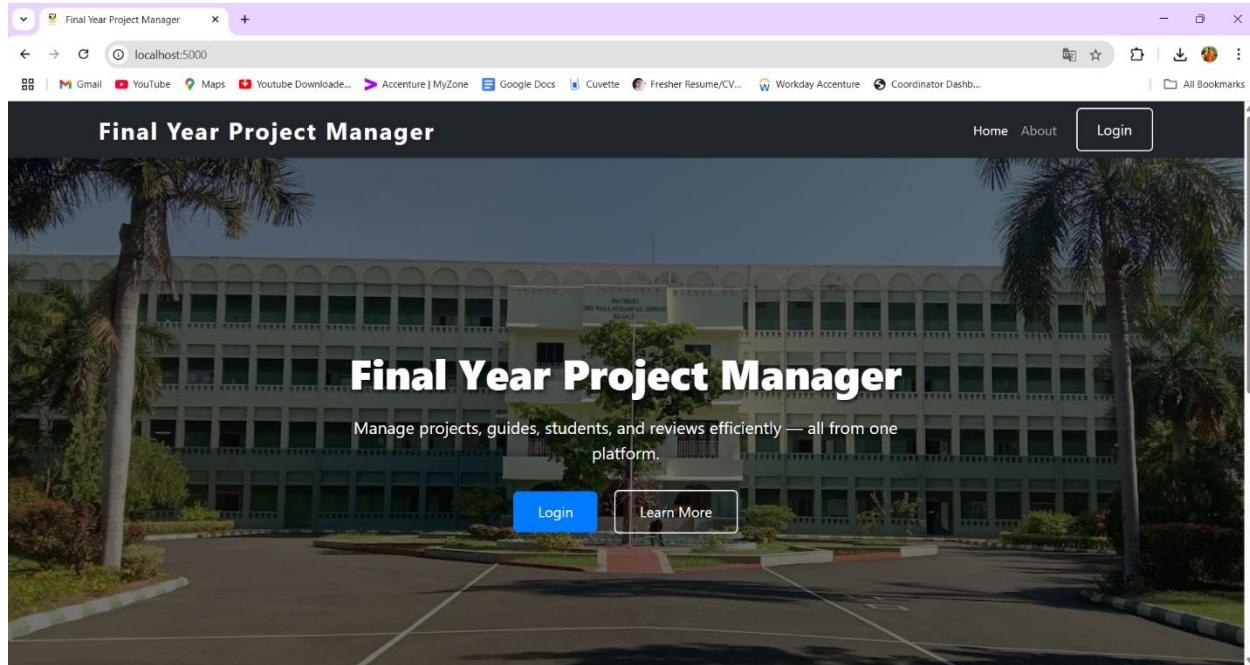
In conclusion, the comprehensive design and development of the Final Year Project (FYP) Web App is rooted in a user-centric approach, employing Bootstrap for Front-End responsiveness and Python with Flask for Back-End efficiency. Leveraging MySQL ensures structured data storage, a critical aspect for housing essential project information. The system's key features, such as project title recommendation, team prioritization, and deduplication mechanisms, aim to optimize the project selection process and enhance collaboration among students, advisors, and administrators. The End User Dashboard, Mentor Assignment, Title Processing and Deduplication, Digital Signature Integration, Review Material Submission, Online/Offline Review, Review Evaluation and Grading, Attendance Tracking, Report Generation, and Notification System modules collectively contribute to streamlining the entire FYP process. The FYP Web App, with its user-friendly interfaces and robust functionalities, stands poised to revolutionize the traditional manual approach to final year project management, ensuring a more streamlined, transparent, and collaborative experience for all stakeholders involved.

### **6.2 FUTURE ENHANCEMENT**

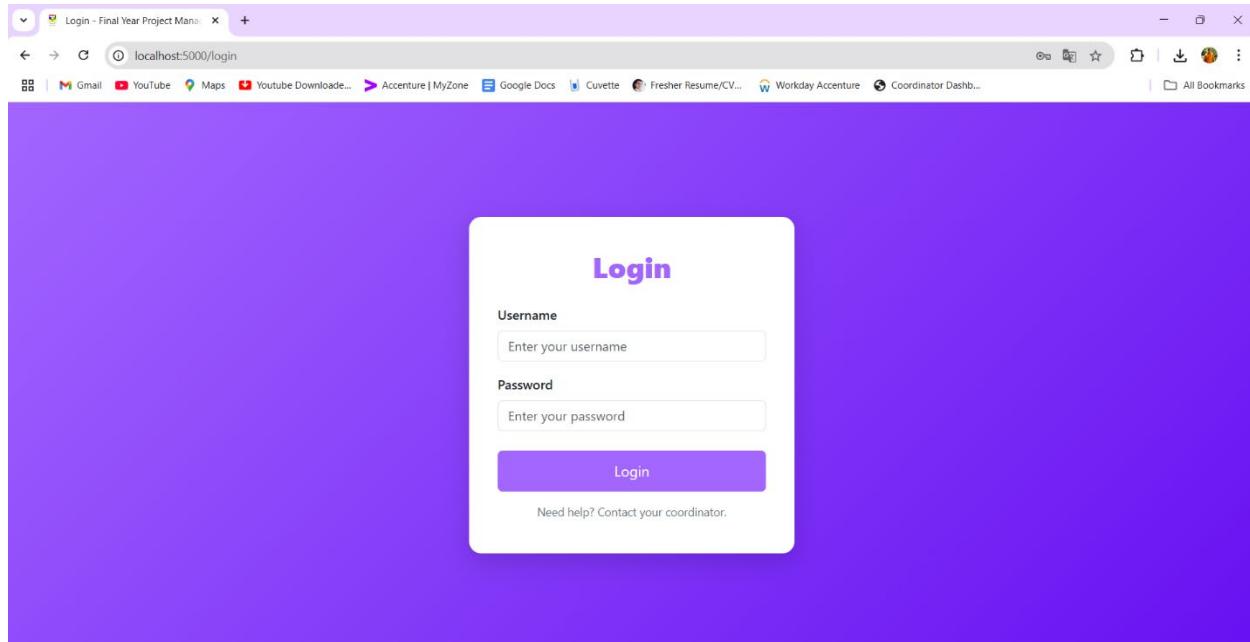
Future enhancements, a dedicated mobile application is in development to provide users with enhanced accessibility and real-time project management capabilities on Android and iOS platforms. Additionally, an automated code review system for technical projects is being implemented to assess code quality and offer constructive feedback, promoting best coding practices. Furthermore, plans include integrating the Final Year Project Management System with existing Learning Management Systems (LMS) to create a cohesive educational experience, streamlining workflows, and fostering a unified learning environment for students. These enhancements aim to elevate user experience, improve project quality, and integrate project activities seamlessly within the broader educational ecosystem.

## APPENDIX A

### SCREENSHOT



A1: Home Page



A2: Login Page

The screenshot shows a web browser window titled 'Coordinator Dashboard - Final' with the URL 'localhost:5000/coordinator'. The page is titled 'Final Year Project Manager' and shows a user logged in as 'Rajkumar (Project Coordinator)'.

The left sidebar contains a navigation menu with the following items:

- Add Staff (Guides) (highlighted)
- Add Students
- Allocate Students to Staff
- Schedule PRC Meeting
- Schedule Review Dates
- Project Overview
- Generate Report
- Manage Staff
- Manage Students

The main content area is titled 'Add Staff (Guides)' and contains the following form fields:

- Staff ID: [Text input field]
- Staff Name: [Text input field]
- Gender: [Select dropdown] (options: Select Gender)
- Department: [Select dropdown] (options: Select Department)
- Designation: [Select dropdown] (options: Select Designation)
- Email: [Text input field]
- Contact Number: [Text input field]

### A3: Add Staff Page

The screenshot shows a web browser window titled 'Coordinator Dashboard - Final' with the URL 'localhost:5000/coordinator'. The page is titled 'Final Year Project Manager' and shows a user logged in as 'Rajkumar (Project Coordinator)'.

The left sidebar contains a navigation menu with the following items:

- Add Staff (Guides)
- Add Students
- Allocate Students to Staff (highlighted)
- Schedule PRC Meeting
- Schedule Review Dates
- Project Overview
- Generate Report
- Manage Staff
- Manage Students

The main content area is titled 'Allocate Students to Staff' and contains the following form fields:

- Select Staff (Guide): [Select dropdown] (options: Select Staff)
- Select Students: [List box]
  - 2303921362221001 - Charan
  - 2303921362221011 - Hari Vignesh
  - 2303921362221012 - Deepak Mayil
  - 2303921362221013 - Anushya S
- Hold Ctrl (or Cmd) to select multiple students
- Allocate** [Blue button]

### A4: Student Allocation Page

**Schedule Review Dates**

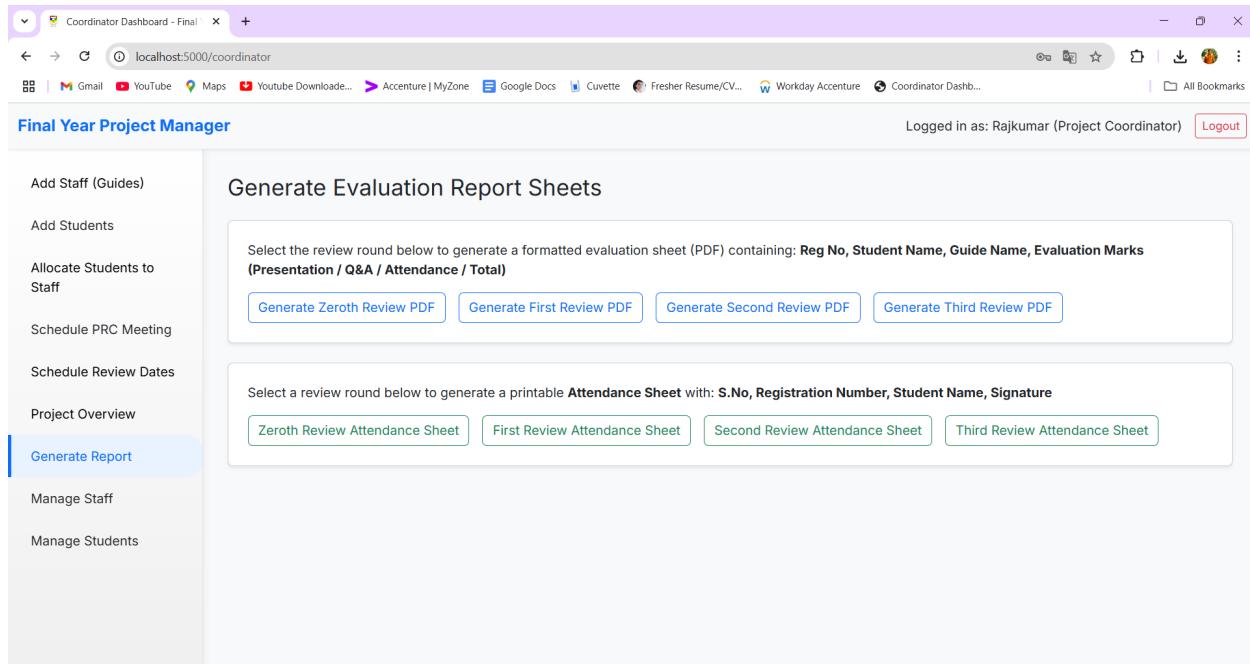
Zeroth Review	First Review	Second Review	Third Review
Review Date mm/dd/yyyy	Review Date mm/dd/yyyy	Review Date mm/dd/yyyy	Review Date mm/dd/yyyy
Time --::--	Time --::--	Time --::--	Time --::--
Online Meeting Link Paste meeting link (Google Meet, Zoom, etc.)	Online Meeting Link Paste meeting link (Google Meet, Zoom, etc.)	Online Meeting Link Paste meeting link (Google Meet, Zoom, etc.)	Online Meeting Link Paste meeting link (Google Meet, Zoom, etc.)
<b>Schedule Zeroth Review</b>	<b>Schedule First Review</b>	<b>Schedule Second Review</b>	<b>Schedule Third Review</b>

## A5: Schedule Review Page

**Project Overview**

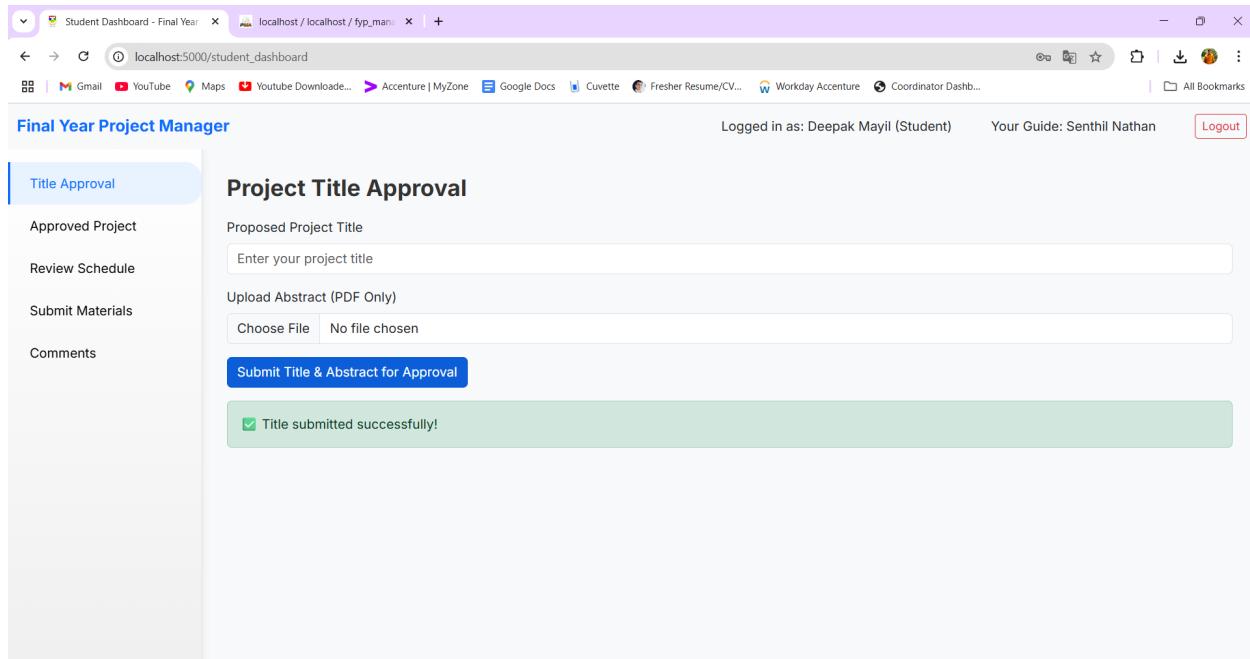
S.No	Reg No	Student Name	Guide Name	Approved Title	Signed Abstract	Attendance / Materials			
						Zeroth Review	First Review	Second Review	Third Review
1	2303921362221001	Charan	Jayakishor	FYP Manager	Not Available	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted
2	2303921362221011	Hari Vignesh	Not Assigned	Not Submitted	Not Available	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted
3	2303921362221012	Deepak Mayil	Senthil Nathan	Not Submitted	Not Available	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted
4	2303921362221013	Anushiya S	Senthil Nathan	SmartDine QR - Restaurant ordering system	<a href="#">View</a>	Attendance: Present Materials : <a href="#">View</a>	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted	Attendance: NA Materials : Not Yet Submitted

## A6: Project Overview Page



The screenshot shows the 'Coordinator Dashboard - Final' page. The left sidebar contains links: Add Staff (Guides), Add Students, Allocate Students to Staff, Schedule PRC Meeting, Schedule Review Dates, Project Overview, Generate Report (which is selected and highlighted in blue), Manage Staff, and Manage Students. The main content area is titled 'Generate Evaluation Report Sheets'. It includes a section for selecting a review round to generate a formatted evaluation sheet (PDF) containing Reg No, Student Name, Guide Name, Evaluation Marks (Presentation / Q&A / Attendance / Total). Buttons for 'Generate Zeroth Review PDF', 'Generate First Review PDF', 'Generate Second Review PDF', and 'Generate Third Review PDF' are available. Below this is another section for generating an 'Attendance Sheet' with S.No, Registration Number, Student Name, and Signature. Buttons for 'Zeroth Review Attendance Sheet', 'First Review Attendance Sheet', 'Second Review Attendance Sheet', and 'Third Review Attendance Sheet' are shown. The top right corner shows the user is logged in as Rajkumar (Project Coordinator) with a 'Logout' button.

## A7: Report Generation Page



The screenshot shows the 'Student Dashboard - Final Year' page. The left sidebar contains links: Title Approval (which is selected and highlighted in blue), Approved Project, Review Schedule, Submit Materials, and Comments. The main content area is titled 'Project Title Approval'. It has fields for 'Proposed Project Title' (with a placeholder 'Enter your project title') and 'Upload Abstract (PDF Only)' (with a file input field showing 'Choose File' and 'No file chosen'). A 'Submit Title & Abstract for Approval' button is present. A green success message at the bottom states 'Title submitted successfully!'. The top right corner shows the user is logged in as Deepak Mayil (Student) with Your Guide: Senthil Nathan and a 'Logout' button.

## A8: Project Title Approval Page

The screenshot shows a web browser window titled "Final Year Project Manager". The left sidebar contains links for "Allocated Students", "Approve Title" (which is highlighted in blue), "Scheduled PRC Meeting", "Reviews & Attendance", "Materials", and "Attendance & Evaluation". The main content area is titled "Approve Project Titles" and includes a sub-instruction: "Review and accept or reject project titles proposed by students. Duplicate titles across the class are not allowed." A table lists a single project title:

Registration Number	Name	Project Title	Abstract	Action
2303921362221012	Deepak Mayil	Cloud Cargo	<a href="#">View</a>	<a href="#">Accept</a> <a href="#">Reject</a>

## A9: Approve Project Title Page

The screenshot shows a web browser window titled "Final Year Project Manager". The left sidebar contains links for "Title Approval", "Approved Project" (which is highlighted in blue), "Review Schedule", "Submit Materials", and "Comments". The main content area is titled "Approved Project" and displays the following details:

**Project Title:** Cloud Cargo  
**Status:** Approved  
**Signed Abstract:** [View Signed Abstract](#)

**Project Description:**

Cloud-based Cargo Management and Live Location Tracking  
Abstract  
The Indian banking industry today is faced with issues such as rising costs of operations, increasing susceptibility to fraudulent attacks on centralized servers and challenges in ensuring transparency. All this, primarily because most of the banking transactions – from opening customer accounts to making global payments – may require intensive manual processing and documentation, involve costly intermediaries and is time-consuming as these transactions need to be validated by various participants at various point in time causing the delay thereby resulting in almost lack of fraud proof real time solution. Banks are continuously exploring new ways to perform transactions quicker for an enhanced customer service, while ensuring cost efficiency in its operations and assuring transparency to customers and regulators. For this,

## A10: Approved Project Page

**Final Year Project Manager**

Logged in as: Deepak Mayil (Student) Your Guide: Senthil Nathan [Logout](#)

**Submit Project Materials**

Select Review Round: Review 0

Choose Files: Choose Files No file chosen  
CloudCargo - Abstract.docx

**Submit Materials**

## A11: Material Submission Page

**Final Year Project Manager**

Logged in as: Senthil Nathan (Staff) [Logout](#)

**Attendance & Evaluation**

Select Student	Select Review Round	Attendance Status	
Anushiya S (2303921362221013)	Zeroth	Present	
Presentation (Max 40)	Q/A (Max 40)	Attendance (Max 20)	Total Marks
40	38	20	98

Comments: Good Presentation

**Submit Evaluation**

## A12: Attendance and Evaluation Page

## APPENDIX B

### SAMPLE CODE

#### Main.py

```
from flask import Flask, render_template, redirect, request, session, url_for
from datetime import datetime
import datetime
import os
from werkzeug.utils import secure_filename
from flask import send_from_directory, abort
from docx import Document
import mysql.connector
from docx.shared import Inches
from docx2pdf import convert
import pythoncom
import time
import base64
from io import BytesIO
import io
import fitz # PyMuPDF
from fpdf import FPDF
from flask import send_file
from flask import Response
from flask_socketio import SocketIO, emit, join_room, leave_room
from engineio.payload import Payload
Payload.max_decode_packets = 200
from werkzeug.utils import secure_filename
```

```

app = Flask(__name__)
app.secret_key = 'abcdef'

socketio = SocketIO(app, async_mode='eventlet') # Explicitly set async_mode

_users_in_room = {} # stores room wise user list
_room_of_sid = {} # stores room joined by an used
_name_of_sid = {} # stores display name of users

mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    password="",
    charset="utf8",
    use_pure=True,
    database="fyp_db_update"
)

@app.route('/hod_generate_reports', methods=['GET'])

def hod_generate_reports():

    if 'staff_id' not in session or session.get('user_type') != 'hod':
        return redirect(url_for('hodlog'))

    staff_id = session['staff_id']

    cursor = mydb.cursor()

    cursor.execute("SELECT dept FROM fyp_staff WHERE staff_id = %s", (staff_id,))
    dept = cursor.fetchone()[0]

    cursor.execute("SELECT * FROM fyp_student WHERE dept = %s", (dept,))
    students = cursor.fetchall()

    cursor.close()

    return render_template('hod_generate_reports.html', students=students)

from fpdf import FPDF

```

```

from flask import send_file

@app.route('/generate_student_report/<reg_no>')
def generate_student_report(reg_no):
    cursor = mydb.cursor()

    # Get student info
    cursor.execute("SELECT * FROM fyp_student WHERE reg_no = %s", (reg_no,))
    student = cursor.fetchone()

    # Get title info
    cursor.execute("SELECT * FROM fyp_title WHERE reg_no = %s", (reg_no,))
    title = cursor.fetchone()

    # Get review submissions
    cursor.execute("SELECT * FROM fyp_review_report WHERE reg_no = %s", (reg_no,))
    review_reports = cursor.fetchall()

    # Get review results
    cursor.execute("SELECT * FROM fyp_review_result WHERE reg_no = %s", (reg_no,))
    review_results = cursor.fetchall()

    # Get staff allocation info
    cursor.execute("SELECT staff_id FROM fyp_staff_allocation WHERE reg_no = %s",
    (reg_no,))
    staff_alloc = cursor.fetchone()
    staff_id = staff_alloc[0] if staff_alloc else "N/A"
    cursor.execute("SELECT name FROM fyp_staff WHERE staff_id = %s", (staff_id,))
    staff_name_data = cursor.fetchone()
    staff_name = staff_name_data[0] if staff_name_data else "Unknown"

    # Get review sessions created by staff or hod for student dept
    cursor.execute("SELECT dept FROM fyp_student WHERE reg_no = %s", (reg_no,))
    dept_data = cursor.fetchone()

```

```

dept = dept_data[0] if dept_data else ""
cursor.execute("""
    SELECT review_date, review_time, name, requirement FROM fyp_review
    WHERE dept = %s ORDER BY review_date
""", (dept,))
review_schedule = cursor.fetchall()
cursor.close()

# Create PDF
pdf = FPDF()
pdf.add_page()
pdf.set_font("Arial", size=12)

# Header
pdf.cell(200, 10, txt=f"Student Report: {student[1]} ({reg_no})", ln=True, align='C')
pdf.ln(10)

# Basic student info
pdf.multi_cell(0, 10, f"""
Name: {student[1]}
Reg No: {student[2]}
Department: {student[8]}
Email: {student[6]}
Allocated Staff: {staff_name} ({staff_id})
""")

# Project Title
if title:
    pdf.set_font("Arial", 'B', 12)
    pdf.cell(200, 10, "Project Title", ln=True)
    pdf.set_font("Arial", size=12)

```

```

pdf.multi_cell(0, 10, f"Title: {title[6]}\nDocument: {title[5]}\n")

# Review Schedule

if review_schedule:

    pdf.set_font("Arial", 'B', 12)

    pdf.cell(200, 10, "Review Schedule", ln=True)

    pdf.set_font("Arial", size=12)

    for r in review_schedule:

        pdf.multi_cell(0, 10, f"Date: {r[0]}, Time: {r[1]}, Name: {r[2]}, Requirement: {r[3]}")

# Review Submissions

if review_reports:

    pdf.set_font("Arial", 'B', 12)

    pdf.cell(200, 10, "Submitted Review Files", ln=True)

    pdf.set_font("Arial", size=12)

    for r in review_reports:

        pdf.multi_cell(0, 10, f"Review ID: {r[1]}, Date: {r[2]}, File(s): {r[5]}")

# Review Results

if review_results:

    pdf.set_font("Arial", 'B', 12)

    pdf.cell(200, 10, "Review Results", ln=True)

    pdf.set_font("Arial", size=12)

    for res in review_results:

        pdf.multi_cell(0, 10, f"Title: {res[2]}, Marks: {res[4]}, Remark: {res[5]}")

# Output PDF

path = f'static/reports/student_report_{reg_no}.pdf'

pdf.output(path)

return send_file(path, as_attachment=True)

```

```

@app.route('/submit_instruction', methods=['GET', 'POST'])

def submit_instruction():

    if 'staff_id' not in session or session.get('user_type') != 'hod':
        return redirect(url_for('hodlog'))

    staff_id = session['staff_id']

    cursor = mydb.cursor()

    cursor.execute("SELECT dept FROM fyp_staff WHERE staff_id = %s", (staff_id,))

    dept = cursor.fetchone()[0]

    msg = ""

    if request.method == 'POST':

        instruction = request.form['instruction']

        cursor.execute("""
            INSERT INTO fyp_review_instruction (dept, instruction, created_by)
            VALUES (%s, %s, %s)
            """, (dept, instruction, staff_id))

        mydb.commit()

        msg = "Instruction submitted successfully."

    cursor.execute("SELECT * FROM fyp_review_instruction WHERE dept = %s ORDER BY
    created_at DESC", (dept,))

    instructions = cursor.fetchall()

    cursor.close()

    return render_template('submit_instruction.html', instructions=instructions, msg=msg)

```

```

@app.route('/hod_generate_reports', methods=['GET'])

def hod_generate_reports():
    if 'staff_id' not in session or session.get('user_type') != 'hod':
        return redirect(url_for('hodlog'))

    staff_id = session['staff_id']

    cursor = mydb.cursor()

    cursor.execute("SELECT dept FROM fyp_staff WHERE staff_id = %s", (staff_id,))

    dept = cursor.fetchone()[0]

    cursor.execute("SELECT * FROM fyp_student WHERE dept = %s", (dept,))

    students = cursor.fetchall()

    cursor.close()

    return render_template('hod_generate_reports.html', students=students)

@app.route('/generate_student_report/<reg_no>')

def generate_student_report(reg_no):
    cursor = mydb.cursor()

    # Get student info

    cursor.execute("SELECT * FROM fyp_student WHERE reg_no = %s", (reg_no,))

    student = cursor.fetchone()

    # Get title info

    cursor.execute("SELECT * FROM fyp_title WHERE reg_no = %s", (reg_no,))

    title = cursor.fetchone()

    # Get review submissions

    cursor.execute("SELECT * FROM fyp_review_report WHERE reg_no = %s", (reg_no,))

    review_reports = cursor.fetchall()

    # Get review results

    cursor.execute("SELECT * FROM fyp_review_result WHERE reg_no = %s", (reg_no,))

    review_results = cursor.fetchall()

```

```

# Get staff allocation info

cursor.execute("SELECT staff_id FROM fyp_staff_allocation WHERE reg_no = %s",
(reg_no,))

staff_alloc = cursor.fetchone()

staff_id = staff_alloc[0] if staff_alloc else "N/A"

cursor.execute("SELECT name FROM fyp_staff WHERE staff_id = %s", (staff_id,))

staff_name_data = cursor.fetchone()

staff_name = staff_name_data[0] if staff_name_data else "Unknown"

# Get review sessions created by staff or hod for student dept

cursor.execute("SELECT dept FROM fyp_student WHERE reg_no = %s", (reg_no,))

dept_data = cursor.fetchone()

dept = dept_data[0] if dept_data else ""

cursor.execute(""""

    SELECT review_date, review_time, name, requirement FROM fyp_review
    WHERE dept = %s ORDER BY review_date
""", (dept,))

review_schedule = cursor.fetchall()

# Get student info

cursor.execute("SELECT * FROM fyp_student WHERE reg_no = %s", (reg_no,))

student = cursor.fetchone()

# Get title info

cursor.execute("SELECT * FROM fyp_title WHERE reg_no = %s", (reg_no,))

cursor.close()@app.route('/schedule_prc_meeting', methods=['POST'])

def schedule_prc_meeting():

    # Get data from the request

    data = request.get_json()

    meeting_date = data.get('meetingDate')

```

```

meeting_time = data.get('meetingTime')
meeting_link = data.get('meetingLink')

if not meeting_date or not meeting_time or not meeting_link:
    return jsonify({'success': False, 'message': 'All fields are required.'}), 400

try:
    # Establish MySQL connection
    conn = get_db_connection()
    cursor = conn.cursor()

    # Insert meeting details into MySQL table
    cursor.execute("""
        INSERT INTO prc_meetings (meeting_date, meeting_time, meeting_link)
        VALUES (%s, %s, %s)
    """, (meeting_date, meeting_time, meeting_link))

    # Commit and close the connection
    conn.commit()
    cursor.close()
    conn.close()

    return jsonify({'success': True, 'message': 'PRC Meeting scheduled successfully.'})
except mysql.connector.Error as e:
    return jsonify({'success': False, 'message': str(e)}), 500

if __name__ == '__main__':
    app.run(debug=True)

```

## REFERENCES

- [1] Aldriye, H., A. Alkhalfaf and M. Alkhalfaf, "Automated grading systems for programming assignments: A literature review", Int. J. Adv. Computer Sci. Appl., Vol. 10, No. 3, pp. 215-222, 2019.
- [2] Alves, N. d. C., C. G. von Wangenheim, J. C. R. Hauck and A. F. Borgatto, A Large-Scale Evaluation of a Rubric for the Automatic Assessment of Algorithms and Programming Concepts, New York, NY, USA: ACM, pp. 556-562, 2020.
- [3] Dey, R., R. Dickler, L. Hirshfield, W. Goss, M. Tissenbaum and S. Puntambekar, "Using participatory design studies to collaboratively create teacher dashboards", Proc. Int. Conf. Artif. Intell. Educ., pp. 506-509, 2022.
- [4] Fu, Q.-K., D. Zou, H. Xie and G. Cheng, "A review of AWE feedback: Types learning outcomes and implications", Computer Assist. Lang. Learn., pp. 1-43, 2022.
- [5] Gomez-del Rio, T. and J. Rodriguez, "Design and assessment of a project-based learning laboratory for integrating knowledge and improving engineering design skills", Educ. Eng., Vol. 40, pp. 17-28, 2022.
- [6] Guo, P., N. Saab, L. S. Post and W. Admiraal, "A review of project-based learning in higher education: Student outcomes and measures", Int. J. Educ. Res., Vol. 102, 2020.
- [7] Hao, Q., J. P. Wilson, C. Ottaway, N. Iriumi, K. Arakawa and D. H. S. IV, "Investigating the F, essential of meaningful automated formative feedback for programming assignments", Proc. IEEE Symp. Vis. Lang. Hum-Centric Computer., pp. 151-155, 2019.
- [8] Israel-Fishelson, R. and D. Kohen-Vacs, "Toward enhancing learning experiences via student-centered dashboards", 2023.
- [9] Lee, S.W., "Deploying Flask Applications on WampServer: A Case Study in Local Development", International Journal of Web Development, Vol. 12, No. 1, pp. 45–60, 2021.
- [10] McKinney, W., "Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython", O'Reilly Media, pp. 1–579, 2022.
- [11] Naveed, M. S. and M. Sarim, "Didactic strategy for learning theory of automata & formal languages", Proc. Pakistan Acad. Sci. A. Phys. Computer Sci., Vol. 55, No. 2, pp. 55-67, 2018.

- [12] Rasooli, H., Zandi and C. DeLuca, "Re-conceptualizing classroom assessment fairness: A systematic meta-ethnography of assessment literature and beyond", *Stud. Educ. Eval.*, Vol. 56, pp. 164-181, 2018.
- [13] Saito, D., R. Yajima, H. Washizaki and Y. Fukazawa, "Validation of rubric evaluation for programming education", *Educ. Sci.*, Vol. 11, pp. 656-10, 2021.
- [14] Smith, S., "Adaptive Layout Techniques in Bootstrap 4 for Mobile-First Web Design", *International Journal of Web Engineering*, Vol. 5, No. 2, pp. 45–60, 2018.
- [15] Vaswani, V., "MySQL: The Complete Reference", McGraw-Hill, Vol. 12, No. 8, pp. 1–784, 2002.
- [16] Veluvali, P. and J. Surisetti, "Learning management system for greater learner engagement in higher education a review", *High. Educ. Future*, Vol.9, No. 1, pp. 107-121, Jan. 2022.
- [17] Vernon, V., Accelerated Software Development: Agile Methods for Rapid Delivery, Publisher, 2<sup>nd</sup> Edition, 2016.
- [18] Wen, M., K. Maki, S. Dow, J. D. Herbsleb and C. Rose, "Supporting virtual team formation through community-wide deliberation", *Proc. ACM Hum. Computer Interaction*, Vol. 1, pp. 1-19, 2017.
- [19] Xu, J., "Performance Optimization Techniques in MySQL Database Design", *Journal of Data Engineering*, Vol. 8, No. 2, pp. 85–102, 2021.
- [20] Yun, W. S., "Digitalization challenges in education during COVID-19: A systematic review", *Cogent Educ.*, Vol. 10, No. 1, 2023.